**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**TRUNG TÂM THÍ NGHIỆM THỰC HÀNH**

**BÀI THÍ NGHIỆM MÔN HỌC IoT VÀ ỨNG DỤNG**

*(Tài liệu dành cho sinh viên nghành CNTT)*

**Họ tên sinh viên:**....................................

**Mã sinh viên:**…………………………..

**Lớp:**…………………………………….

**BÀI THÍ NGHIỆM SỐ 3**

**ĐIỀU KHIỂN THIẾT BỊ IoT**

1. **MỤC ĐÍCH, YÊU CẦU**
   1. **Mục đích:**

* Giúp sinh viên nắm được một số phương pháp điều khiển thiết bị IoT dựa trên các nền tảng phát triển IoT như Blynk, FireBase,..hoặc AWS, ThingSpeak..vv.
* Giúp sinh viên hiểu rõ hơn về cách thức hoạt động của một mô hình IoT và có thể tự xây dựng được một hệ thống IoT đơn giản như trạm dự báo thời tiết, hệ thống giám sát thông số môi trường, hệ thống nhà thông minh, điều khiển thiết bị từ xa qua Internet.
  1. **Yêu cầu:**
* Sinh viên đã được học lý thuyết Chương 1÷4 của môn học *IoT và ứng dụng*.
* Sinh viên đã có tài liệu thí nghiệm cho môn học *IoT và ứng dụng.*
* Sinh viên đã được học về ngôn ngữ lập trình C/C++. Kỹ thuật vi xử lý và kiến trúc máy tính.

1. **CHUẨN BỊ**

- Sinh viên đọc trước ở nhà nội dung bài thí nghiệm.

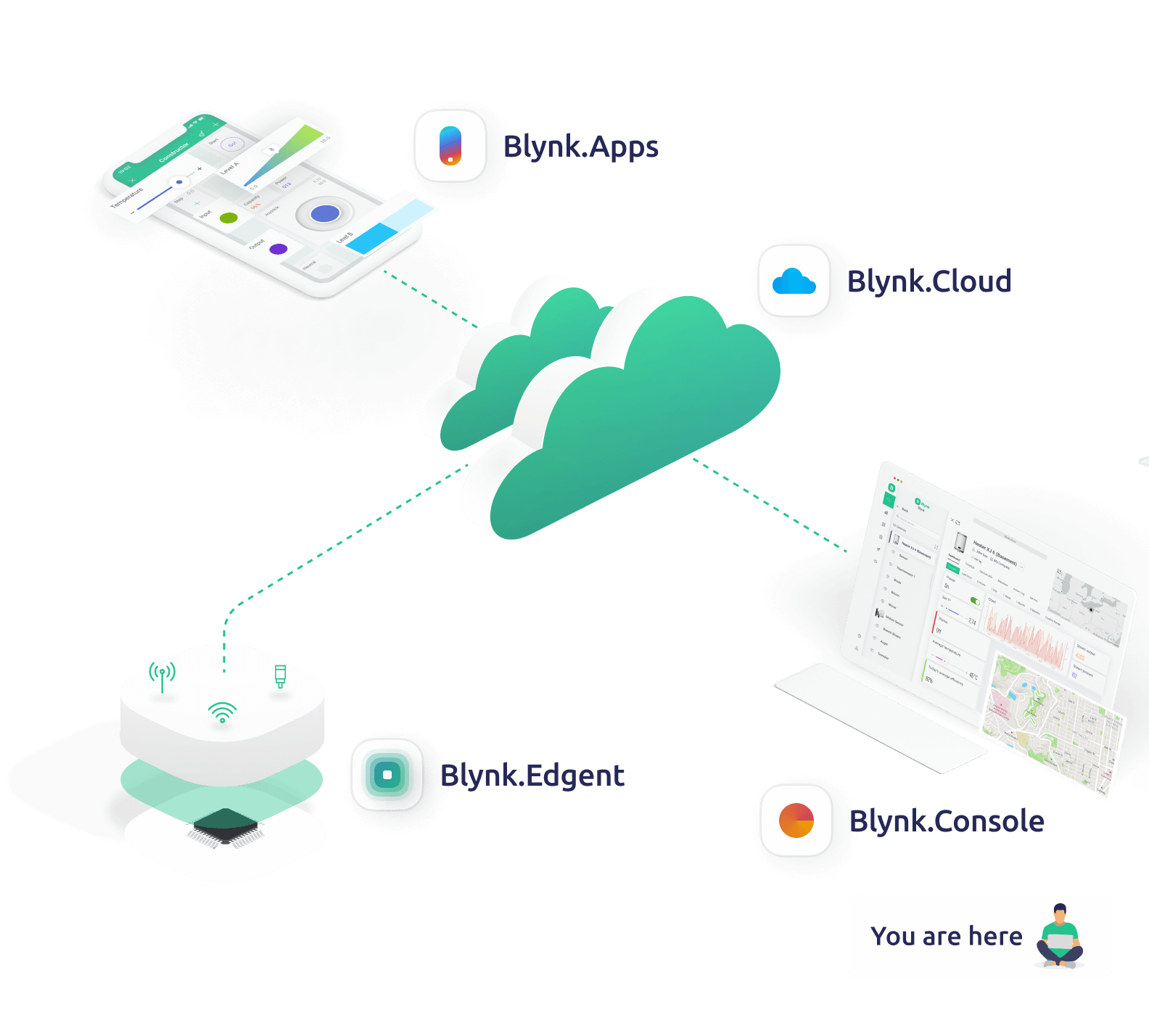
- Xem lại nội dung lý thuyết Chương 1, 2, 3 môn học *IoT và ứng dụng*.

- Xem lại lý thuyết của bài thí nghiệm số 1 (phần giới thiệu về KIT thí nghiệm IoT)

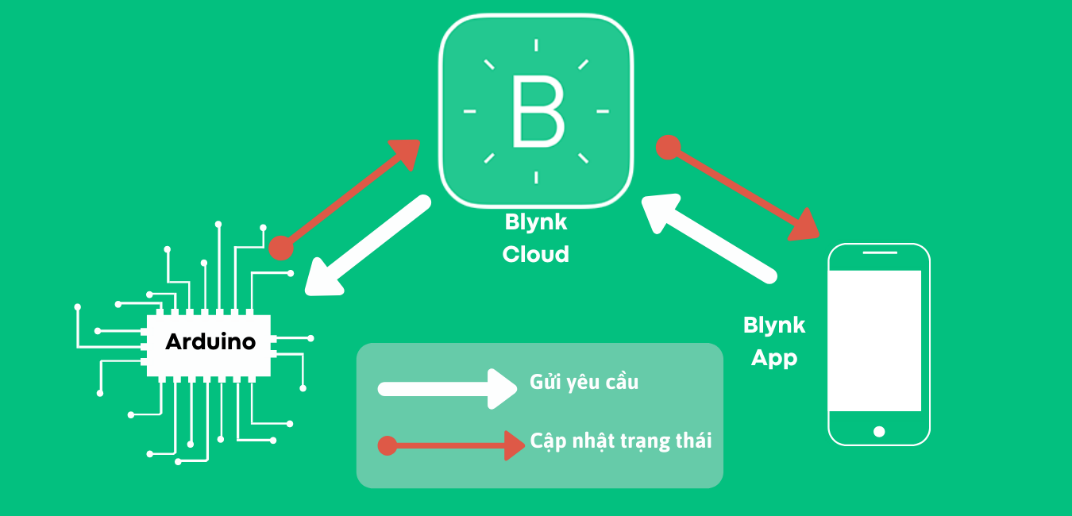
**Giới thiệu về Blynk**

Blynk là một nền tảng phần mềm toàn diện cho phép tạo nguyên mẫu, triển khai và quản lý từ xa các thiết bị điện tử được kết nối ở mọi quy mô.

Cho dù đó là dự án IoT cá nhân hay hàng triệu sản phẩm được kết nối thương mại, Blynk trao quyền cho người dùng kết nối phần cứng của họ với đám mây và tạo ứng dụng iOS, Android và web, phân tích dữ liệu lịch sử và thời gian thực từ các thiết bị, điều khiển chúng từ xa từ mọi nơi, nhận thông báo quan trọng và nhiều hơn nữa.



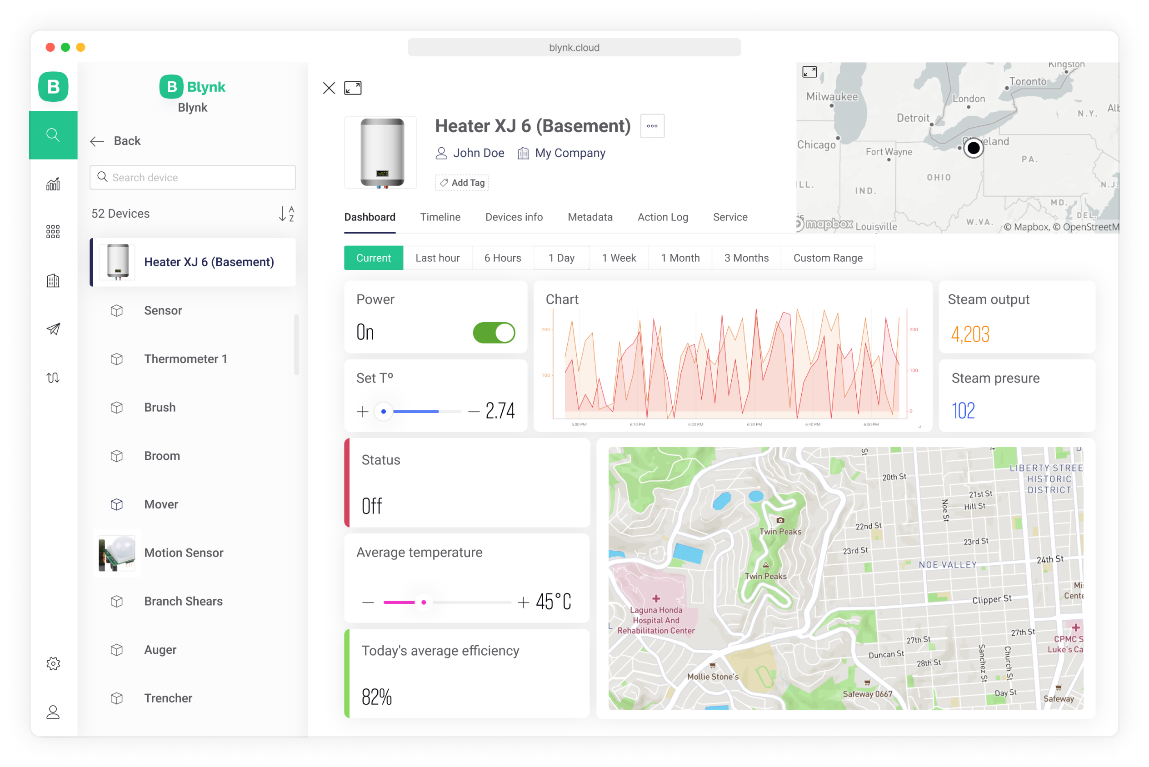
*Hình 3.1. Các thành phần trong hệ thống Blynk*



*Hình vẽ 3.2. Mô hình hệ thống thu nhận dữ liệu và điều khiển thiết bị sử dụng Blynk*

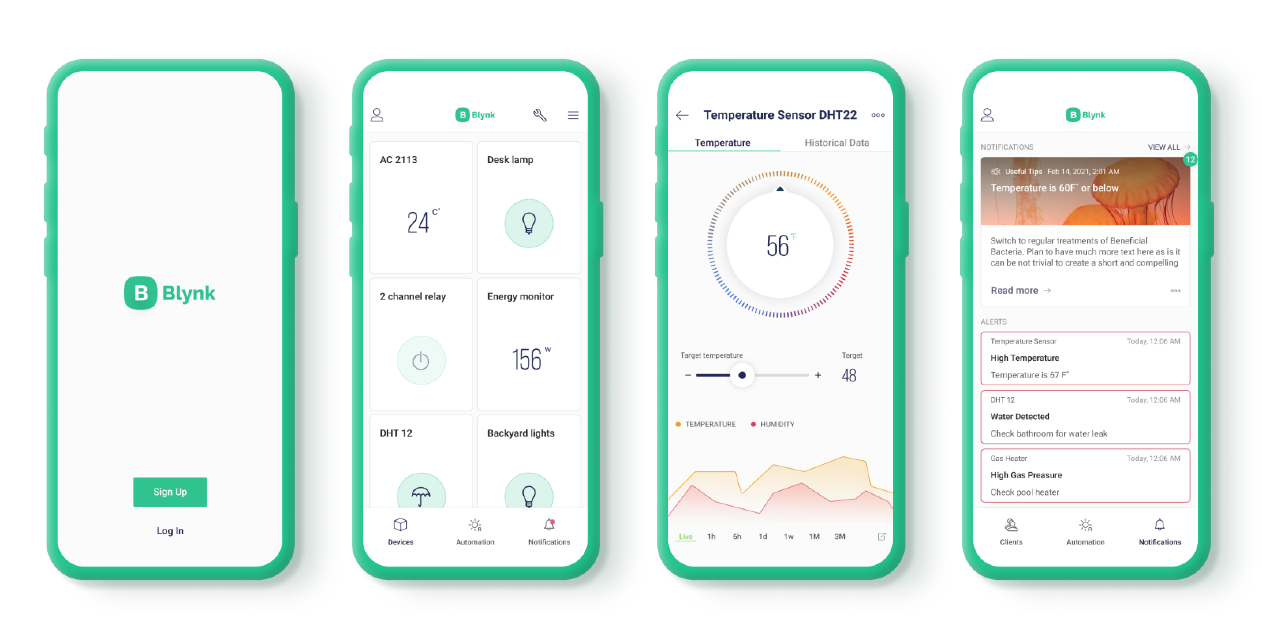
Nền tảng Blink Cloud bao gồm các thành phần chính sau đây:

* **Blynk.Console** là ứng dụng Web cho phép cấu hình, kết nối thiết bị, phân tích dữ liệu cảm biến, cập nhật firmware cho thiết bị thông qua mạng internet, quản lý cách các tổ chức hoặc cá nhân truy cập vào thiết bị IoT.



*Hình 3.3 Blynk. Console - Bảng điểu khiển Blynk*

* **Blynk.App** là ứng dụng dành cho các thiết bị cầm tay với các hệ điều hành iOS và Android. Với ứng dụng này rất dễ dàng để tạo ra các giao diện người dùng (UI) cho thiết bị của người dùng mà không cần phải lập trình, đồng thời có thể chia sẻ giao diện này cho những người dùng khác.



*Hình 3.4. Blynk.App - Ứng dụng Blynk trên thiết bị di động*

* **Blynk.Edgent** **và Blynk. Library (**Edge + Agent = Edgent**)**

**Blynk. Edgent** là một giải pháp đóng gói được thiết kế để đơn giản hóa việc kết nối các thiết bị được hỗ trợ với nền tảng Blynk, cung cấp quyền truy cập vào tất cả các tính năng nâng cao của nó mà không cần viết mã nhiều. Các tính năng chính của Blynk.Edgent bao gồm:

* + Thiết bị được kết nối với mạng Wi-Fi (đưa thiết bị kết nối Internet và xác thực chúng với một người dùng nhất định).
  + Quản lý kết nối cho Wi-Fi, Cellular và Ethernet.
  + Truyền dữ liệu giữa thiết bị và đám mây.
  + Tích hợp API với các tính năng của Blynk.Apps và Blynk.Cloud.
  + Cập nhật chương trình cơ sở qua mạng cho các kiểu phần cứng được chọn (FOTA).

**Blynk. Library** (thư viện Blynk) là một thư viện C++ di động và thân thiện với người dùng, được cấu hình sẵn để hoạt động với hàng trăm loại KIT điện tử phát triển ứng dụng. Nó thực hiện một giao thức kết nối trực tuyến, cho phép giao tiếp hai chiều và độ trễ thấp.

* **Blynk.Cloud l**à cơ sở hạ tầng máy chủ đóng vai trò là trung tâm của nền tảng Blynk IoT liên kết tất cả các thành phần của Blynk IoT lại với nhau.

1. **THÍ NGHIỆM**

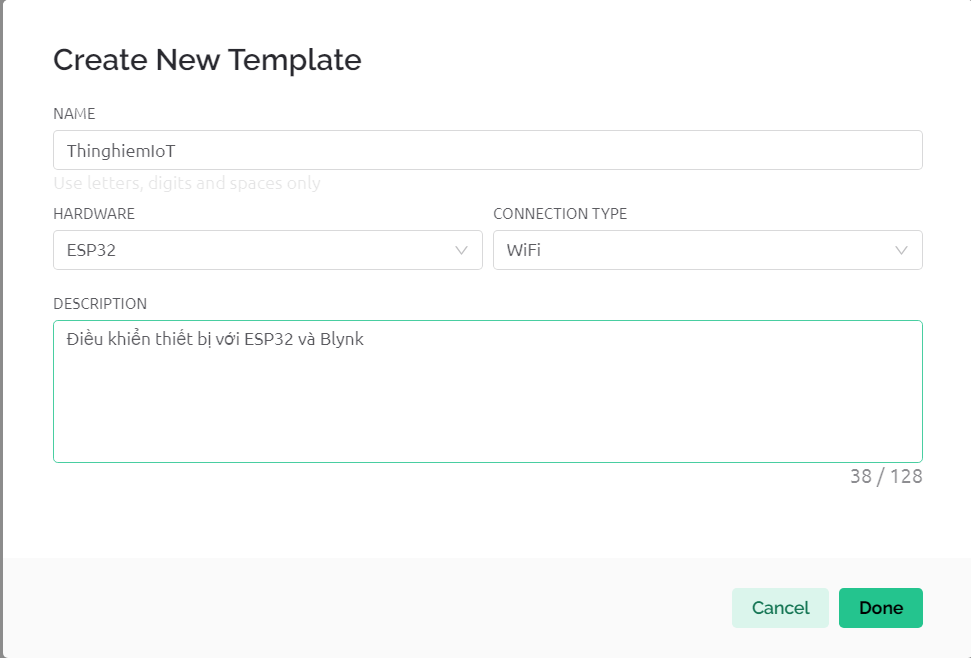
**3.1 Điều khiển thiết bị IoT với Blynk và ESP32**

Mục tiêu của thí nghiệm này là sử dụng KIT Arduino ESP32 kết nối với Blynk qua mạng WiFi nhằm điều khiển bật, tắt động cơ một chiều (hoặc các đền LED đơn trên KIT thí nghiệm IoT). Các bước tiến hành như sau:

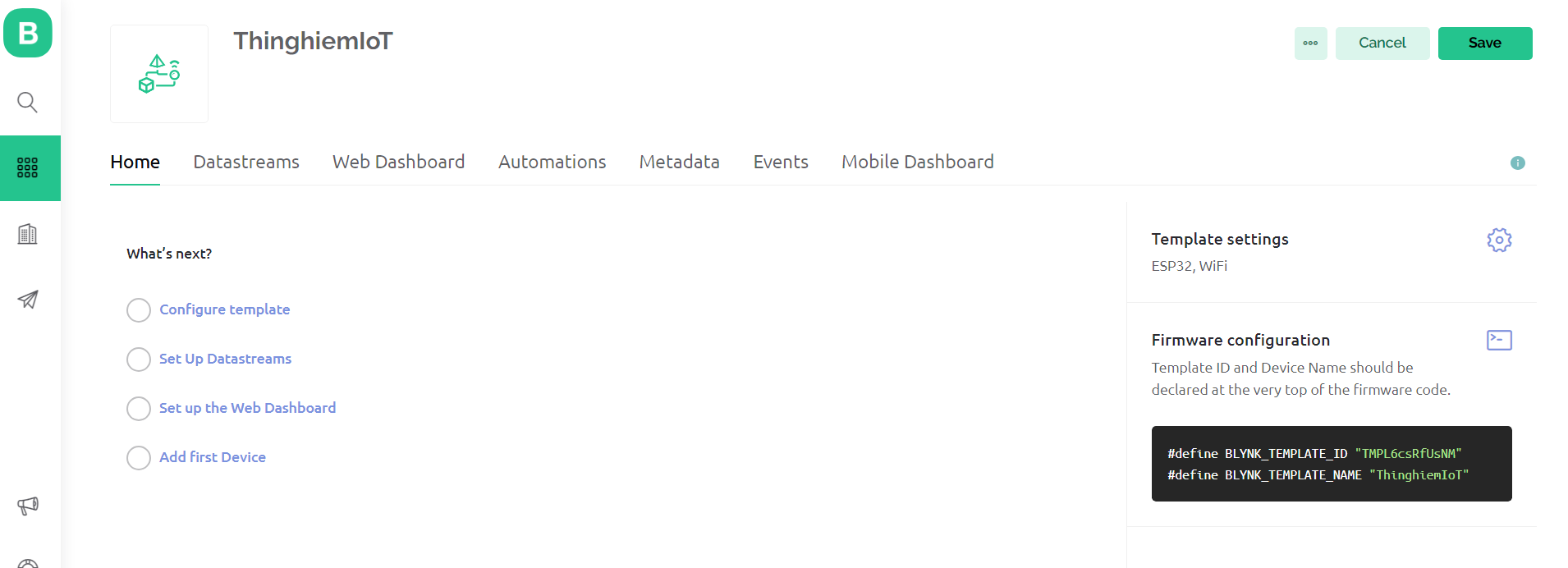
***Bước 1***: Đăng ký tài khoản miễn phí trên trang web tại địa chỉ *blynk.io*. Đăng nhập vào Blynk.io bằng tài khoản vừa tạo ra.

***Bước 2****:*Tạo một *Template* cho dự án của mình bằng cách chọn *Create New Template*. *Template* chính là một bản mẫu thiết bị được sử dụng (ở đây chính là ESP32), từ *Template* có thể tạo ra các Things có thuộc tính giống nhau mà không cần khởi tạo lại nhiều lần.

*Chú ý*: chọn HARDWARE là ESP32 và CONNECTION TYPE là WiFi.

**

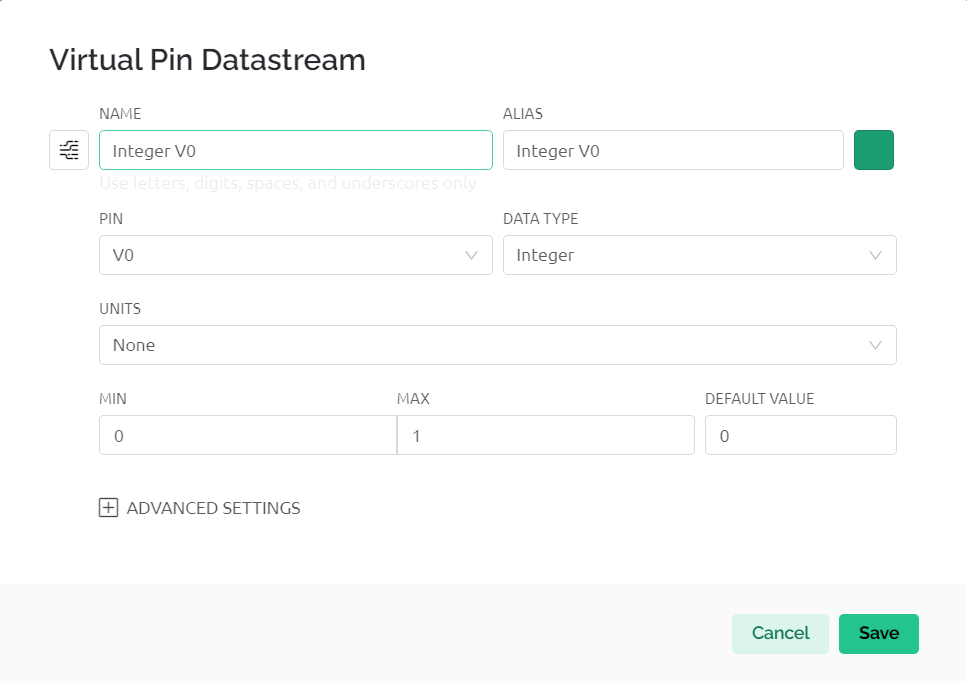
*Hình vẽ 3.5. Tạo một template mới*



*Hình vẽ 3.6. Một template mới được tạo ra*

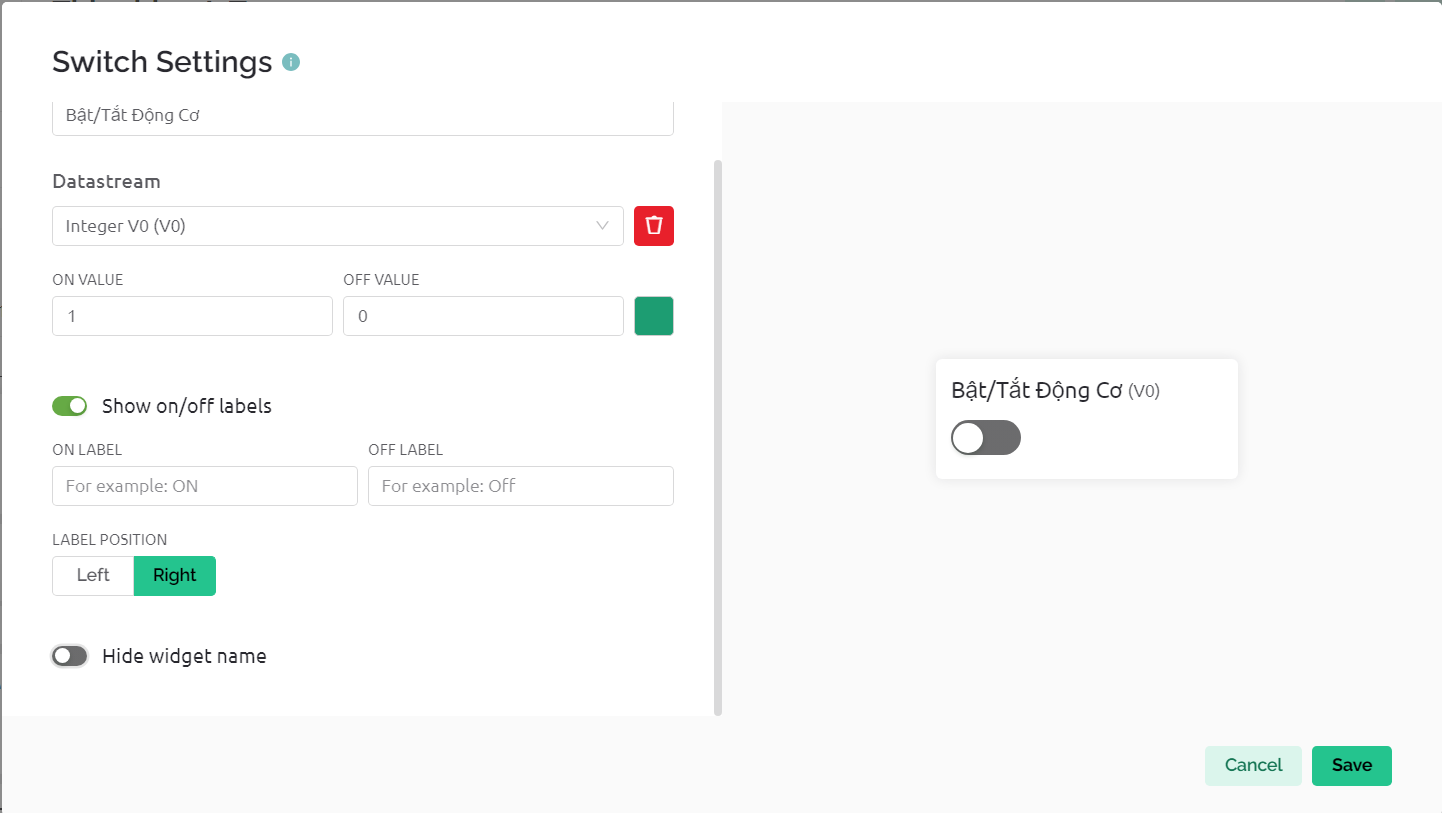
***Bước 3****:* Thiết lập các tham số của *Template*

* + *Setup Data Stream:* Đây chính là biến dùng để truyền dữ liệu giữa chương trình firmware chạy trên phần cứng nhúng ESP32 và Blynk. Chọn *New Data Stream.* Sau đó chọn và cài đặt *Virtual Pin*. Sau khi cài đặt ta được như hình vẽ dưới đây:



*Hình vẽ 3.7. Cài đặt Data Stream*

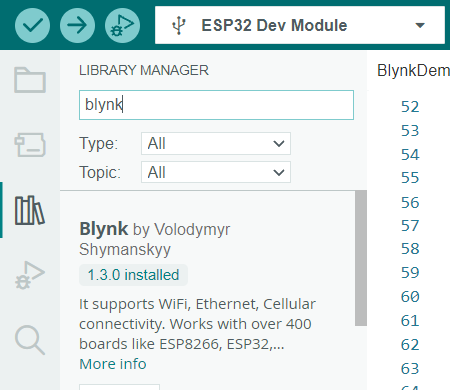
* + Cài đặt *Web Dashboard:* Để điểu khiển Bật/Tắt đèn LED đơn hoặc động cơ một chiều cần có một nút điều khiển có thể thực hiện chức năng ON/OFF do vậy hãy chọn *Switch widget* và tiến hành gán *Data Stream* vào *Widget* đó. Sau đó *Save* lại.



*Hình vẽ 3.8. Các thiết lập cho nút điều khiển Switch*

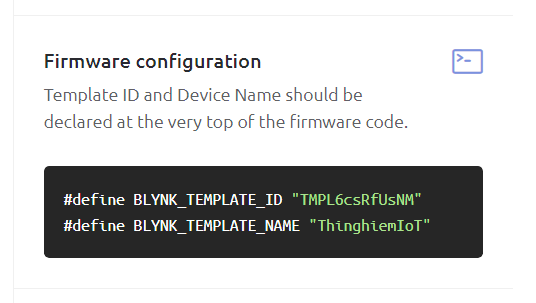
***Bước 4:*** Lập trình firmware cho ESP32

* + Mở chương trình Arduino IDE version 2.1.1và tạo một Sketch mới, đặt tên là *Bai3.ino* sau đó cài đặt thư viện Blynk bằng cách vào Library Manager, trong ô tìm kiếm gõ Blynk và Install gói thư viện Blynk



*Hình vẽ 3.9. Cài đặt thư viện Blynk cho Arduino*

* + Quay trở lại Blynk, vào tab Home và copy đoạn mã firmware configuration sau đó paste vào chương trình trong Arduino



*Hình vẽ 3.10. Đoạn mã cấu hình cho firmware trên Blynk*

BLYNK\_TEMPLATE\_ID chính là ID template vừa tạo ra.

BLYNK\_DEVICE\_NAME là tên Things tạo ra dựa trên Template đó.

* + Trong Arduino IDE chọn Board là ESP32 dev module, chọn đúng cổng COM ảo được kết nối với KIT ESP32 (*số hiệu cổng COM khác nhau đối với từng máy tính khác nhau*). Chọn Upload Speed là 115200.

***Bước 5:*** *Kết nối dây trên KIT thí nghiệm IoT*

Trên KIT thí nghiệm IoT động cơ một chiều có thể được điều khiển quay thuận, ngược và tốc độ nhanh hay chậm bằng cách phát các xung điều khiển và xung điều chế độ rộng PWM thông qua 4 chân được cho ở bảng sau:

|  |  |
| --- | --- |
| **Tên chân điều khiển** | **Ý nghĩa** |
| **R\_EN** | Điều khiển cho phép động cơ quay phải |
| **L\_EN** | Điều khiển cho phép động cơ quay trái |
| **RPWM** | Điều khiển tốc độ quay phải nhanh hay chậm |
| **LPWM** | Điều khiển tốc độ quay trái nhanh hay chậm |

*Bảng 1. Tên các chân điều khiển động cơ 1 chiều*

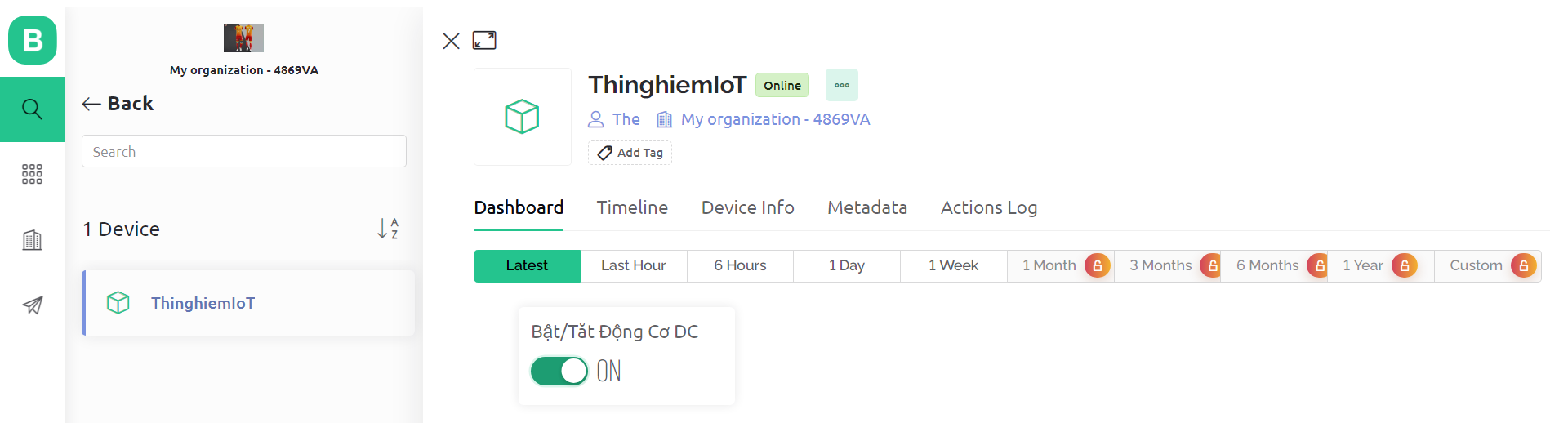
Trong bài này động cơ được điều khiển quay từ trái sang phải với một tốc độ cố định, động cơ sẽ quay hay dừng lại khi nhận tín hiệu điều khiển có hay không có xung đưa tới 2 chân điều khiển **R\_EN** và **L\_EN**. ESP32 sẽ điều khiển động cơ quay hay dừng thông qua chân IO14. Tốc độ động cơ không thay đổi được đưa tới chân RPWM bằng chân IO27 của ESP32. Sơ đồ kết nối dây được cho như bảng sau đây:

|  |  |
| --- | --- |
| **Chân của ESP32** | **Chân điều khiển động cơ DC** |
| **IO14** | **R\_EN** |
| **IO14** | **L\_EN** |
| **IO27** | **RPWM** |

*Bảng 2: Sơ đồ kết nối dây giữa ESP32 và mạch điều khiển động cơ DC*

***Bước 6***: Viết chương trình điều khiển động cơ DC với Blynk trên Arduino IDE xuống dưới đây ?

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………



*Hình vẽ 3.11. Bảng điều khiển động cơ DC trên Blynk*

***Bước 7***. Điều khiển động cơ DC bằng ứng dụng Blynk IoT trên điện thoại di động

* + Cài đặt ứng dụng Blynk IoT trên điện thoại di động.
  + Đăng nhập vào ứng dụng với username và password như đã đăng ký
  + Xây dựng giao diện điều khiển động cơ DC trên điện thoại với *data stream* tương ứng như được thiết lập trên web.
  + Thực hiện điều khiển bật/tắt động cơ DC và quan sát kết quả.

**3.2 Ghi nhận kết quả và trả lời câu hỏi**

*Câu 1*: Số thiết bị mà Blynk cho phép sử dụng trong một tài khoản đăng ký miễn phí là bao nhiêu ? ………………Số thiết bị nhiều nhất mà Blynk cho phép một tài khoản mất phí kết nối đến Cloud là bao nhiêu ?.......................... Từ đây rút ra kết luận về việc có hay không có thể sử dụng Blynk để phát triển ứng dụng IoT trong thực tế ?.........................................

………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

*Câu 2*: Viết chương trình cho ESP32 và sử dụng Blyn để xây dựng ứng dụng IoT chăm sóc cây trồng trong vườn nhà. Ứng dụng có thể giám sát các thông số nhiệt độ, độ ẩm của môi trường, vẽ đồ thị nhiệt độ, độ ẩm, hiển thị nhiệt độ, độ ẩm dưới dạng Widget *Gauge* trên Blynk và trên điện thoại, đồng thời có thể điều khiển bật/tắt máy bơm nước tưới cây. Viết mã nguồn chương trình xuống dưới đây.

*Gợi ý: sử dụng cảm biến nhiệt độ, độ ẩm DHT11 và động cơ một chiều trên KIT thí nghiệm IoT (xem lại bài thí nghiệm số 2 phần giao tiếp giữa ESP32 và cảm biến DHT11).*

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………

………………………………………………………………………………………………