

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN THỰC HÀNH
MÔN: VẬT LÝ CHO CÔNG NGHỆ THÔNG TIN
(PHY00007 – 21CLC01)
ĐỒ ÁN: Ổ KHÓA THÔNG MINH (DEADLOCK)

GIẢNG VIÊN LÝ THUYẾT

CAO XUÂN NAM

GIẢNG VIÊN THỰC HÀNH

ĐẶNG HOÀI THƯƠNG

SINH VIÊN THỰC HIỆN

21127621 – ÂU DƯƠNG KHANG
21127205 – CAO HOÀI YẾN VY
21127559 – TRẦN ĐÌNH TRUNG

Mục lục

I) Thông tin nhóm.....	1
II) Phân công công việc:.....	1
III) Giới thiệu sản phẩm.....	2
1) Lý do thực hiện sản phẩm.....	2
2) Mô tả chức năng toàn hệ thống	3
A) Mô tả sản phẩm	3
B) Website:	5
B) Mô tả cloud lưu trữ	11
C) Gửi thông báo trên điện thoại:.....	14
IV) Thiết kế 3D của sản phẩm:.....	17
V) Sơ đồ truyền nhận dữ liệu.....	22

I) Thông tin nhóm

Mã số nhóm: 9

STT	MSSV	Họ và tên	Email
1	21127621	Âu Dương Khang	adkhang21@clc.fitus.edu.vn
2	21127205	Cao Hoài Yến Vy	chyvy21@clc.fitus.edu.vn
3	21127559	Trần Đình Trung	tdtrung21@clc.fitus.edu.vn

II) Phân công công việc:

Khang: Lập trình Node-RED và MQTT

- Tạo giao diện node – red chính.

- Lập trình một hệ thống giao tiếp giữa thiết bị ESP8266 và Node-RED thông qua giao thức MQTT. Cài đặt MQTT Server để làm trung gian giữa ESP8266 và Node-RED.
- Bạn cần lập trình trên ESP8266 để gửi dữ liệu cảm biến, trạng thái cửa, và thông tin hoạt động lên MQTT Server.
- Trong Node-RED, tạo các luồng (flows) để đọc dữ liệu từ MQTT Server và hiển thị trạng thái, hoạt động của ổ khóa, buzzer, cảm biến trên giao diện Node-RED.
- Lập trình tải dữ liệu lên cloud từ ESP.

Trung: Lập trình chức năng mở bằng keypad, RFID, lắp mạch

- Vẽ mô hình 3D sản phẩm.
- Lập trình chức năng mở cửa thông qua các phương pháp khác nhau: keypad và thẻ RFID.
- Lắp mạch và cài đặt keypad và đọc thẻ RFID để đảm bảo việc nhận dạng thẻ và mật khẩu.
- Lập trình để khi người dùng nhập mật khẩu chính xác hoặc quét thẻ RFID hợp lệ, cửa sẽ mở và thực hiện các hoạt động khác như đóng cửa sau thời gian xác định, âm thanh khi có các tác động đến ổ khóa từ buzzer.

Vy: Lập trình giao tiếp UART giữa Node-RED và ESP

- Vẽ sơ đồ truyền nhận dữ liệu.
- Lập trình giao tiếp UART giữa Node-RED và ESP8266.
- Lập trình truyền nhận dữ liệu giữa hai thiết bị, bao gồm các tín hiệu để điều khiển cửa, gửi thông tin cảm biến và trạng thái sang ESP.
- Lập trình gửi thông báo khi có khẩn cấp sang điện thoại người dùng.
- Lập trình tải dữ liệu cloud lên node-red.

III) Giới thiệu sản phẩm

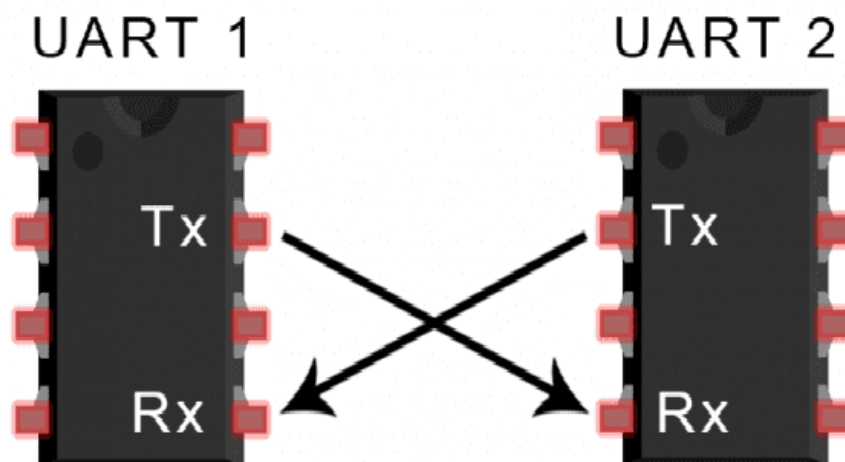
1) Lý do thực hiện sản phẩm

- **Tên sản phẩm:** Khóa thông minh DEADLOCK
- Sản phẩm "DEADLOCK" là một giải pháp an ninh và quản lý ra vào nhà thông minh, được phát triển để đáp ứng nhu cầu ngày càng cao về an ninh và tiện ích trong các khu vực dân cư. Với sự gia tăng của hệ thống giám sát và an ninh, nhà thông minh đã trở thành xu hướng phát triển quan trọng. DEADLOCK ra đời để cung cấp giải pháp an toàn, dễ dàng quản lý và tăng cường sự riêng tư cho người dùng.
- DEADLOCK sử dụng các công nghệ thông minh như camera, cảm biến chuyển động và các tính năng khác để cho phép người dùng xác định ai đang đến và đi từ ngôi nhà của họ. Sản phẩm này giúp giảm thiểu rủi ro mất trộm và cung cấp sự an toàn hơn cho gia đình. Hơn nữa, DEADLOCK có khả năng tích hợp với các hệ thống nhà thông minh khác, như điều khiển ánh sáng, nhiệt độ và âm thanh, tạo ra một môi trường sống thông minh và tiện ích cho người dùng.
- Tóm lại, DEADLOCK mang lại lợi ích đáng kể cho người dùng bao gồm an toàn, tiện ích, ngăn ngừa trộm cắp và cải thiện sự riêng tư. Sản phẩm này hứa hẹn trở thành một sản phẩm thông minh phổ biến trong tương lai, đóng góp vào sự phát triển của ngành công nghệ an ninh và nhà thông minh.

2) Mô tả chức năng toàn hệ thống

A) Mô tả sản phẩm

- Sản phẩm này tập trung vào việc phát triển một ổ khóa thông minh sử dụng board ESP8266 và các thiết bị ngoại vi như arduino, các cảm biến, keypad,... Hệ thống này cho phép người dùng kiểm soát và giám sát từ xa qua internet, bao gồm chức năng quét thẻ từ RFID và mật khẩu, điều khiển thiết bị đóng mở cửa, cảnh báo khoảng cách, lưu trữ dữ liệu cảm biến và lịch sử hoạt động lên cloud, và thông báo tới điện thoại/email trong tình huống khẩn cấp.
- Sử dụng board ESP8266:
 - + Ở đồ án này, em chúng em sử dụng cả ESP8266 và Arduino Uno để kết nối với thiết bị. Arduino hỗ trợ điều khiển thiết bị, và ESP8266 dùng để hỗ trợ kết nối internet và truyền nhận dữ liệu lên internet.
 - + Chúng em sử dụng giao tiếp UART để giao tiếp giữa ESP8266 và Arduino. Chân RX của ESP nối với chân TX của Arduino và ngược lại. Giao tiếp bằng SoftwareSerial, giao tiếp từ Arduino -> ESP bằng cách sử dụng lệnh `Serial.println()` và giao tiếp từ ESP -> Arduino bằng cách sử dụng `mySerial.readStringUntil('\n')`.
 - + Mô tả kết nối UART:



Hình 1: Mô tả giao tiếp UART

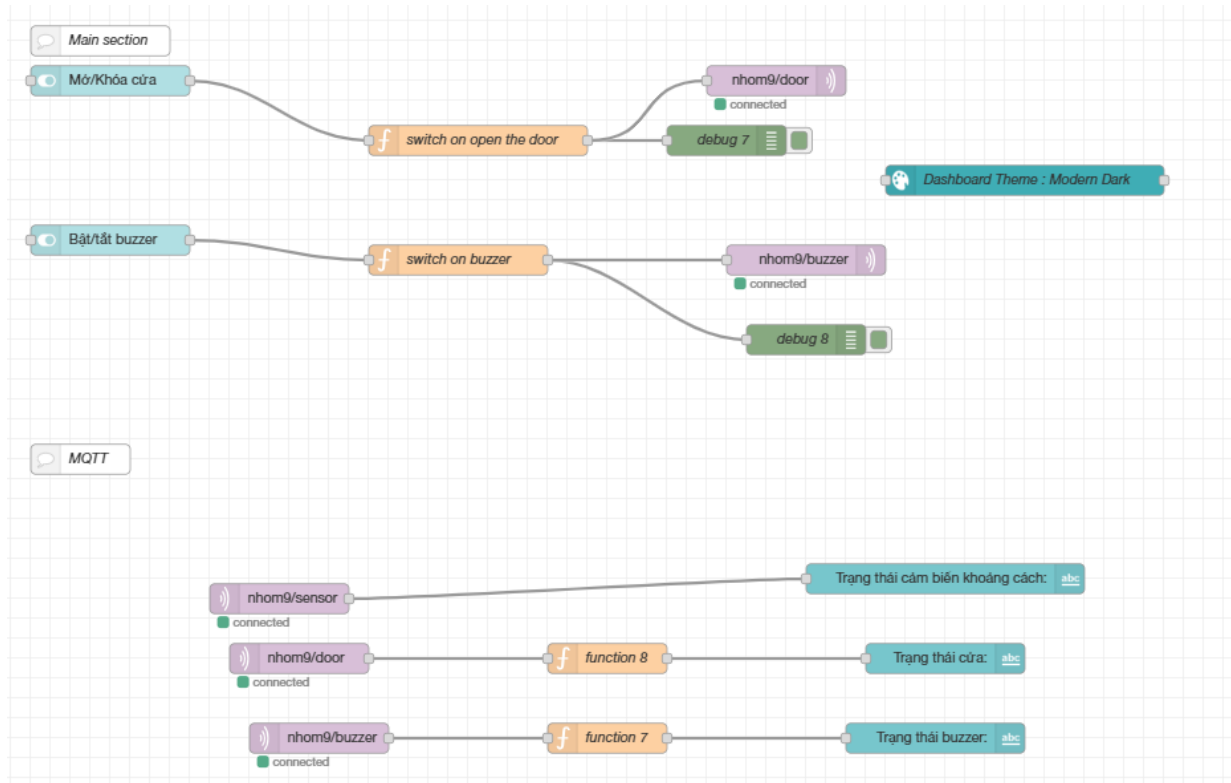
- Phạm vi sử dụng: sử dụng gắn trực tiếp lên cửa của ngôi nhà để kéo chốt khóa
- Sản phẩm có các thiết bị input và output như sau:
 - + INPUT:
 - Cảm biến chuyển động.
 - Keypad
 - Thẻ từ
 - + OUTPUT:
 - LED
 - Buzzer

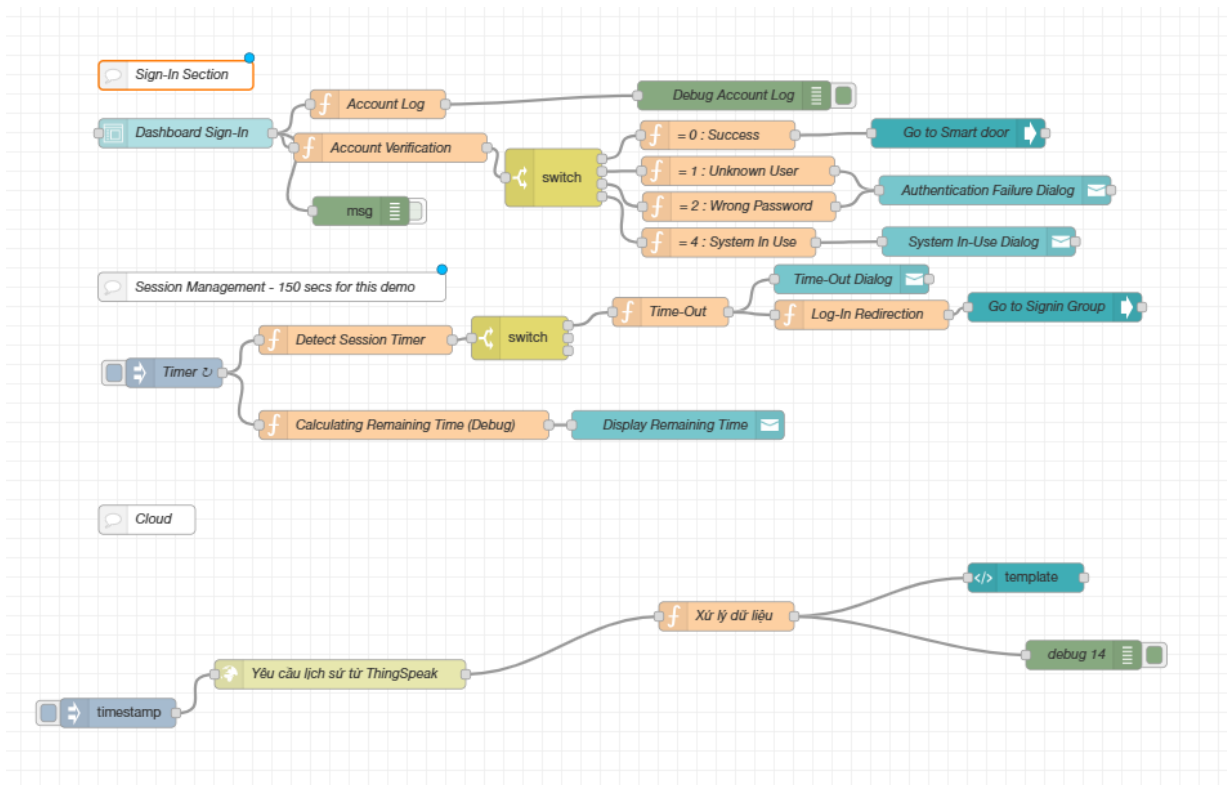
- Servo
- Ta cũng có thể xem thiết bị input cảm biến chuyển động trên web với 2 trạng thái là (ON/OFF).
- Các chức năng của sản phẩm:
 - + Mở khóa bằng RFID và mật khẩu:
 - Sử dụng thiết bị RFID để quét thẻ từ. Nếu thẻ từ hợp lệ, chuyển sang bước tiếp theo. Tại bước này nếu RFID sai buzzer sẽ phát ra tiếng để cảnh báo sai thẻ từ, nếu đúng sẽ kêu “tít tít” từ buzzer để báo hiệu chuyển sang bước sử dụng keypad để mở khóa.
 - Sử dụng keypad để nhập mật khẩu. Nếu mật khẩu đúng, tiến hành mở khóa, mật khẩu sai sẽ phát ra tiếng kêu liên tục và không mở cửa.
 - Nếu thẻ từ sai quá 3 lần hoặc mật khẩu sai quá 3 lần thì sẽ gửi thông báo về điện thoại người dùng thông qua IFTTT với thông báo rằng “Failed attempt to unlock 3 times”
 - Điều khiển thiết bị đóng mở:
 - Trường hợp mở cửa bằng thẻ từ và mật khẩu:
 - Nếu hoàn thành 2 bước trên thì cửa sẽ mở và đếm ngược 5s kèm theo tiếng kêu đếm giây từ buzzer để đóng cửa lại sau khi người dùng đã đi vào nhà.
 - Trường hợp mở cửa bằng kết nối wifi:
 - Có thể mở cửa hoặc đóng cửa từ xa trên web bằng cách sử dụng các lựa chọn.
 - Cửa sẽ được mở/đóng bằng cách xoay servo 180 độ để kéo chốt cửa. (điều khiển output Servo trên website).
 - Sau khi vào nhà thì có thể mở cửa từ bên trong, còn bên ngoài muốn mở cửa thì cần thẻ từ và keypad hoặc thao tác thông qua điện thoại được kết nối wifi của chủ nhà.
 - Mỗi lần servo quay thì dữ liệu về thời gian cũng như trạng thái đóng/mở cửa cũng được gửi lên cloud và hiển thị lên web để chủ nhà theo dõi tình hình ra vào của nhà mình.
 - + Cảm biến khoảng cách và cảnh báo:
 - Sử dụng cảm biến khoảng cách để kiểm tra môi trường. Cảm biến chuyển động sẽ ghi nhận chuyển động xung quanh cửa. Đặc biệt là khi trời tối, nếu cảm biến chuyển động ghi nhận có chuyển động quanh cửa thì sẽ gửi thông báo về điện thoại người dùng cảnh báo có trộm hoặc chuyển động không mong muốn.
 - Thông báo sẽ được gửi qua app IFTTT trên điện thoại của chủ nhà.
 - + Gửi dữ liệu và lịch sử hoạt động lên cloud:
 - Sử dụng board ESP8266 để gửi dữ liệu cảm biến và lịch sử hoạt động của cửa lên dịch vụ đám mây và gửi thông kê hàng tuần để khi có bất thường chủ nhà sẽ xử lý kịp.
 - Gửi giá trị cảm biến khoảng cách.
 - Gửi trạng thái của cửa (đóng/mở - đóng là 0, mở là 1) và trạng thái mở cửa thành công hay thất bại (thất bại - 0, thành công - 1)
 - + Hiển thị yêu cầu và thông báo trên LCD:

- Khi sử dụng ổ khóa, các yêu cầu như quét thẻ từ, nhập mật khẩu và thông báo mở cửa, đóng cửa, mật khẩu sai, đúng sẽ được hiển thị lên LCD giúp người dùng nắm bắt thông tin tốt hơn.

B) Website:

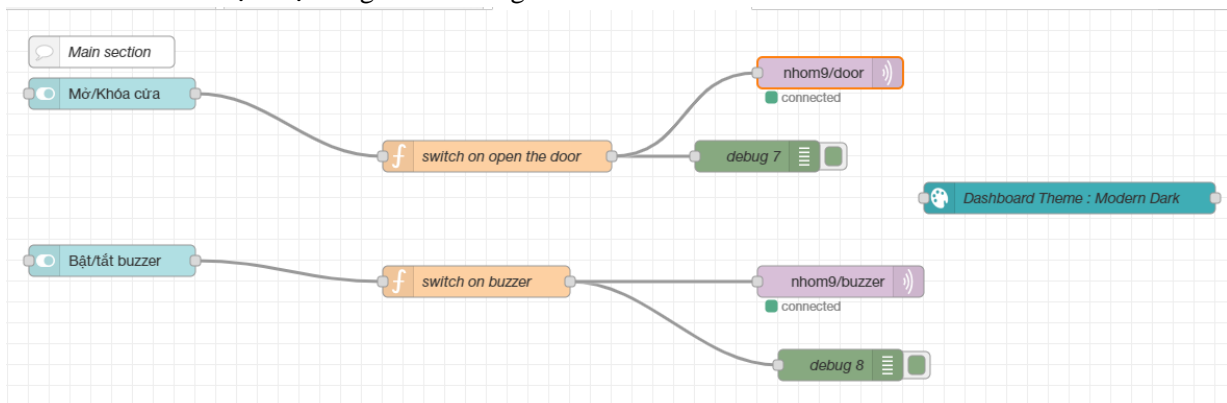
a) Mô tả flow node - red:





Hình 2: Mô tả flow node-red

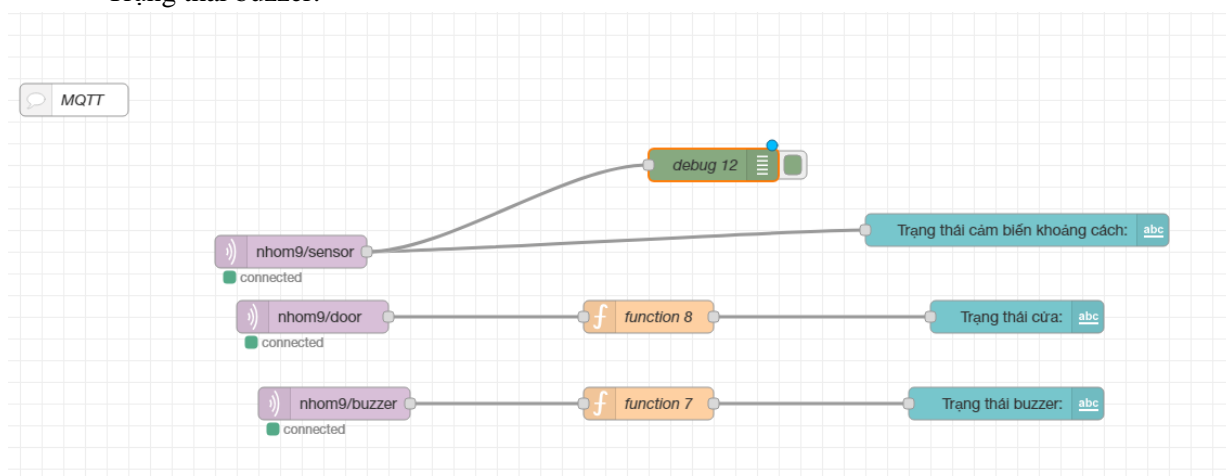
- Ở phần main section: Đây là flow miêu tả các chức năng chính được mô tả bên dưới của ổ khóa gồm nút để mở khóa cửa, nút để bật buzzer. Sau khi mở cửa hoặc bật buzzer, sẽ có các hàm xử lý thông tin đầu vào của các nút để đưa lên MQTT truyền về thiết bị. Kèm với đó là node Dashboard Theme nhằm cung cấp giao diện cho trang web. Chúng ta có 2 node switch gồm node Mở/Khóa cửa và Bật/tắt buzzer có công dụng khi mở (switch = true) sẽ mở cửa hay bật buzzer. Function switch on open the door có nhiệm vụ chuyển hóa thông tin từ node switch thành thông tin ESP8266 có thể đọc được, từ đó thông tin được truyền qua node MQTT in đưa về server MQTT với topic nhom9/door. Sau đó nhiệm vụ của thiết bị chỉ cần đọc thông tin từ server và thực hiện theo nút switch từ trên node red. Tương tự cho function switch on buzzer để bật buzzer. Node Dashboard theme có nhiệm vụ trang trí cho trang web.



Hình 3: Flow main-section

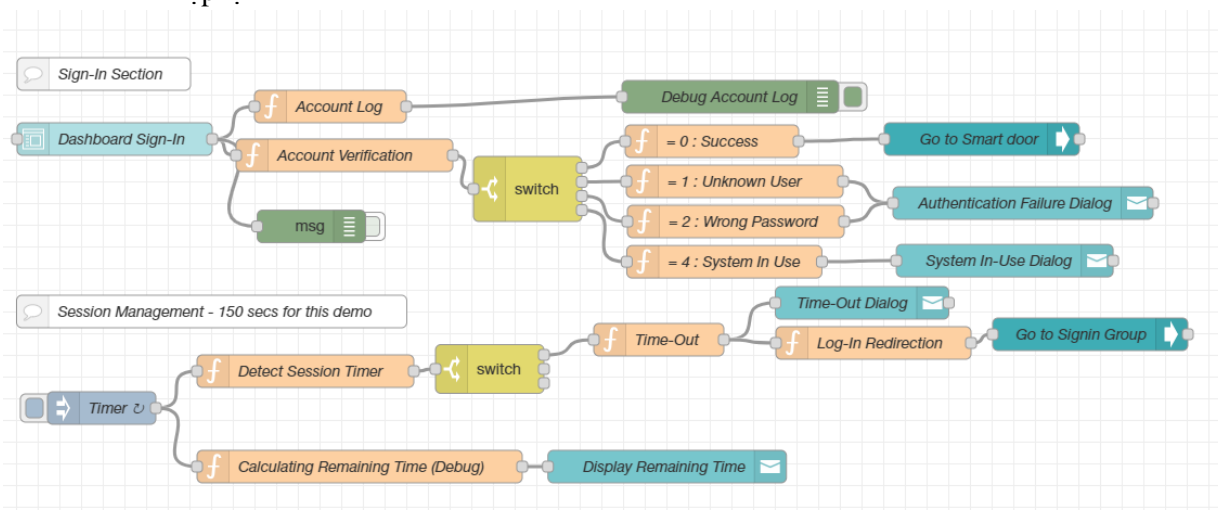
- Ở phần MQTT: Đây là flow miêu tả các trạng thái của cửa, buzzer và cảm biến chuyển động. Chúng được cập nhật liên tục từ thiết bị thông qua MQTT và ESP8266. Ở trạng thái door và buzzer có thêm các function nhằm chuyển đổi dữ liệu để hiện lên website

một cách rõ ràng, dễ hiểu cho người dùng. Node MQTT out để nhận dữ liệu từ ESP8266 truyền lên và cho nó xuất hiện dưới dạng text của node red dashboard. Chúng ta cũng có các function 7 và 8 có nhiệm vụ thông tin truyền dẫn từ server MQTT đưa về các dạng chữ Open hoặc Closed đối với node Trạng thái cửa, hay ON hoặc OFF đối với node Trạng thái buzzer.



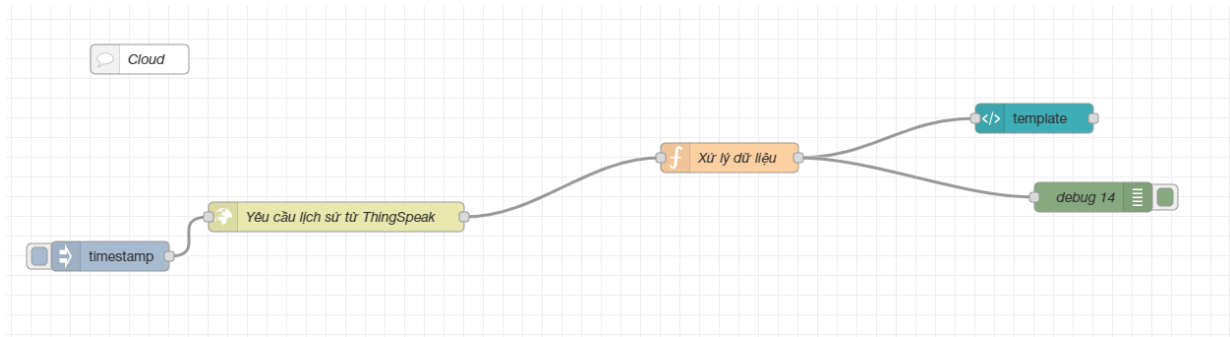
Hình 4: Flow MQTT

- Ở phần Sign-In Section: Ta có dashboard của phần đăng nhập gồm có tên đăng nhập và mật khẩu. Khi nhập vào tên đăng nhập và mật khẩu, function account verification có nhiệm vụ xử lý phần đăng nhập đó.
- Có 4 trạng thái đăng nhập gồm: Success - đăng nhập thành công, Unknown user - sai tên đăng nhập, Wrong password, mật khẩu sai và System in use, đăng nhập thành công nhưng nó đang được sử dụng ở máy khác. Tùy vào các trạng thái mà ta sẽ di chuyển tới các tab và hiển thị thông báo khác nhau và có các function có tên tương ứng xử lý các trạng thái. Nếu như đăng nhập thành công ta sẽ vào tab điều khiển Smart door do phần flow ở main section tạo nên, nếu sai tên đăng nhập hoặc sai mật khẩu sẽ hiển thị thông báo đăng nhập không thành công, nếu đăng nhập vào khi có người khác đang sử dụng cũng sẽ có thông báo tương ứng. Ngoài ra, chức năng chỉ cho phép đăng nhập 1 tài khoản trong vòng 150s. Khi kích hoạt tài khoản thành công và vào trang web, người dùng được sử dụng trong vòng 150s, khi hết thời gian cho phép trước đó một lúc sẽ hiện một box thông báo phía bên phải website để báo cho người dùng biết đã hết thời gian sử dụng web, yêu cầu đăng nhập lại



Hình 5: Flow Sign-In

- Ở phần Cloud: sử dụng node http request để gửi một request lên thingspeak yêu cầu đọc dữ liệu từ field 2 (thành công hay thất bại) sau đó sử dụng một node function nhằm đọc và giải quyết dữ liệu từ thingspeak như chuyển đổi múi giờ, đổi 0 - 1 thành thành công, thất bại, parse dữ liệu thành kiểu float và trả về để gửi vào widget hiển thị trên website bao gồm 2 mục là thời gian và hoạt động để người dùng có thể cập nhật trạng thái cửa liên tục trên web.



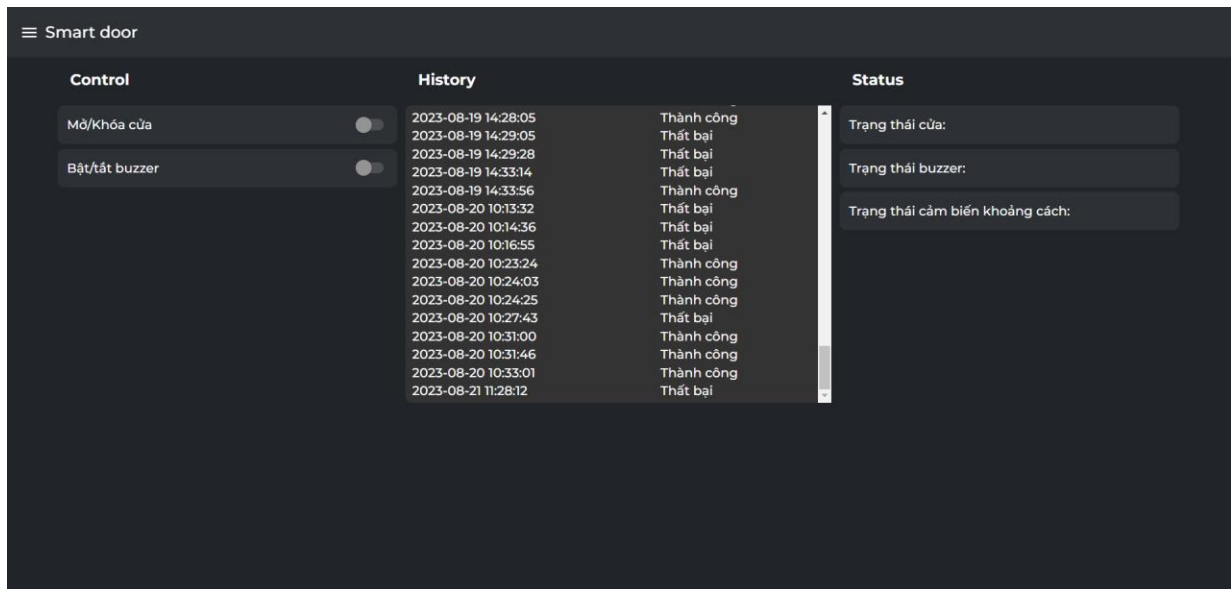
Hình 6: Mô tả flow Cloud

b) Mô tả website:

- Người dùng khi sử dụng sản phẩm khóa thông minh DEADLOCK sẽ được cung cấp một tài khoản (password và tên tài khoản). Người dùng có thể dùng tài khoản này để đăng nhập vào web và điều khiển cửa, theo dõi mọi hoạt động của cửa từ xa. Trang web có các chức năng cơ bản như sau.
- Gồm 3 group chính là: Control, History, Status
- Phần Control: Phần này dùng để điều khiển output từ xa.
Bao gồm 2 button:
 - + Điều khiển mở/khóa cửa từ xa (điều khiển servo từ xa)
 - + Bật/tắt buzzer từ xa.
- Phần History:
 - + Xem lịch sử mở cửa thành công hay thất bại trong widget.
- Phần Status:
 - + Xem trạng thái cửa (OPEN hay CLOSE)
 - + Xem trạng thái buzzer (ON hay OFF)
 - + Xem trạng thái cảm biến khoảng cách (ON hay OFF)

c) Giao diện website

Gồm 3 group chính: Control, History và Status

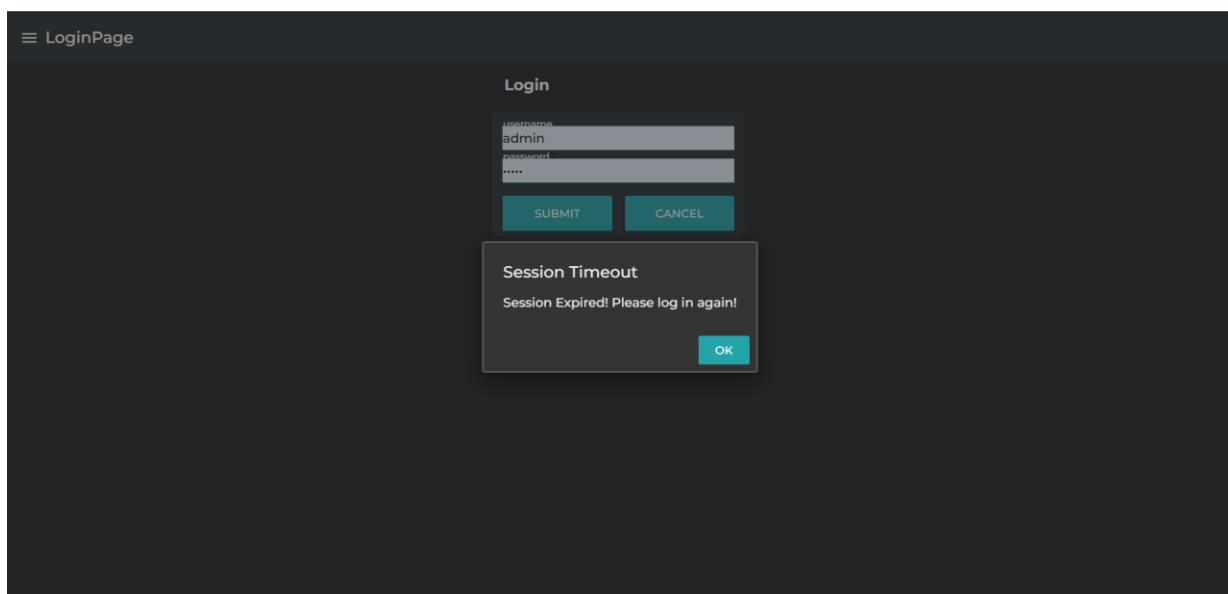


Hình 7: Giao diện web

d) Bảo mật website

Giao diện login:

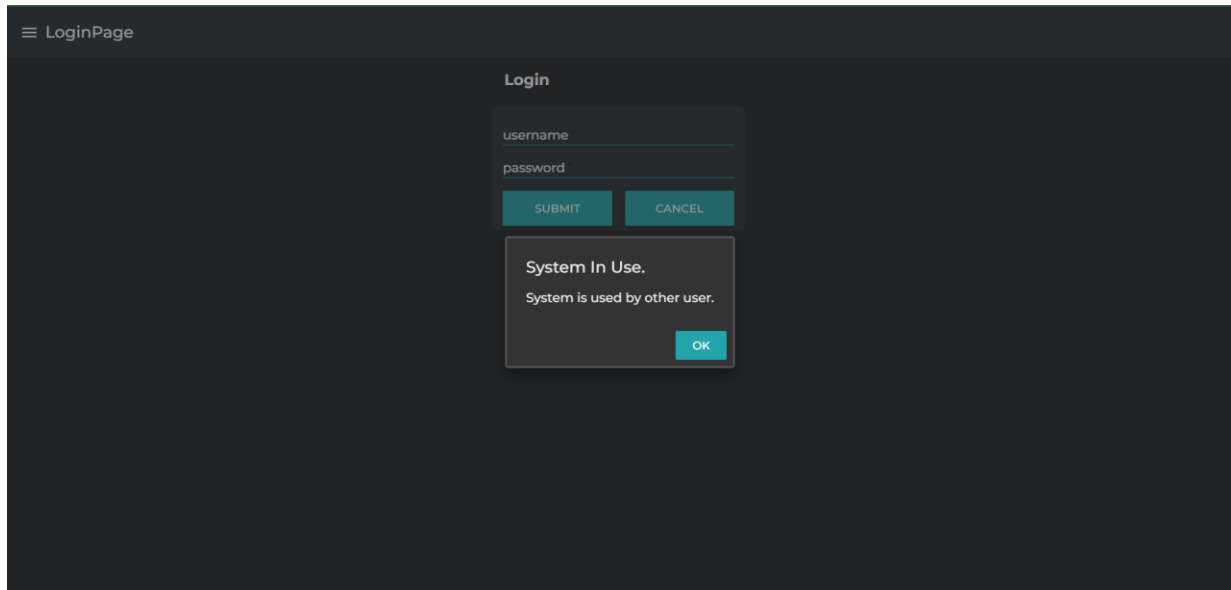
- Người dùng phải nhập mật khẩu và tài khoản khớp với tài khoản và mật khẩu cho trước thì mới vào website được.
- Khi nhập đúng, người dùng sẽ được chuyển tới trang chính để có thể điều khiển và theo dõi hiện trạng của ổ khóa.
- Login có thêm 2 chức năng phụ là session timeout và system is in use
- Đối với chức năng session timeout, khi người dùng đăng nhập vào trang và sử dụng qua thời gian quy định (150s - có thể điều chỉnh trong node function) thì hệ thống sẽ tự động đăng xuất và người dùng sẽ đăng nhập lại để có thể tiếp tục sử dụng.



Hình 8: Giao diện session timeout

- Đối với chức năng system is in use, cho phép mỗi lần chỉ được một người dùng vào trang web vào cùng một thời điểm nhằm tránh xung đột điều khiển trên web lần thiết bị.

Giao diện system is in use:



Hình 9: System in use

e) Các chức năng của website

Website có các chức năng:

- Mở/đóng cửa: Khi bấm nút mở, servo quay 1 góc 180 độ để mở cửa (OUTPUT).
- Bật/tắt buzzer: Khi bấm nút mở, buzzer sẽ mở lên để xuất hiện âm thanh (OUTPUT). Âm thanh sẽ vang lên khi thao tác với các chức năng của ổ khóa (Ví dụ: quét thẻ từ thành công, thất bại; Âm thanh của keypad; Mở đóng cửa ...)
- Xem trạng thái cửa: Cho biết trạng thái cửa đang hoạt động như thế nào (Mở hoặc đóng).
- Xem trạng thái buzzer: Cho biết trạng thái buzzer đang hoạt động như thế nào (Bật hoặc tắt).
- Xem trạng thái cảm biến: Cho biết xung quanh ổ khóa hay cửa có chuyển động xung quanh hay không.

Ngoài ra, website còn có khả năng cho biết lịch sử các lần sử dụng ổ khóa để mở cửa. (Các trạng thái mở cửa như thành công hay thất bại).

f) Đánh giá độ phức tạp của website

Dựa vào mô tả chi tiết về chức năng, giao diện và tính năng bảo mật của website quản lý khóa thông minh DEADLOCK, có thể đánh giá độ phức tạp của website như sau:

- Chức năng và Luồng làm việc:

Website cung cấp nhiều chức năng quản lý khóa thông minh, bao gồm điều khiển cửa, quản lý trạng thái cửa, bật/tắt buzzer và xem lịch sử hoạt động.

Luồng làm việc với nhiều phân chức năng khác nhau như Control, History, và Status, mỗi phân có nhiều tính năng nhỏ.

- Giao diện:

Giao diện trang web được thiết kế với 3 group chính: Control, History và Status.

Mỗi group chứa nhiều widget và chức năng con như mở/đóng cửa, bật/tắt buzzer, xem trạng thái cảm biến, xem lịch sử hoạt động.

Giao diện đòi hỏi cân nhắc về trải nghiệm người dùng để đảm bảo dễ sử dụng và trực quan.

- Bảo mật:

Giao diện đăng nhập yêu cầu xác thực thông qua tên đăng nhập và mật khẩu.

Có chức năng bảo mật "session timeout" để đảm bảo an toàn, người dùng sẽ phải đăng nhập lại sau một khoảng thời gian không sử dụng.

Có chức năng "system is in use" để chỉ cho phép một người dùng sử dụng trang web tại một thời điểm.

Cả hai tính năng này đòi hỏi xử lý logic và quản lý trạng thái phức tạp.

- Tích hợp MQTT và Cloud:

Tích hợp MQTT để cập nhật trạng thái của cửa, buzzer và cảm biến chuyển động từ thiết bị thông qua giao thức MQTT.

Sử dụng HTTP request để gửi và nhận dữ liệu từ Thingspeak, cần xử lý dữ liệu nhận về và hiển thị một cách trực quan cho người dùng.

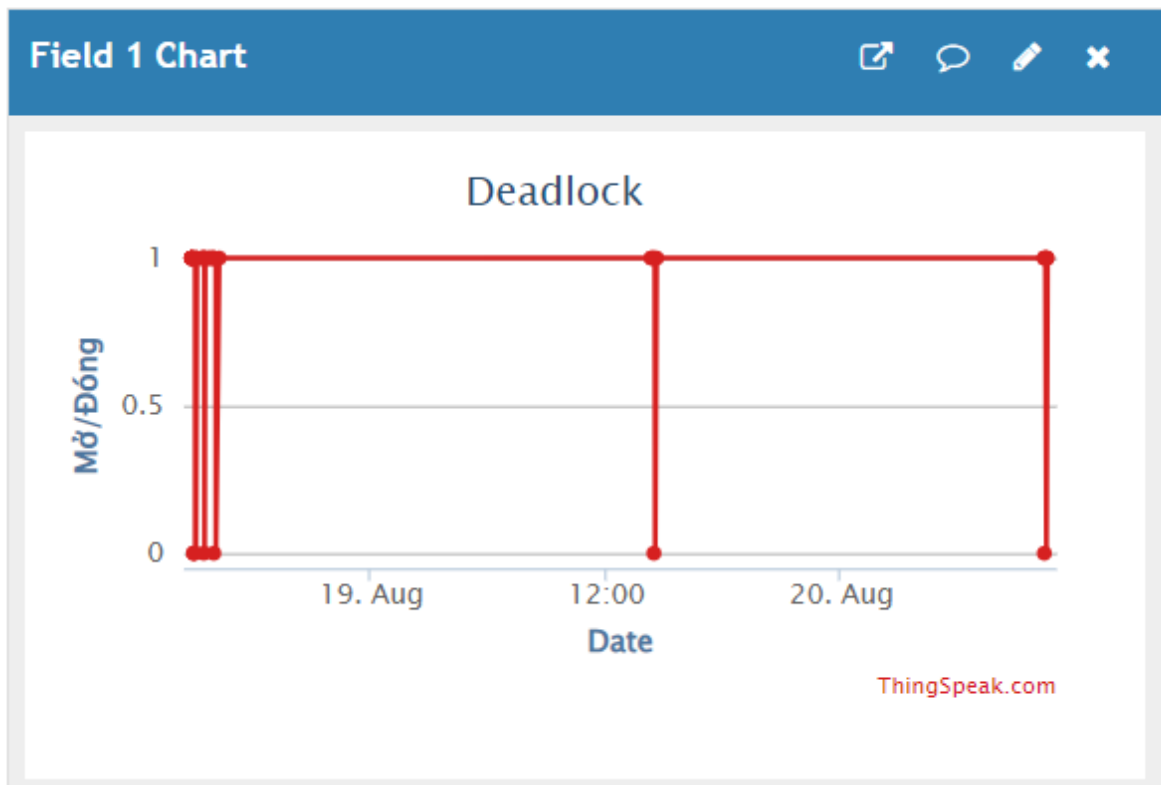
- Các tính năng phụ:

Có các chức năng phụ như hiển thị lịch sử hoạt động, cảnh báo khi cửa đang mở hoặc thiết bị đang được sử dụng bởi người khác.

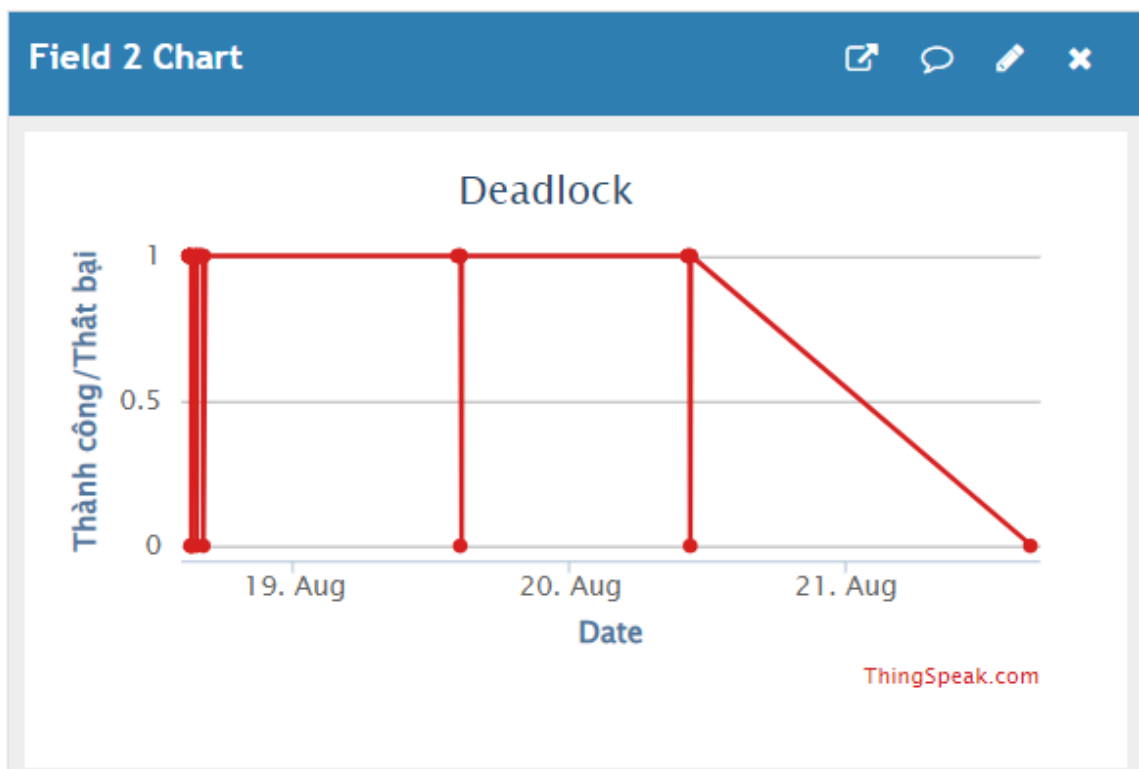
Dựa trên những yếu tố trên, có thể thấy rằng website quản lý khóa thông minh DEADLOCK có độ phức tạp trung bình. Cần phải xử lý nhiều chức năng, tích hợp giao thức MQTT và HTTP, quản lý trạng thái và bảo mật.

B) Mô tả cloud lưu trữ

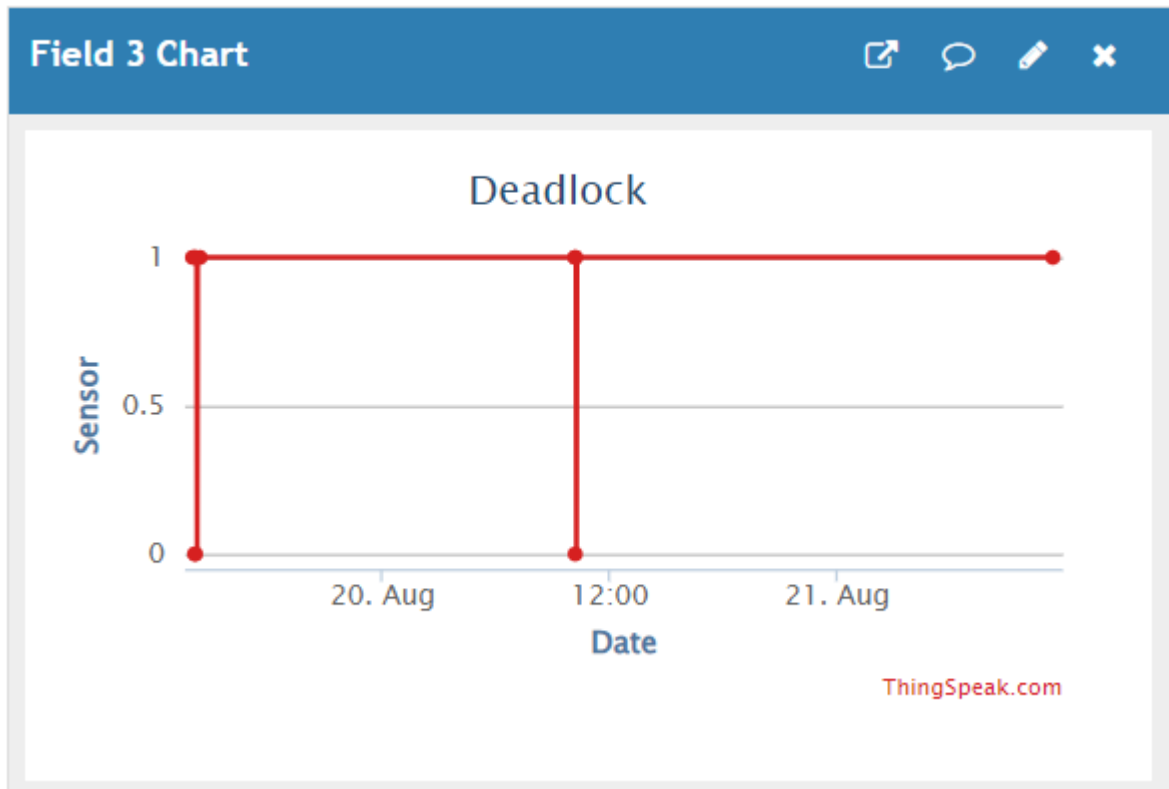
Các biểu đồ được miêu tả theo dạng đường giúp cho người dùng dễ dàng theo dõi trạng thái của smart lock theo thời gian thực



Hình 10: Biểu đồ lưu trữ trạng thái mở hoặc đóng (1 - Mở, 0 - Đóng)



Hình 11: Biểu đồ lưu trữ và biểu diễn trạng thái mở của thành công (1) hay thất bại (0)



Hình 12: Biểu đồ lưu trữ và biểu diễn dữ liệu từ cảm biến khoảng cách (0 - có chuyển động, 1 - không có chuyển động)

Gửi thông tin từ cloud lên node - red:

History	
Thời gian	Trạng thái
2023-08-18 12:13:00	Thất bại
2023-08-18 12:14:55	Thất bại
2023-08-18 12:15:31	Thất bại
2023-08-18 12:15:50	Thành công
2023-08-18 12:17:14	Thất bại
2023-08-18 12:17:30	Thất bại
2023-08-18 12:19:08	Thất bại
2023-08-18 12:19:55	Thất bại
2023-08-18 12:20:13	Thất bại
2023-08-18 12:20:59	Thành công
2023-08-18 12:30:47	Thành công
2023-08-18 12:31:17	Thành công
2023-08-18 12:47:41	Thất bại
2023-08-18 12:48:08	Thất bại
2023-08-18 12:49:27	Thất bại

Hình 13: Gửi thông tin từ Cloud lên Node-red

Gửi các thông tin từ URL field2 (thành công hay thất bại) để hiện lên website các thông tin của hành động mở cửa giúp người dùng dễ dàng quản lý.

C) Gửi thông báo trên điện thoại:


a) Gửi thông báo khi có chuyển động

Khi trời tối hoặc khi đã đóng cửa đi ngủ hoặc ra khỏi nhà, nếu nhận được cảm biến chuyển động báo có chuyển động thì sẽ gửi cho người dùng muốn thông báo về chuyển động xung quanh cửa.

Với value1 là giá trị của cảm biến chuyển động.

Và message gửi về sẽ là “Motion detected!”

Cấu hình IF (Webhooks) và cấu hình then (notifications) như sau:

**Webhooks**

Receive a web request

- Trigger ran, 9:26 PM

Value2

(none)

Value1

Motion_detect

EventName


Motion_detect

Value3

(none)

OccurredAt

August 16, 2023 at 09:26PM

**Notifications**

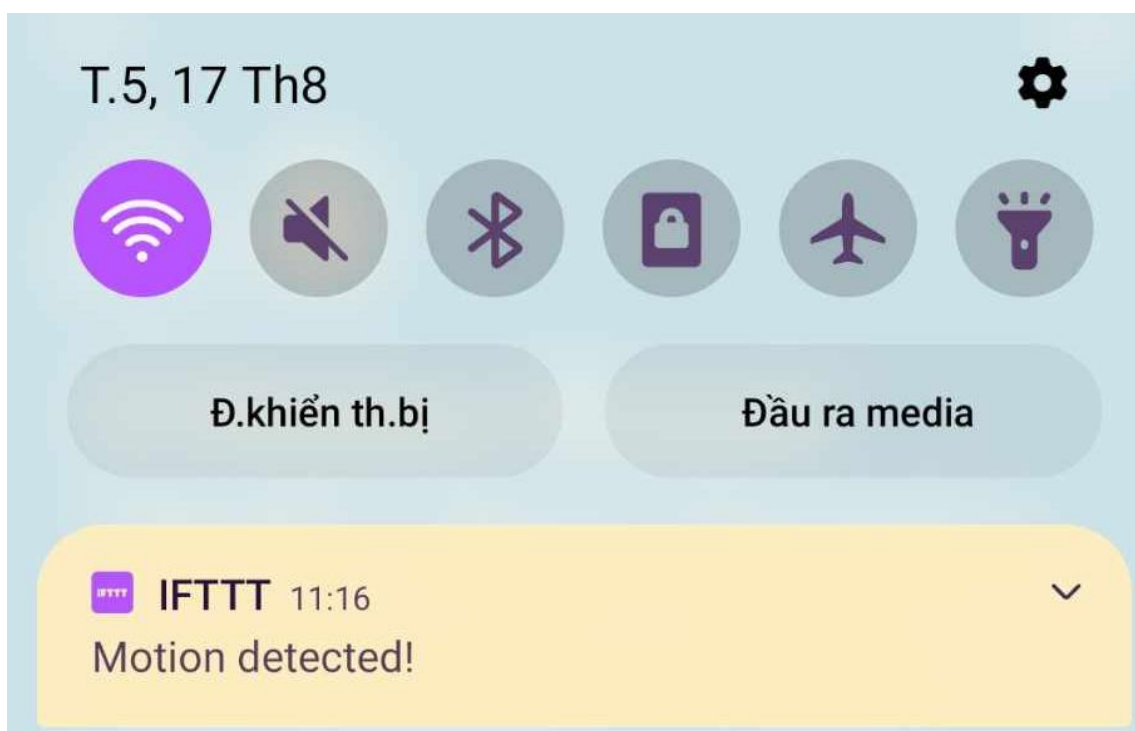
Send a notification from the IFTTT app

- Action ran, 9:26 PM

message

Motion detected!

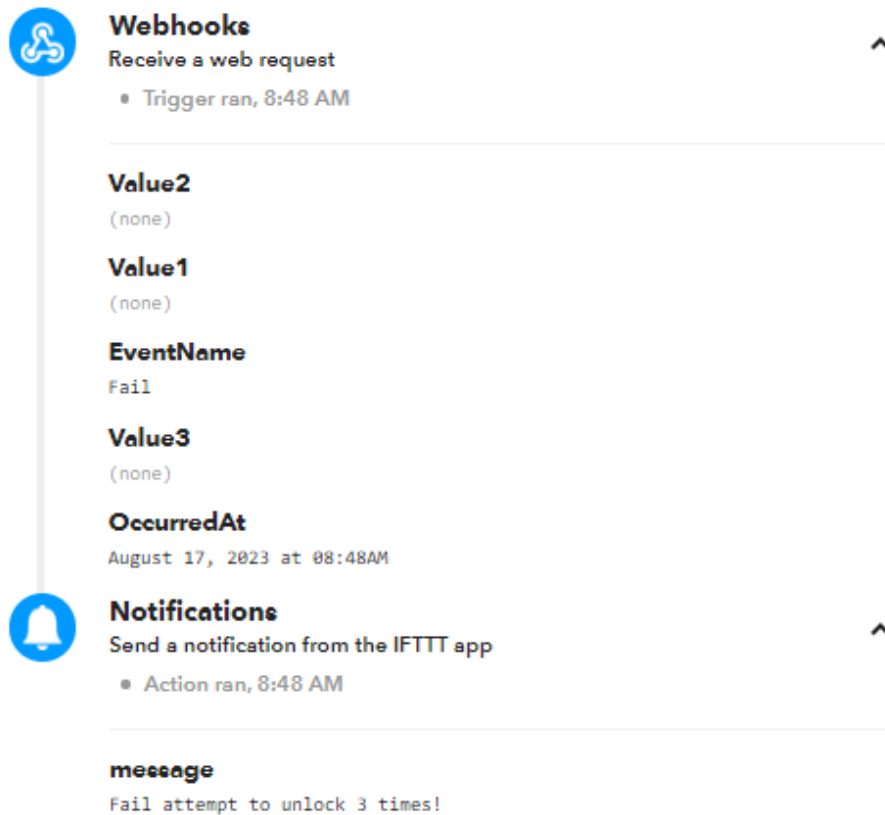
Hình 14.1: Cấu hình thông báo IFTTT



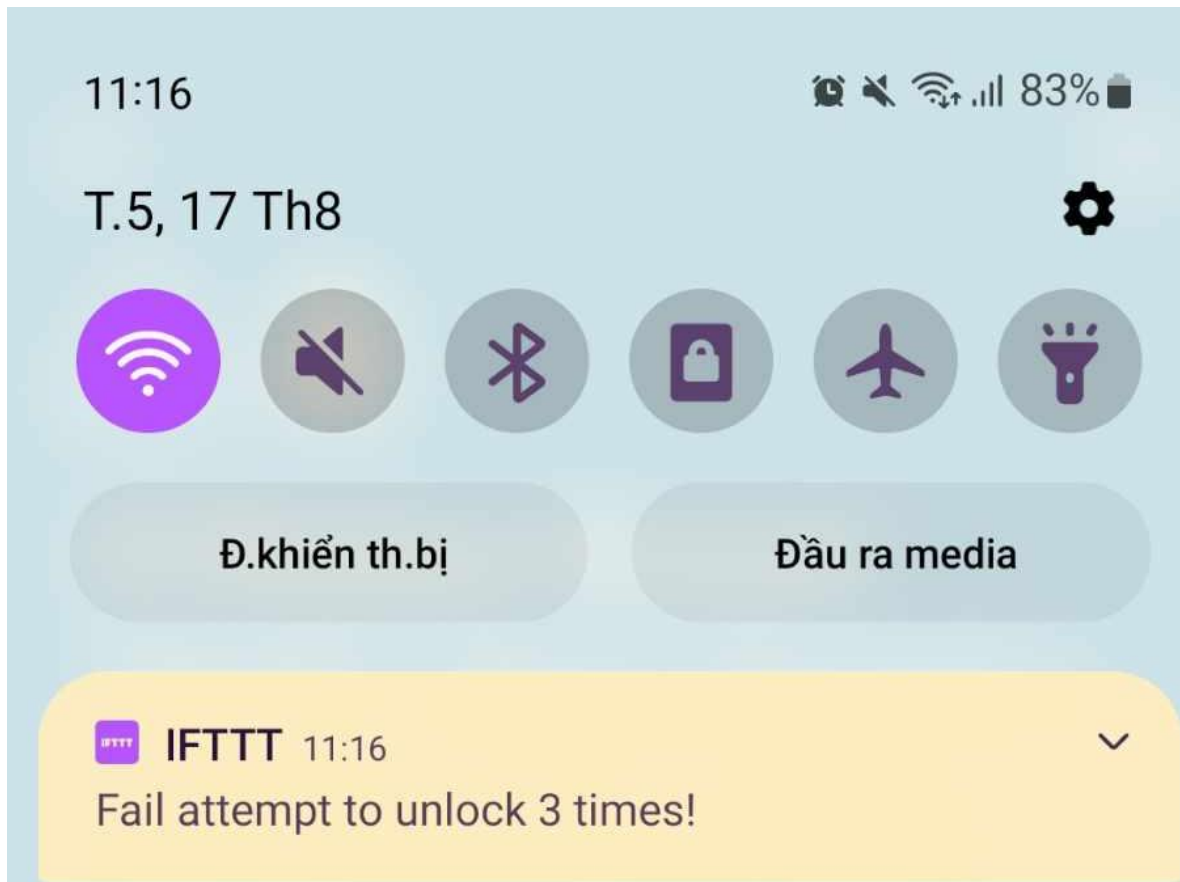
Hình 15: Thông báo IFTTT về motion detected

b) Gửi thông báo khi mở cửa sai 3 lần (sai mật khẩu hoặc thẻ từ quá 3 lần)

- Khi có người cố tình xâm nhập trái phép và sai thẻ từ hoặc mật khẩu quá 3 lần. IFTTT sẽ gửi thông báo về điện thoại rằng có cố gắng xâm nhập 3 lần để người dùng có thể kịp thời ngăn chặn.
- Cấu hình IF (Webhooks) và cấu hình then (notifications) như sau:
- Với message được gửi về điện thoại là “Fail attempt to unlock 3 times”



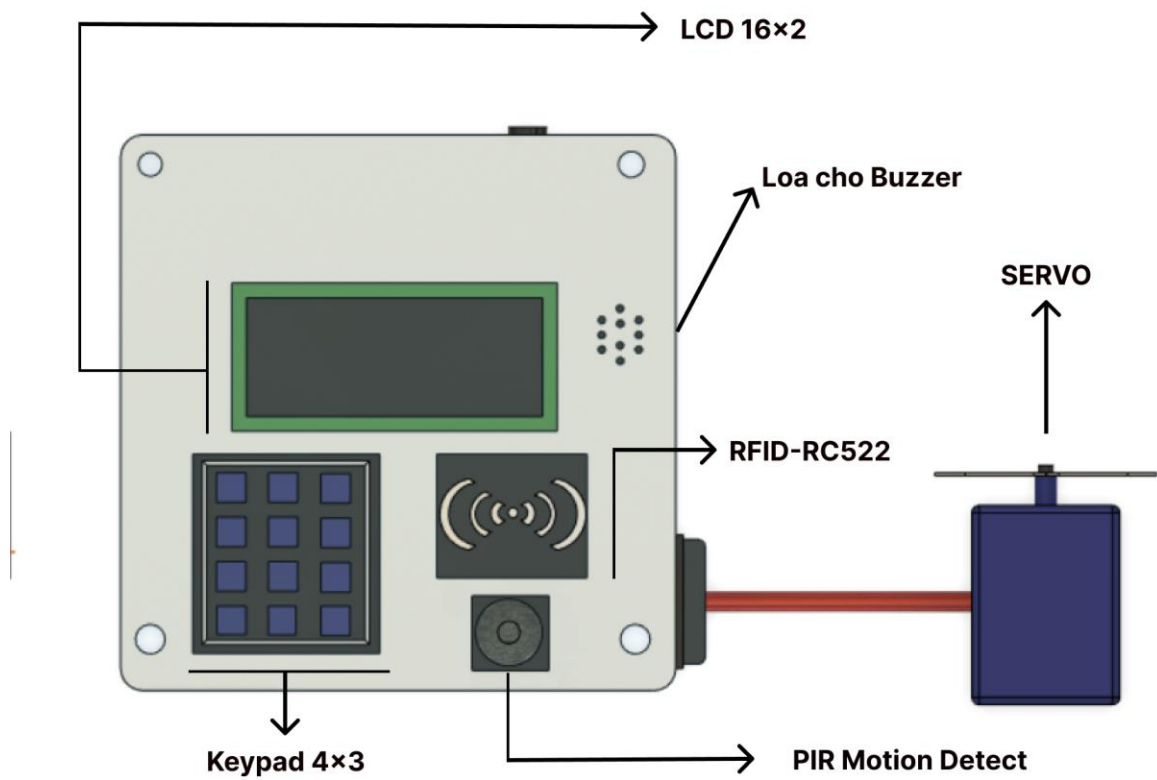
Hình 16.2: Cấu hình thông báo IFTTT



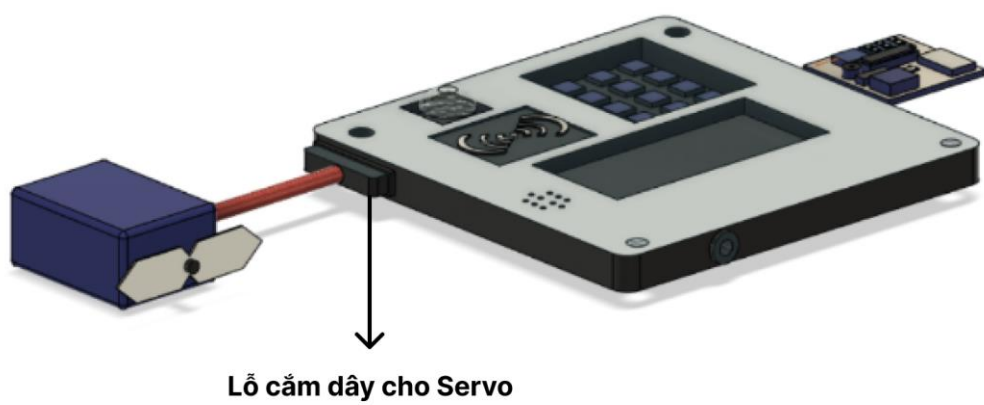
Hình 17: Cấu hình thông báo IFTTT thông báo IFTTT về failed 3 times

IV) Thiết kế 3D của sản phẩm:

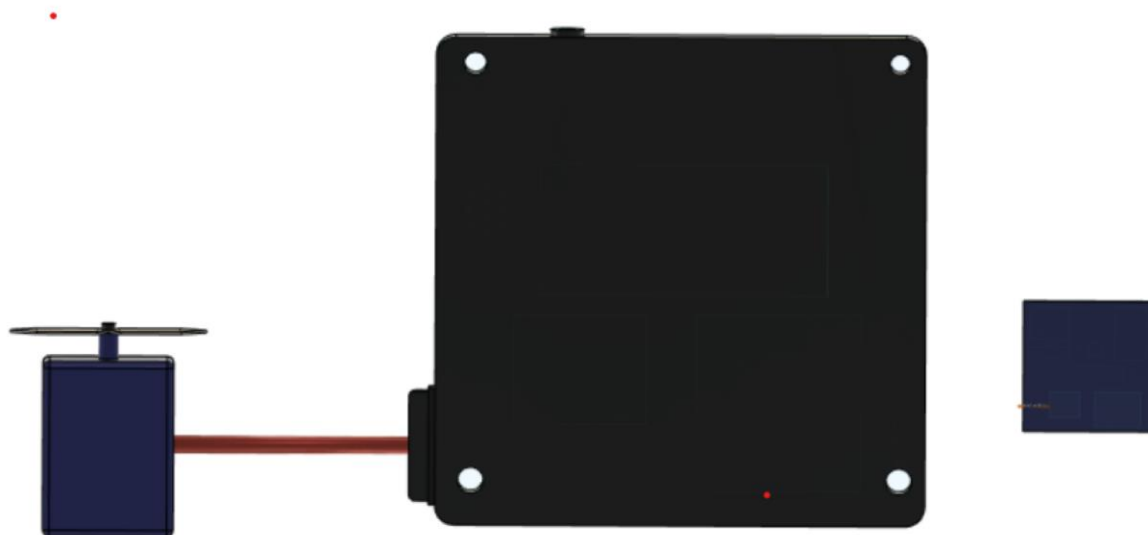
- Giao diện chính diện của sản phẩm:



- Giao diện góc nhìn chéo:



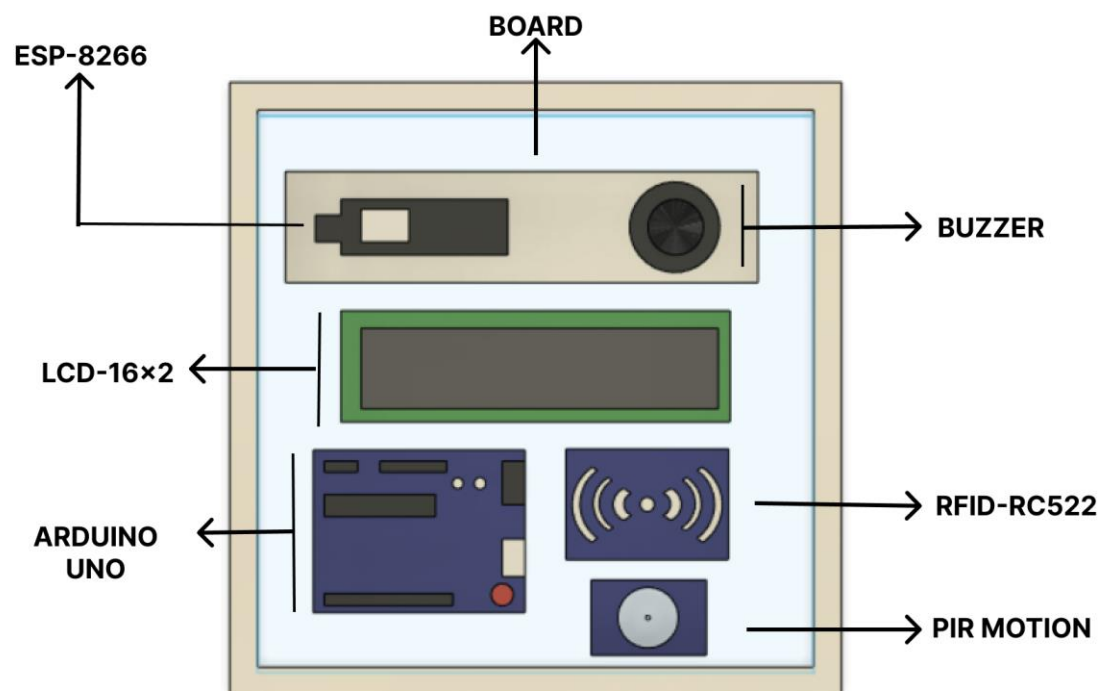
- Giao diện mặt sau:



- Giao diện nhìn từ trên xuống



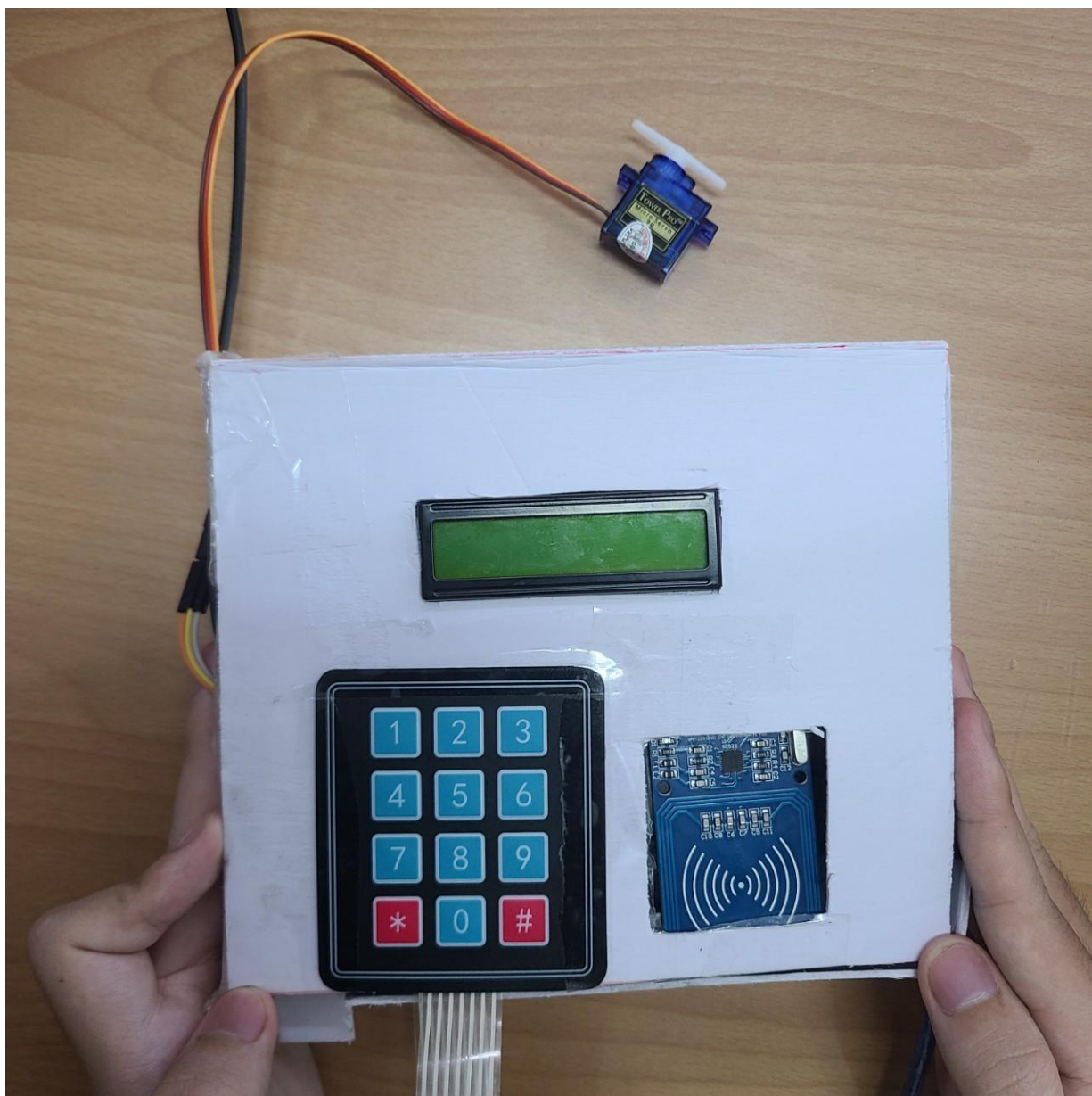
- Giao diện bên trong sản phẩm:



Giao diện phía sau bên trong:



Hình ảnh thực tế sản phẩm:



V) Sơ đồ truyền nhận dữ liệu

- Bản sơ đồ trình bày cách dữ liệu được truyền từ các thiết bị như ESP8266, keypad, RFID đến Node-RED và MQTT Server, sau đó hiển thị trên giao diện web.
- Sự truyền nhận này quan trọng để đảm bảo thông tin đúng đắn và kịp thời đến cho người dùng và hệ thống

