TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

VÀ TRUYỀN THÔNG VIỆT – HÀN

**KHOA KỸ THUẬT MÁY TÍNH & ĐIỆN TỬ**



**BÁO CÁO**

**ĐỒ ÁN CƠ SỞ IV**

**ĐỀ TÀI:** Thiết kế và lập trình cho hệ thống

thông minh sử dụng ESP32 & Blynk

**Sinh viên thực hiện     : Thái Văn Hòa**

**Lớp : 21IR**

**Niên khoá : 2021 - 2026**

**Giảng viên hướng dẫn : ThS. Trần Thu Thủy**

***Đà Nẵng, ngày 10 tháng 12 năm 2023***

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

VÀ TRUYỀN THÔNG VIỆT – HÀN

**KHOA KỸ THUẬT MÁY TÍNH & ĐIỆN TỬ**



**BÁO CÁO**

**ĐỒ ÁN CƠ SỞ IV**

**ĐỀ TÀI:** Thiết kế và lập trình cho hệ thống

thông minh sử dụng ESP32 & Blynk

**Sinh viên thực hiện     : Thái Văn Hòa**

**Lớp : 21IR**

**Niên khoá : 2021 - 2026**

**Giảng viên hướng dẫn : ThS. Trần Thu Thủy**

***Đà Nẵng, ngày 10 tháng 12 năm 2023***

# NHẬN XÉT

**(Của Giảng viên hướng dẫn)**

....................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

Đà Nẵng,….. tháng … năm 2023

Giảng viên hướng dẫn

# LỜI CẢM ƠN

Lời đầu tiên cho phép em gửi lời cảm ơn tới các Thầy Cô giáo, các cán bộ công tác tại Trường ĐH Công Nghệ Thông Tin Và Truyền Thông Việt Hàn đã tạo mọi điều kiện giúp đỡ em trong thời gian xây dựng và hoàn thành báo cáo

Đặc biệt em xin bày tỏ lòng biết ơn sâu sắc tới cô **ThS.Trần Thu Thủy** giảng viên hướng dẫn môn Đồ Án Cơ Sở 4 đã tận tình giúp đỡ, chỉ bảo về nghiệp vụ và trực tiếp hướng dẫn em trong suốt quá trình hoàn thành báo cáo này.

Tuy nhiên do thời gian có hạn và cùng với nhiều nguyên nhân khác, mặc dù em đã nỗ lực hết mình xong đồ án của em vẫn còn mắc phải những thiếu sót và hạn chế. Em rất mong nhận được sự thông cảm và chỉ bảo của các Thầy Cô cùng tất cả các bạn.

Em xin chân thành cảm ơn!

***Đà Nẵng, 10 tháng 12 năm 2023***

*Sinh viên thực hiện*

Thái Văn Hòa

**MỤC LỤC**

[NHẬN XÉT 3](#_Toc152620190)

[LỜI CẢM ƠN 4](#_Toc152620191)

[DANH MỤC HÌNH VẼ 7](#_Toc152620192)

[DANH MỤC CÁC TỪ VIẾT TẮT 8](#_Toc152620193)

[MỞ ĐẦU 9](#_Toc152620194)

[CHƯƠNG 1 - TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc152620195)

[1.1Tổng quan 1](#_Toc152620196)

[1.1.1 Lý do chọn đề tài 1](#_Toc152620197)

[1.1.2 Mục tiêu và nội dung thực hiện đề tài 1](#_Toc152620198)

[1.1.3 Phương pháp nghiên cứu của đề tàia 2](#_Toc152620199)

[1.2 Kết quả và ý nghĩa đề tài 2](#_Toc152620200)

[1.2.1. Kết quả 2](#_Toc152620201)

[1.2.2. Ý nghĩa đề tài 2](#_Toc152620202)

[1.3 Công cụ 2](#_Toc152620203)

[1.3.1. Phần mềm lấp trình 2](#_Toc152620204)

[1.3.2. Blynk 3](#_Toc152620205)

[CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT 4](#_Toc152620206)

[2.1 Giới thiệu chung về ESP32 4](#_Toc152620207)

[2.1.1 Cấu Trúc của ESP32 4](#_Toc152620208)

[2.1.2 Nguyên Lý Hoạt Động 8](#_Toc152620209)

[2.1.3 Ứng Dụng của ESP32 8](#_Toc152620210)

[2.1.4 Lợi Ích của Việc Sử Dụng ESP32 trong Hệ Thống Thông Minh 9](#_Toc152620211)

[2.2 Blynk cloud 9](#_Toc152620212)

[2.2.1 Blynk là gì? 9](#_Toc152620213)

[2.3 Các công nghệ liên quan khác 10](#_Toc152620214)

[CHƯƠNG 3 – THIẾT KẾ & LẬP TRÌNH HỆ THỐNG 11](#_Toc152620215)

[3.1Các Module cần có 11](#_Toc152620216)

[3.1.1. ESP32 11](#_Toc152620217)

[3.1.2. Cảm biến nhiệt độ và độ ẩm DHT11 12](#_Toc152620218)

[3.1.3. Màn hình OLED I2C 0,96 inch 13](#_Toc152620219)

[3.1.4. Relay 2 kênh 5V 14](#_Toc152620220)

[3.2Thiết kế hệ thống và ráp mạch và lập trình 15](#_Toc152620221)

[3.3Sản phẩm hoàn thiện 17](#_Toc152620222)

[CHƯƠNG 4 – KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 19](#_Toc152620223)

[4.1 Kết quả đạt được 19](#_Toc152620224)

[4.2 Hạn chế 19](#_Toc152620225)

[4.3 Hướng phát triển 19](#_Toc152620226)

[TÀI LIỆU THAM KHẢO 21](#_Toc152620227)

# DANH MỤC HÌNH VẼ

[Hình 1. 1 Phần mềm Arduno IDE 3](#_Toc152624597)

[Hình 1. 2 Blynk cloud 3](#_Toc152624598)

[Hình 2. 1 Datasheet ESP32 4](#_Toc152624599)

[Hình 2. 2 Các loại ESP32 8](#_Toc152624600)

[Hình 2. 3 Apps blynk 9](#_Toc152624601)

[Hình 3. 1 ESP32 11](#_Toc152624602)

[Hình 3. 2 Cảm biến nhiệt độ và độ ẩm DHT11 13](#_Toc152624603)

[Hình 3. 3 Màn hình OLED I2C 13](#_Toc152624604)

[Hình 3. 4 Module relay 2 chaner 14](#_Toc152624605)

[Hình 3. 5 Mô phỏng hệ thống 15](#_Toc152624606)

[Hình 3. 6 Sản phẩm hoàn thiện 17](#_Toc152624607)

[Hình 3. 7 Demo sản phẩm 17](#_Toc152624608)

[Hình 3. 8 Web dashboard 18](#_Toc152624609)

[Hình 3. 9 Apps dashboard 18](#_Toc152624610)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| STT | Nội dung | Viết tắt |
| 1 | Espressif System Platform | ESP |
| 2 | Internet of Think | IoT |

# MỞ ĐẦU

Trong thời đại số hóa ngày nay, việc sử dụng công nghệ để cải thiện chất lượng cuộc sống đã trở thành một xu hướng không thể tránh khỏi. Đặc biệt, việc tạo ra các hệ thống thông minh, nhờ vào sự phát triển của các công nghệ như ESP32 và Blynk, đã mở ra những cơ hội mới cho chúng ta để tạo ra những giải pháp tối ưu cho cuộc sống hàng ngày.

ESP32, một chip có khả năng kết nối không dây và nhiều tính năng khác, đã trở thành một công cụ quan trọng trong việc phát triển các hệ thống thông minh. Bên cạnh đó, Blynk, một nền tảng cho phép người dùng tạo ra các ứng dụng IoT một cách dễ dàng và nhanh chóng, cũng đã đóng góp một phần quan trọng trong việc tạo ra những giải pháp IoT hiệu quả.

Đề tài sẽ tập trung vào việc khám phá sức mạnh của ESP32 và Blynk, cũng như cách chúng có thể được sử dụng để tạo ra các hệ thống thông minh. Chúng ta sẽ đi sâu vào việc tìm hiểu về cách hoạt động của ESP32 và Blynk, cũng như cách chúng có thể được lập trình để tạo ra các hệ thống thông minh.

Kết quả , chúng ta sẽ áp dụng những kiến thức đã học để thiết kế và lập trình một hệ thống thông minh cụ thể, minh họa cho khả năng và tiềm năng của ESP32 và Blynk trong việc tạo ra các giải pháp IoT hiệu quả. Thông qua đề tài này, ta sẽ có thể mở rộng hiểu biết về công nghệ và tạo ra những giải pháp tốt nhất cho cuộc sống của chúng ta.

# CHƯƠNG 1 - TỔNG QUAN VỀ ĐỀ TÀI

## Tổng quan

### Lý do chọn đề tài

Hệ thống thông minh là một xu hướng phát triển mạnh mẽ trong nhiều lĩnh vực của đời sống xã hội. Hệ thống thông minh có khả năng tự động hóa các nhiệm vụ, xử lý thông tin và đưa ra quyết định một cách thông minh.

ESP32 và Blynk là hai công nghệ hiện đại có thể được sử dụng để tạo ra các hệ thống thông minh với chi phí thấp và dễ triển khai. ESP32 là một vi điều khiển mạnh mẽ, có khả năng kết nối Wi-Fi và Bluetooth. Blynk là một nền tảng phát triển ứng dụng IoT dựa trên đám mây.

Đề tài này có tính ứng dụng cao. Các hệ thống thông minh sử dụng ESP32 và Blynk có thể được ứng dụng trong nhiều lĩnh vực khác nhau, bao gồm:

- Quản lý nhà thông minh

- An ninh thông minh

- Nông nghiệp thông minh

- Công nghiệp thông minh

Với những lý do trên, đề tài "Thiết kế và lập trình cho hệ thống thông minh sử dụng ESP32 & Blynk" là một đề tài có ý nghĩa khoa học và thực tiễn. Đề tài này sẽ giúp sinh viên hiểu rõ hơn về hệ thống thông minh, ESP32 và Blynk. Ngoài ra, đề tài này cũng sẽ giúp sinh viên có cơ hội thực hành thiết kế và lập trình các hệ thống thông minh sử dụng ESP32 và Blynk.

### Mục tiêu và nội dung thực hiện đề tài

* Hiểu rõ về hệ thống thông minh, ESP32 và Blynk.
* Thiết kế một hệ thống thông minh sử dụng ESP32 và Blynk.
* Lapp trình hệ thống thing minh sử dụng ESP32 và Blynk.
* Thực nghiệm và đánh giá hệ thống thông minh sử dụng ESP32 và Blynk.

### Phương pháp nghiên cứu của đề tàia

* Nghiên cứu tài liệu: Nghiên cứu các tài liệu liên quan đến hệ thống thông minh, ESP32 và Blynk. Các tài liệu này có thể bao gồm sách, bài báo, tài liệu kỹ thuật,...
* Thực hành: Thực hành thiết kế và lập trình các hệ thống thông minh sử dụng ESP32 và Blynk. Sinh viên có thể sử dụng các phần mềm mô phỏng hoặc các thiết bị thực tế để thực hành.
* Áp dụng kiến thức đã học: Áp dụng các kiến thức đã học về hệ thống thông minh, ESP32 và Blynk để thiết kế và lập trình hệ thống thông minh.

### 1.2 Kết quả và ý nghĩa đề tài

### 1.2.1. Kết quả

* Hiểu rõ về hệ thống thông minh, ESP32 và Blynk.
* Thiết kế được một hệ thống thông minh sử dụng ESP32 và Blynk.
* Lập trình được hệ thống thông minh sử dụng ESP32 và Blynk.
* Thực nghiệm và đánh giá được hệ thống thông minh sử dụng ESP32 và Blynk.

### 1.2.2. Ý nghĩa đề tài

* Giúp sinh viên hiểu rõ hơn về hệ thống thông minh, ESP32 và Blynk.
* Giúp sinh viên có cơ hội thực hành thiết kế và lập trình các hệ thống thông minh sử dụng ESP32 và Blynk.
* Đề tài có tính ứng dụng cao, có thể được sử dụng trong nhiều lĩnh vực khác nhau

### 1.3 Công cụ

### 1.3.1. Phần mềm lấp trình

Arduino IDE: là một phần mềm với một mã nguồn mở, được sử dụng chủ yếu để viết và biên dịch mã vào module Arduino. Nó bao gồm phần cứng và phần mềm. Phần cứng chứa đến 300,000 board mạch được thiết kế sẵn với các cảm biến, linh kiện. Phần mềm giúp bạn có thể sử dụng các cảm biến, linh kiện ấy của Arduino một cách linh hoạt phù hợp với mục đích sử dụng.



Hình 1. 1 Phần mềm Arduno IDE

### 1.3.2. Blynk

Blynk là một nền tảng với các ứng dụng iOS và Android để điều khiển Arduino, Raspberry Pi và các ứng dụng tương tự qua Internet.

Nó là một bảng điều khiển kỹ thuật số nhờ đó bạn có thể xây dựng giao diện đồ họa cho dự án của mình bằng cách kéo và thả các widget.

Việc thiết lập mọi thứ rất đơn giản và bạn sẽ bắt đầu sau chưa đầy 5 phút.

Blynk không bị ràng buộc với một số bo hoặc shield cụ thể. Thay vào đó, nó hỗ trợ phần cứng mà bạn lựa chọn. Cho dù Arduino hoặc Raspberry Pi của bạn được liên kết với Internet qua Wi-Fi, Ethernet hoặc chip ESP8266, Blynk sẽ giúp bạn online và sẵn sàng cho IoT.



Hình 1. 2 Blynk cloud

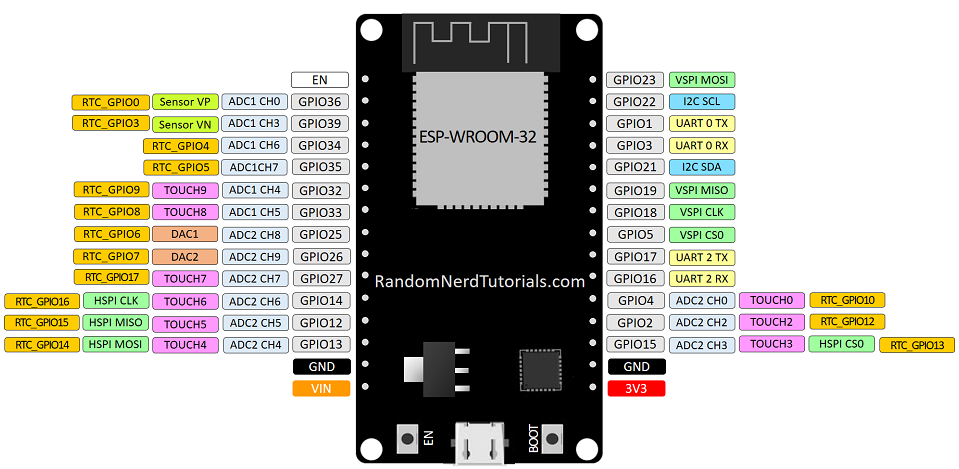
# CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT

## 2.1 Giới thiệu chung về ESP32

### 2.1.1 Cấu Trúc của ESP32

ESP32 bao gồm nhiều thành phần chính, bao gồm:

* SoC (System on a Chip): Nền tảng tích hợp nhiều chức năng trên một chip, bao gồm vi xử lý, bộ nhớ, module WiFi và Bluetooth, GPIO (General Purpose Input/Output), các cổng giao tiếp khác như UART, SPI, I2C, v.v.
* Bộ xử lý kép (Dual-core): ESP32 có hai nhân xử lý, giúp chia công việc và tối ưu hiệu suất.
* WiFi và Bluetooth: Cung cấp khả năng kết nối không dây với mạng WiFi và các thiết bị Bluetooth khác



Hình 2. 1 Datasheet ESP32

Mô tả các chân ESP32

Chân Input Only

GPIO từ 34 đến 39 là GPI – chân chỉ đầu vào. Các chân này không có điện trở kéo lên hoặc kéo xuống bên trong. Chúng không thể được sử dụng làm đầu ra, vì vậy chỉ sử dụng các chân này làm đầu vào:

* + GPIO 34
  + GPIO 35
  + GPIO 36
  + GPIO 39

Chân tích hợp Flash trên ESP32

GPIO 6 đến GPIO 11 dùng để kết nối Flash SPI, không khuyến khích sử dụng trong các ứng dụng khác

* + GPIO 6 (SCK/CLK)
  + GPIO 7 (SDO/SD0)
  + GPIO 8 (SDI/SD1)
  + GPIO 9 (SHD/SD2)
  + GPIO 10 (SWP/SD3)
  + GPIO 11 (CSC/CMD)

Chân cám biến điện dung

Các chân ESP32 này có chức năng như 1 nút nhấn cảm ứng, có thể phát hiện sự thay đổi về điện áp cảm ứng trên chân.

Các cảm biến cảm ứng bên trong đó được kết nối với các GPIO sau:

* T0 (GPIO 4)
* T1 (GPIO 0)
* T2 (GPIO 2)
* T3 (GPIO 15)
* T4 (GPIO 13)
* T5 (GPIO 12)
* T6 (GPIO 14)
* T7 (GPIO 27)
* T8 (GPIO 33)
* T9 (GPIO 32)

Analog to Digital Converter (ADC)

ESP32 có các kênh đầu vào ADC 18 x 12 bit (trong khi ESP8266 chỉ có ADC 1x 10 bit). Đây là các GPIO có thể được sử dụng làm ADC và các kênh tương ứng:

* + ADC1\_CH0 (GPIO 36)
  + ADC1\_CH1 (GPIO 37)
  + ADC1\_CH2 (GPIO 38)
  + ADC1\_CH3 (GPIO 39)
  + ADC1\_CH4 (GPIO 32)
  + ADC1\_CH5 (GPIO 33)
  + ADC1\_CH6 (GPIO 34)
  + ADC1\_CH7 (GPIO 35)
  + ADC2\_CH0 (GPIO 4)
  + ADC2\_CH1 (GPIO 0)
  + ADC2\_CH2 (GPIO 2)
  + ADC2\_CH3 (GPIO 15)
  + ADC2\_CH4 (GPIO 13)
  + ADC2\_CH5 (GPIO 12)
  + ADC2\_CH6 (GPIO 14)
  + ADC2\_CH7 (GPIO 27)
  + ADC2\_CH8 (GPIO 25)
  + ADC2\_CH9 (GPIO 26)

Các kênh đầu vào ADC có độ phân giải 12 bit. Điều này có nghĩa là bạn có thể nhận được các số đọc tương tự từ 0 đến 4095, trong đó 0 tương ứng với 0V và 4095 đến 3,3V. Bạn cũng có thể lập trình độ phân giải của các kênh của mình trên code.

Digital to Analog Converter (DAC)

Có các kênh DAC 2 x 8 bit trên ESP32 để chuyển đổi tín hiệu kỹ thuật số thành đầu ra tín hiệu điện áp tương tự. Các kênh này chỉ có độ phân giải 8 bit, nghĩa là có giá trị từ 0 – 255 tương ứng với 0 – 3.3V

Đây là các kênh DAC:

* + DAC1 (GPIO25)
  + DAC2 (GPIO26)

Các chân thời gian thực RTC

Các chân này có tác dụng đánh thức ESP32 khi trong chế độ Low Power Mode. Sử dụng như 1 chân ngắt ngoài.

Các chân RTC:

* + RTC\_GPIO0 (GPIO36)
  + RTC\_GPIO3 (GPIO39)
  + RTC\_GPIO4 (GPIO34)
  + RTC\_GPIO5 (GPIO35)
  + RTC\_GPIO6 (GPIO25)
  + RTC\_GPIO7 (GPIO26)
  + RTC\_GPIO8 (GPIO33)
  + RTC\_GPIO9 (GPIO32)
  + RTC\_GPIO10 (GPIO4)
  + RTC\_GPIO11 (GPIO0)
  + RTC\_GPIO12 (GPIO2)
  + RTC\_GPIO13 (GPIO15)
  + RTC\_GPIO14 (GPIO13)
  + RTC\_GPIO15 (GPIO12)
  + RTC\_GPIO16 (GPIO14)
  + RTC\_GPIO17 (GPIO27)

Chân PWM

ESP32 LED PWM có 16 kênh độc lập có thể được định cấu hình để tạo tín hiệu PWM với các thuộc tính khác nhau. Tất cả các chân có thể hoạt động như đầu ra đều có thể được sử dụng làm chân PWM (GPIO từ 34 đến 39 không thể tạo PWM).

Để xuất PWM, bạn cần xác định các thông số này trong code:

* + Frequency – tần số
  + Duty cycle
  + Kênh PWM
  + Chân GPIO nơi bạn muốn xuất tín hiệu

Chân I2C

ESP32 có hai kênh I2C và bất kỳ chân nào cũng có thể được đặt làm SDA hoặc SCL. Khi sử dụng ESP32 với Arduino IDE, các chân I2C mặc định là:

* + GPIO 21 (SDA)
  + GPIO 22 (SCL)

### 2.1.2 Nguyên Lý Hoạt Động

* WiFi: ESP32 sử dụng giao thức WiFi để kết nối với mạng không dây, cho phép truyền dữ liệu giữa thiết bị và các điểm truy cập WiFi.
* Bluetooth: ESP32 hỗ trợ Bluetooth Classic và Bluetooth Low Energy (BLE), mở rộng khả năng kết nối với các thiết bị khác như điện thoại di động, cảm biến, và thiết bị thông minh khác.
* GPIO và Cổng Giao Tiếp: GPIO cho phép ESP32 tương tác với nhiều loại thiết bị ngoại vi thông qua các giao tiếp như UART, SPI, I2C

### 2.1.3 Ứng Dụng của ESP32

* IoT (Internet of Things): ESP32 thường được sử dụng trong các ứng dụng IoT như đọc dữ liệu từ cảm biến và truyền tải chúng qua WiFi hoặc Bluetooth.
* Điều Khiển Thiết Bị: ESP32 có thể được sử dụng để điều khiển và giám sát thiết bị qua mạng không dây.
* Dự Án Mô Hình: Được ứng dụng rộng rãi trong các dự án mô hình và prototype nhờ tính linh hoạt và hiệu suất.



Hình 2. 2 Các loại ESP32

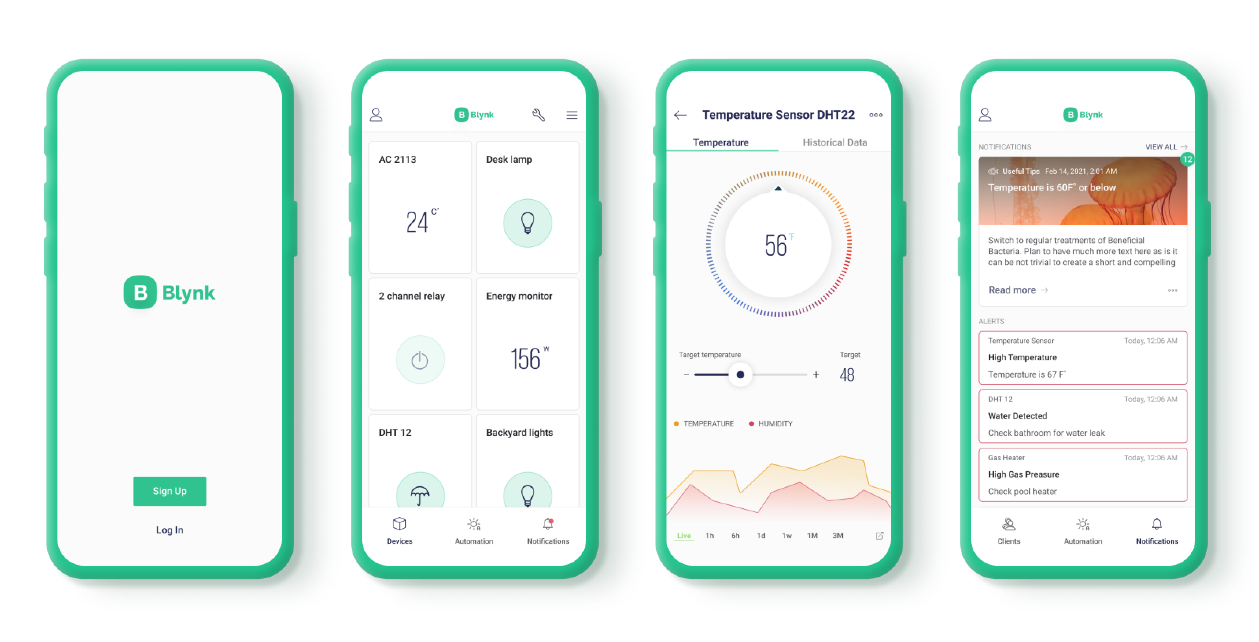
### 2.1.4 Lợi Ích của Việc Sử Dụng ESP32 trong Hệ Thống Thông Minh

* Tiết Kiệm Năng Lượng: ESP32 được thiết kế để tiêu thụ ít năng lượng, phù hợp cho các ứng dụng di động và pin-kéo dự án.
* Kết Nối Đa Dạng: Với khả năng kết nối WiFi và Bluetooth, ESP32 có thể tương tác với nhiều loại thiết bị khác nhau.
* Hiệu Suất Cao: Bộ xử lý kép giúp tối ưu hiệu suất và xử lý đa nhiệm một cách hiệu quả.
* Phát Triển Nhanh Chóng: Sự hỗ trợ mạnh mẽ từ cộng đồng và sự linh hoạt của ESP32 giúp phát triển nhanh chóng các ứng dụng và dự án.

## 2.2 Blynk cloud

### 2.2.1 Blynk là gì?

Blynk là một nền tảng IoT (Internet of Things) cho phép người dùng tạo ứng dụng di động để giám sát và điều khiển các thiết bị IoT. Nền tảng này cung cấp một giao diện đồ họa trực quan để tương tác với thiết bị thông qua điện thoại di động hoặc máy tính bảng một cách dễ dàng.



Hình 2. 3 Apps blynk

## 2.3 Các công nghệ liên quan khác

Một số công nghệ quan trọng được sử dụng trong IOT

* LoRa (Long Range): là một công nghệ truyền thông không dây được thiết kế để cung cấp phạm vi kết nối rộng lớn và tiêu thụ năng lượng thấp, thích hợp cho các ứng dụng IoT yêu cầu truyền thông xa và pin-kéo.
* NB-IoT (Narrowband IoT): là một tiêu chuẩn kết nối di động dựa trên mạng di động LTE, được tối ưu hóa cho các thiết bị IoT yêu cầu băng thông thấp và tiêu thụ năng lượng thấp.
* Zigbee và Z-Wave: là hai công nghệ kết nối không dây được thiết kế đặc biệt cho các ứng dụng điều khiển nhà thông minh. Chúng hỗ trợ mạng lưới cho việc kết nối và điều khiển các thiết bị trong ngôi nhà.
* MQTT (Message Queuing Telemetry Transport): là một giao thức truyền thông nhẹ và linh hoạt, thường được sử dụng trong các ứng dụng IoT để truyền tải dữ liệu giữa thiết bị và máy chủ.
* AWS IoT và Azure IoT: là hai nền tảng lớn của Amazon và Microsoft, cung cấp các dịch vụ đám mây để quản lý, giám sát và triển khai các ứng dụng IoT.
* Sigfox: là một mạng truyền thông toàn cầu dựa trên công nghệ kết nối không dây, thiết kế để hỗ trợ các ứng dụng IoT yêu cầu gửi lượng dữ liệu nhỏ như vị trí, nhiệt độ, và cảm biến đơn giản.
* 6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks): là một tiêu chuẩn cho việc truyền tải giao thức IPv6 qua mạng không dây có công suất thấp, thích hợp cho các thiết bị IoT với tài nguyên hạn chế.
* Thread: là một giao thức mạng lưới có khả năng tự sắp xếp, thiết kế để kết nối các thiết bị IoT trong ngôi nhà thông minh và các môi trường tương tự.
* Edge Computing: giúp xử lý dữ liệu ngay tại nguồn, giảm độ trễ và giảm áp lực trên mạng, đặc biệt quan trọng trong các hệ thống IoT có nhiều dữ liệu sinh ra từ nhiều thiết bị.

# CHƯƠNG 3 – THIẾT KẾ & LẬP TRÌNH HỆ THỐNG

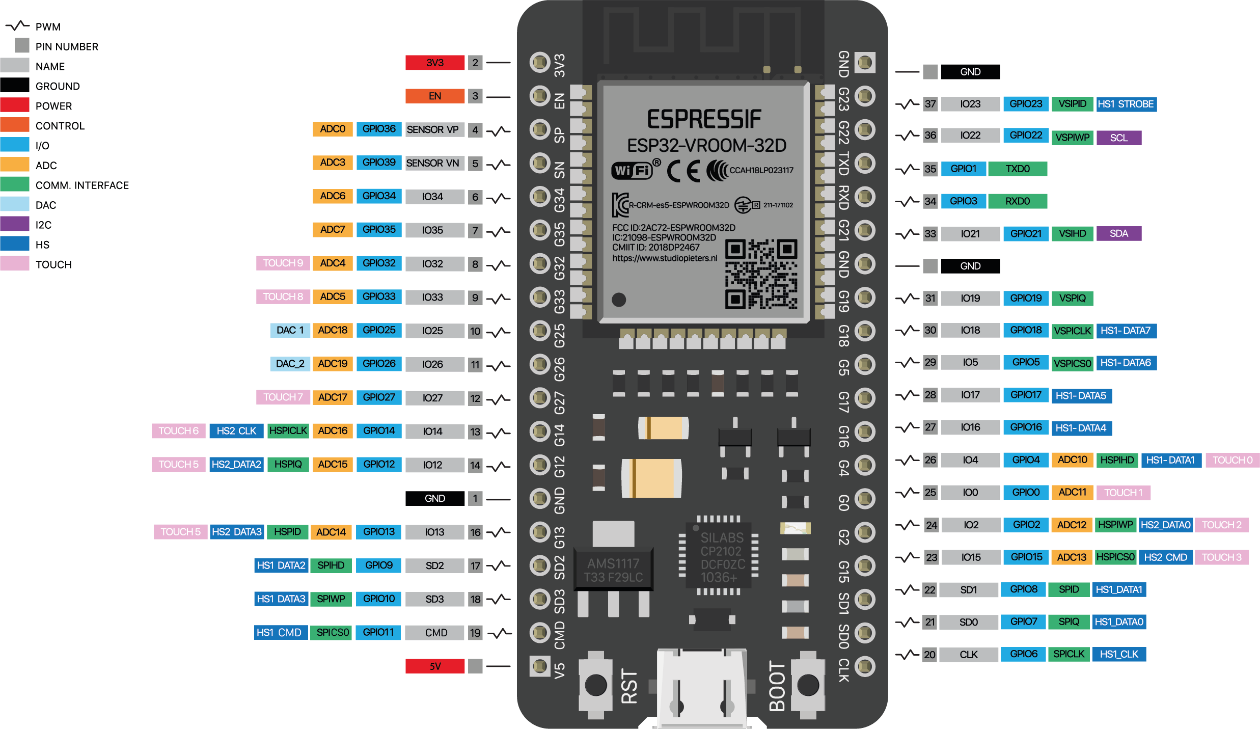
* 1. Các Module cần có

### 3.1.1. ESP32

ESP32 là một bộ vi điều khiển thuộc danh mục vi điều khiển trên chip công suất thấp và tiết kiệm chi phí. Hầu hết tất cả các biến thể ESP32 đều tích hợp Bluetooth và Wi-Fi chế độ kép, làm cho nó có tính linh hoạt cao, mạnh mẽ và đáng tin cậy cho nhiều ứng dụng.

Nó là sự kế thừa của vi điều khiển NodeMCU ESP8266 phổ biến và cung cấp hiệu suất và tính năng tốt hơn. Bộ vi điều khiển ESP32 được sản xuất bởi Espressif Systems và được sử dụng rộng rãi trong nhiều ứng dụng khác nhau như IoT, robot và tự động hóa.

ESP32 cũng được thiết kế để tiêu thụ điện năng thấp, lý tưởng cho các ứng dụng chạy bằng pin. Nó có hệ thống quản lý năng lượng cho phép nó hoạt động ở chế độ ngủ và chỉ thức dậy khi cần thiết, điều này có thể kéo dài tuổi thọ pin rất nhiều.



Hình 3. 1 ESP32

### 3.1.2. Cảm biến nhiệt độ và độ ẩm DHT11

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

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

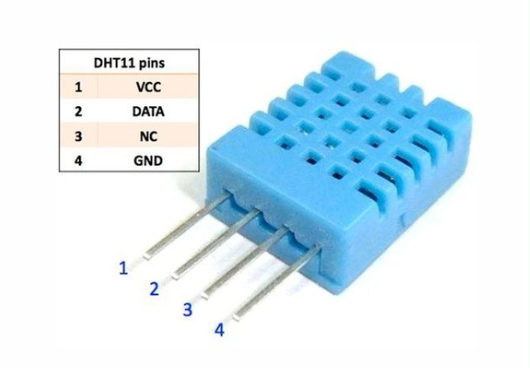
Cấu tạo cảm biến nhiệt độ độ ẩm DHT11

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

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

**Sơ đồ chân DHT11**

|  |  |  |
| --- | --- | --- |
| **Số chân** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Nguồn 3.5V đến 5.5V |
| 2 | Data | Đầu ra cả nhiệt độ và độ ẩm thông qua dữ liệu nối tiếp |
| 3 | NC | Không có kết nối và do đó không sử dụng |
| 4 | Ground | Nối đất |



Hình 3. 2 Cảm biến nhiệt độ và độ ẩm DHT11

### 3.1.3. Màn hình OLED I2C 0,96 inch

màn hình Oled 0.96inch sử dụng giao tiếp I2C cho chất lượng đường truyền ổn định và rất dễ giao tiếp chỉ với 2 chân GPIO.

Sơ đồ chân:

- VCC: là nguồn cấp cho màn hình có thể là 3.3V hoặc 5V

- GND: chân nối đất

- SCL: chân xung clock

- SDA: chân dữ liệu nối tiếp

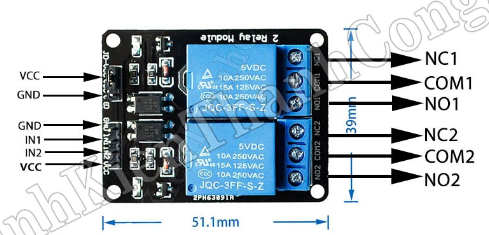
Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

Hình 3. 3 Màn hình OLED I2C

### 3.1.4. Relay 2 kênh 5V

Module Relay 2-Channel 5V được dùng nhiều trong các ứng dụng đóng ngắt các thiết bị tiêu thụ dòng điện lớn (<10A). Module có thể đóng ngắt cùng lúc hai kênh bằng tín hiệu điều khiển ( với mức điện áp 3V3 hoặc 5V) từ các vi điều khiển khác nhau như: Arduino, 8051, AVR, PIC, DSP, ARM, ARM, MSP430, logic TTL, đồng thời module được cách ly bằng optocoupler giúp bảo vệ tốt hơn cho các vi điều khiển.



Hình 3. 4 Module relay 2 chaner

Module được kết nối với các board điều khiển bằng 4 chân header như sau:

* **VCC** cung cấp nguồn cho các opto.
* **GND** kết nối với GND của board điều khiển.
* **IN1** và **IN2** dùng để điều khiển relay 1 và relay 2, tích cực mức thấp

Ngoài ra còn một 3 chân header được dùng để cấp nguồn cho relay, header này sẽ có một jumper dùng để kết nối chân VCC với chân RY\_VCC mục đích dùng chung nguồn VCC (5V) từ header 4 chân cho relay, thông thường jumper được nối lại với nhau. Nếu như muốn cách ly tín hiệu điều khiển với nguồn cấp cho relay thì có thể bỏ jumper này ra và cấp nguồn riêng 5V cho chân RY\_VCC.

**Thông số kỹ thuật**

* Đóng ngắt được dòng điện cao: AC250V 10A, DC30V 10A
* 2 led báo trạng thái relay
* Điện áp điều khiển: 5V
* Mạch cách ly bằng opto

## Thiết kế hệ thống và ráp mạch và lập trình

Ảnh có chứa văn bản, biểu đồ, Kỹ thuật điện, ảnh chụp màn hình

Mô tả được tạo tự động

Hình 3. 5 Mô phỏng hệ thống

**Sơ Đồ Mạch Kết Nối:**

1. Kết Nối Màn Hình OLED với ESP32:

* Sử dụng chân SDA của màn hình OLED kết nối với chân D21 của ESP32.
* Sử dụng chân SCL của màn hình OLED kết nối với chân D22 của ESP32.
* Hai chân GND và VCC của màn hình OLED được kết nối đến chân nguồn 3,3V của ESP32.

2. Kết Nối Cảm Biến DHT11 với ESP32:

* Sử dụng chân D23 của ESP32 để kết nối với chân data của cảm biến DHT11.

3. Kết Nối Relay Hai Kênh với ESP32:

* Sử dụng chân D18 của ESP32 để kết nối với một kênh của relay.
* Sử dụng chân D19 của ESP32 để kết nối với kênh còn lại của relay.
* Chân VCC của relay được kết nối với chân Vin của ESP32 cấp nguồn 5V.
* Kết nối chân đất của relay với chân GND của ESP32.

4. Kết Nối Nút Giao Diện với ESP32:

* Sử dụng chân D25 của ESP32 để kết nối với một nút trên giao diện.
* Sử dụng chân D26 của ESP32 để kết nối với một nút trên giao diện.
* Sử dụng chân D27 của ESP32 để kết nối với một nút trên giao diện.

5. Kết Nối Nguồn và Đất:

* Chân GND của (OLED, DHT11, Relay, Nút) được kết nối với chân GND của ESP32.
* Chân VCC của (OLED, DHT11, Relay, Nút) được kết nối với chân 3,3V của ESP32.
* Kết nối chân VCC và GND vào chân của ESP32

**Lập trình hệ thống**

* + 1. **Thư viện Sử Dụng**:
* WiFi.h: Kết nối thiết bị vào mạng WiFi.
* BlynkSimpleEsp32.h: Kích hoạt giao tiếp với dịch vụ và ứng dụng Blynk.
* DHT.h: Đọc dữ liệu từ cảm biến nhiệt độ và độ ẩm DHT11.
* Preferences.h: Lưu trữ dữ liệu liên tục, đảm bảo giữ giá trị ngay cả khi thiết bị tắt nguồn.
* AceButton.h: Hỗ trợ việc sử dụng nút và định nghĩa sự kiện nhấn nút.
* Wire.h: Giao tiếp với màn hình OLED.
* Adafruit\_GFX.h và Adafruit\_SSD1306.h: Các hàm điều khiển cho màn hình OLED.
  + 1. **Cấu Hình và Kết Nối:**
* WiFi: Hệ thống kết nối vào mạng WiFi thông qua tên và mật khẩu.
* char ssid[] = "";
* char pass[] = "";
* Blynk Auth Token: Để kết nối với ứng dụng Blynk
* #define BLYNK\_AUTH\_TOKEN"sglbMgWeSFAvnxRg4UTUKUMTBSHrrY\_t"
* Template ID và Template Name của Blynk.Cloud:
* #define BLYNK\_TEMPLATE\_ID "TMPL6Hvve0kKA"
* #define BLYNK\_TEMPLATE\_NAME "IOT controller ESP32"

## Sản phẩm hoàn thiện

Ảnh có chứa Dây điện, đồ điện tử, Kỹ thuật điện, dây cáp

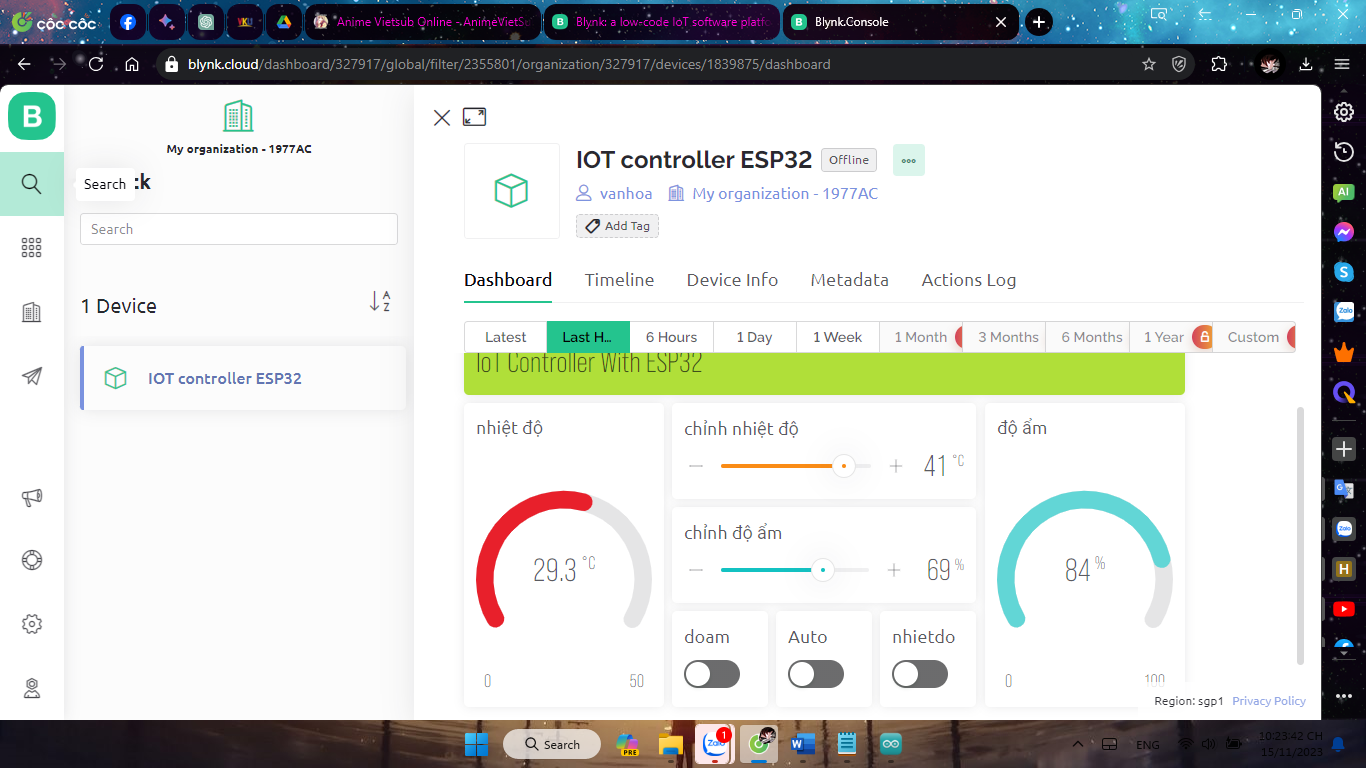
Mô tả được tạo tự động

Hình 3. 6 Sản phẩm hoàn thiện

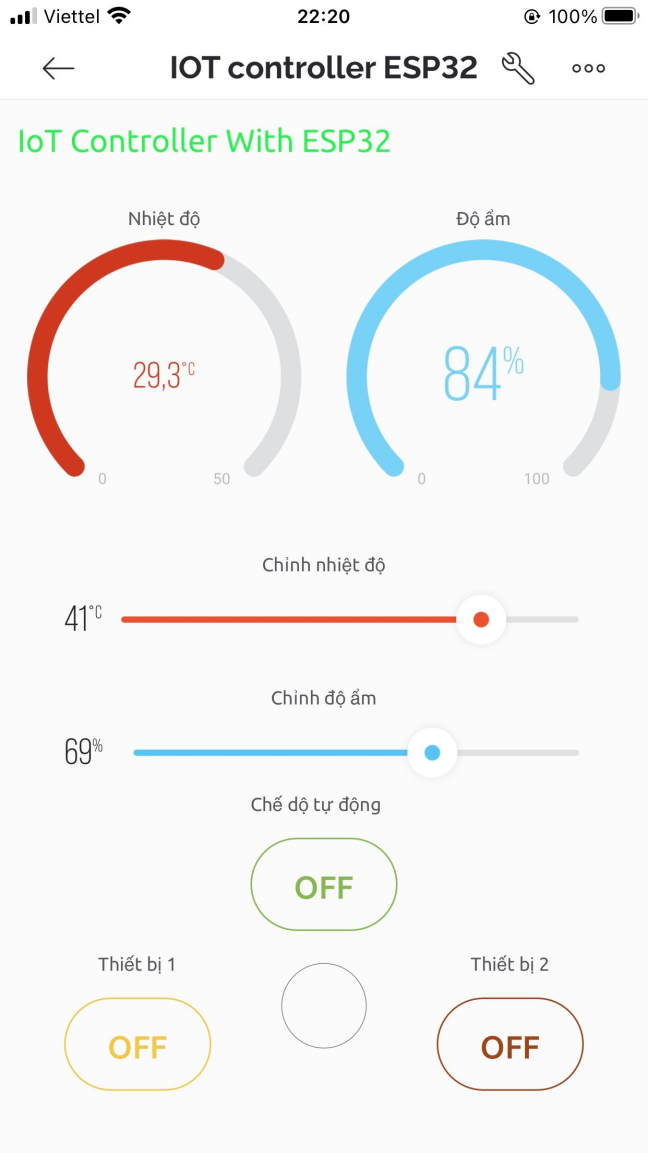
Ảnh có chứa đồ điện tử, Dây điện, dây cáp, Kỹ thuật điện

Mô tả được tạo tự động

Hình 3. 7 Demo sản phẩm



Hình 3. 8 Web dashboard



Hình 3. 9 Apps dashboard

# CHƯƠNG 4 – KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 4.1 Kết quả đạt được

* Dự án đã đạt được mục tiêu ban đầu của việc xây dựng một hệ thống thông minh sử dụng ESP32 và Blynk. Điều này bao gồm việc kiểm soát nhiệt độ và độ ẩm tự động, tích hợp công tắc và màn hình OLED để theo dõi thông tin.
* Quá trình lập trình và triển khai cho dự án đã là cơ hội tuyệt vời để học và áp dụng kiến thức về lập trình ESP32, sử dụng thư viện Blynk, và tương tác với các linh kiện IoT khác nhau.
* Mãnguồn :

https://github.com/hoathaivan20062002/ESP32withBlynk.git

## 4.2 Hạn chế

* Gặp phải một số thách thức như ổn định kết nối Blynk và xử lý sự cố của cảm biến. Đã giải quyết bằng cách thực hiện các kiểm tra liên tục và xử lý lỗi thông qua việc sử dụng các hàm chuyển đổi trạng thái.

## 4.3 Hướng phát triển

* Tích Hợp Nhiều Cảm Biến:
* Mở rộng chức năng cảm biến bằng cách tích hợp các cảm biến khác như cảm biến khí CO2, ánh sáng, hoặc cảm biến chất lượng không khí để có cái nhìn toàn diện về môi trường.
* Tối Ưu Hóa Giao Diện Người Dùng:
* Nâng cao trải nghiệm người dùng bằng cách phát triển một giao diện người dùng trực quan và dễ sử dụng trên ứng dụng Blynk, bao gồm biểu đồ, thông báo và chức năng điều khiển nhanh.
* Bảo Mật Tăng Cường:
* Nâng cấp bảo mật thông qua mã hóa dữ liệu truyền và lưu trữ, xác thực hai yếu tố, và giải pháp bảo vệ khác để ngăn chặn truy cập trái phép.
* Tương Tác Đa Thiết Bị:
* Hỗ trợ tương tác đa thiết bị, cho phép người dùng kiểm soát hệ thống từ nhiều thiết bị khác nhau, từ điện thoại di động đến máy tính bảng.
* Quản Lý Năng Lượng:
* Phát triển tính năng quản lý năng lượng để giảm tiêu thụ năng lượng và tăng hiệu suất của hệ thống.
* Giao Thức Kết Nối Linh Hoạt:
* Hỗ trợ nhiều giao thức kết nối như MQTT để tương tác với nhiều hệ thống IoT và các nền tảng khác.
* Tích Hợp Thêm Chức Năng Mở Rộng:
* Cho phép người dùng mở rộng hệ thống thông qua các cổng mở rộng, chẳng hạn như kết nối với các thiết bị thông minh khác.
* Nghiên Cứu Và Áp Dụng Công Nghệ Mới:
* Liên tục theo dõi và áp dụng các công nghệ mới như 5G, LoRaWAN, để cải thiện kết nối và hiệu suất.
* Thực Hiện Kiểm Soát Từ Xa:
* Phát triển khả năng kiểm soát hệ thống từ xa, giúp người dùng có thể kiểm soát và giám sát từ xa mọi lúc, mọi nơi.
* Tối Ưu Hóa Hiệu Năng:
* Nghiên cứu và triển khai cách tối ưu hóa hiệu suất, đặc biệt là trong việc tiêu thụ năng lượng để làm cho hệ thống trở nên bền vững và hiệu quả hơn.

# TÀI LIỆU THAM KHẢO

[1] https://docs.espressif.com/

[2] https://docs.wokwi.com/

[3] https://www.arduino.cc/reference/en/libraries/