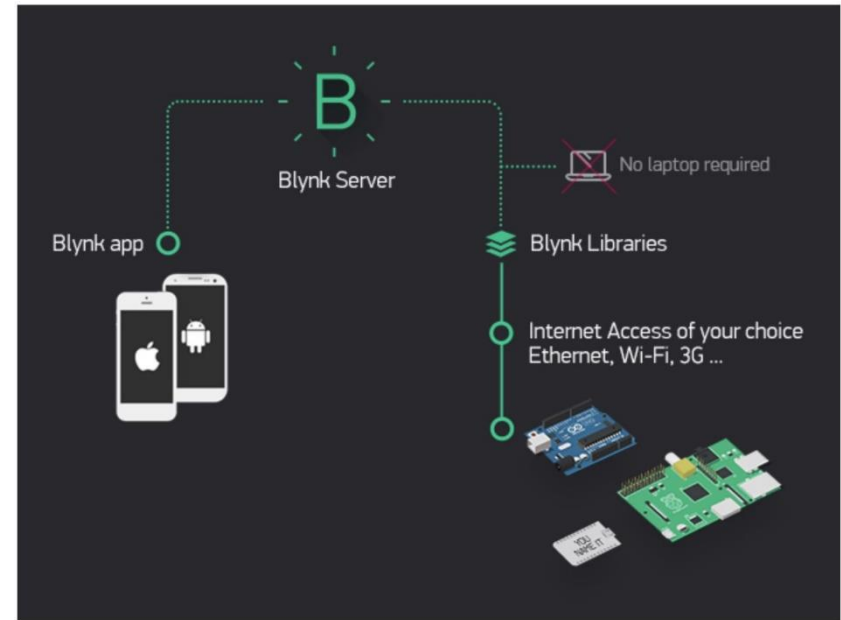


# Blynk



# Blynk

- IoT แพลตฟอร์มที่ค่อนข้างใช้ง่าย
- ประกอบด้วย 3 ส่วน ได้แก่
- Blynk App เพื่อใช้สร้างอินเตอร์เฟซที่ต้องการ โดยมี widgets ที่หลากหลายจัดเตรียมไว้แล้ว
- Blynk Server ทำหน้าที่สื่อสารระหว่างอุปกรณ์และมือถือแบบสมาร์ทโฟน นอกจากการใช้เซิร์ฟเวอร์ของทาง Blynk ผู้พัฒนาสามารถติดตั้ง Blynk Server ของตนเองได้
- Blynk ไลบารีสำหรับทำงานบนฮาร์ดแวร์สื่อสารกับเซิร์ฟเวอร์ทำงานตามที่ได้โปรแกรมไว้การสื่อสารที่ใช้อาจเป็นไวไฟ (Wi-Fi) บลูทูธ (Bluetooth) หรือ อื่น ๆ



# การเชื่อมต่อ Wi-Fi |

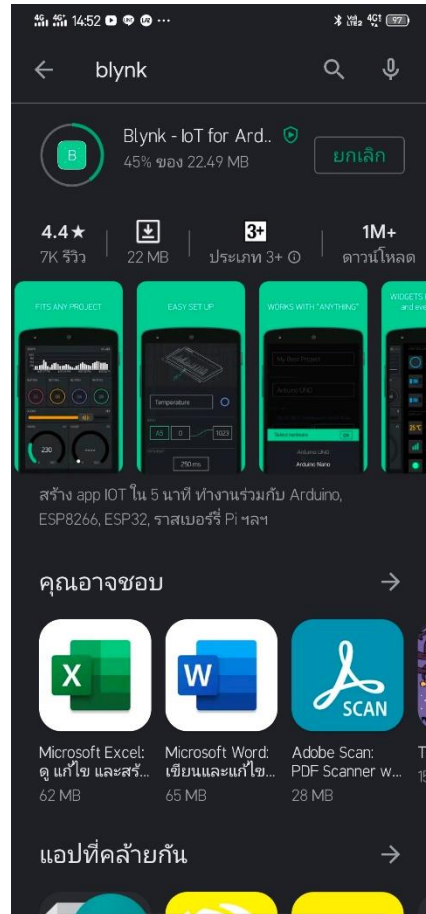
- กรอกโค้ดต่อไปนี้

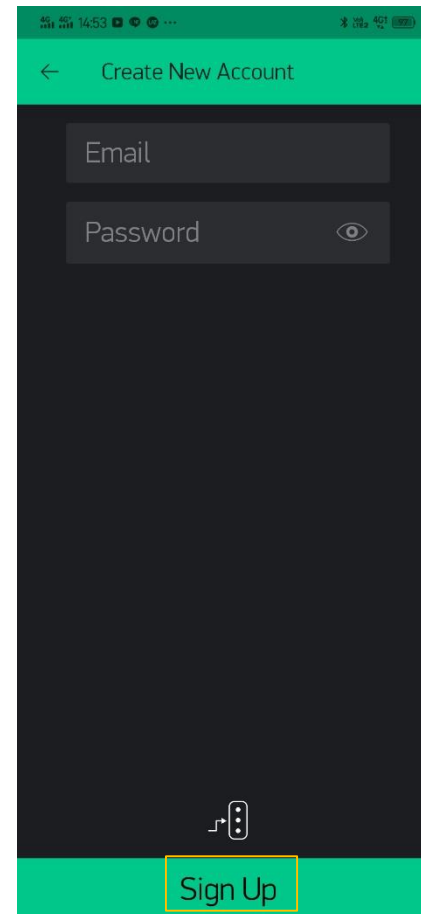
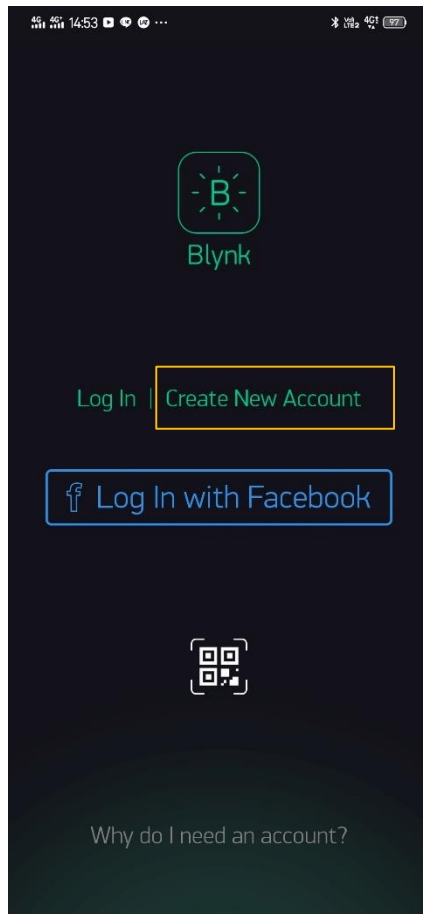
```
import network
station = network.WLAN(network.STA_IF)
station.active(True)
station.connect("<AP_Name>", "<password>")
station.isconnected()
```

- ทดสอบคำสั่งต่อไปนี้

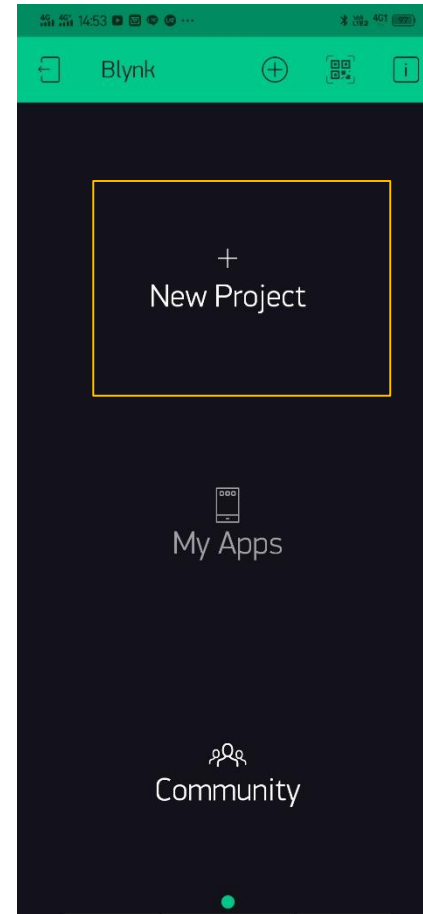
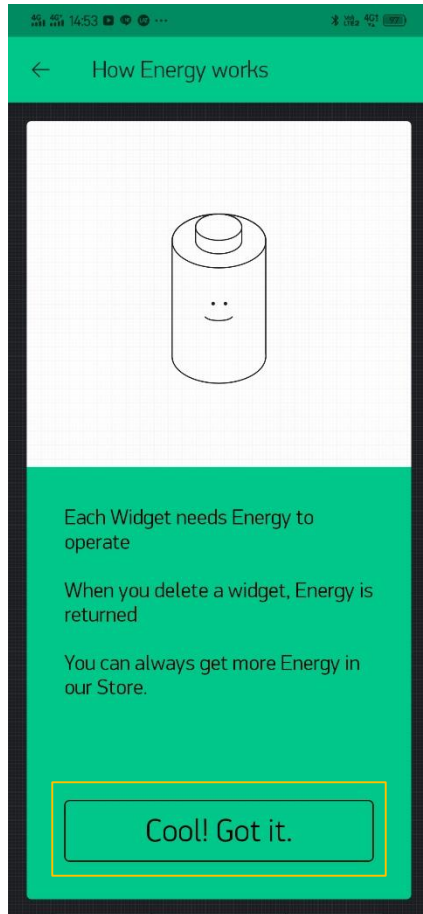
```
station.config('mac')
station.ifconfig()
```

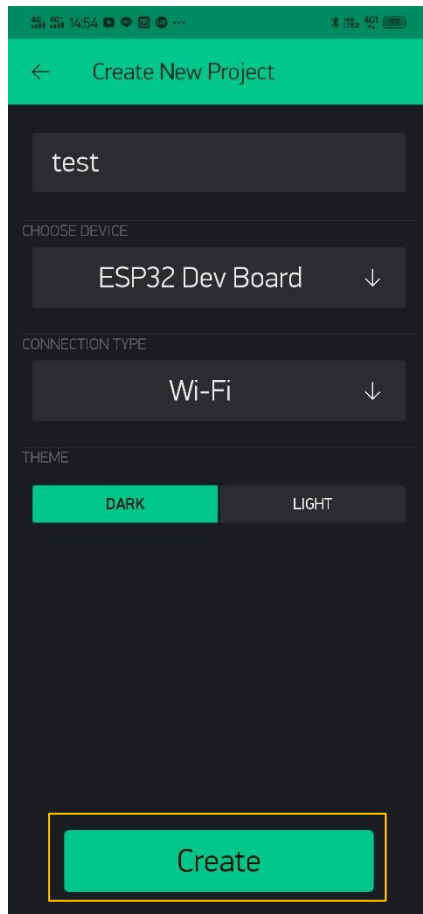
# ติดตั้ง Blynk



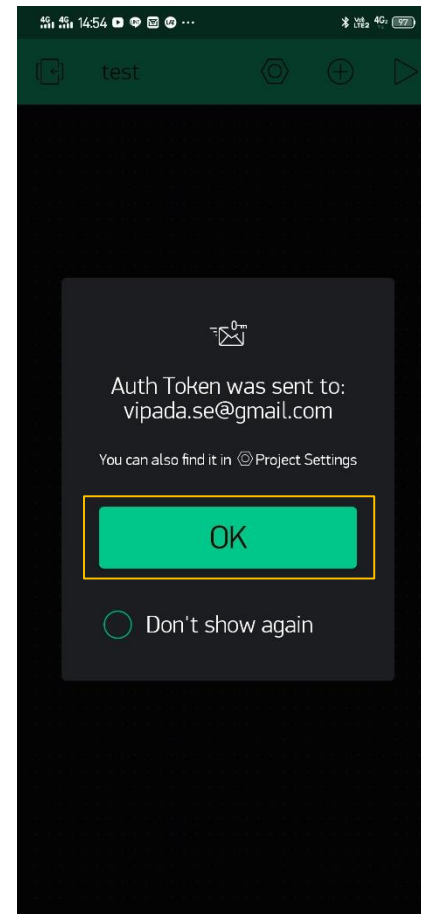


- กรอก email
- กรอก รหัสผ่าน

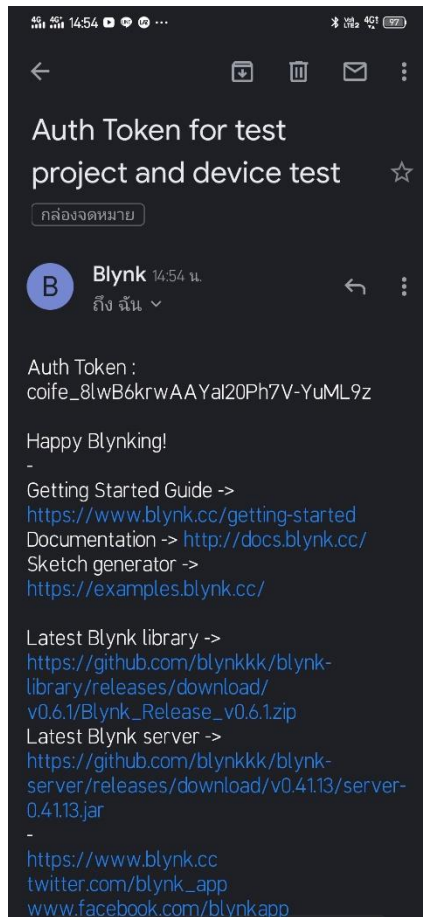




- ชื่อโปรเจก
- บอร์ดที่ใช้
- ประเภทการเชื่อมต่อ



รอรับ Auth Token  
ทางอีเมลที่เรากรอก  
ตอนสมัคร



Auth Token

ถูกส่งมาทางอีเมล



# ทดสอบการเชื่อมต่อกับเซิร์ฟเวอร์ Blynk ผ่านไวไฟ

- เชื่อมต่อไวไฟ หากไม่ได้ค่า True ให้ตรวจสอบ ssid

```
>>> import network
```

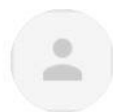
```
>>> station = network.WLAN(network.STA_IF)
```

```
>>> station.active(True)
```

```
>>> station.connect(ssid, password)
```

```
>>> station.isconnected()
```

```
True
```



**Blynk** <dispatcher@blynk.io> ยกเลิกการสมัคร

ถึง ฉัน ▾



อังกฤษ ▾



ไทย ▾

[แปลข้อความ](#)

Auth Token : gTuoJT18C7OhBGBChvp-qzcs8dklvQXn

Happy Blynking!

-

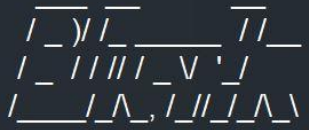
- ทดสอบพิมพ์ต่อไปนี้จะโดยใช้ Token ที่ได้รับใน email

```
>>> import blynklib_mp as bk
```

```
>>> BLYNK_AUTH = "gTuoJT18C7OhBGBChvpqzcs8dklvQXn"
```

```
>>> blynk = bk.Blynk(BLYNK_AUTH)
```

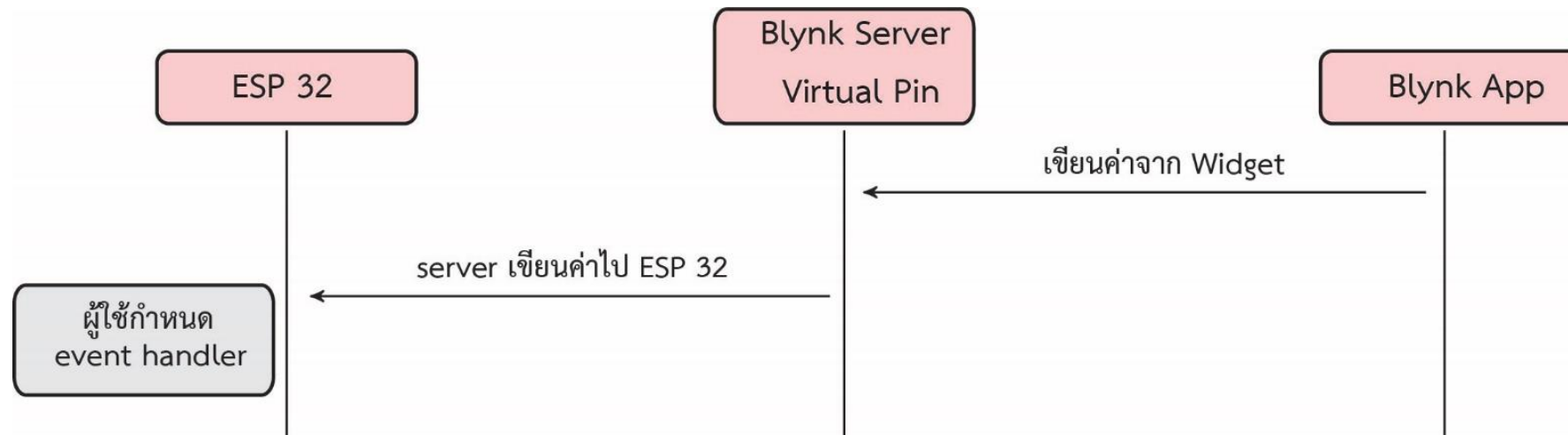
- หาก Logo ขึ้นแสดงว่า การเชื่อมต่อถูกต้อง หากไม่ได้ให้ตรวจสอบ Token อีกครั้ง



for Python v0.2.6

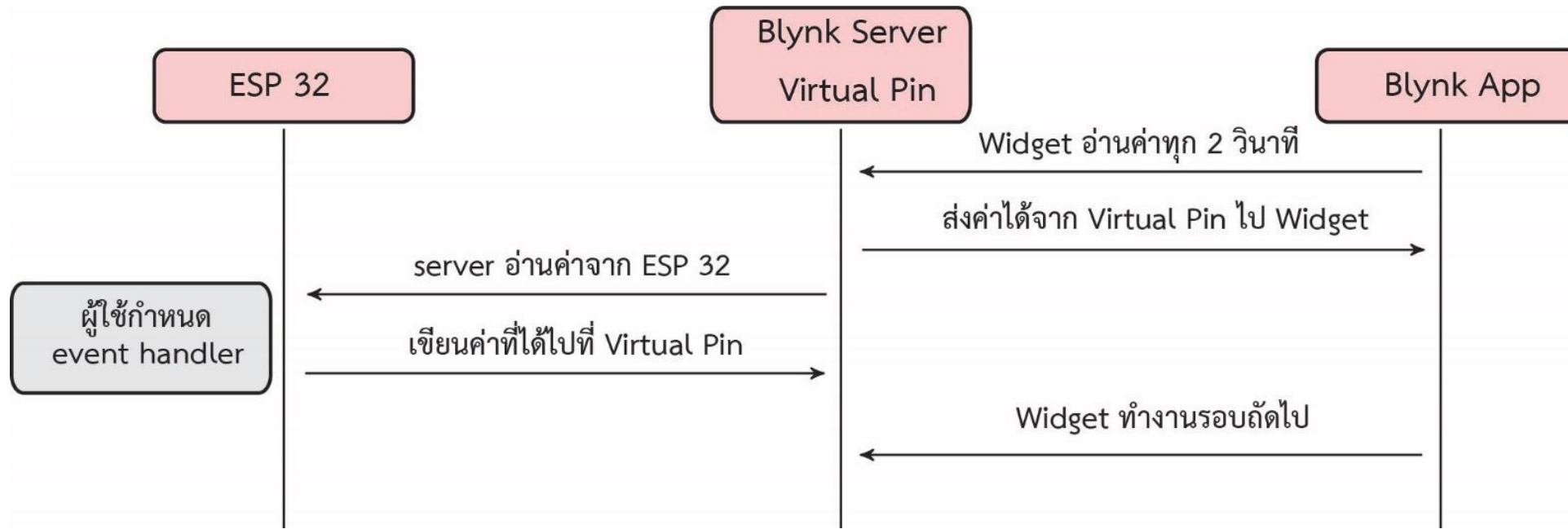
# การสื่อสารระหว่าง Blynk App และอุปกรณ์

- การส่งข้อมูลจาก Blynk App ไปยังอุปกรณ์



```
@blynk.handle_event('write V1')
def write_virtual_pin_handler(pin, value):
    print("Sever->esp32".format(pin, value))
```

# การส่งข้อมูลจากอุปกรณ์ไปยัง Blynk App

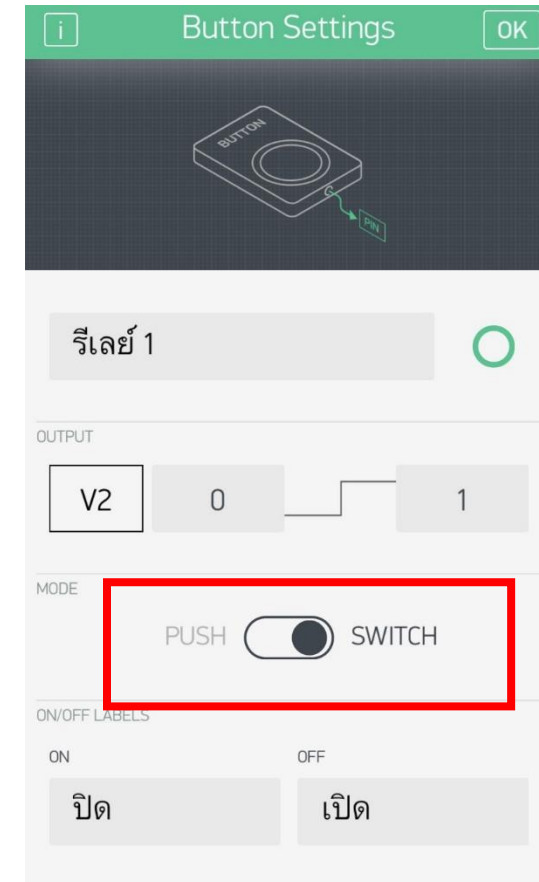
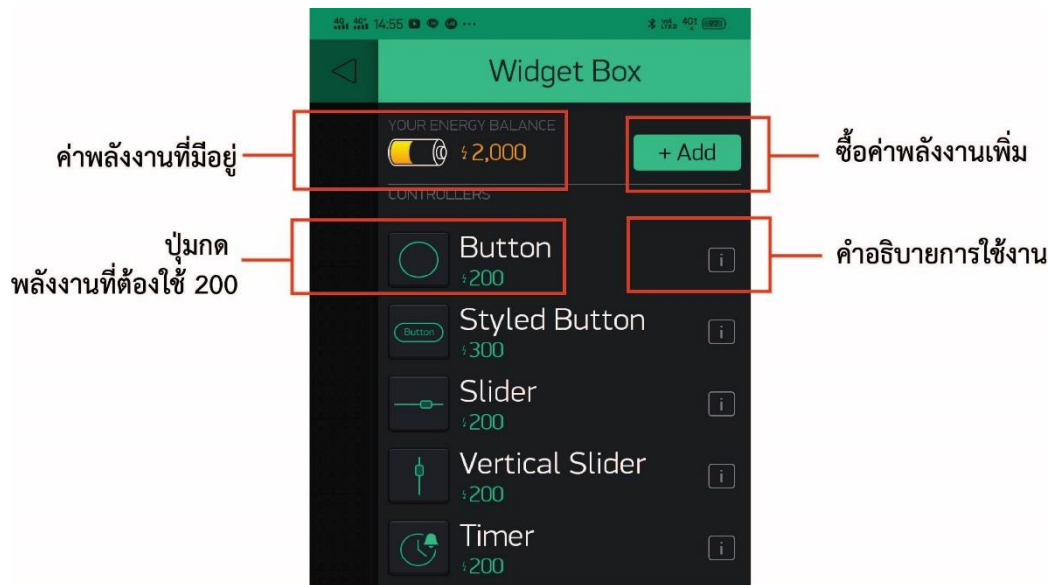


```
@blynk.handle_event('read V0')
def read_virtual_pin_handler(pin):
    print("esp32->server".format(pin))
    blynk.virtual_write(pin, random.randint(0, 255))
```

# สร้าง Gauge และระบบรอก



- กำหนดระบบ
  - V0 เป็นค่าของอุณหภูมิ
  - V1 เป็นค่าของความชื้น
  - V2-V5 เป็นรีเลย์ 1-4 ตามลำดับ



simple\_1\_blynk.py

```
1  import network
2  import blynklib_mp as bk
3  import random
4
5  # connect wifi
6  wifi = network.WLAN(network.STA_IF)
7  wifi.active(True)
8  wifi.connect("ssid", "password")
9
10 print("connected")
11 #connect Blynk
12 BLYNK_AUTH = 'gTuoJT18C70hBGBChvp-qzcs8dklvQXn'
13 blynk = bk.Blynk(BLYNK_AUTH)
14
15
16 @blynk.handle_event('read V0') # Guage Temperature
17 def read_virtual_pin_handler(pin):
18     print("ESP->Server")
19     blynk.virtual_write(pin, random.randint(0, 100))
20
21 @blynk.handle_event('write V2') # button 1
22 def write_virtual_pin_handler(pin, value):
23     print("Server->ESP".format(pin, value))
24
25 print("OK start");
26 while True:
27     blynk.run()
```

- เปลี่ยนค่าเป็น ssid
- เปลี่ยนค่า password
- เปลี่ยนค่า Token
- ทดสอบด้วยค่า random ก่อน

```

simple_1_blynk_with_1_relay.py
1  import network
2  import blynklib_mp as bk
3  import random
4  from machine import Pin
5
6  # conect wifi
7  wifi = network.WLAN(network.STA_IF)
8  wifi.active(True)
9  wifi.connect("ssid", "password")
10 #add relay ..
11 relay1 = Pin(26, Pin.OUT)
12
13 print("conected")
14 #connect Blynk
15 BLYNK_AUTH = 'gTuoJT18C70hBGBChvp-qzcs8dklvQXn'
16 blynk = bk.Blynk(BLYNK_AUTH)
17
18 print("authenOK")
19 @blynk.handle_event('read V0') # Guage Temperature
20 def read_virtual_pin_handler(pin):
21     print("ESP->Server")
22     blynk.virtual_write(pin, random.randint(0, 255))
23
24 @blynk.handle_event('write V2') # button 1
25 def write_virtual_pin_handler(pin, value):
26     print("Server->ESP".format(pin, value))
27     # check the value here
28     if(int(value[0]) == 0):
29         print("on {0}".format(value[0]))
30         relay1.value(0)
31     else:
32         print("off {0}".format(value[0]))
33         relay1.value(1)
34
35 print("OK start");
36 while True:
37     blynk.run()
38

```

- เพิ่มรีเลย์ 1 ตัว
- เพิ่มส่วนการตรวจสอบค่าที่ได้รับได้



```

12
13 V2 = Pin(26, Pin.OUT)
14 V3 = Pin(27, Pin.OUT)
15 V4 = Pin(32, Pin.OUT)
16 V5 = Pin(33, Pin.OUT)
17
18 relay = [V2, V3, V4, V5]
19 relay_name = ['V2', 'V3', 'V4', 'V5']
20
21 print("conected")
22 #connect Blynk
23 BLYNK_AUTH = 'gTuoJT18C70hBGBChvp-qzcs8dklvQXn'
24 blynk = bk.Blynk(BLYNK_AUTH)
25
26 print("authen OK")
27
28 def relay_onoff(relay_no, onoff):
29     print("func Pin: {} value {}".format(relay_no, onoff))
30     relay[relay_name.index(str(relay_no))].value(int(onoff))
31
32 @blynk.handle_event('read V*') # Guage Temperature
33 def read_virtual_pin_handler(pin):
34     print("ESP -> Server V{}".format(pin))
35     blynk.virtual_write(pin, random.randint(0, 100))
36
37
38 @blynk.handle_event('write V*') # button 1
39 def write_virtual_pin_handler(pin, value):
40     print("Server->ESP Write V{} Value{} ".format(pin, value))
41     relay_onoff("V{}".format(pin), "{}".format(value[0]))
42
43 print("OK start");
44 while True:
45     blynk.run()

```

- เขียนเป็นฟังก์ชัน
- ใช้ค่า index

```
# init sensor and display
sensor = dht.DHT22(Pin(16))
i2c = SoftI2C(scl=Pin(22), sda=Pin(21))
display = ssd1306.SSD1306_I2C(128, 64, i2c)
adc35 = ADC(Pin(35))
```

```
print("conected")
#connect Blynk
BLYNK_AUTH = 'gTuoJT18C70hBGBChvp-qzcs8dklvQXn'
blynk = bk.Blynk(BLYNK_AUTH)

print("authen OK")
```

```
def relay_display():
    relay_text1 = "relay 1:{} 2:{}".format(relay_state[0], relay_state[1])
    relay_text2 = "          3:{} 4:{}".format(relay_state[2], relay_state[3])
    display.text(relay_text1, 0, 40)
    display.text(relay_text2, 0, 50)
    display.show()
```

```
def reading_sensor():
    print("sensor reading")
    display.fill(0)
    sensor.measure()
    temp = sensor.temperature()
    humi = sensor.humidity()
    soil = adc35.read()
    display.text("Temp %3.1f" %temp, 0, 0)
    display.text("Humi %3.1f" %humi, 0, 10)
    display.text("Soil %d" %soil, 0, 20)
    blynk.virtual_write(0, temp)
    blynk.virtual_write(1, humi)
    blynk.virtual_write(6, soil)
    relay_display()
    display.show()
```

- เพิ่มเซ็นเซอร์ และการแสดงผลออก OLED