

## HƯỚNG DẪN THỰC HÀNH

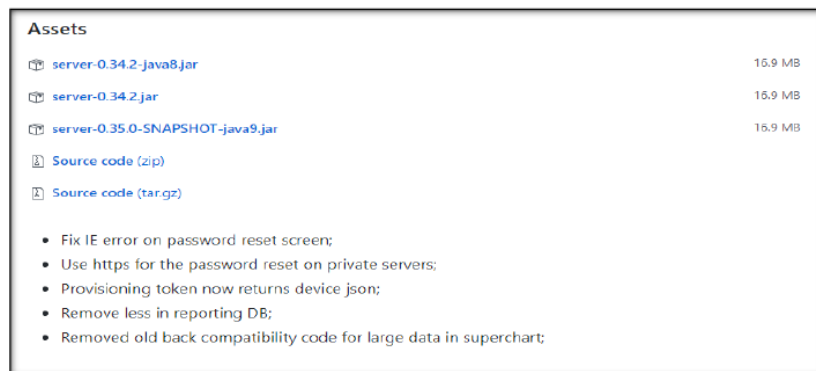
### Lab 2-3: IoT Platform – Blynk

- Học viên Học viên thực hành tất cả các nội dung có trong bài học lý thuyết đồng thời hoàn thành các nội dung bài hướng dẫn thực hành sau đây.
- Bài 1: Cài đặt Blynk trên Window
- Bài 2: Bật đèn LED bằng Blynk với mạch ESP8266 NodeMCU
- Bài 3: Get Data trong Blynk
- Bài 4: Push Data trong Blynk
- Bài 5: Đọc chân ảo trong Blynk
- Bài tập về nhà: Người dạy có thể cung cấp thêm một số bài tập về nhà cho học viên để hoàn thiện các kiến thức liên quan đến nội dung bài học.

#### Bài 1: Cài đặt Blynk trên Window

Mỗi một hệ thống được xây dựng dựa trên cloud blynk thì đều cần một tài khoản ứng dụng để có thể truy cập dữ liệu cũng như quản lý, điều khiển hệ thống đó. Tuy nhiên, không phải tất cả hệ thống đều chỉ tạo tài khoản trên cloud blynk và sử dụng như vậy có thể rất bất tiện khi tạo ra nhiều tài khoản khác nhau để quản lý chúng vì tốn kém hơn, chúng ta có thể xây dựng một server riêng cho hệ thống đó và tự do tạo tài khoản hay quản lý hệ thống của mình một cách chủ động hơn như thế chúng ta sẽ không phụ thuộc hệ thống vào bất kỳ thứ gì. Hơn nữa tạo một server quản lý cũng không là quá khó khăn.

Trước hết để cài đặt server cloud cần cài đặt một môi trường để tạo server cũng như để cài đặt cấu hình từ link sau: <https://github.com/blynkkk/blynk-server/releases>



Sau khi đã tải thành công server thì ta bắt đầu cài đặt môi trường java để tạo các kết nối đến server này: <https://java.com/en/download/>



Tiếp theo ta mở giao diện “Command Prompt” và kiểm tra phiên bản java theo lệnh sau : “java -version” nếu như java có phiên bản lớn hơn 9 là được. Sau đó, chạy lệnh cài đặt server, tập lệnh này sẽ cài đặt mặc định cổng “8080” là cổng phân cứng tức là cổng ta sử dụng NodeMCU để kết nối đến blynk còn cổng “9443” là cổng ứng dụng tức là cổng sử dụng để chạy trên ứng dụng điện thoại.

Chạy tập lệnh sau để có thể cài đặt server trong đó “server-0.35.0.jar” là tên file server đã tải về và “path” là đường dẫn hiện tại đến thư mục muốn lưu trữ dữ liệu.

```
java -jar server-0.35.0.jar -dataFolder /path
```

Sau khi chạy tập lệnh trên nếu như xuất hiện một dòng lệnh như sau thì tức là đã cài đặt xong server giờ chỉ cần sử dụng là xong.

```
Blynk Server successfully started.  
All server output is stored in current folder in 'logs/blynk.log' file.
```

Sau khi đã cài đặt thành công server ta vào đường dẫn lưu trữ dữ liệu thì sẽ thấy một tập tin admin@blynk.cc.Blynk đây sẽ là tên của tài khoản quản trị hệ thống và mật khẩu là “admin”

backup	17/04/2018 8:18 SA	File folder	
clone	12/04/2018 1:57 CH	File folder	
data	13/04/2018 8:24 SA	File folder	
deleted	12/04/2018 1:57 CH	File folder	
logs	16/04/2018 10:21 ...	File folder	
static	16/04/2018 10:21 ...	File folder	
admin@blynk.cc.Blynk	12/04/2018 1:59 CH	Visual Studio Proj...	1 KB
Blynk	12/04/2018 2:43 CH	WinRAR archive	16.977 KB
dangthuhai111096@gmail.com.Blynk	17/04/2018 8:19 SA	Visual Studio Proj...	5 KB
server-0.34.1-java8	12/04/2018 1:52 CH	Executable Jar File	17.346 KB
start-blynk	12/04/2018 3:07 CH	Windows Batch File	1 KB
thanhthangbmt151@gmail.com.Blynk	12/04/2018 2:36 CH	Visual Studio Proj...	6 KB

Bây giờ để kiểm tra các tài khoản đã được tạo trên server này ta chạy trên trình duyệt bất kỳ câu lệnh “localhost:9443/admin” sẽ xuất hiện một cửa sổ yêu cầu nhập tài khoản quản trị, sau khi nhập vào sẽ xuất hiện cửa sổ quản trị server như vậy là ta đã xây dựng và quản lý thành công server blynk của chính mình. Hoặc nếu chúng ta muốn server này lớn hơn hãy tạo cho mình một domain rồi cập nhật server này lên đó thì không cần máy tính chúng ta cũng có thể truy cập vào server này theo địa chỉ domain này.

Blynk Administration

Users

Stats

Hardware Info

Config

Dashboard

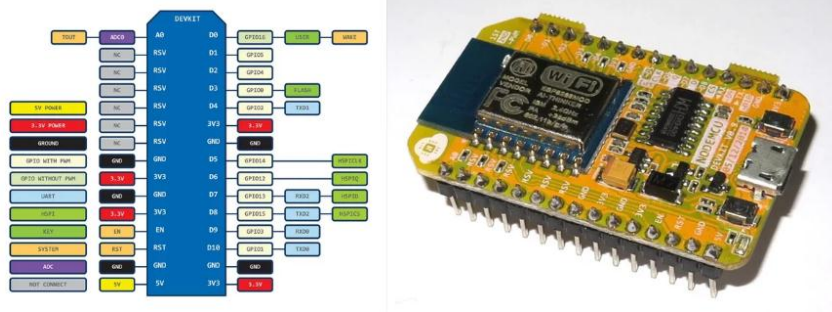
Realtime stats

1 min request rate	Total registrations	Logged In 24h	Logged In 72h	Hard and App connected	App connections	Hardware connections
0		0		0	0	0

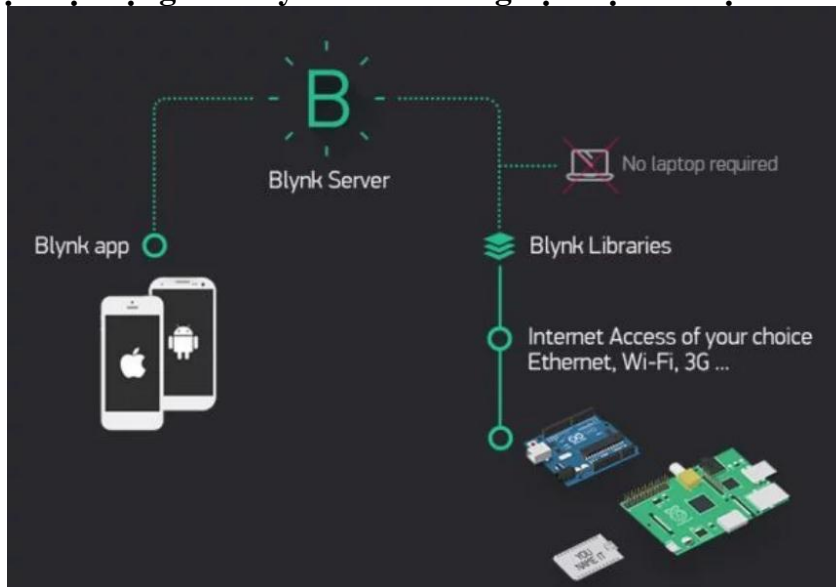
## Bài 2: Bật/Tắt đèn LED bằng Blynk với ESP8266 NodeMCU

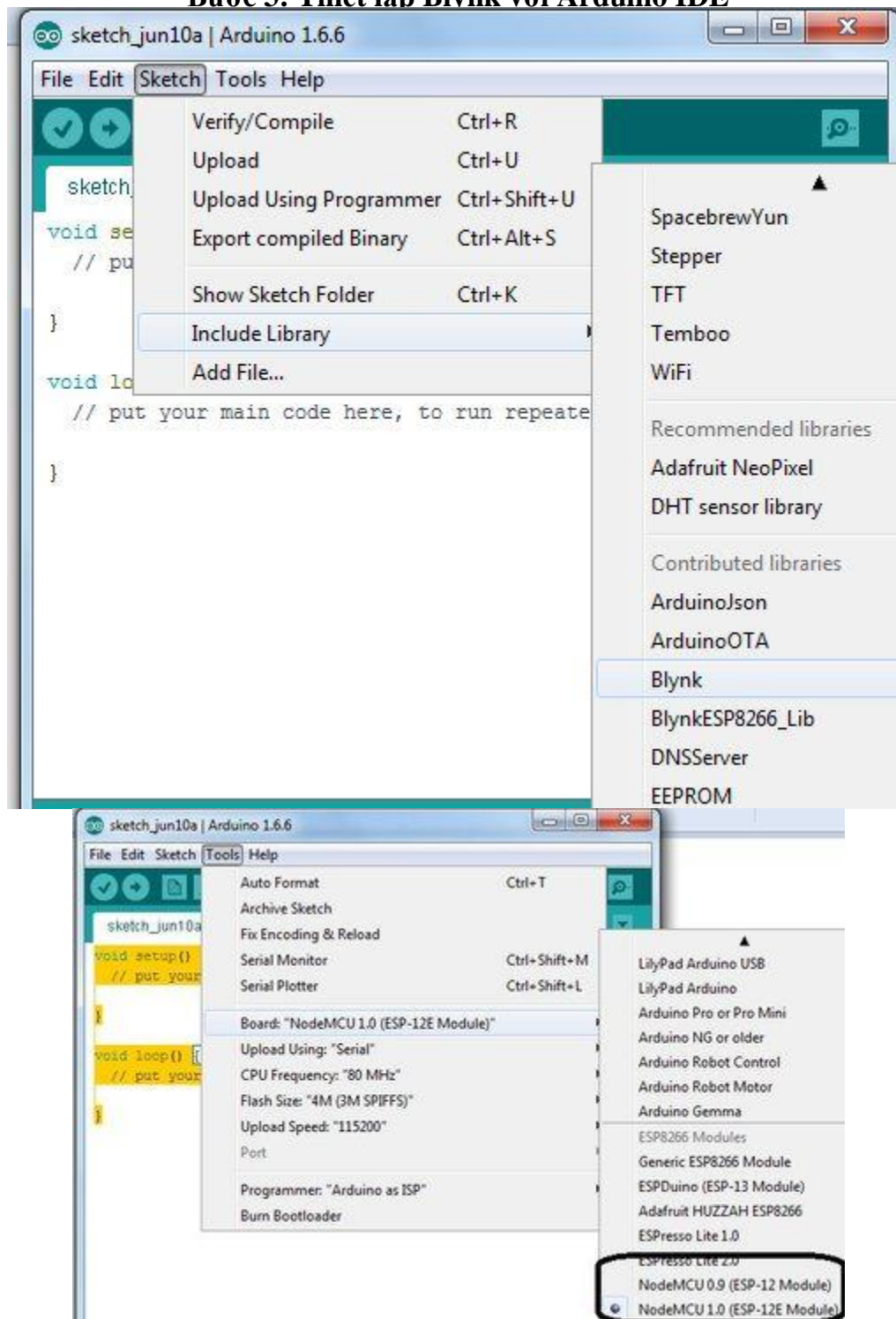
Trong bài này sẽ hướng dẫn Điều khiển đèn LED đơn giản với Blynk và NodeMCU ESP8266 12E.

## Bước 1: Tìm hiểu về Bo NodeMCU



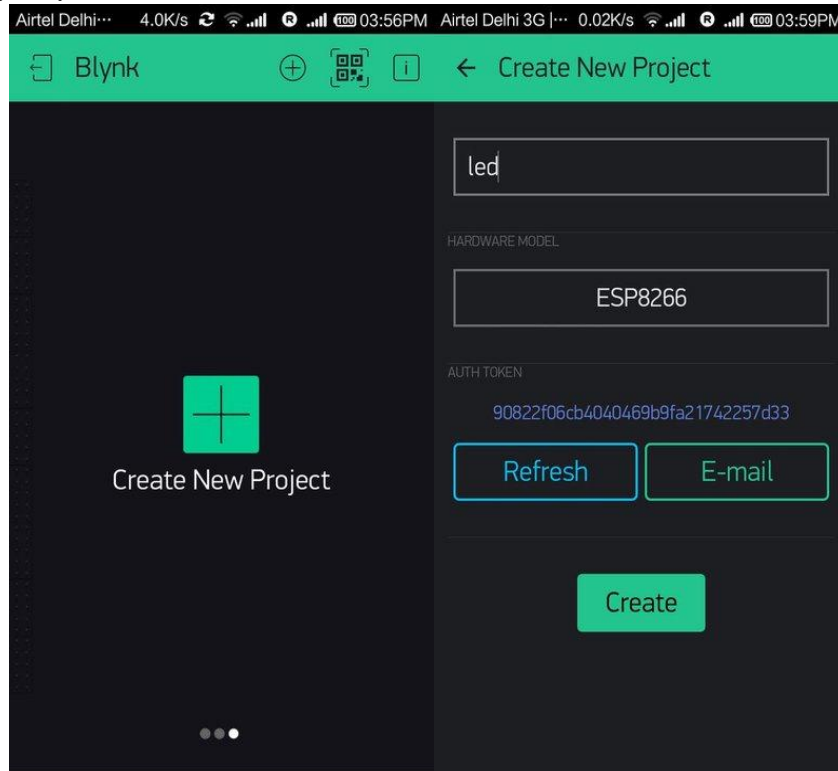
## Bước 2: Hiểu về sự hoạt động của Blynk và các công cụ được hỗ trợ



**Bước 3: Thiết lập Blynk với Arduino IDE****Bước 4: Thiết lập Blynk**

- 1) Tải phần mềm Blynk trên thiết bị di động và sau đó đăng nhập (vào Blynk local server hoặc Blynk có trả phí)

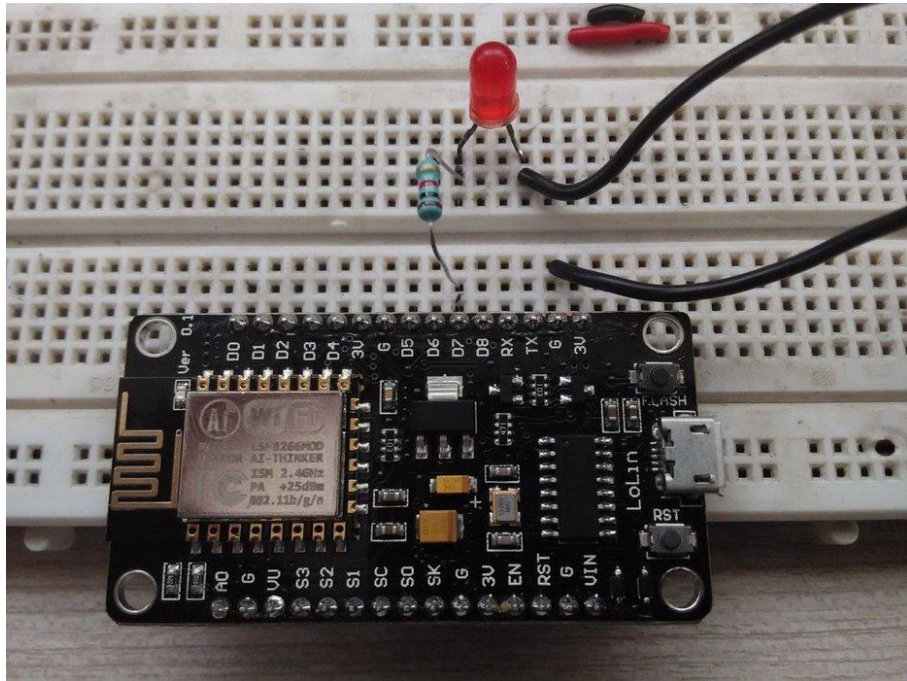
2) Sau đó bấm tạo dự án mới:



- 3) Nhập tên dự án: **led**
- 4) Sau đó, chọn board như **ESP8266**
- 5) Lấy mã xác thực (token) bằng Email hoặc Text
- 6) Nhấn nút Create
- 7) Nhấn vào nút dấu "+" ở góc bên phải để thêm tiện ích vào dự án.
- 8) Chúng ta thêm một nút và cấu hình là pin Digital GP13
- 9) Chọn chức năng công tắc (switch)
- 10) Gán nhãn cho nút là BẬT và TẮT tương ứng



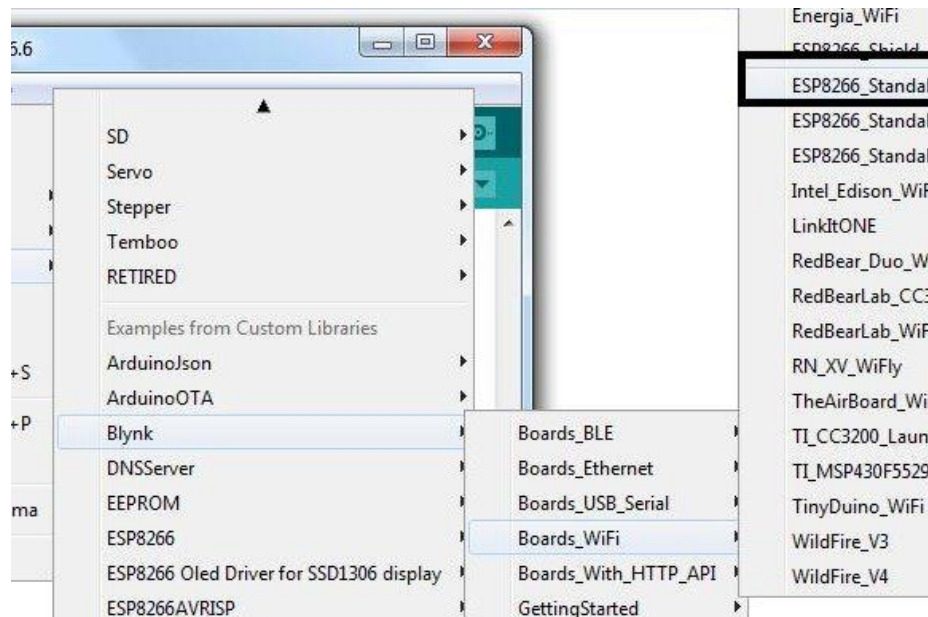
### Bước 5: Sơ đồ mạch



- 1) Pin của bo mạch ESP8266 có D7 là chân thứ 13.
- 2) Kết nối pin LED với D7 qua điện trở 330 / 220 Ohm

### Bước 6: Tải code lên bo mạch

- 1) Kết nối Wi8266 Wifi với PC
- 2) Mở Arduino IDE
- 3) Sau đó vào File → Eamples → Blynk-Boards\_Wifi → Esp8266Standalone



- 4) Chọn NodeMCU 1.0 và cổng COM từ Tools Menu
- 5) Gõ code:

```
Serial.begin (9600); // Thay đổi Baud thành 115200  
Blynk.begin (auth, "ssid", "pass"); // Nhập token, SSID và mật khẩu Wifi  
...
```

Lưu tệp và Tải code lên mạch

### Bước 7: Bật/Tắt đèn LED bằng điện thoại





**Bài 3: Lấy giá trị trong Blynk**

- Ta có thể sử dụng hàm lấy giá trị này để gỡ lỗi và in tất cả các giá trị gửi được gửi bởi một tiện ích có kết nối với Pin ảo trong Ứng dụng của Blynk.
- Thiết lập dự án ứng dụng: Tiện ích trackBar giá trị từ 0 đến 100 trên V1

```
// Hàm này sẽ được gọi mỗi khi Tiện ích trackBar thay đổi
// trong ứng dụng Blynk và cập nhật cho Pin V1
BLYNK_WRITE(V1)
{
  int pinValue = param.asInt(); // gán giá trị đến từ pin V1 vào biến
                                // Bạn có thể dùng:
                                // String i = param.asStr();
                                // double d = param.asDouble();
  Serial.print("V1 Slider value is: ");
  Serial.println(pinValue);
}
```

**Bài 4: Đẩy dữ liệu lên Blynk**

- Ví dụ này giá trị có thể được đẩy từ Arduino lên ứng dụng Blynk như thế nào.
- BlynkTimer cung cấp chức năng SimpleTimer, tham khảo thêm tại: <http://playground.arduino.cc/Code/SimpleTimer>
- Thiết lập dự án ứng dụng: Widget Hiển thị giá trị tương ứng với Pin ảo V5

```
BlynkTimer timer;
```

```
// Hàm này gửi thời gian lên của Arduino mỗi giây lên Virtual Pin (5).
// Trong ứng dụng, tần suất đọc của Widget phải được đặt thành PUSH. Điều này có nghĩa là
// chúng ta xác định tần suất gửi dữ liệu đến Blynk App.
```

```
void myTimerEvent()
{
  // Chúng ta có thể gửi bất kỳ nội dung nào vào thời gian nào, tuy nhiên không gửi quá 10 giá
  // trị trong một giây
  Blynk.virtualWrite(V5, millis() / 1000);
}
```

```
void setup()
{
  ...
  // Hàm sẽ được gọi theo mỗi giây
  timer.setInterval(1000L, myTimerEvent);
}
```



### Bài 5: Đọc chân ảo trong Blynk

- Bài này hướng dẫn cách đọc các giá trị từ các Pin ảo
- Thiết lập dự án ứng dụng: Tiện ích trackBar (0 ... 100) trên Pin ảo V1

// Hàm này sẽ được gọi mỗi khi Tiện ích trackBar thay đổi giá trị

// trong ứng dụng Blynk và ghi giá trị cho Pin ảo V1

**BLYNK\_WRITE(V1)**

{

**int** pinValue = param.asInt(); // nhận giá trị thay đổi từ V1 và gán vào biến để xử lý

}