

TIẾP CẬN MÔ HÌNH SMART HOME VỚI NỀN TẢNG IOT E-RA

Lâm Giang Sơn¹

¹Công ty TNHH EoH

Email: son.lg@eoh.io

Keywords:

iot, e-ra, gateway, unit, widget.

TÓM TẮT:

Trong cuộc sống hiện đại hóa hiện nay, công nghệ là một trong những lĩnh vực mang tính chủ chốt, một trong những công nghệ đang được chú ý không thể không kể đến là IoT. Ứng dụng tiêu biểu của công nghệ IoT hiện nay là Smart Home, nhằm hướng đến việc tối ưu hóa không gian sống thông qua việc kết nối và tự động hóa các thiết bị và công nghệ trong nhà.

Sự tiện lợi trong cuộc sống nhờ IoT đã được cải thiện đáng kể khi người dùng có thể điều khiển từ xa các thiết bị trong nhà thông qua điện thoại thông minh hoặc trợ lý ảo. Ngoài ra, người dùng sẽ tiết kiệm năng lượng đáng kể nhờ tối ưu hóa sử dụng điện và giảm lãng phí. Người dùng có thể lập lịch hoạt động cho các thiết bị trong thời gian ngoài giờ cao điểm và nhận thông tin tiêu thụ năng lượng thời gian thực để đưa ra quyết định thông minh.

Mặc dù mang những lợi ích không thể chối cãi, công nghệ Smart Home vẫn có một số khó khăn trong việc tiếp cận. Qua tham đề E-Ra Smart Home sẽ giúp cho việc kết nối dữ liệu cảm biến, các thiết bị điều khiển một cách đơn giản hơn từ đó giúp người dùng dễ dàng quản lý và nâng cao chất lượng sống.

1. Mở đầu

Ngày nay, các tổ chức trong nhiều ngành khác nhau đã và đang sử dụng IoT để hoạt động hiệu quả và nhằm cải thiện việc ra quyết định, giúp gia tăng giá trị của doanh nghiệp. Với sự kết nối của các thiết bị thông minh và dữ liệu, IoT có thể giúp chúng ta giải quyết các vấn đề về an ninh, sức khỏe, môi trường và nhiều lĩnh vực khác. Trong bối cảnh cuộc sống ngày càng hiện đại, IoT đang trở thành một yếu tố không thể thiếu trong cuộc sống của chúng ta.

IoT là viết tắt của Internet of Things, tức là mạng lưới các thiết bị kết nối với nhau thông qua Internet. IoT cho phép các thiết bị như máy tính, điện thoại, máy in, đồ gia dụng, xe hơi và các thiết bị khác có thể kết nối với nhau và trao đổi thông tin một cách tự động và thông minh. Các thiết bị này có thể được điều khiển từ xa và chia sẻ dữ liệu với nhau để tạo ra các ứng dụng thông minh và tiện ích cho người dùng.

Một số ứng dụng IoT tiêu biểu có thể kể đến như Smart Home, Smart Farm, Smart Parking, Smart Health,... Và Smart Home là một trong những lĩnh vực đang được IoT hóa mạnh nhất hiện nay.

Smart Home với các chức năng quản lý điều khiển các thiết bị trong nhà từ đó người dùng có theo dõi môi trường sống của mình một cách dễ dàng và thuận tiện trong thời đại công nghệ IoT hiện nay.

Đầu tiên, thiết bị đo nhiệt độ và độ ẩm thông minh có thể giúp người dùng kiểm soát môi trường sống của mình. Người dùng có thể theo dõi nhiệt độ độ ẩm trong nhà và điều chỉnh thiết bị điều hòa hoặc máy sưởi để tạo ra một môi trường thoải mái và khỏe mạnh cho cả gia đình.

Ngoài ra, các thiết bị thông minh còn có thể được sử dụng để điều khiển các thiết bị trong nhà. Ví dụ, người dùng có thể điều khiển đèn, quạt, máy lạnh và các thiết bị khác thông qua điện thoại hoặc máy tính bảng của mình.

Điều này giúp người dùng tiết kiệm thời gian và năng lượng, đồng thời tạo ra một môi trường sống thông minh và hiện đại.

Với kinh nghiệm nhiều năm trong lĩnh vực IoT cho mảng công nghiệp, công ty EoH đang dần áp dụng IoT vào dân dụng, sau cùng là mở ra cho cộng đồng và giáo dục với nền tảng IoT E-Ra, hướng đến mục tiêu có thể kết nối tất cả các thiết bị, một cách dễ dàng và nhanh chóng nhất.

Tóm lại, Smart Home là xu hướng công nghệ đang phát triển mạnh mẽ nhất hiện nay. Với sự phát triển của các thiết bị cùng với E-Ra, người dùng có thể quản lý môi trường sống của mình một cách dễ dàng và thuận tiện hơn bao giờ hết.

Với tham đề “E-Ra trong lĩnh vực Smart Home” chúng tôi mong muốn cung cấp cho mọi người một cách tổng quan nhất trong việc kết nối thiết bị, có thể theo dõi cũng như điều khiển từ xa một cách tiện lợi nhanh chóng và có thể dễ dàng quản lý hơn.

2. Kết quả nghiên cứu

2.1 Khái niệm Unit / Widget

Unit là một đơn vị dùng để thể hiện cho một địa điểm, đơn vị có thể hình dung là một tòa nhà, có thể là một nơi làm việc nói chung hoặc có thể là 1 trường học.

Widget là những khối hiển thị dữ liệu hoặc là các Khối điều khiển.

Unit và Widget là phần quan trọng trong việc hiển thị dữ liệu cho người dùng. Đối với web, mỗi Unit sẽ có 1 trang Dashboard để hiển thị toàn bộ Widget này.

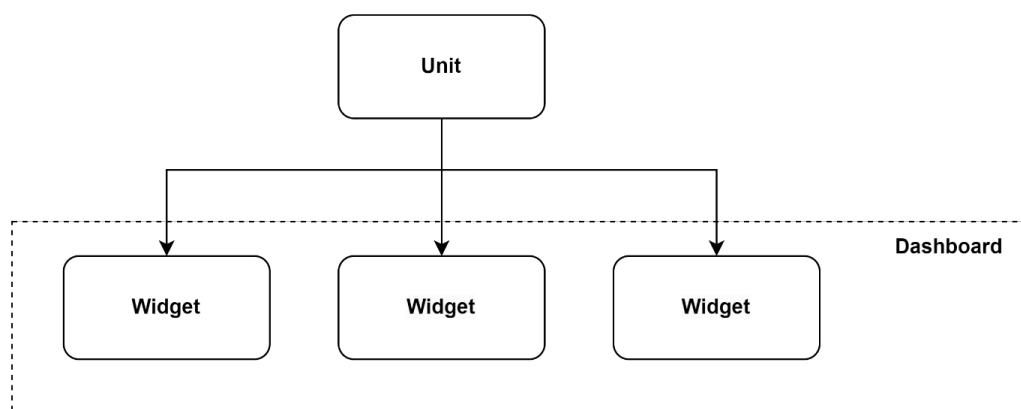
Ví dụ: Unit EoH

Trong dashboard Unit EoH có nhiều Widget như

- Widget hiển thị dữ liệu nhiệt độ, độ ẩm
- Widget hiển thị biểu đồ dữ liệu nhiệt độ, độ ẩm
- Widget nút nhấn để điều khiển thiết bị

Các dữ liệu được gắn vào các Widget được lấy từ Config (hay còn gọi là Datastream), dữ liệu này được lấy Config từ Gateway / Device / Config.

Mô hình Unit và Widget như sau:



2.2 Khái niệm Sub-unit / Device display

Chúng ta có thể dễ dàng quản lý và theo dõi các dữ liệu từ Widget trên web, tuy nhiên để theo dõi toàn bộ Widget trên điện thoại sẽ khó hơn. Vì thế khái niệm Sub-unit được ra đời dùng để phân loại các device vào từng Sub-unit khác nhau và Widget sẽ được đưa vào các device này.

Sub-unit ở đây được hiểu là các phòng, các đơn vị nhỏ trong Unit.

Device display ở đây được hiểu là các device sẽ hiển thị cho người dùng, trong device display sẽ gồm nhiều Widget.

Vì vậy Widget sẽ cần được phân bổ vào một Device display và Sub-unit cụ thể nào đó.

Ví dụ:

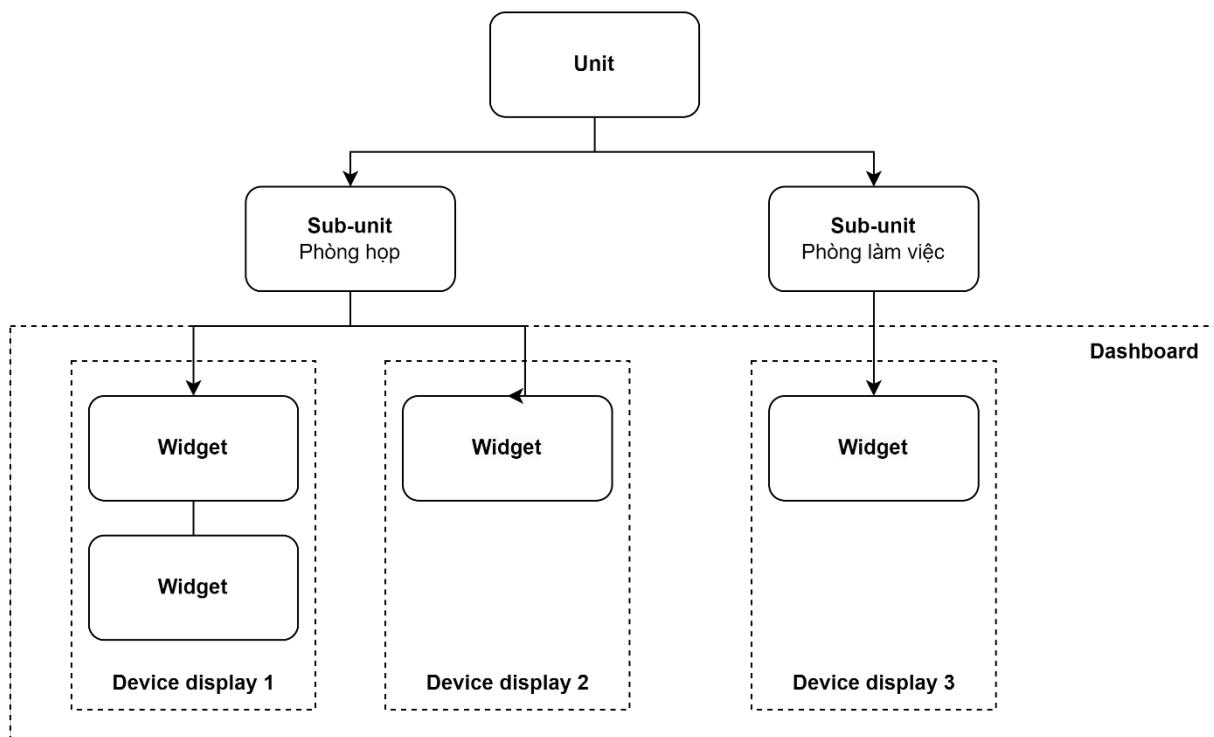
Unit công ty EoH

Sub-unit

- Phòng làm việc
- Phòng họp

Việc này sẽ giúp trong việc quản lý và theo dõi dữ liệu từ thiết bị một cách dễ dàng hơn trên điện thoại.

Mô hình sau khi thêm Sub-unit và Device display như sau:



2.3 Khái niệm Gateway / Device / Config

Gateway:

- Gateway đại diện cho chip mà mình sẽ kết nối trong thực tế.
- Các loại chip hỗ trợ hiện tại là ESP8266, ESP32, STM32, RASPBERRYPI.
- Một Gateway có thể kết nối nhiều Device.

Device:

- Device có thể là các thiết bị cảm biến như: Cảm biến nhiệt độ, độ ẩm, Cảm biến khí gas,...
- Device cũng có thể là các thiết bị điều khiển (qua relay hoặc hồng ngoại) như: Đèn quạt, Máy lạnh,...
- Một Device có thể có nhiều Config, có Config read và Config write.

Config hay còn gọi là Datastream, đại diện cho mỗi giá trị trong từng cảm biến.

Ví dụ:

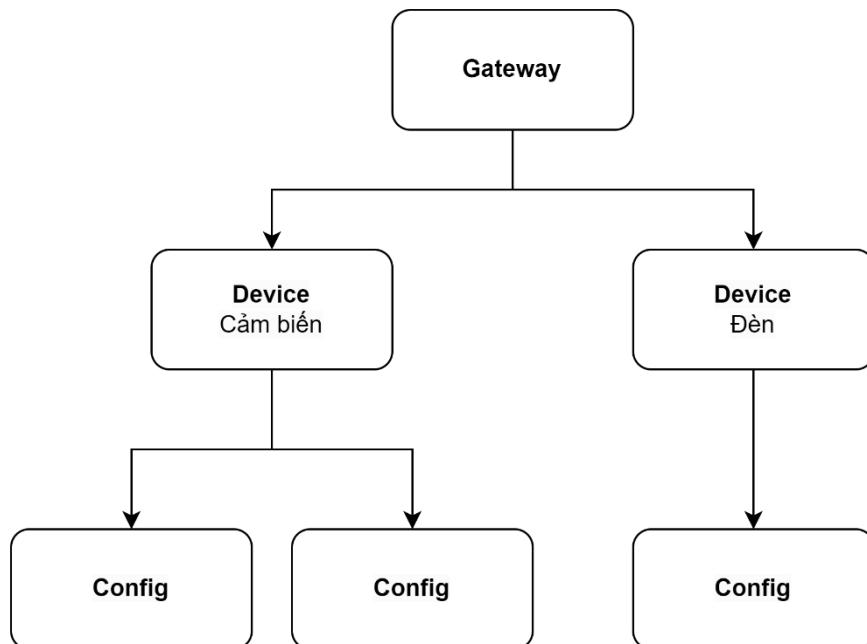
Đọc cảm biến cảm biến nhiệt độ, độ ẩm qua IO Pins (Virtual Pins)

- ➔ Device cảm biến nhiệt độ, độ ẩm
- ➔ Có 2 config là nhiệt độ và độ ẩm

Điều khiển đèn dân dụng thông qua relay

- ➔ Device Đèn
- ➔ Config write cho chân IO nào

Mô hình Gateway / Device / Config như sau:



2.4 Phần cứng

Nền tảng IoT E-Ra hiện tại hỗ trợ nhiều phần cứng đa dạng khác nhau, là những phần cứng đang được sử dụng nhiều trên thị trường để làm gateway trong lĩnh vực IoT với các kết nối như wifi, gsm hoặc bluetooth.

Một số loại board nền tảng IoT E-Ra đang hỗ trợ có thể kể đến như là ESP32, ESP8266, STM32, Raspberry Pi, Module Sim. Các loại board này có khả năng xử lý từ thấp đến cao, dễ dàng cho người dùng mới cũng như các chuyên gia để tiếp cận.

Với các đặc trưng của board là khả năng xử lý nhanh, hỗ trợ kết nối wifi, được thiết kế nhỏ gọn để dễ dàng mang đi, tiêu tốn ít điện năng và nhiều ngoại vi tiện dụng để dễ dàng trong việc mở rộng như là các ngõ io, ngoài ra ngoại vi UART thông qua mạch chuyển UART – MODBUS để kết nối đến các thiết bị dung modbus trong công nghiệp.



ESP32



ESP8266



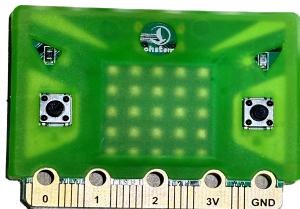
STM32



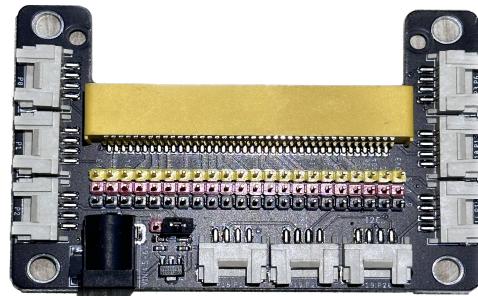
Raspberry Pi

Để tiện lợi trong việc kết nối Device và Gateway, bộ kit được sử dụng trong tham đề này sẽ là Yolo:Bit, từ Ohstem, với board cốt lõi bên trong sẽ là ESP32. Việc kết nối đơn giản nhờ các cổng đã được thiết kế với nguồn và chân tín hiệu bên trong, được kết nối thông qua 1 jack 3 chân.

Gateway trung tâm Yolo:Bit và board ra chân, dễ dàng cho việc kết nối...



Cảm biến nhiệt độ, độ ẩm



Relay



Đèn 12V



Nguồn 12V



2.5 Thực hành

2.5.1 Tạo Unit

Đăng nhập

Chọn mục Dashboard Units, sau đó click Create Unit để khởi tạo 1 Unit mới
Popup hiện lên. Đặt tên Unit > Chọn vị trí > Nhấn Create

The image consists of two vertically stacked screenshots of the E-Ra developer interface.

Screenshot 1: Dashboard Units

- The top navigation bar shows "E-Ra DEVELOPER" and "Dashboard Units".
- A red box highlights the "Dashboard Units" button in the top left of the main content area.
- A red box highlights the "Create unit" button in the top right corner.
- The main content area displays a table with columns "Name", "ID", and "Description". A message "No data" is shown below the table.
- A search bar at the bottom left contains the placeholder "Create Wokwi demo".
- The footer contains links to "Subscription", "Server status", "Documentation", "General Trading Conditions", "Privacy Policy", and copyright information "© 2020 E-Ra v 0.1.643".

Screenshot 2: Create New Unit

- The top navigation bar shows "E-Ra DEVELOPER" and "Dashboard Units".
- A red box highlights the "Create unit" button in the top right corner.
- A modal window titled "Create New Unit" is open. It has fields for "Name" (containing "Trường Lý Tự Trọng") and "Choose location" (containing "38 Đ. Phạm Thận Duật, Phường Thạnh Mỹ Lợi, Quận 2, TP HCM").
- A red box highlights the location map area, which shows a street view of "Đ. Phạm Thận Duật" and "Đ. Phạm Văn Đồng".
- A red box highlights the "Create" button at the bottom right of the modal.
- The footer contains links to "Subscription", "Server status", "Documentation", "General Trading Conditions", "Privacy Policy", and copyright information "© 2020 E-Ra v 0.1.643".

Your subscription: Nâng cao - Khuyến mãi
End date: 14/07/2023

Dashboard Units

Create unit

All gateways Manage unit

What are you looking for ?

Name	ID	Description
Trường Lý Tự Trọng	3834	

Create Wokwi demo

< 1 >

Subscription Server status Documentation General Trading Conditions Privacy Policy © 2020 E-Ra v 0.1.643

Help

2.5.2 Tạo và kích hoạt gateway đến E-Ra

Chọn mục All gateways, sau đó click New gateway để tạo gateway mới. Chờ gateway online kích hoạt thành công

Your subscription: Nâng cao - Khuyến mãi
End date: 14/07/2023

All gateways

New gateway

All gateways Manage unit

What are you looking for ?

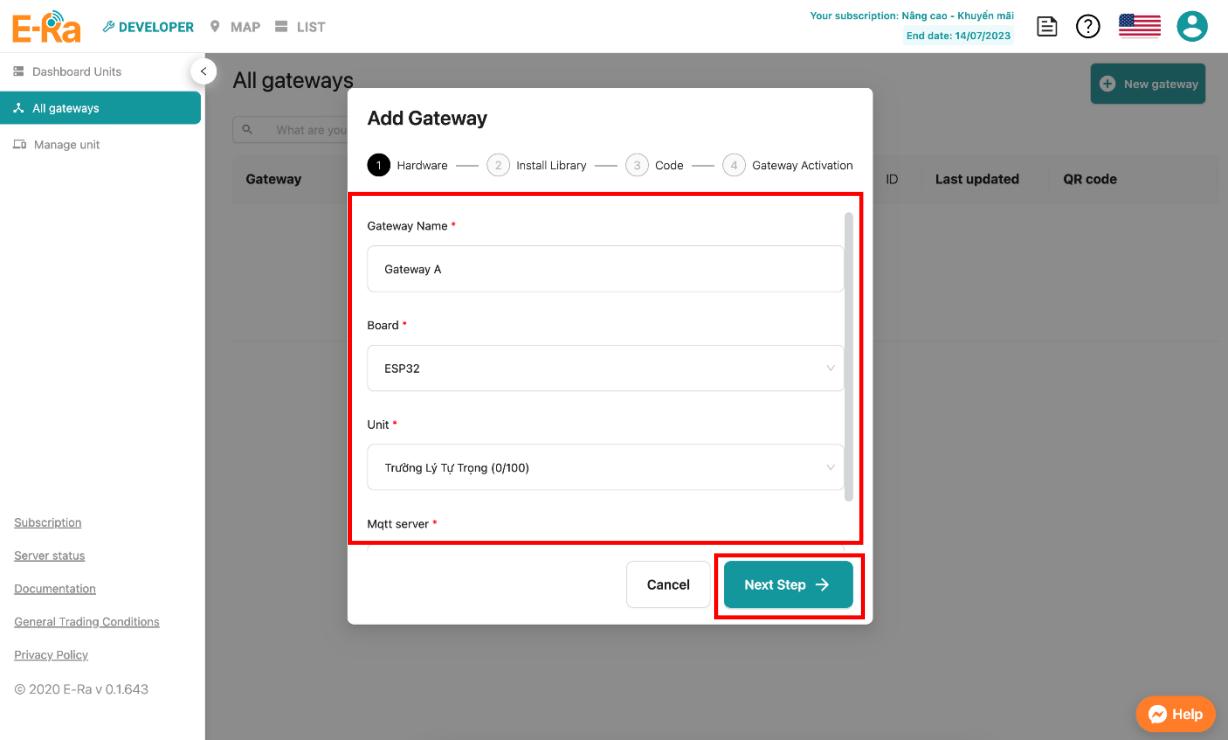
Gateway	Status	Unit	Type	Mqtt server	ID	Last updated	QR code
							No data

Subscription Server status Documentation General Trading Conditions Privacy Policy © 2020 E-Ra v 0.1.643

Help

Bước 1: Khai báo thông tin Gateway

- Đặt tên gateway
- Chọn loại board, vì gateway sử dụng chip là ESP32 nên chọn ESP32.
- Chọn unit đã tạo từ bước trước đó
- Mqtt server mặc định là Việt Nam (các option khác của gói nâng cao)
- Nhấn Next Step



Bước 2: Hướng dẫn cài đặt Arduino IDE

Ở bước này sẽ hướng dẫn cài đặt các phần mềm và thư viện cần thiết.

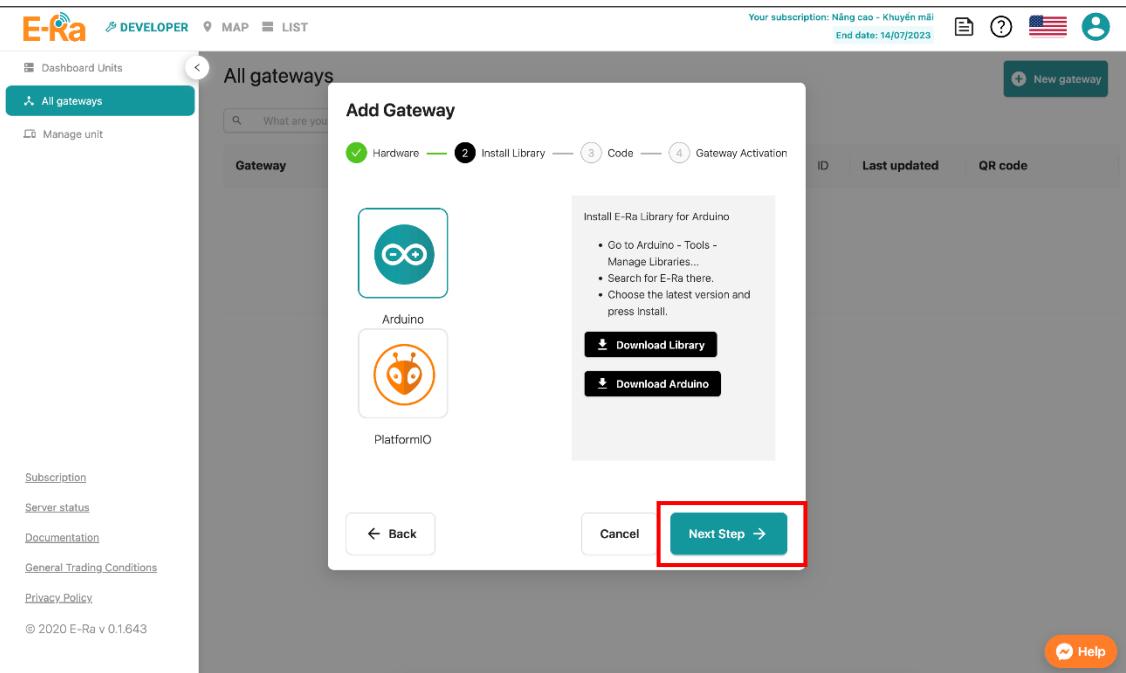
Cụ thể trong bài thực hành này sẽ gồm:

- Phần mềm Arduino IDE để biên dịch và nạp code cho gateway
- Thư viện E-Ra Eoh để kết nối đến E-Ra
- Thư viện DHT20 cho cảm biến nhiệt độ, độ ẩm

Link hướng dẫn cài đặt:

<https://era-open-iot-platform.gitbook.io/documentation/E-Ra-workshop/topic-3-xay-dung-ngoi-nha-thong-minh-voi-bo-kit-ohstem>

Sau khi cài đặt thành công, chúng ta bấm next step.

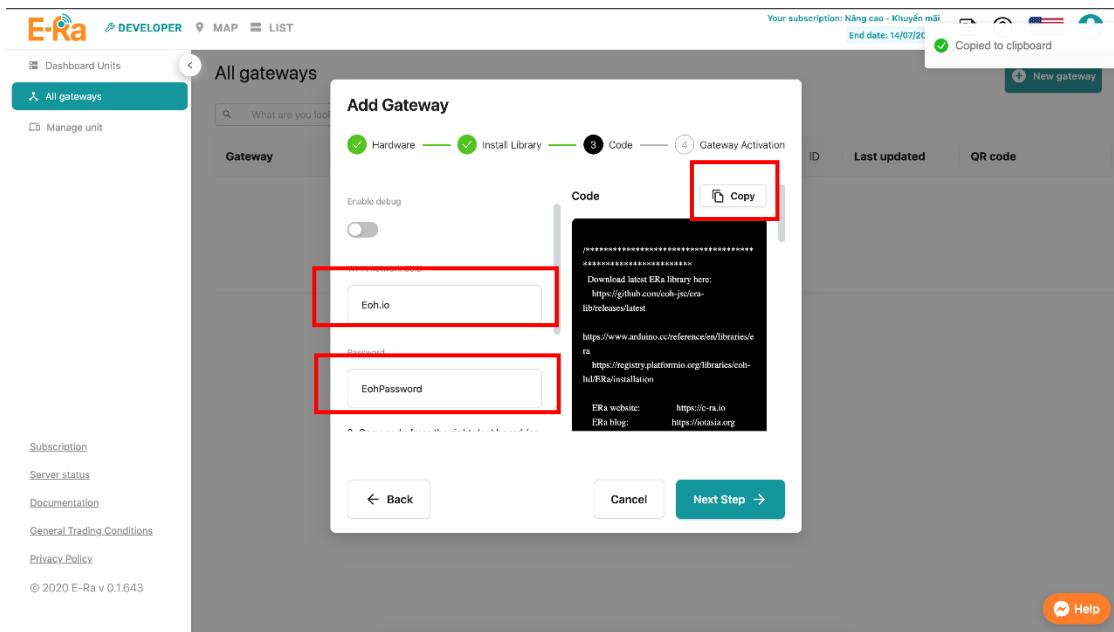


Bước 3: Cập nhật thông tin wifi và password và update source code

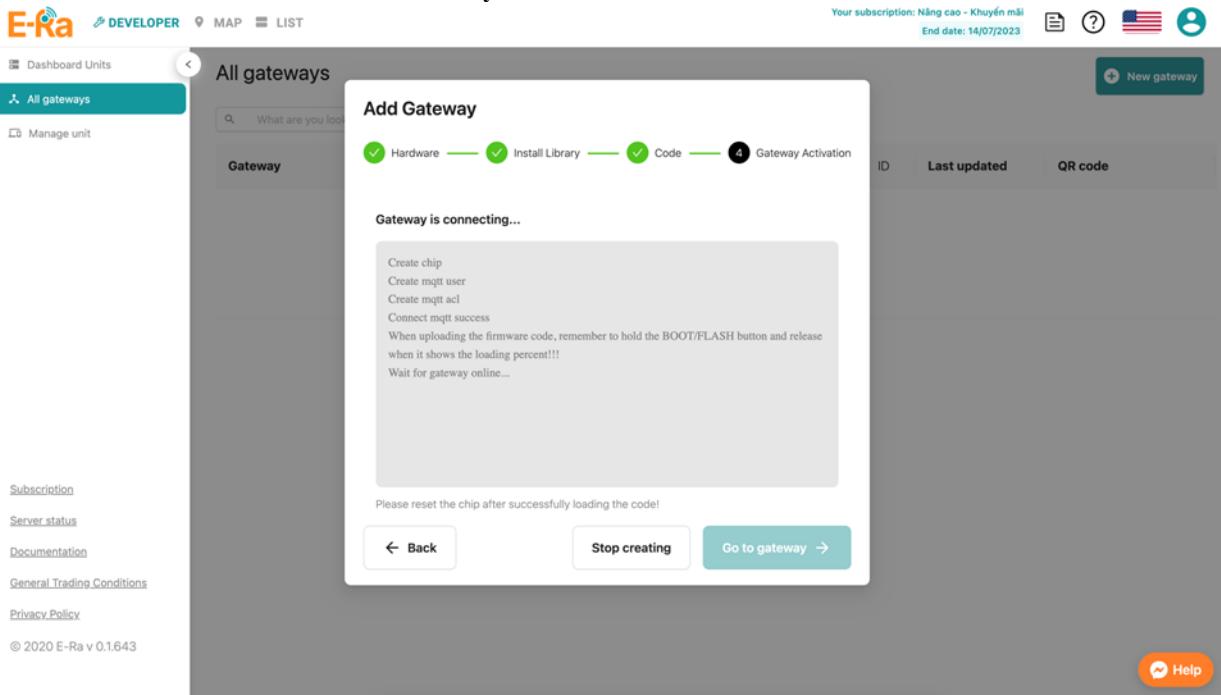
Điền thông tin wifi / password.

Sau đó bấm nút Copy source code để sao chép source code này. Nhấn Next Step.

Phần thông tin wifi / password sau khi điền sẽ được cập nhật vào phần source code bên phải, phần code này sẽ được nạp xuống gateway, gateway sẽ sử dụng wifi / password này để kết nối mạng wifi.



Bước 4: Tiến trình theo dõi kết nối Gateway với E-Ra



Source code đã copy

```
*****
```

Download latest ERA library here:

<https://github.com/eoh-jsc/era-lib/releases/latest>

<https://www.arduino.cc/reference/en/libraries/era>

<https://registry.platformio.org/libraries/eoh-ltd/ERA/installation>

ERA website: <https://E-Ra.io>

ERA blog: <https://iotasia.org>

ERA forum: <https://forum.eoh.io>

Follow us: <https://www.fb.com/EoHPlatform>

```
*****
```

```
// Enable debug console
```

```
// Set CORE_DEBUG_LEVEL = 3 first
```

```
// #define ERA_DEBUG
```

```
#define DEFAULT_MQTT_HOST "mqtt1.eoh.io"
```

```
// You should get Auth Token in the ERA App or ERA Dashboard
```

```
#define ERA_AUTH_TOKEN "62746a4b-3dd9-4fad-bdb5-710a0dcef123"
```

```
#include <Arduino.h>
```

```
#include <ERa.hpp>
#include <ERa/ERaTimer.hpp>

const char ssid[] = "Eoh.io";
const char pass[] = "EohPassword";

ERaTimer timer;

/* This function print uptime every second */
void timerEvent() {
    ERA_LOG("Timer", "Uptime: %d", ERaMillis() / 1000L);
}

void setup() {
    /* Setup debug console */
    Serial.begin(115200);

    ERa.begin(ssid, pass);

    /* Setup timer called function every second */
    timer.setInterval(1000L, timerEvent);
}

void loop() {
    ERa.run();
    timer.run();
}
```

Cảm biến nhiệt độ, độ ẩm được sử dụng là DHT20, để đọc được cảm biến này ta sẽ sử dụng thư viện là DHT20.

Source code cần thêm

```
#include <DHT20.h>
DHT20 dht;

dht.begin();

void readDHT() {
    if (dht.isMeasuring()) {
        return;
    }
    if (dht.readData() < 0) {
        return;
    }
    if (dht.convert() == DHT20_OK) {
        ERa.virtualWrite(V0, dht.getHumidity());
        ERa.virtualWrite(V1, dht.getTemperature());
    }
    dht.requestData();
}
```

Source code sau cùng sẽ là

```
*****
```

Download latest ERA library here:

<https://github.com/eoh-jsc/era-lib/releases/latest>

<https://www.arduino.cc/reference/en/libraries/era>

<https://registry.platformio.org/libraries/eoh-ltd/ERA/installation>

ERA website: <https://E-Ra.io>

ERA blog: <https://iotasia.org>

ERA forum: <https://forum.eoh.io>

Follow us: <https://www.fb.com/EoHPlatform>

```
*****
```

```
// Enable debug console  
// Set CORE_DEBUG_LEVEL = 3 first  
// #define ERA_DEBUG
```

```
#define DEFAULT_MQTT_HOST "mqtt1.eoh.io"
```

```
// You should get Auth Token in the ERA App or ERA Dashboard
```

```
#define ERA_AUTH_TOKEN "62746a4b-3dd9-4fad-bdb5-710a0dcef123"
```

```
#include <Arduino.h>  
#include <ERa.hpp>  
#include <ERa/ERaTimer.hpp>
```

```
#include <DHT20.h>
```

```
DHT20 dht;
```

```
const char ssid[] = "Eoh.io";  
const char pass[] = "EohPassword";
```

```
ERaTimer timer;
```

```
void readDHT() {  
    if (dht.isMeasuring()) {  
        return;  
    }  
    if (dht.readData() < 0) {  
        return;  
    }  
    if (dht.convert() == DHT20_OK) {  
        ERa.virtualWrite(V0, dht.getHumidity());  
        ERa.virtualWrite(V1, dht.getTemperature());  
    }  
    dht.requestData();  
}
```

```
/* This function print uptime every second */
```

```
void timerEvent() {
    readDHT();
    ERA_LOG("Timer", "Uptime: %d", ERaMillis() / 1000L);
}

void setup() {
    /* Setup debug console */
    Serial.begin(115200);

    ERa.begin(ssid, pass);
    dht.begin();

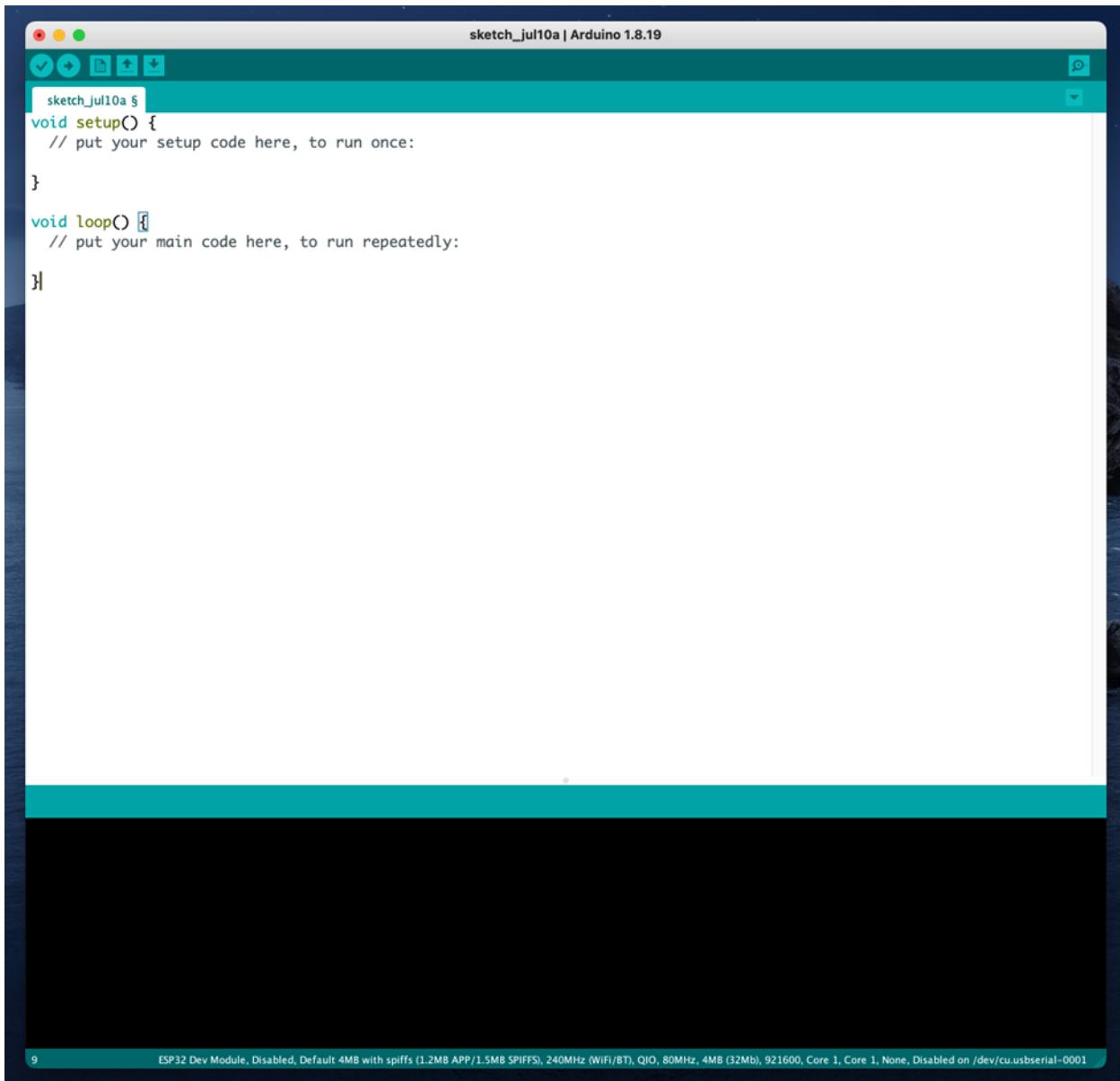
    /* Setup timer called function every second */
    timer.setInterval(1000L, timerEvent);
}

void loop() {
    ERa.run();
    timer.run();
}
```

Tiến hành nạp code để kích hoạt Gateway kết nối đến E-Ra

Mở phần mềm Arduino IDE

Chúng ta sẽ xóa đoạn code mặc định ví dụ của Arduino cung cấp và thay thế bằng đoạn code vừa được Copy và thêm phần đọc cảm biến DHT20 (Source code sau cùng)



The screenshot shows the Arduino IDE interface with the title bar "sketch_jul10a | Arduino 1.8.19". The code editor contains the following code:

```
sketch_jul10a §
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

In the status bar at the bottom, it says "9 ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WIFI/BT), QIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None, Disabled on /dev/cu.usbserial-0001".

Sau khi copy đoạn code vào

```
sketch_jul10a | Arduino 1.8.19

sketch_jul10a §

***** Download latest ERA library here:
https://github.com/eoh-jsc/era-lib/releases/latest
https://www.arduino.cc/reference/en/libraries/era
https://registry.platformio.org/libraries/eoh-ltd/ERA/installation

ERA website: https://e-ra.io
ERA blog: https://iotasia.org
ERA forum: https://forum.eoh.io
Follow us: https://www.fb.com/EoHPlatform
***** /



// Enable debug console
// Set CORE_DEBUG_LEVEL = 3 first
// #define ERA_DEBUG

#define DEFAULT_MQTT_HOST "mqtt1.eoh.io"

// You should get Auth Token in the ERA App or ERA Dashboard
#define ERA_AUTH_TOKEN "62746a4b-3dd9-4fad-bdb5-710a0dcefe0d"

#include <Arduino.h>
#include <ERA.hpp>
#include <ERA/ERaTimer.hpp>

const char ssid[] = "Eoh.io";
const char pass[] = "EohPassword";

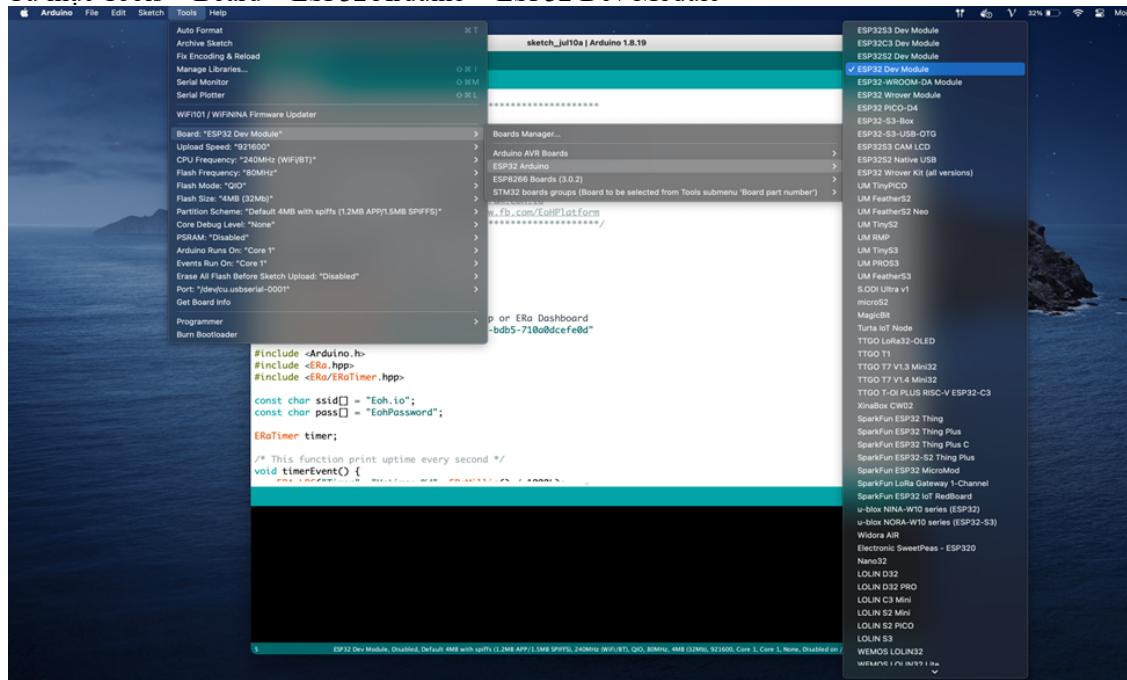
ERaTimer timer;

/* This function print uptime every second */
void timerEvent() {
    Serial.println("Uptime: " + String((millis() - startUpTime) / 1000));
}

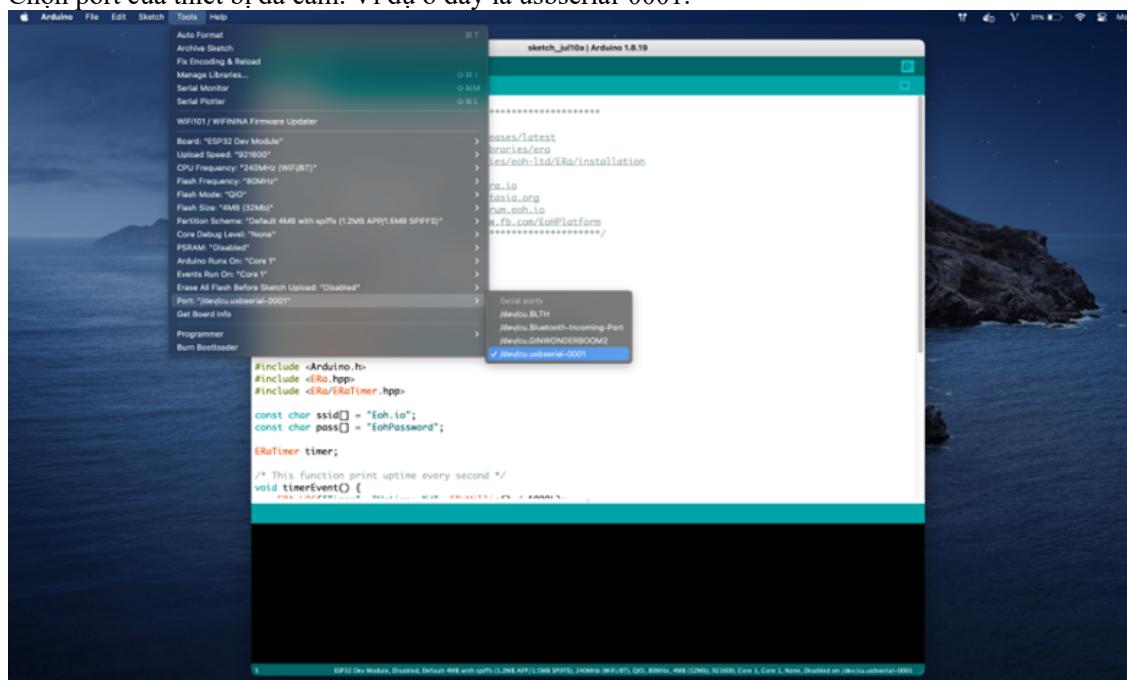
46          ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None, Disabled on /dev/cu.usbserial-0001
```

Chọn loại board Esp32 Dev Module

Tùy mục Tools > Board > ESP32 Arduino > ESP32 Dev Module

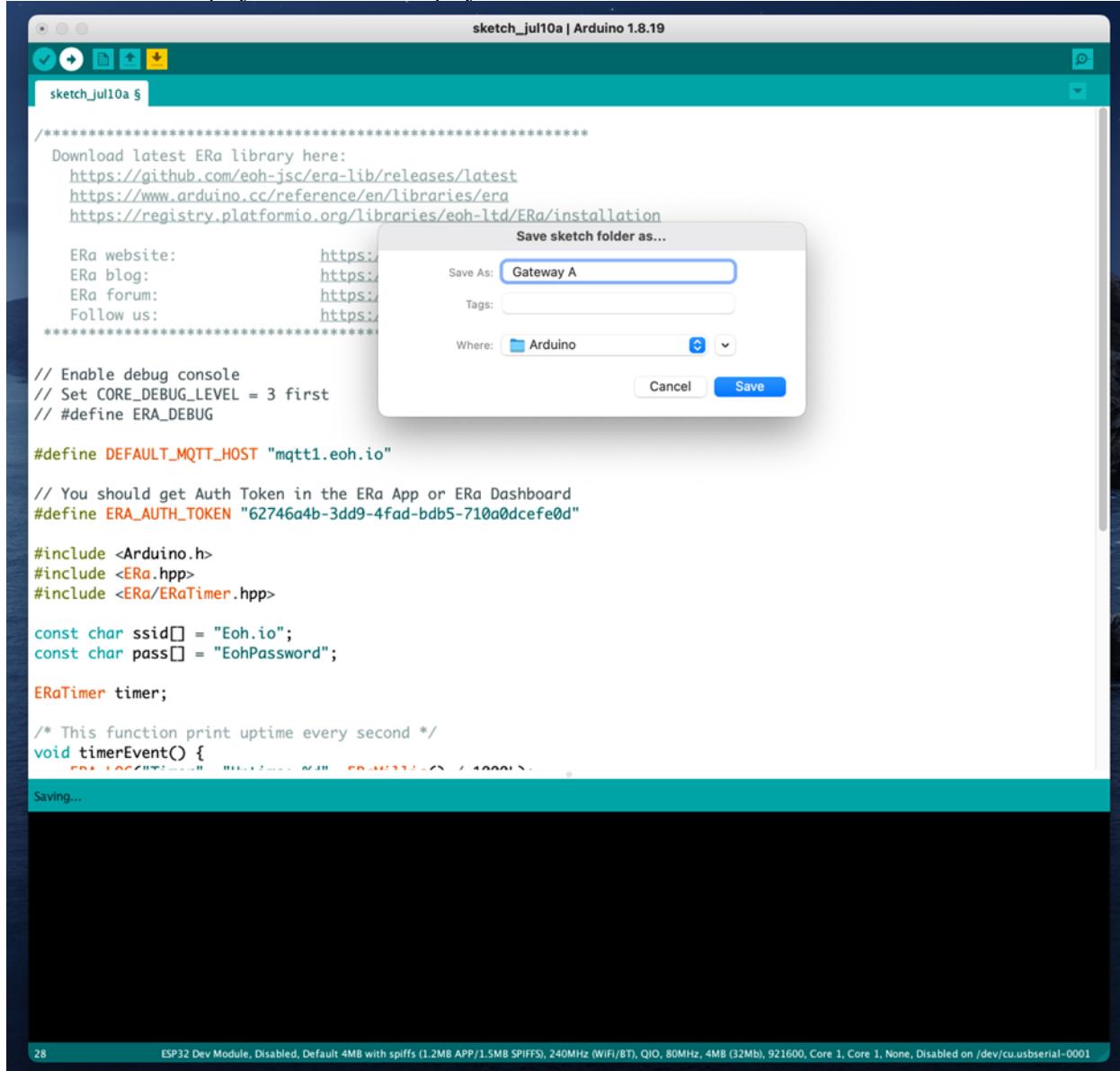


Cắm dây usb kết nối từ gateway vào máy tính hoặc laptop qua cổng usb.
Chọn port của thiết bị đã cắm. Ví dụ ở đây là usbserial-0001.



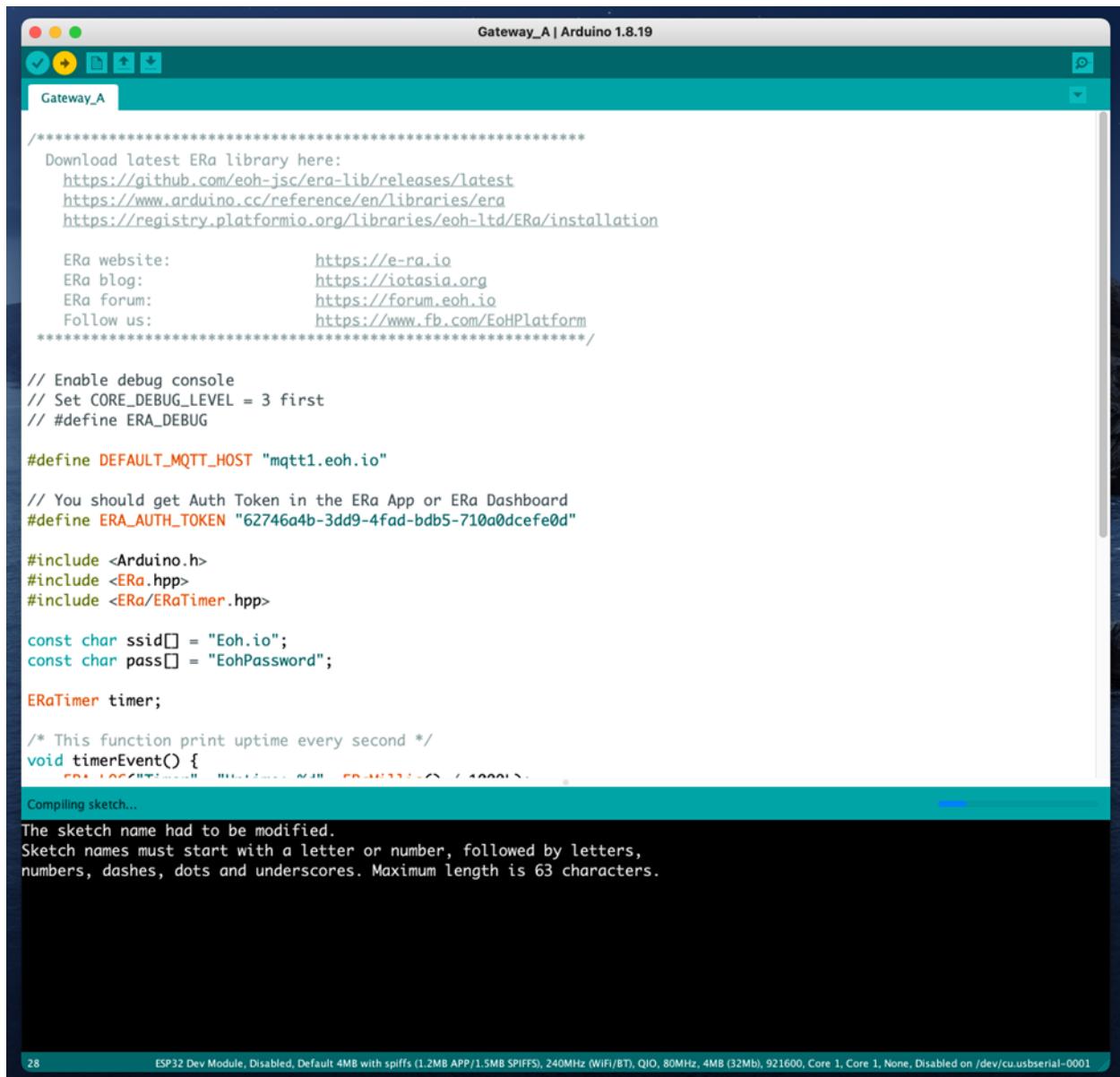
Nhấn Upload

Phần mềm hỏi save project thì đặt tên cho project và nhấn Save.



Việc upload sẽ mất thời gian tầm 2 đến 3 phút.

Phần mềm sẽ biên dịch (compiling) đoạn code sau đó sẽ đến bước nạp code (upload).



The screenshot shows the Arduino IDE interface with the sketch named "Gateway_A". The code is as follows:

```
Gateway_A | Arduino 1.8.19

Gateway_A

/*
Download latest ERA library here:
https://github.com/eoh-jsc/era-lib/releases/latest
https://www.arduino.cc/reference/en/libraries/era
https://registry.platformio.org/libraries/eoh-ltd/ERa/installation

ERA website: https://e-ra.io
ERA blog: https://iotasia.org
ERA forum: https://forum.eoh.io
Follow us: https://www.fb.com/EoHPlatform
*/

// Enable debug console
// Set CORE_DEBUG_LEVEL = 3 first
// #define ERA_DEBUG

#define DEFAULT_MQTT_HOST "mqtt1.eoh.io"

// You should get Auth Token in the ERA App or ERA Dashboard
#define ERA_AUTH_TOKEN "62746a4b-3dd9-4fad-bdb5-710a0dcef0d"

#include <Arduino.h>
#include <ERa.hpp>
#include <ERa/ERaTimer.hpp>

const char ssid[] = "Eoh.io";
const char pass[] = "EohPassword";

ERaTimer timer;

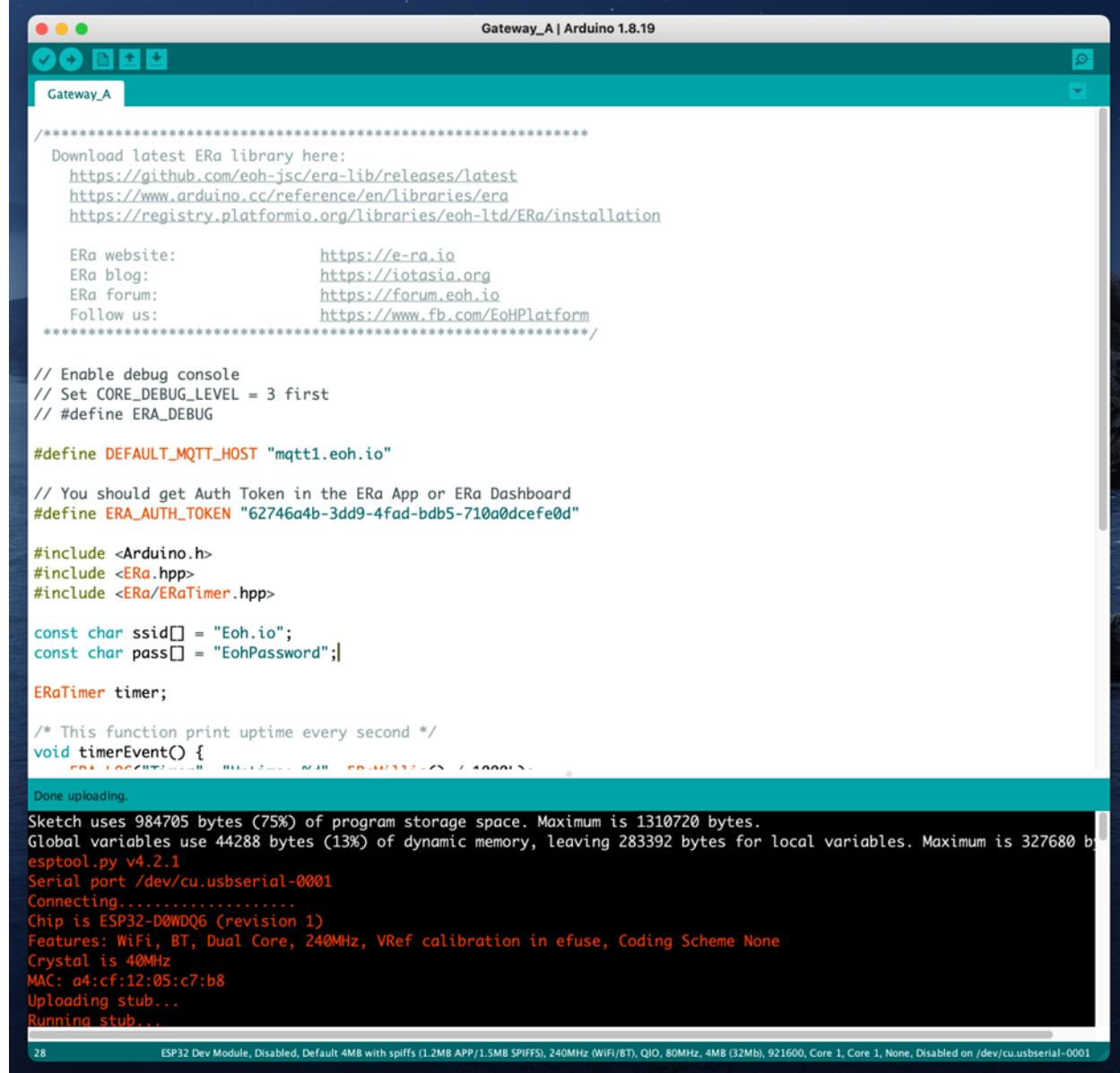
/* This function print uptime every second */
void timerEvent() {
    // ...
}

Compiling sketch...
The sketch name had to be modified.
Sketch names must start with a letter or number, followed by letters,
numbers, dashes, dots and underscores. Maximum length is 63 characters.

28 ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WIFI/BT), QIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None, Disabled on /dev/cu.usbserial-0001
```

Với 1 số board việc nạp code sẽ cần đợi hiển thị Connecting thì nhấn giữ nút Boot trên ESP32 để đưa chip vào chế độ nạp code.

Với board sử dụng trong bài viết này thì không.



```
Gateway_A | Arduino 1.8.19

Gateway_A

/********************* Download latest ERA library here: *****/
https://github.com/eoh-jsc/era-lib/releases/latest
https://www.arduino.cc/reference/en/libraries/era
https://registry.platformio.org/libraries/eoh-ltd/ERa/installation

ERa website: https://e-ra.io
ERa blog: https://iotasia.org
ERa forum: https://forum.eoh.io
Follow us: https://www.fb.com/EoHPlatform
*********************/




// Enable debug console
// Set CORE_DEBUG_LEVEL = 3 first
// #define ERA_DEBUG

#define DEFAULT_MQTT_HOST "mqtt1.eoh.io"

// You should get Auth Token in the ERA App or ERA Dashboard
#define ERA_AUTH_TOKEN "62746a4b-3dd9-4fad-bdb5-710a0dcefe0d"

#include <Arduino.h>
#include <ERa.hpp>
#include <ERa/ERaTimer.hpp>

const char ssid[] = "Eoh.io";
const char pass[] = "EohPassword";|



ERaTimer timer;

/* This function print uptime every second */
void timerEvent() {
    // Code here
}

Done uploading.

Sketch uses 984705 bytes (75%) of program storage space. Maximum is 1310720 bytes.
Global variables use 44288 bytes (13%) of dynamic memory, leaving 283392 bytes for local variables. Maximum is 327680 bytes.
esptool.py v4.2.1
Serial port /dev/cu.usbserial-0001
Connecting.....
Chip is ESP32-D0WDQ6 (revision 1)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
Crystal is 40MHz
MAC: a4:cf:12:05:c7:b8
Uploading stub...
Running stub...|


28           ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP / 1.5MB SPIFFS), 240MHz (WIFI/BT), QIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None, Disabled on /dev/cu.usbserial-0001
```

Khi bắt đầu nạp code sẽ hiển thị tiến độ nạp code.

Hiển thị 100% thể hiện hoàn tất việc nạp code cho Gateway.

The screenshot shows the Arduino IDE interface with the title bar "Gateway_A | Arduino 1.8.19". The main window displays C++ code for an ESP32 Dev Module. The code includes comments for ERA library download, links to ERA website, blog, forum, and Facebook page. It defines MQTT host as "mqtts1.eoh.io" and auth token as "62746a4b-3dd9-4fad-bdb5-710a0dcefe0d". The code includes includes for Arduino.h, ERA.hpp, and ERA/ERaTimer.hpp. It defines ssid and password, creates an ERaTimer object, and defines a timerEvent() function that prints uptime every second. The upload process is shown as completed, writing data to memory at addresses like 0x000eb52c, 0x000f0ca0, etc., at 876.4 kbit/s. The final message is "Hash of data verified." followed by "Leaving..." and "Hard resetting via RTS pin...".

```
Gateway_A | Arduino 1.8.19

Gateway_A

***** Download latest ERA library here:
https://github.com/eoh-jsc/era-lib/releases/latest
https://www.arduino.cc/reference/en/libraries/era
https://registry.platformio.org/libraries/eoh-ltd/ERA/installation

ERa website: https://e-ra.io
ERa blog: https://iotasia.org
ERa forum: https://forum.eoh.io
Follow us: https://www.fb.com/EoHPlatform
*****// Enable debug console
// Set CORE_DEBUG_LEVEL = 3 first
// #define ERA_DEBUG

#define DEFAULT_MQTT_HOST "mqtts1.eoh.io"

// You should get Auth Token in the ERA App or ERA Dashboard
#define ERA_AUTH_TOKEN "62746a4b-3dd9-4fad-bdb5-710a0dcefe0d"

#include <Arduino.h>
#include <ERa.hpp>
#include <ERa/ERaTimer.hpp>

const char ssid[] = "Eoh.io";
const char pass[] = "EohPassword";

ERaTimer timer;

/* This function print uptime every second */
void timerEvent() {
    Serial.println("Uptime: " + String((millis() - startUpTime) / 1000));
}

Done uploading.
Writing at 0x000eb52c... (89 %)
Writing at 0x000f0ca0... (92 %)
Writing at 0x000f6647... (94 %)
Writing at 0x000fb801... (97 %)
Writing at 0x0010130f... (100 %)
Wrote 990480 bytes (624217 compressed) at 0x00010000 in 9.0 seconds (effective 876.4 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

28 ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32Mb), 921600, Core 1, Core 1, None, Disabled on /dev/cu.usbserial-0001
```

Quay lại web và đợi gateway hiển thị “Activate gateway success!” là gateway đã kết nối thành công đến E-Ra. Nhấn Go to Gateway.

The screenshot shows two views of the E-Ra developer portal:

- Top View (Gateway A):** Shows the details for 'Gateway A'. It includes sections for General information (Board: ESP32, Unit: Trường Lý Tự Trọng, Connection time: 1800, WiFi: Eoh.io, Gateway ID: 4130, AUTH TOKEN: 6274...), Connection methods (Zigbee method, Modbus method, Input/Output method), and a sidebar with links like Dashboard Units, All gateways, Manage unit, Subscription, Server status, Documentation, General Trading Conditions, Privacy Policy, and Help.
- Bottom View (Add Gateway):** Shows the 'Add Gateway' process. A progress bar indicates steps: Hardware (done), Install Library (done), Code (done), and Gateway Activation (in progress). The status message says "Gateway is connecting...". Below it, a log window shows the following text:


```
eoh/chip/62746a4b-3dd9-4fad-bdb5-710a0dcef0d/is_online: {"ol":1,"wif_i_ssid":"Eoh.io","ask_configuration":1}
eoh/chip/62746a4b-3dd9-4fad-bdb5-710a0dcef0d/info: {"board":"ESP32","model":"ERA","board_id":"62746a4b-3dd9-4fad-bdb5-
```

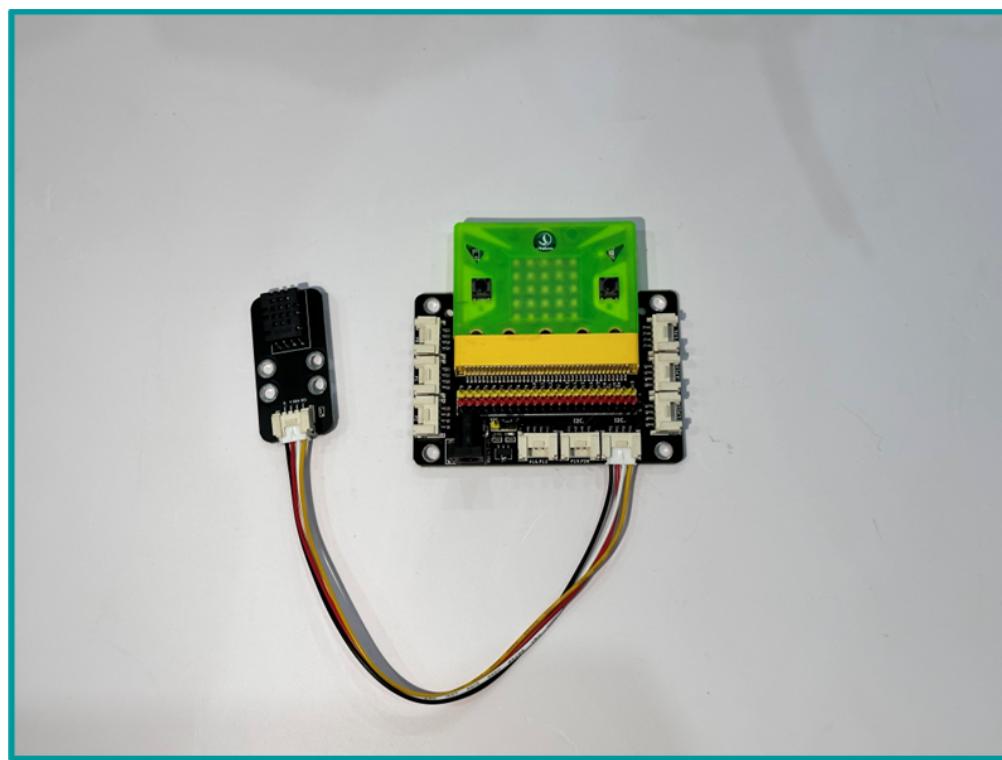
 Buttons at the bottom include Back, Stop creating, and Go to gateway.

2.5.3 Lắp đặt phần cứng

Đọc nhiệt độ độ ẩm

Dùng 1 jack 3 cổng kết nối từ cảm biến đến Kit trung tâm, tại cổng P19.P20.

Gateway sẽ đọc dữ liệu từ chân này và gửi lên E-Ra thông qua Virtual Pin 0 (V0) cho độ ẩm và Virtual Pin 1 (V1) cho nhiệt độ đã được thêm trong code.

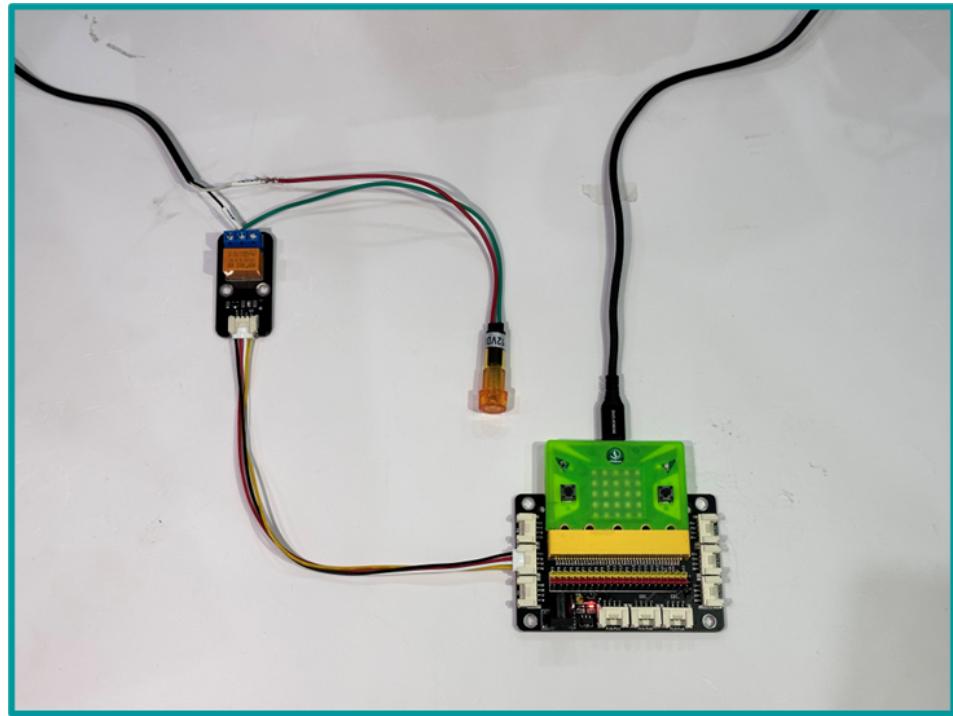
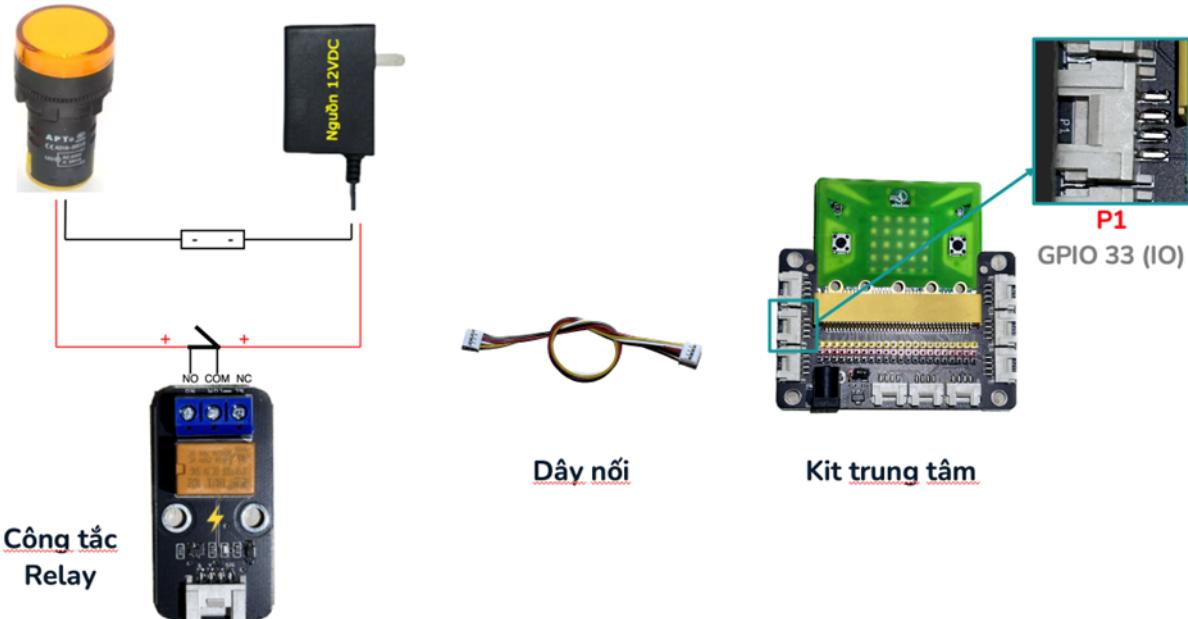


Hình ảnh thực tế

Điều khiển đèn dân dụng

Dùng 1 jack 3 cổng kết nối từ Relay đến Kit trung tâm, tại cổng P1.

Dữ liệu được điều khiển thông qua chân thực tế GPIO33 chỉ với việc cấu hình, không cần thêm code.

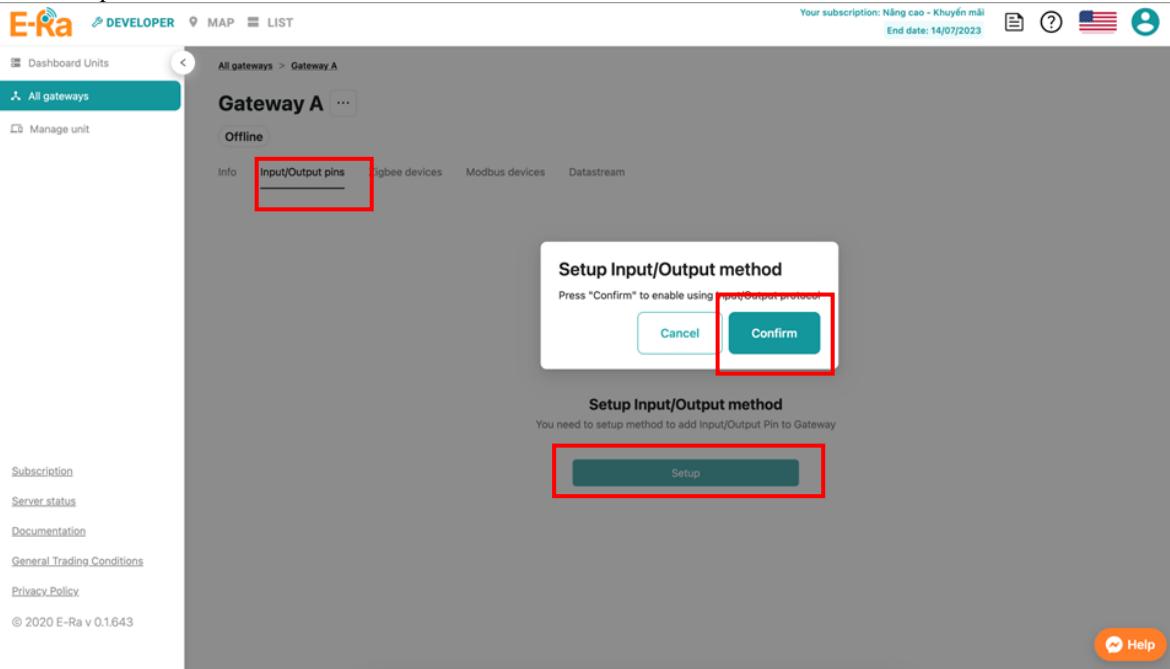


Hình ảnh thực tế

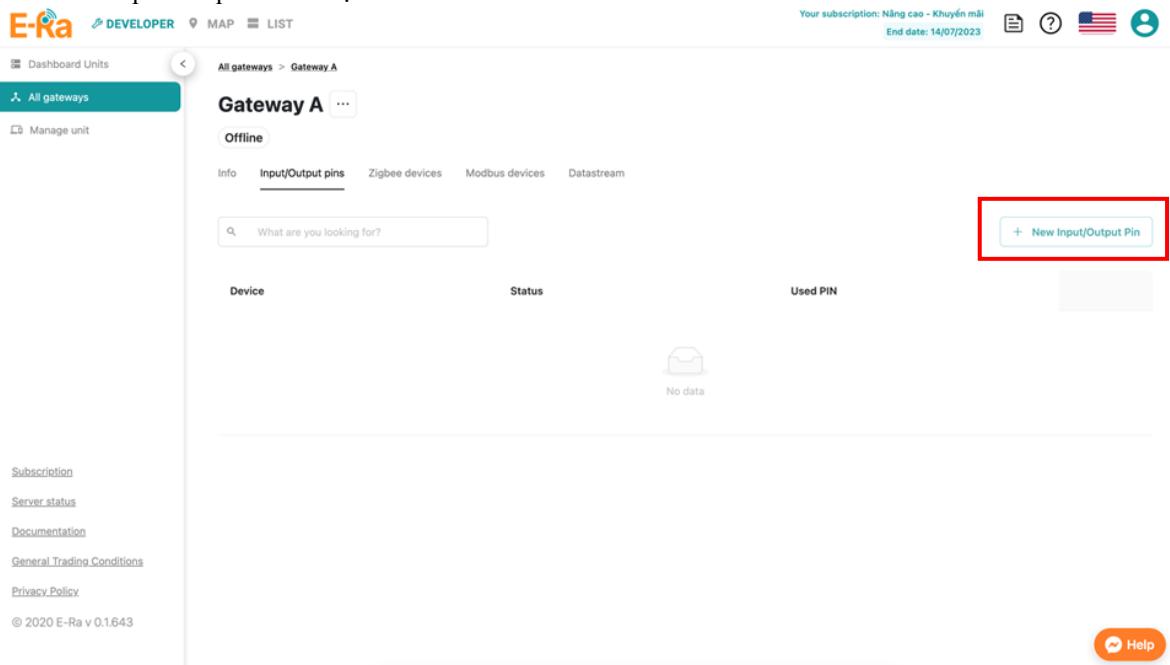
2.5.4 Tạo Device/ Config

Sẽ có nhiều phương thức khác nhau, ở bài thực hành này chúng ta sẽ sử dụng Input/Output pins. Chọn mục Input/Output pins

Nhấn Setup và Confirm

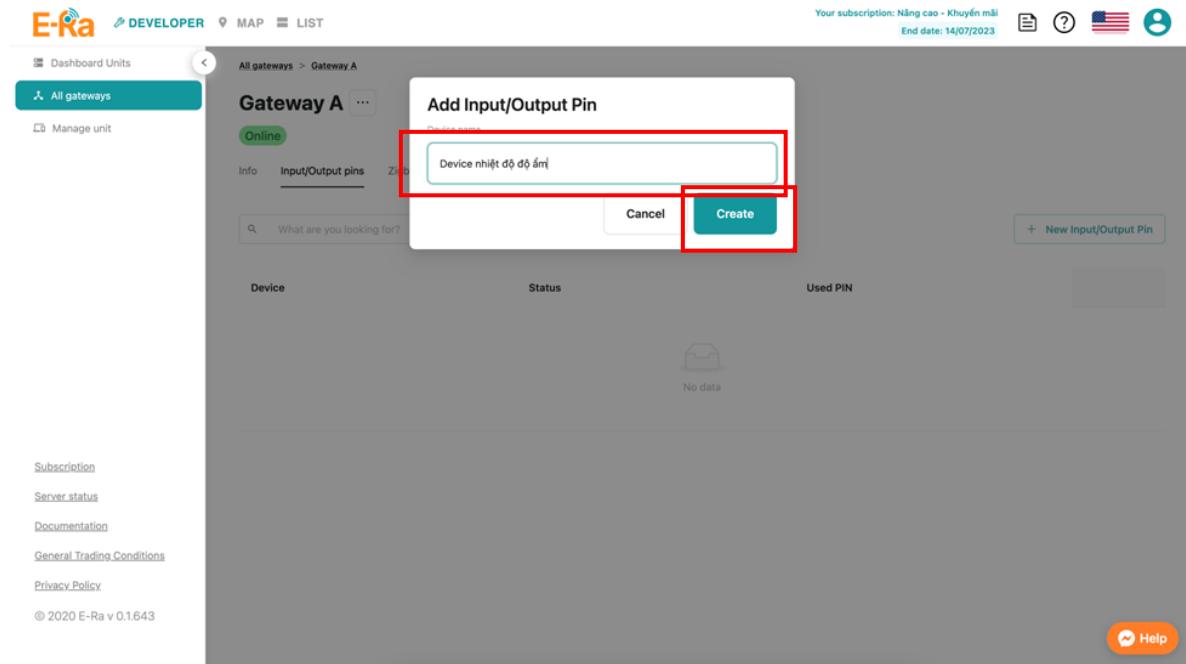


Nhấn New Input/Output Pin để tạo Device mới



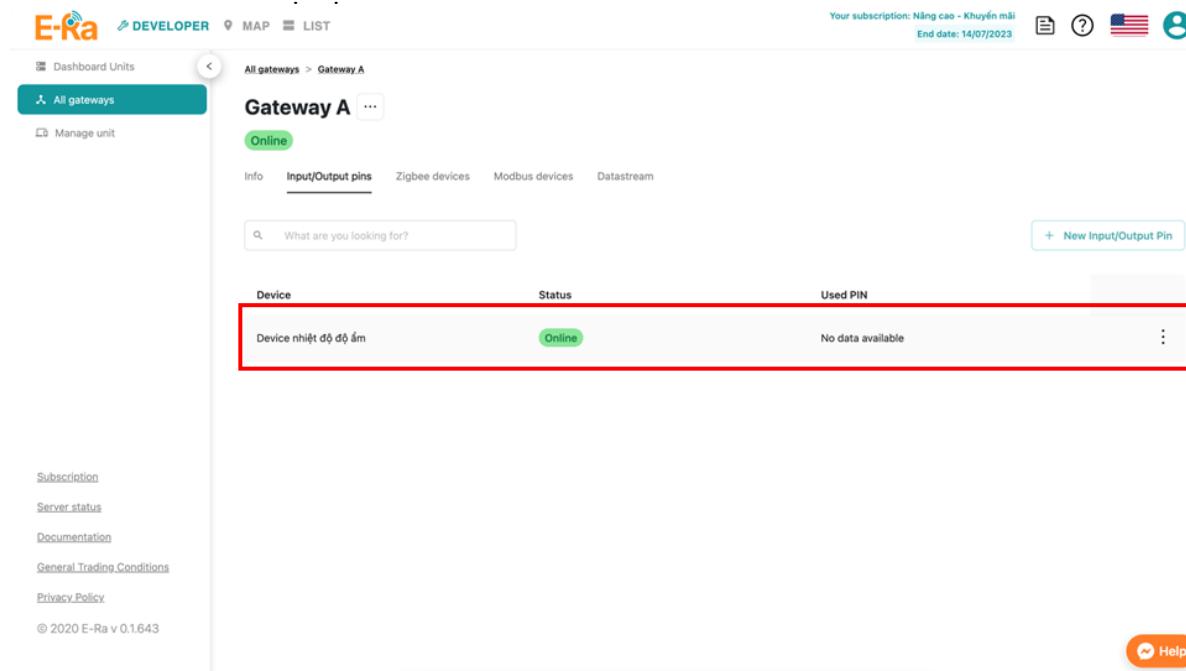
Đặt tên Device nhiệt độ, độ ẩm.

Nhấn Create



Device nhiệt độ độ ẩm được tạo thành công.

Nhấn vào Device vừa được tạo



Chọn mục Virtual pin

Nhấn New virtual pin để tạo chân pin ảo mới

Virtual pin (chân ảo) là 1 biến trung gian để Gateway và E-Ra có thể gửi nhận dữ liệu với nhau.

Trong phần code ta đã thêm đọc dữ liệu từ cảm biến DHT20 và gửi thông qua giá trị độ ẩm thông qua V0, nhiệt độ thông qua V1 nên ở đây ta cũng mapping tương tự.

The screenshot shows the E-Ra developer interface. On the left, there's a sidebar with links like 'Dashboard Units', 'All gateways' (which is selected), 'Manage unit', 'Subscription', 'Server status', 'Documentation', 'General Trading Conditions', and 'Privacy Policy'. The main area has a title 'Input/Output pins' with '1 Device' listed. Below it is a search bar and a section for 'Device nhiệt độ độ ẩm' which says 'None' and has a '+ New device' button. At the top right, there's a message about a subscription and some user icons. In the center, there's a sub-menu for 'Device nhiệt độ độ ẩm' with tabs for 'Info', 'Config write', 'Config read', 'Action', and 'Virtual pin' (which is currently selected). Below this, there's another search bar and a button '+ New virtual pin' which is highlighted with a red box. The main content area shows a table with columns 'Config Name' and 'PIN', both currently empty with a note 'No data'.

Đặt tên cho config chân ảo V0 là Độ ẩm

Nhấn Done

This screenshot shows a modal dialog titled 'Add virtual pin'. It has two input fields: 'Virtual pin name' containing 'Độ ẩm' and 'PIN' containing 'V0'. There are 'Cancel' and 'Done' buttons at the bottom, with 'Done' being highlighted with a red box. The background is the same interface as the previous screenshot, showing the 'Input/Output pins' page for the temperature device.

Tiếp tục nhấn New virtual pin để tạo config chân ảo mới

The screenshot shows the E-Ra developer interface. On the left, there's a sidebar with links like 'Subscription', 'Server status', 'Documentation', 'General Trading Conditions', and 'Privacy Policy'. The main area is titled 'Input/Output pins' and shows one device: 'Device nhiệt độ độ ẩm v0'. Below it is a table with columns 'Config Name' and 'PIN'. A single row is listed: 'Độ ẩm' under 'Config Name' and 'V0' under 'PIN'. At the top of the page, there's a header bar with the E-Ra logo, 'DEVELOPER', 'MAP', 'LIST', and user information: 'Your subscription: Nâng cao - Khuyến mãi', 'End date: 14/07/2023', and icons for help, download, and profile.

Đặt tên cho config chân ảo V1 là Nhiệt độ

This screenshot shows the 'Add virtual pin' dialog box overlaid on the previous interface. The dialog has two input fields: 'Virtual pin name' containing 'Nhiệt độ' and 'PIN' containing 'V1'. Below these fields are 'Cancel' and 'Done' buttons. The 'Done' button is highlighted with a red box. The background shows the same E-Ra developer interface with the temperature device and its configuration table.

Sau khi tạo thành công Độ ẩm và Nhiệt độ
Nhấn Input/Output pins để quay về tạo thêm Device mới

The screenshot shows the E-Ra developer interface. On the left, there's a sidebar with 'Dashboard Units', 'All gateways' (which is selected), and 'Manage unit'. The main area has a header 'Input/Output pins' with '1 Devices'. Below it is a search bar and a list item 'Device nhiệt độ độ ẩm V0, V1'. At the bottom of this list is a button '+ New device'. To the right, there's a breadcrumb navigation 'All gateways > Gateway A > Input/Output pins > Device nhiệt độ độ ẩm'. The main content area is titled 'Device nhiệt độ độ ẩm' with status 'Online'. It has tabs 'Info', 'Config write', 'Config read', 'Action', and 'Virtual pin' (which is selected). There's another search bar and a button '+ New virtual pin'. Below these are two rows of configuration data:

Config Name	PIN	⋮
Độ ẩm	V0	⋮
Nhiệt độ	V1	⋮

At the bottom of the page, there are links for 'Subscription', 'Server status', 'Documentation', 'General Trading Conditions', 'Privacy Policy', and copyright information '© 2020 E-Ra v 0.1.643'. On the far right, there's a 'Help' button.

Nhấn New Input/Output Pin để tạo Device thứ 2.

This screenshot shows the E-Ra developer interface again. The sidebar is identical to the previous one. The main area shows 'Gateway A' with status 'Online'. The 'Input/Output pins' tab is selected, showing a search bar and a button '+ New Input/Output Pin' (which is highlighted with a red box). Below this is a table with three columns: 'Device', 'Status', and 'Used PIN'. One row is listed: 'Device nhiệt độ độ ẩm' with status 'Online' and used pins 'V0, V1'. The bottom of the page has the same footer links and copyright information as the first screenshot, along with a 'Help' button.

Đặt tên Device Đèn. Nhấn Create

Your subscription: Nâng cao - Khuyến mãi
End date: 14/07/2023

Dashboard Units All gateways MAP LIST

All gateways > Gateway A

Gateway A Online

Info Input/Output pins Zigbee

Add Input/Output Pin

Device name: Device Đèn

Create

Cancel

What are you looking for? + New Input/Output Pin

Device	Status	Used PIN
Device nhiệt độ độ ẩm	Online	V0, V1

Subscription Server status Documentation General Trading Conditions Privacy Policy © 2020 E-Ra v 0.1.643 Help

Chọn Device đèn được tạo

Your subscription: Nâng cao - Khuyến mãi
End date: 14/07/2023

Dashboard Units All gateways MAP LIST

All gateways > Gateway A

Gateway A Online

Info Input/Output pins Zigbee devices Modbus devices Datastream

What are you looking for? + New Input/Output Pin

Device	Status	Used PIN
Device nhiệt độ độ ẩm	Online	V0, V1
Device Đèn	Online	No data available

Subscription Server status Documentation General Trading Conditions Privacy Policy © 2020 E-Ra v 0.1.643 Help

Ở đây chúng ta cần điều khiển thiết bị nên sẽ chọn mục Config write.

Sau đó nhấn New config write để tạo config mới.

The screenshot shows the E-Ra developer interface. On the left, there's a sidebar with links like 'Subscription', 'Server status', 'Documentation', 'General Trading Conditions', and 'Privacy Policy'. The main area is titled 'Input/Output pins' and shows two devices: 'Device cảm biến ánh sáng v0' and 'Device đèn None'. A search bar says 'What are you looking for?'. On the right, under 'Device đèn', there are tabs for 'Info', 'Config write' (which is selected), 'Config read', 'Action', and 'Virtual pin'. A red box highlights the 'New Config write' button at the top right of this section. Below it, there's a table with columns 'Config Name', 'Config Type', 'PIN', and 'PIN Mode', showing 'No data'.

Đặt tên Config write là Đèn.

Vì cần điều khiển chân GPIO2 nên Chọn Pin là GPIO2.

Tín hiệu điều khiển là digital và output nên chúng ta giữ nguyên.

Nhấn Done.

This screenshot shows a modal dialog box titled 'Add config write'. It contains fields for 'Config write name' (set to 'Đèn'), 'Type config' (set to 'Digital'), 'PIN' (set to 'GPIO 2'), and 'PIN Mode' (set to 'Output'). At the bottom of the dialog are 'Cancel' and 'Done' buttons, with 'Done' being highlighted by a red box. The background shows the same interface as the previous screenshot, with the 'Input/Output pins' page visible.

Để điều khiển thì chúng ta sẽ cần action, cụ thể ở đây sẽ là 1 action on và 1 action off.

Chọn mục Action.

The screenshot shows the E-Ra developer interface. On the left, there's a sidebar with links like 'Subscription', 'Server status', 'Documentation', 'General Trading Conditions', 'Privacy Policy', and copyright information. The main area has tabs for 'Info', 'Config write', 'Config read', 'Action', and 'Virtual pin'. The 'Action' tab is highlighted with a red box. Below it, there's a table with columns 'Config Name', 'Config Type', 'PIN', and 'PIN Mode'. One row is visible: 'Đèn' (Config Name), 'Digital' (Config Type), 'GPIO 2' (PIN), and 'Output' (PIN Mode). At the bottom right of the main area, there's a 'Help' button.

Chọn New action

Đặt tên Action Đèn ON

Chọn Pin GPIO2

Value điều khiển là 1

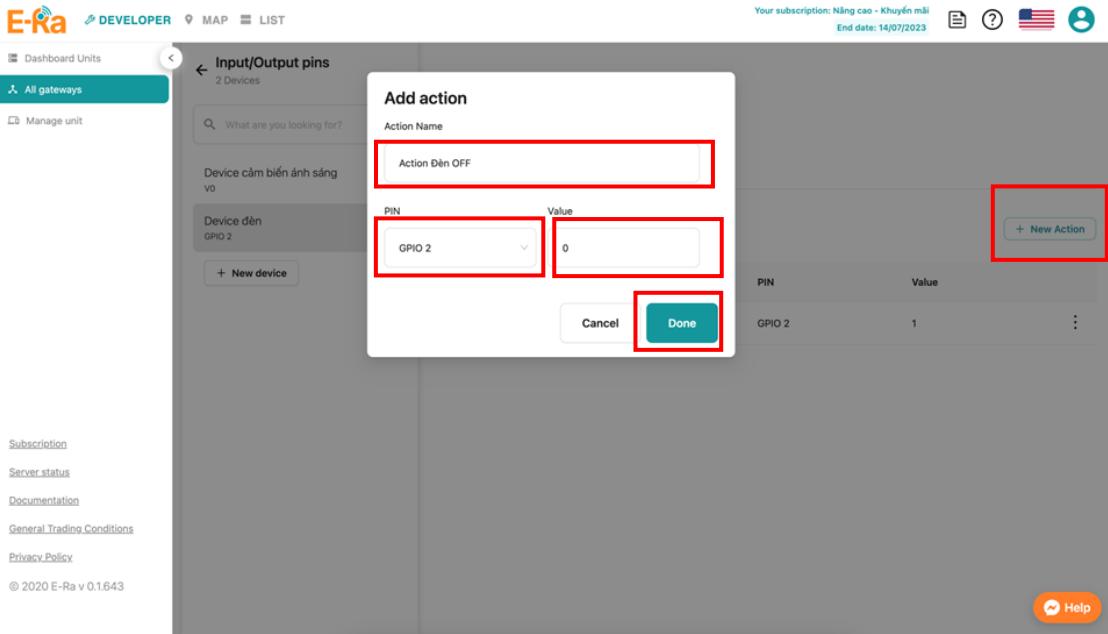
The screenshot shows the 'Add action' dialog box from the E-Ra developer interface. It has fields for 'Action Name' (containing 'Action Đèn ON'), 'PIN' (containing 'GPIO 2'), and 'Value' (containing '1'). At the bottom, there are 'Cancel' and 'Done' buttons, with 'Done' being highlighted with a red box. The background shows the main interface with a table for actions and a 'New Action' button.

Tiếp tục tương tự tạo Action thứ 2

Đặt tên Action Đèn OFF

Chọn Pin GPIO2

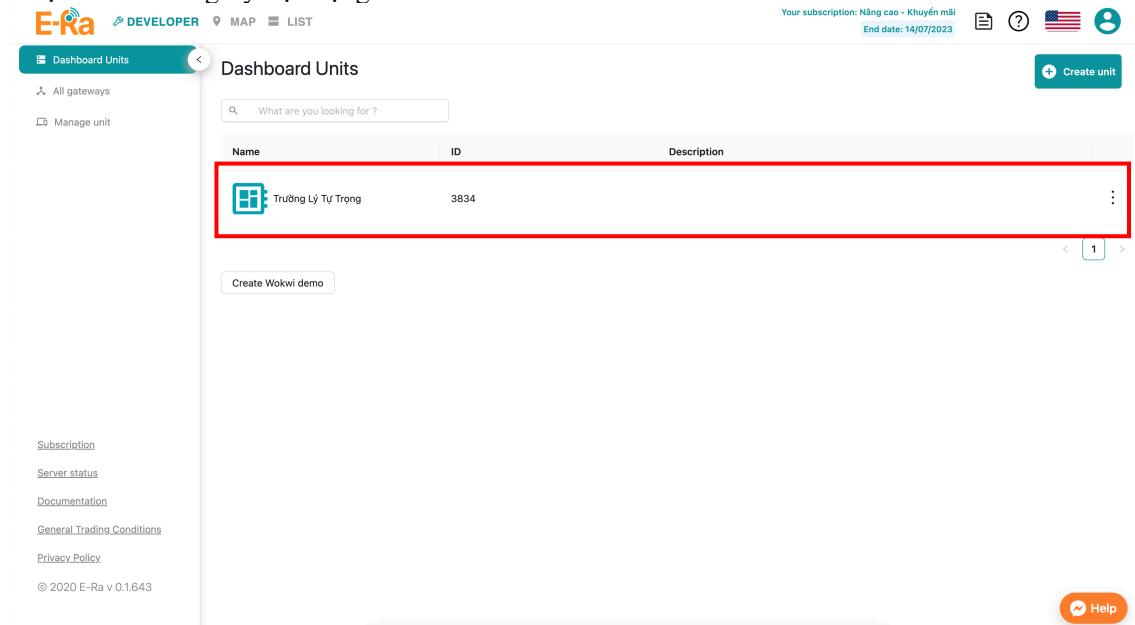
Value điều khiển là 0



Các bước tạo Widget

Quay lại mục Dashboard Units

Chọn Unit Trường Lý Tự Trọng



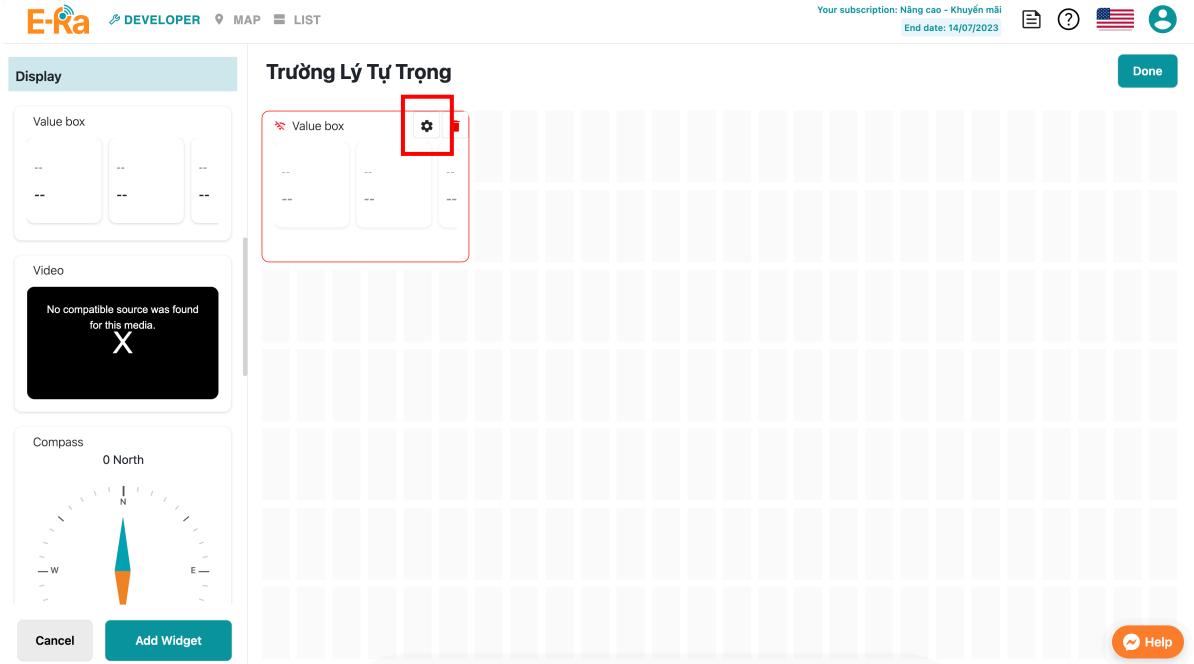
Chọn Edit Dashboard

The screenshot shows the E-Ra developer dashboard. At the top, there's a header with the E-Ra logo, 'DEVELOPER', 'MAP', 'LIST', and user information like 'Your subscription: Nâng cao - Khuyến mãi' and 'End date: 14/07/2023'. Below the header, a breadcrumb navigation shows 'Dashboard Units > Trưởng Lý Tự Trọng'. On the left, a sidebar has 'Dashboard Units' selected. A main content area titled 'Trưởng Lý Tự Trọng' contains a large empty space and a 'Edit Dashboard' button at the top right, which is highlighted with a red box. The bottom of the screen includes links for 'Subscription', 'Server status', 'Documentation', 'General Trading Conditions', 'Privacy Policy', and copyright information '© 2020 E-Ra v 0.1.643'. A 'Help' button is also visible.

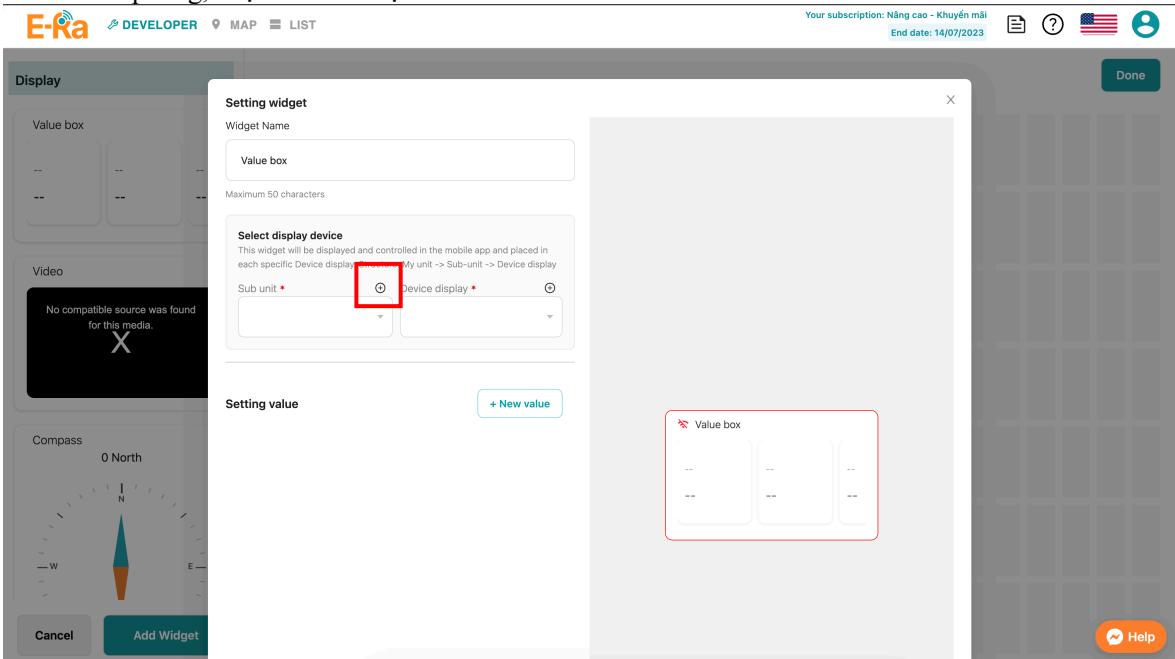
Chọn Widget Value box dùng để hiển thị giá trị của dữ liệu.
Nhấn add Widget

The screenshot shows the 'Add Widget' dialog box. On the left, there are three categories: 'Value box' (highlighted with a red box), 'Video' (with a message 'No compatible source was found for this media.' and a red X icon), and 'Compass'. On the right, there's a large empty grid area for placing widgets. At the bottom left is a 'Cancel' button, and at the bottom center is a prominent 'Add Widget' button, which is also highlighted with a red box. The top right of the dialog has a 'Done' button. The overall interface is clean with a light blue header and a white body.

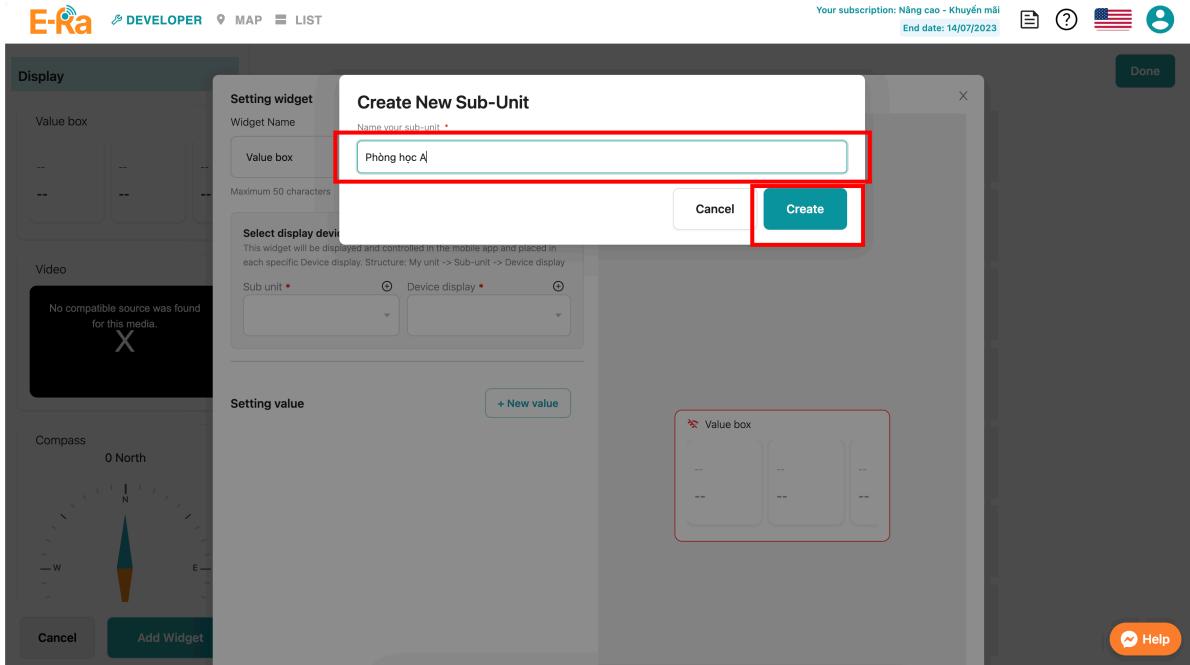
Di chuyển chuột vào Widget vừa được thêm.
Nhấn vào icon răng cưa để cấu hình Widget này.



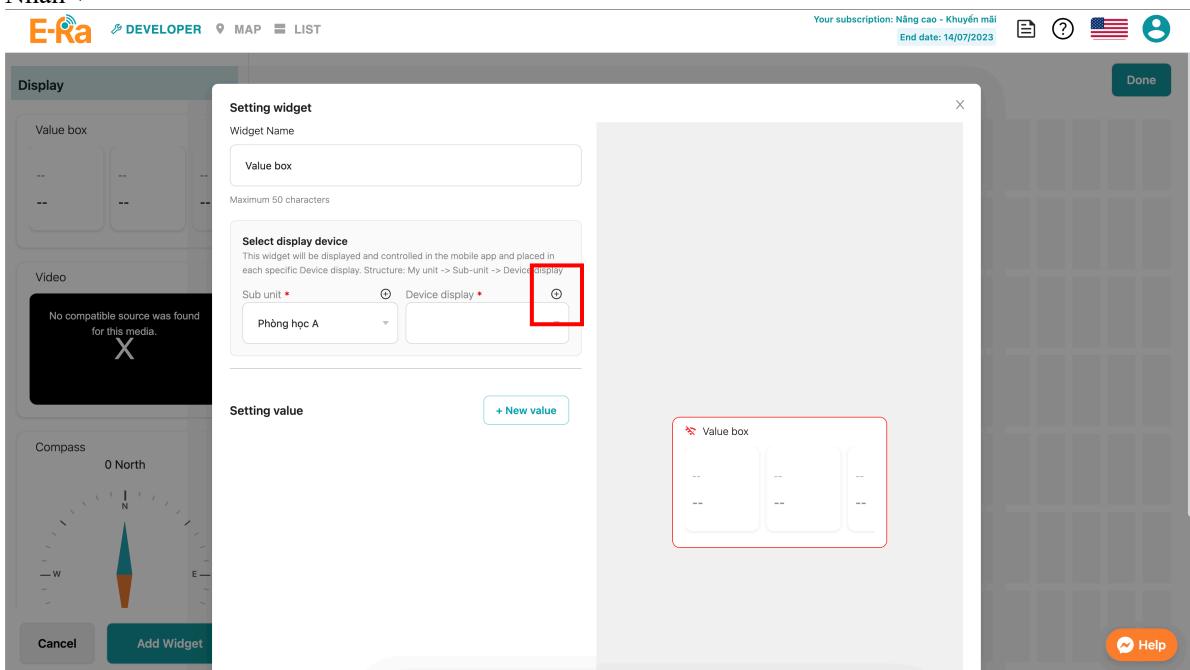
Tại mục Sub-unit, dùng để phân loại phòng cho các Widget này.
Vì chưa có phòng, chọn dấu + để tạo mới Sub-unit.



Đặt tên là Phòng học A.
Nhấn Create.



Device display giúp hiển thị trong mobile.
Tại mục Device display
Nhấn +

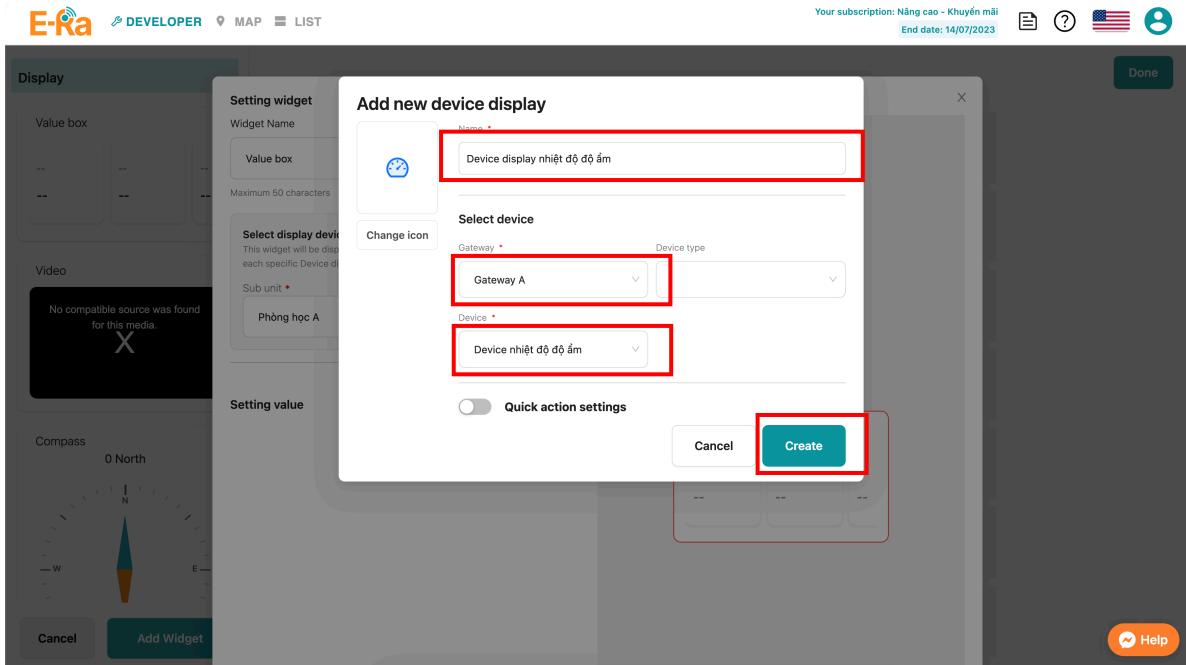


Đặt tên là Device display nhiệt độ độ ẩm

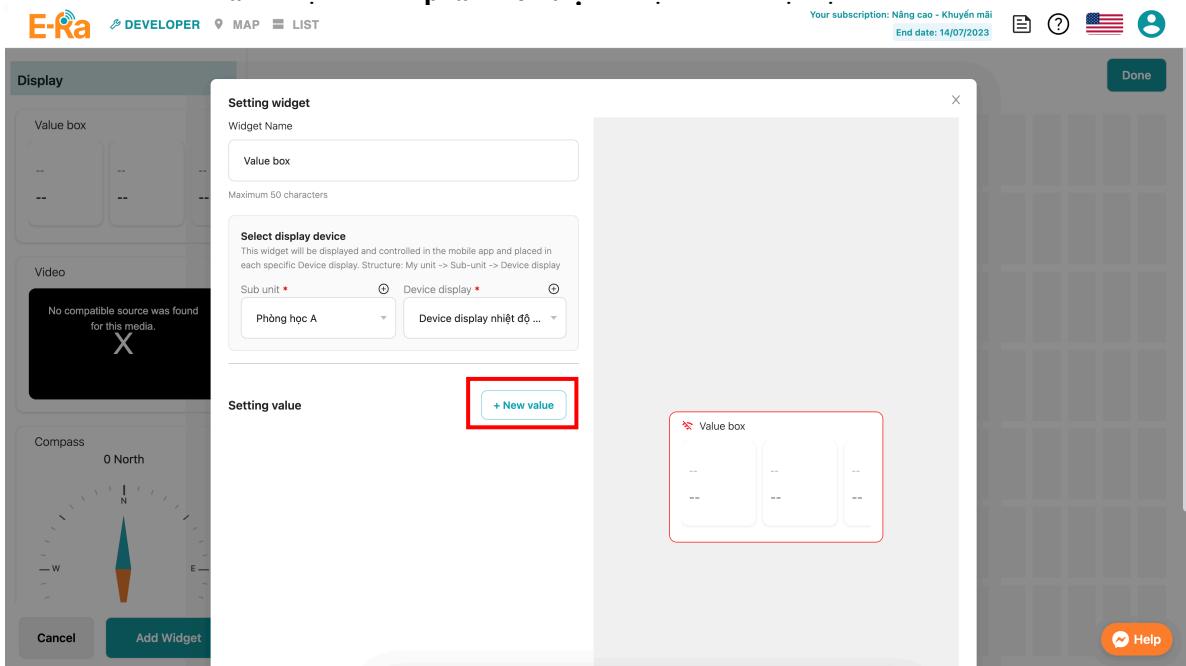
Chọn Gateway A

Chọn Device nhiệt độ độ ẩm

Nhấn Create



Nhấn New value 2 lần để tạo thêm 2 phần hiển thị cho độ ẩm và nhiệt độ.



Box đầu tiên Value 0

Đặt tên là **Độ ẩm**

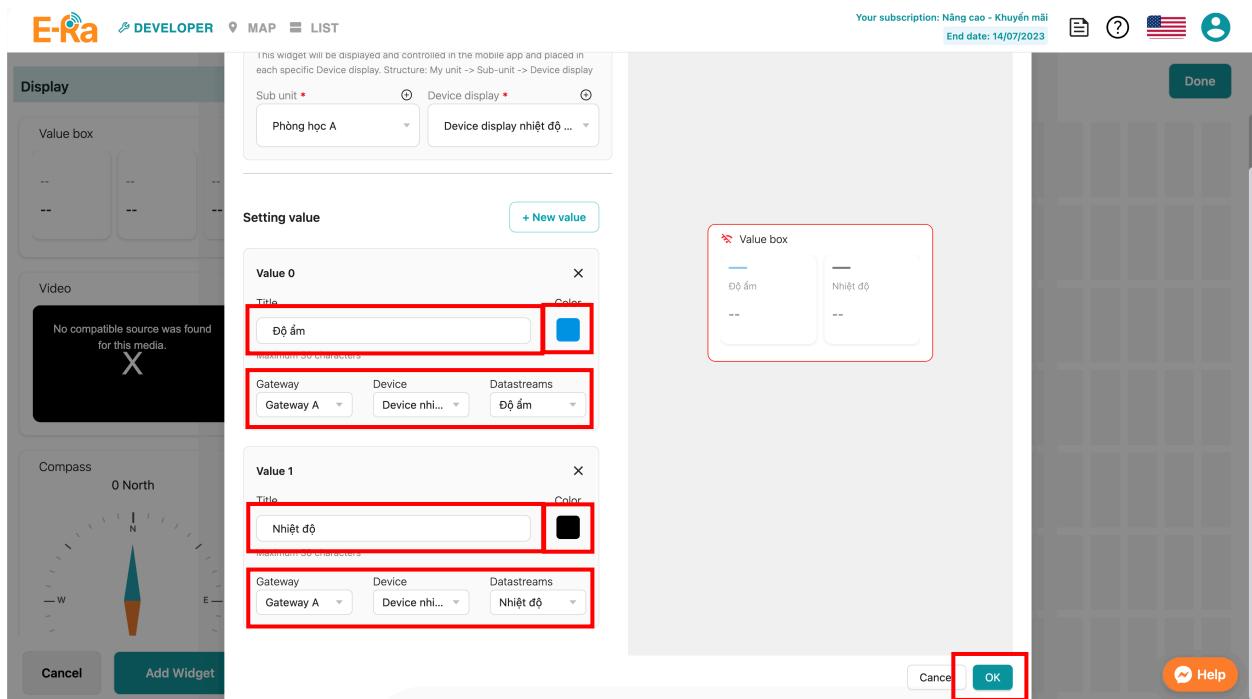
Chọn Gateway A > Device nhiệt độ độ ẩm > Datastream **Độ ẩm**

Box thứ hai Value 1

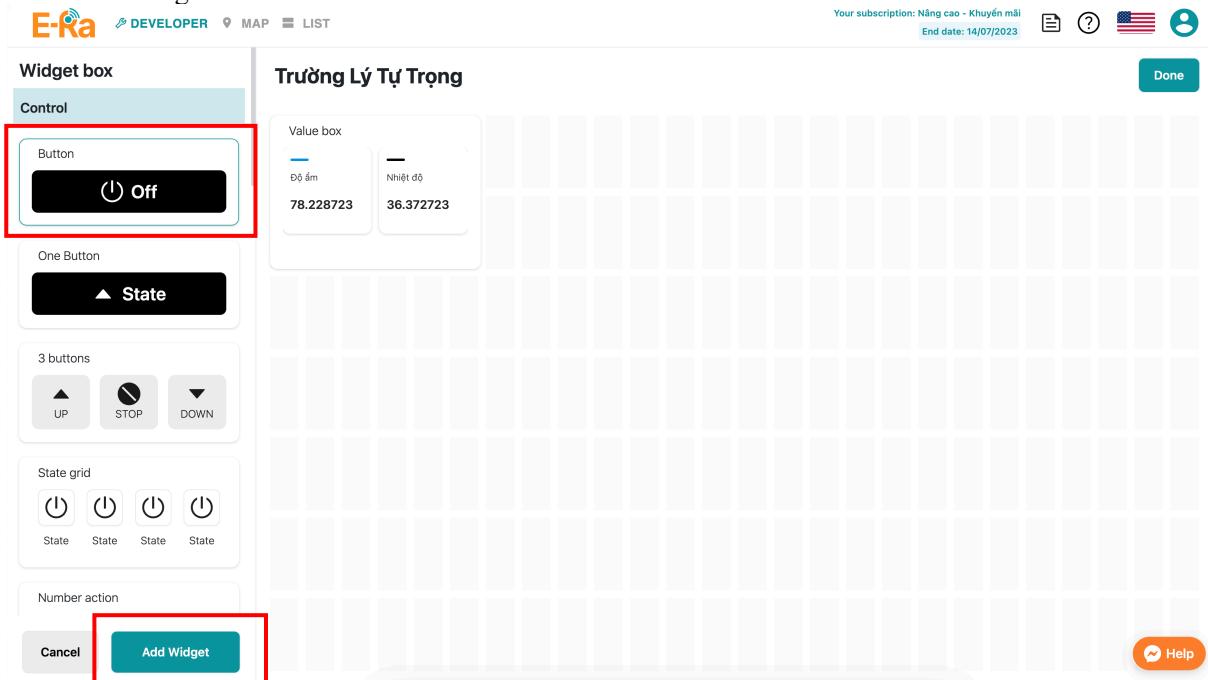
Đặt tên là **Nhiệt độ**

Chọn Gateway A > Device nhiệt độ độ ẩm > Datastream **Nhiệt độ**

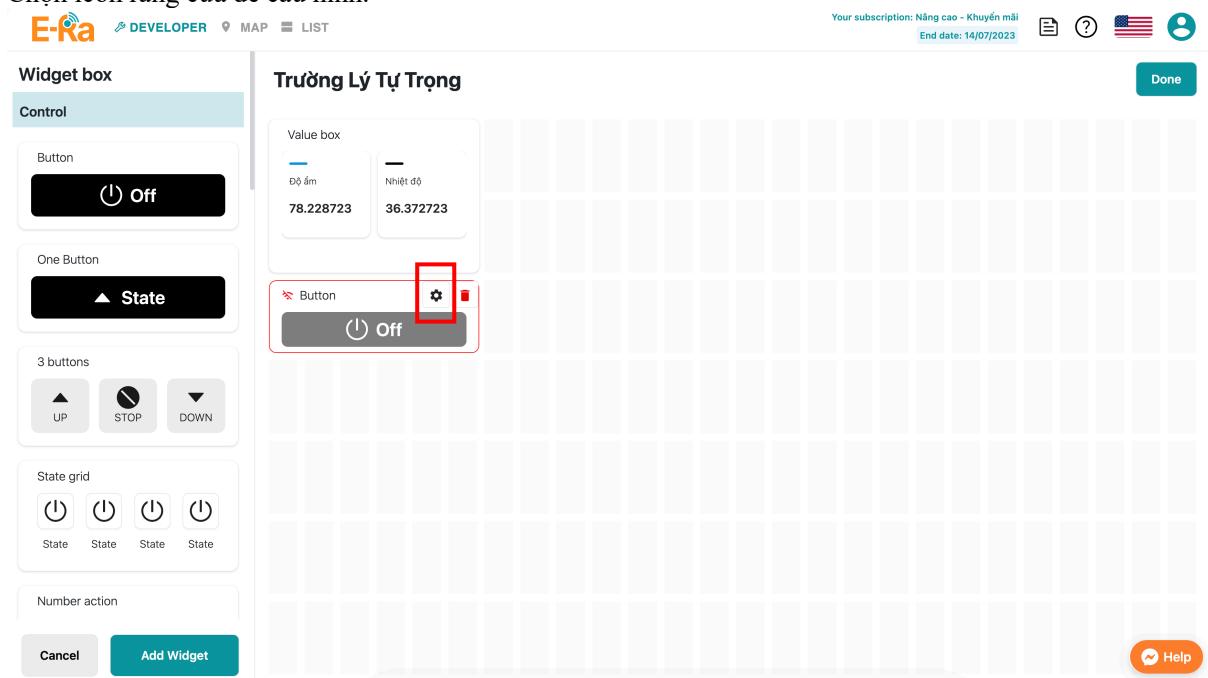
Nhấn Ok



Tiếp tục chọn Widget Button dùng cho việc điều khiển on / off.
Nhấn Add Widget.

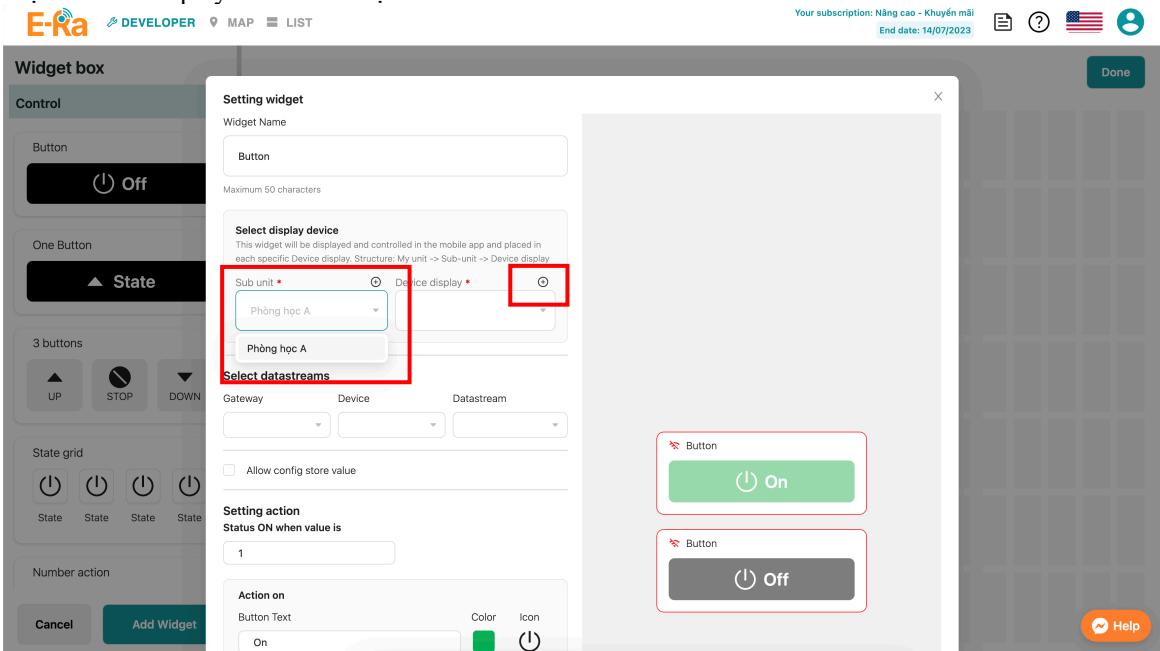


Di chuyển chuột vào Widget Button vừa được thêm.
Chọn icon răng cưa để cấu hình.



Mục Sub-unit: Chọn Phòng học A đã tạo trước đó

Mục Device display: Nhấn + để tạo mới

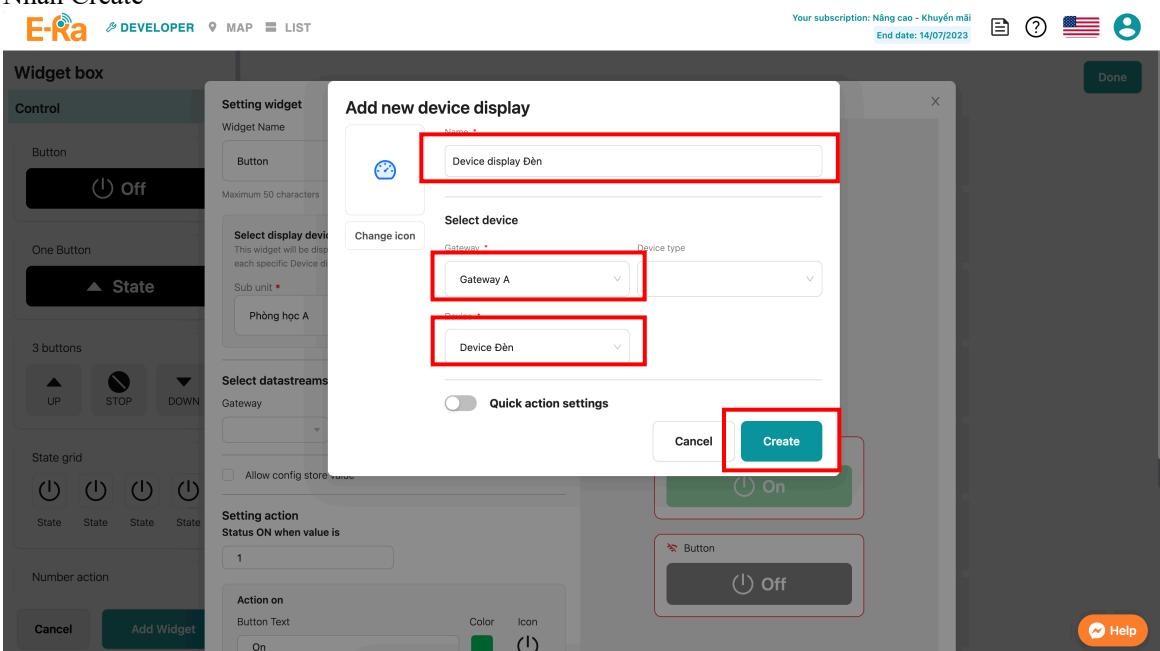


Đặt tên Device display Đèn

Chọn Gateway A

Chọn Device Đèn

Nhấn Create

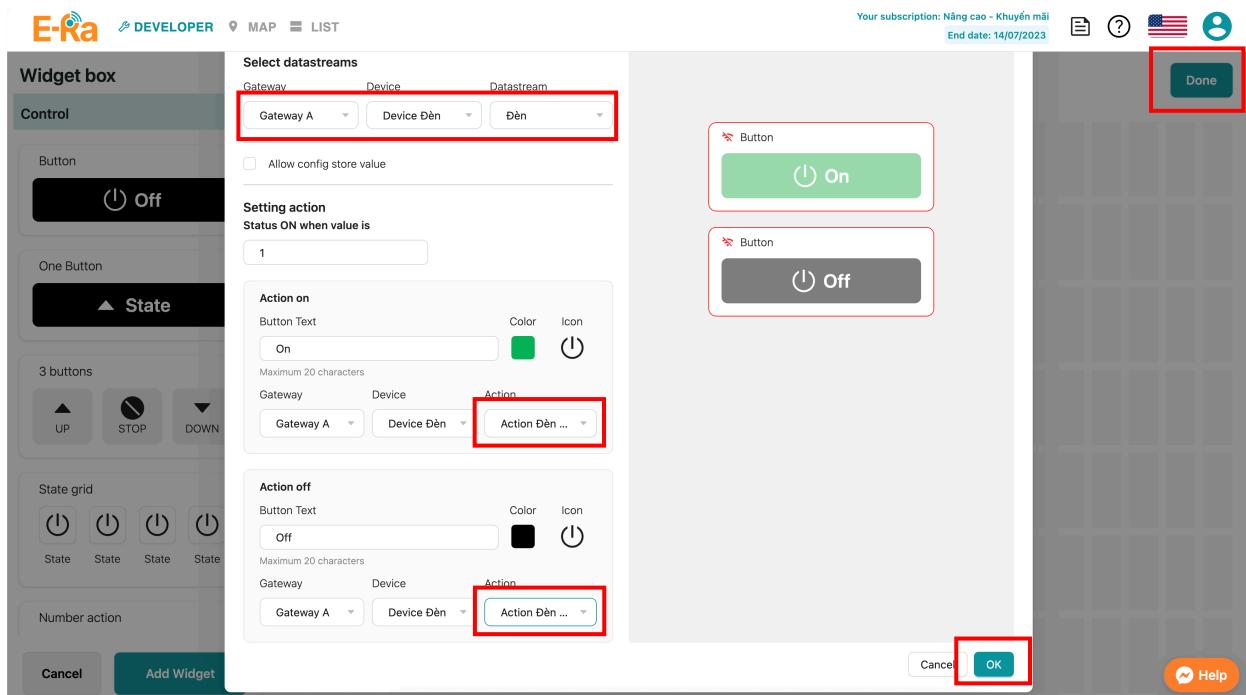


Mục Select datastreams:
Gateway A > Device Đèn > Đèn

Mục Action On:
Đặt tên là On
Gateway A > Device Đèn > Action Đèn ON

Mục Action Off:
Đặt tên là Off
Gateway A > Device Đèn > Action Đèn OFF

Nhấn Ok
Sau khi cấu hình hoàn tất, nhấn Done.



Trang Dashboard Unit sau khi hoàn tất cấu hình
Giá trị nhiệt độ, độ ẩm với số thập phân chưa đẹp và chưa có đơn vị.

The screenshot shows a dashboard titled "Trường Lý Tự Trọng". At the top, there is a "Value box" containing two numerical values: 78.228723 and 36.372723. These values are highlighted with a red rectangular box. Below the value box is a "Button" labeled "Off". The dashboard also includes a navigation bar with links like "Dashboard Units", "MAP", "LIST", "All gateways", and "Manage unit". On the right side, there are user profile icons and a "Help" button. The bottom left corner contains legal links: "Subscription", "Server status", "Documentation", "General Trading Conditions", and "Privacy Policy".

2.5.5 Điều chỉnh số thập phân sau dấu phẩy và đơn vị

Quay lại mục All gateways

Chọn Gateway A

Gateway	Status	Unit	Type	Mqtt server	ID	Last updated	QR code
Gateway A ESP32	Online	Trường Lý Tự Trọng	Input/Output	Viet Nam	4130	06:35 10/07/2023	Create QR code

Chọn mục Datastream

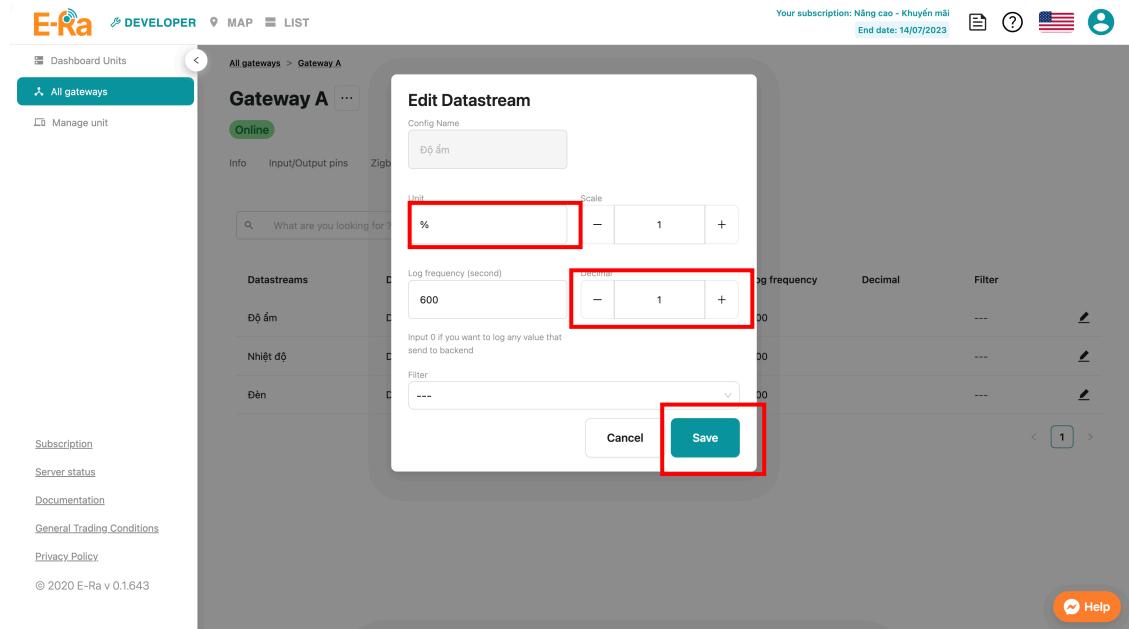
Chọn hình bút chì ở cuối mục Độ ẩm

Datastreams	Device	Unit	Scale	Log frequency	Decimal	Filter
Độ ẩm	Device nhiệt độ độ ẩm	1	600	---	---	
Nhiệt độ	Device nhiệt độ độ ẩm	1	600	---	---	
Đèn	Device Đèn	1	600	---	---	

Điều đơn vị là %.

Điều decimal là 1, để chỉ hiển thị 1 số sau dấu phẩy.

Nhấn Save.

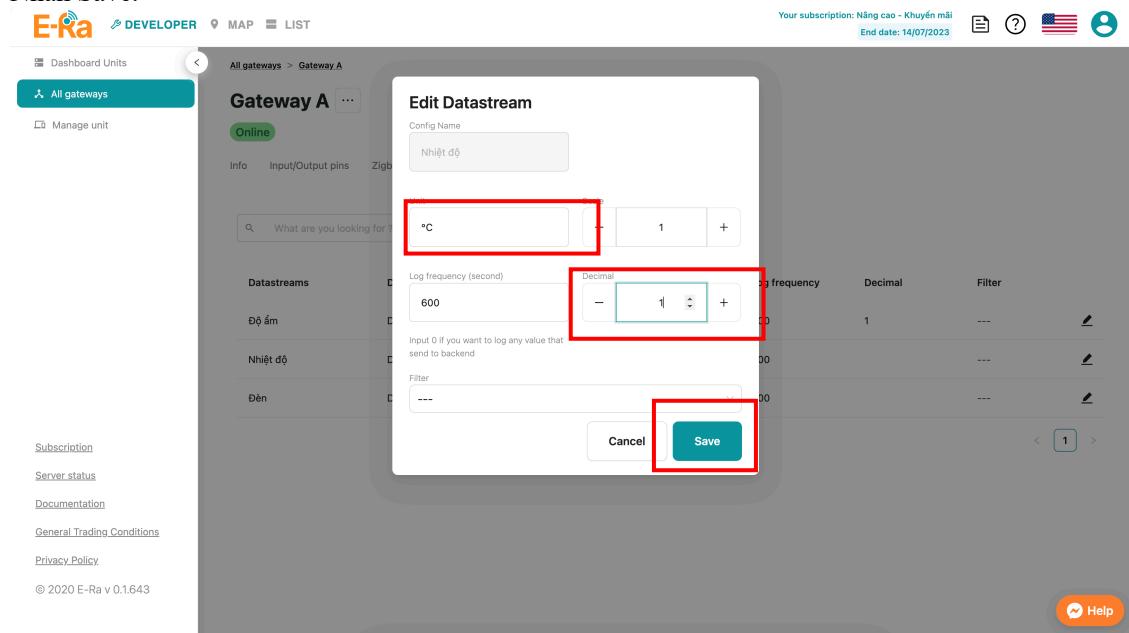


Tương tự cho Nhiệt độ.

Điều đơn vị là °C.

Điều decimal là 1, để chỉ hiển thị 1 số sau dấu phẩy.

Nhấn Save.



Sau khi hoàn tất cấu hình Datastream.
 Quay lại mục Dashboard Units.
 Chọn Unit Trường Lý Tự Trọng để xem kết quả.

Your subscription: Nâng cao - Khuyến mãi
End date: 14/07/2023

Gateway A

Online

Datastream

Datastreams	Device	Unit	Scale	Log frequency	Decimal	Filter
Độ ẩm	Device nhiệt độ độ ẩm	%	1	600	1	---
Nhiệt độ	Device nhiệt độ độ ẩm	°C	1	600	1	---
Đèn	Device Đèn		1	600		---

Subscription Server status Documentation General Trading Conditions Privacy Policy © 2020 E-Ra v 0.1.643

Help

Your subscription: Nâng cao - Khuyến mãi
End date: 14/07/2023

Dashboard Units

Create unit

Name	ID	Description
Trường Lý Tự Trọng	3834	

Create Wokwi demo

Subscription Server status Documentation General Trading Conditions Privacy Policy © 2020 E-Ra v 0.1.643

Help

Cảm biến nhiệt độ độ ẩm

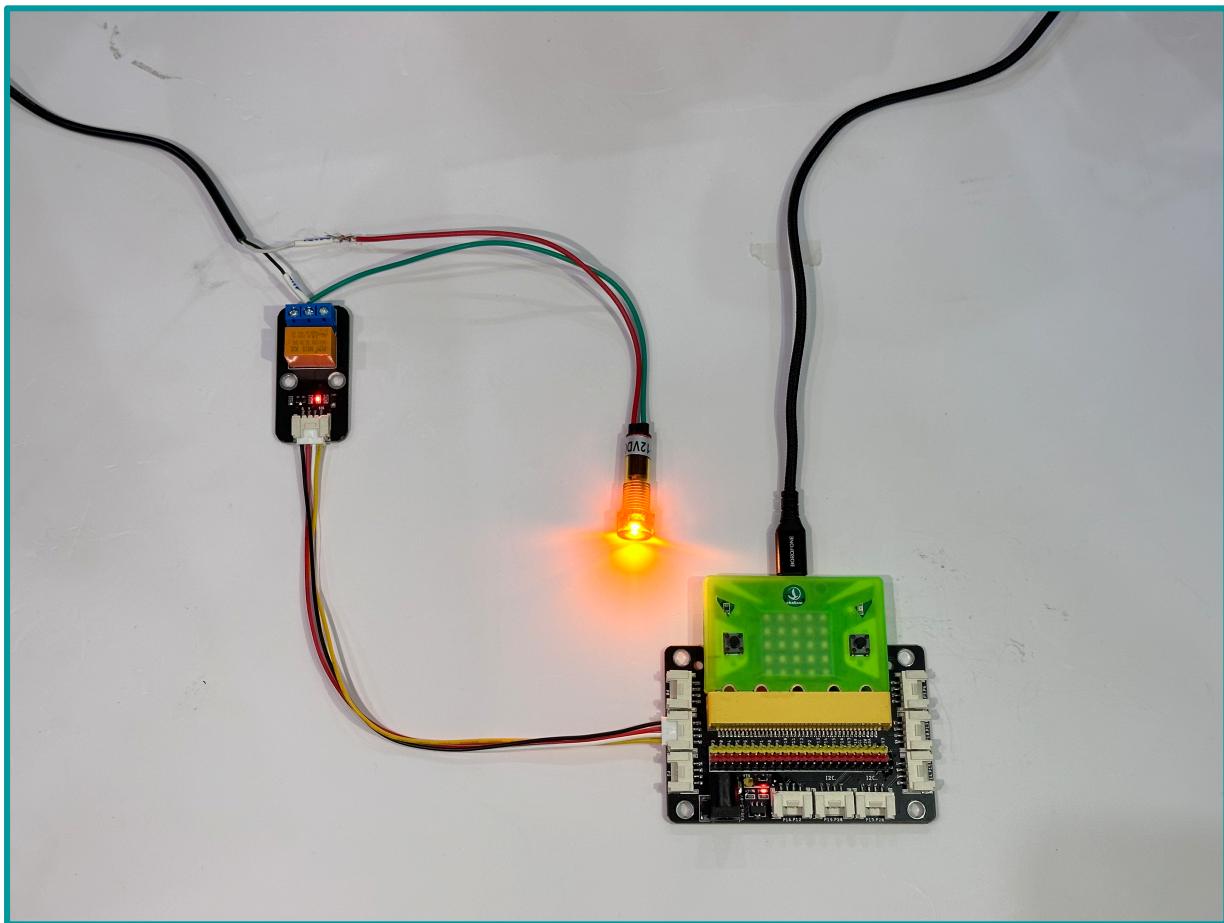
Giá trị Value Box lúc này đã đẹp với 1 số thập phân và có đơn vị.
Và trạng thái button hiện tại là tắt . Nhấn điều khiển để bật đèn.

The screenshot shows a developer dashboard for a smart device named "Trường Lý Tự Trọng". At the top, there's a "Value box" containing two metrics: "Độ ẩm" (Humidity) at 72.6% and "Nhiệt độ" (Temperature) at 35.3 °C. Below the value box is a large button labeled "Off". In the bottom right corner of the dashboard, there's a "Help" button.

Thiết bị điều khiển đèn

Trạng thái đèn đã thay đổi và cập nhật là ON.
Nhấn điều khiển để tắt đèn.

The screenshot shows a developer dashboard for a smart device named "Trường Lý Tự Trọng". At the top, there's a "Value box" containing two metrics: "Độ ẩm" (Humidity) at 72.3% and "Nhiệt độ" (Temperature) at 36.6 °C. Below the value box is a large button labeled "On". In the bottom right corner of the dashboard, there's a "Help" button.



Hình ảnh thực tế đèn được bật

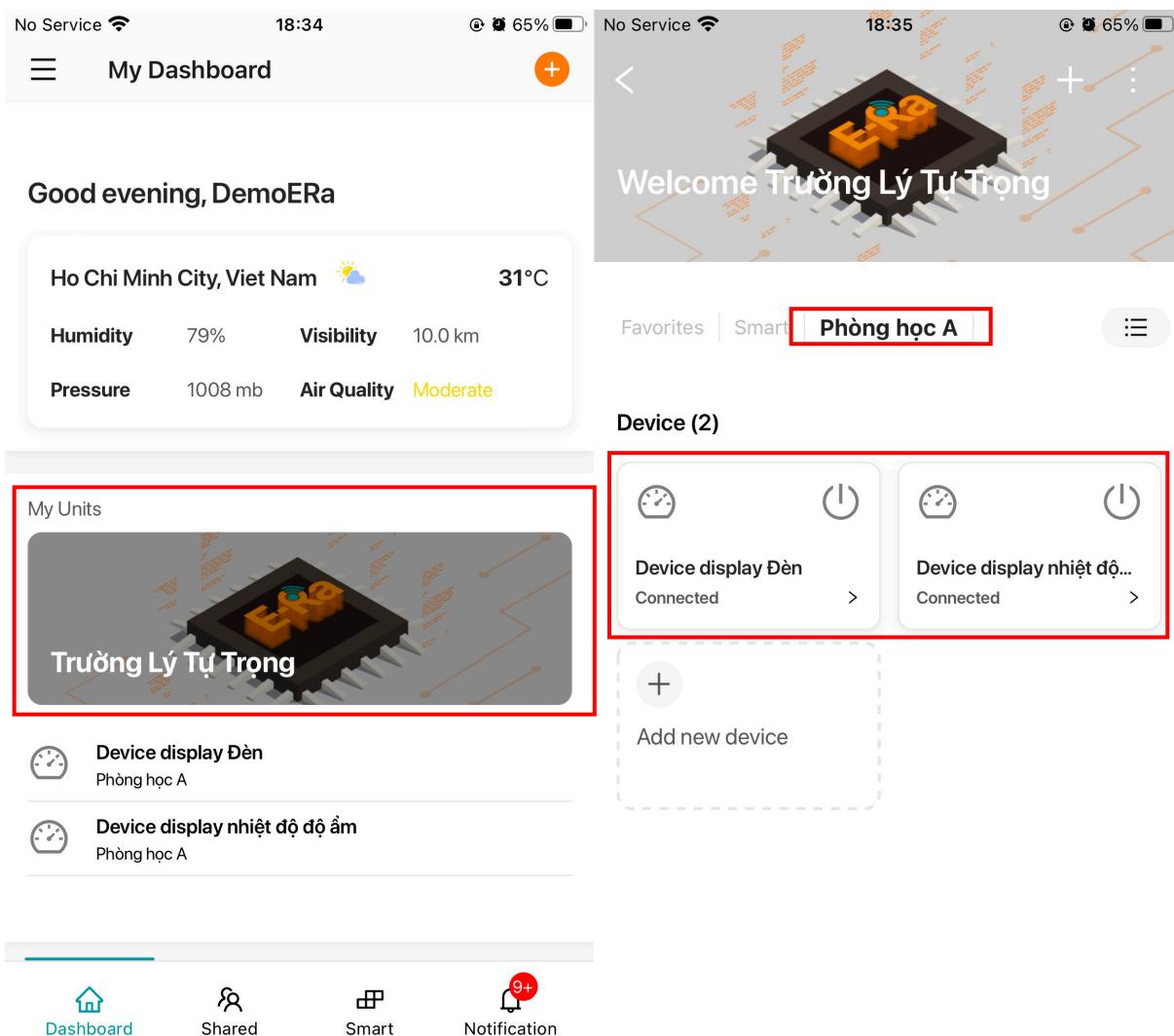
2.5.6 Đồng bộ mobile

E-Ra với tiêu chí tiện lợi cho người dùng, tất cả các cầu hình đều sẽ được đồng bộ lên mobile và được sắp xếp thông qua việc chia phòng Sub-unit và device display. Người dùng sẽ không cần cầu hình lại. Chúng ta cần download app E-Ra Eoh và đăng nhập vào đúng với tài khoản đã sử dụng cho web.

Chọn Unit Trường Lý Tự Trọng.

Chọn Sub-unit là Phòng học A.

Phòng học A sẽ có 2 Device display là Device display Đèn và Device display Nhiệt độ độ ẩm.



Mobifone

18:39

64% No Service

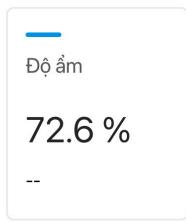
18:35

65%

**Device display nhiệt độ độ ẩm**

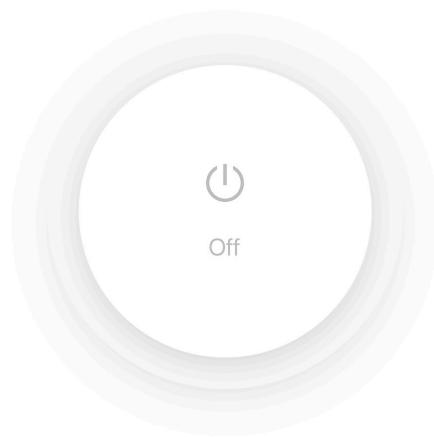
Connected via Internet

Last updated 34 seconds ago

**Device display Đèn**

Connected via Internet

Last updated 5 seconds ago



3. Kết luận

IoT tác động sâu rộng tới cuộc sống cũng như công việc của con người. IoT cho phép máy móc xử lý phần việc nặng nhọc, đảm nhiệm những nhiệm vụ nhảm chán và giúp cuộc sống trở nên lành mạnh, năng suất và thoái mái hơn.

IoT cung cấp cho các doanh nghiệp cái nhìn về thời gian mà hệ thống của họ thực sự hoạt động, cung cấp thông tin chi tiết về mọi thứ từ hiệu suất của máy móc đến chuỗi cung ứng và hoạt động hậu cần.

E-Ra là một nền tảng giúp cho việc kết nối một cách nhanh chóng và dễ dàng, qua tham đề “E-Ra trong lĩnh vực Smart Home” đã kết nối được 2 thiết bị đó là cảm biến nhiệt độ độ ẩm và thiết bị điều khiển đèn led một cách đơn giản hơn, mọi thứ đều có thể theo dõi và điều khiển trực tiếp qua web cũng như tiện lợi trong việc sử dụng trên điện thoại mà không cần cầu hình mọi thứ lại từ đầu.

Ngoài ra, Nền tảng IoT E-Ra còn giúp việc quản lý mọi thiết bị một cách hiệu quả thông qua việc phân chia Sub-unit cũng như có thể lựa chọn các Widget nào có thể hiển thị.

Hiện nay, IoT còn có mặt ở nhiều lĩnh vực khác nhau như y tế, công nghiệp hoặc nông trại, thành phố, vì thế E-Ra với nhiều tính năng sẵn có, nhiều phương thức giao tiếp khác nhau có thể đáp ứng cho các nhu cầu khác trong tương lai.

Tài liệu tham khảo

Các khái niệm, hướng dẫn và thực hành workshop từ E-Ra
<https://era-open-iot-platform.gitbook.io/documentation/>

Facebook Cộng đồng IoT E-Ra
<https://www.facebook.com/groups/567625788148920>.