

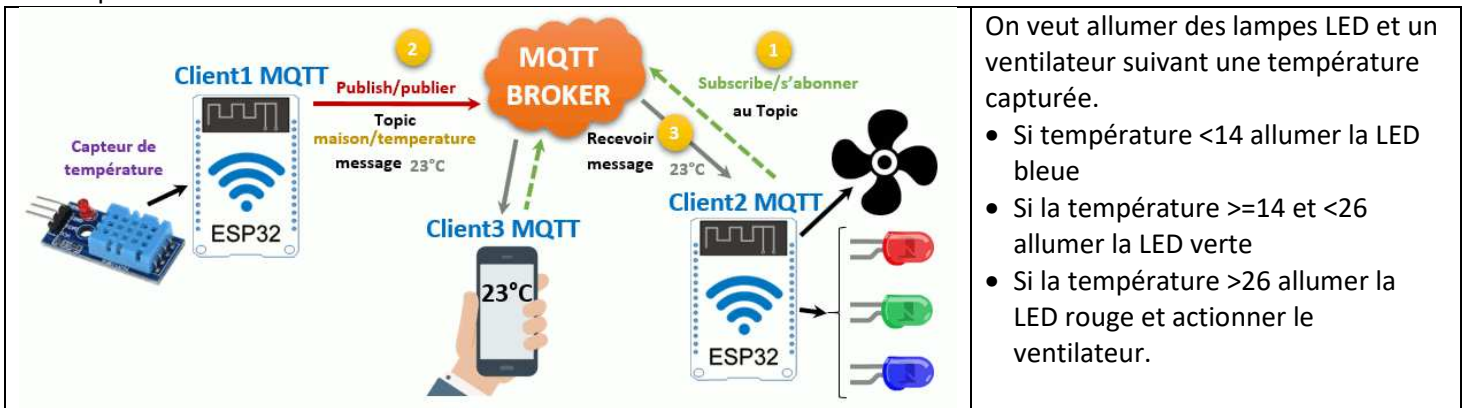
## Mini-projet 3 : (IoT)

Le protocole **MQTT** (Message Queuing Telemetry Transport) est un protocole de connectivité de machine-to-machine (**M2M**) utilisé dans l'**IoT**(Internet of Things). Il s'agit d'un système de publication et d'abonnement (publish and subscribe) léger où vous pouvez publier et recevoir des messages en tant que client, avec une faible bande passante. C'est donc la solution parfaite pour les applications de l'Internet des objets. Ci-après, les concepts de base :

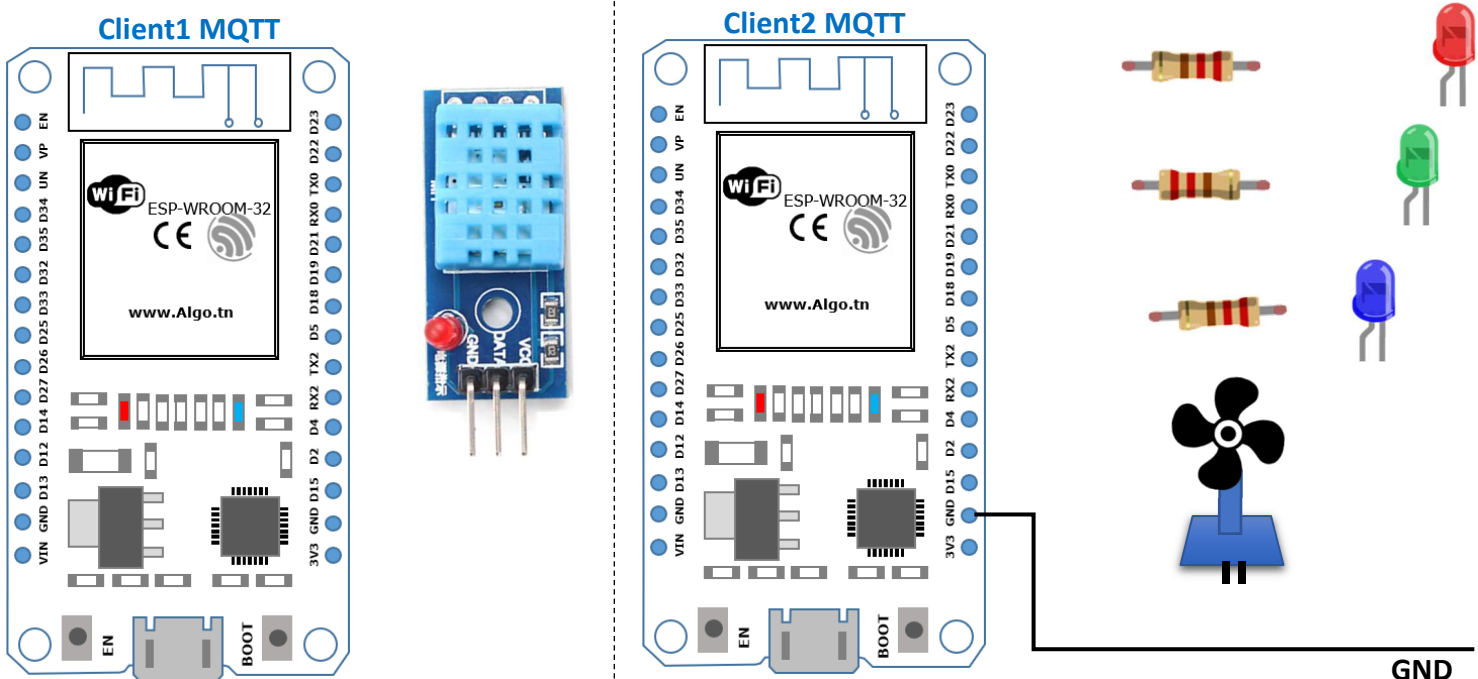
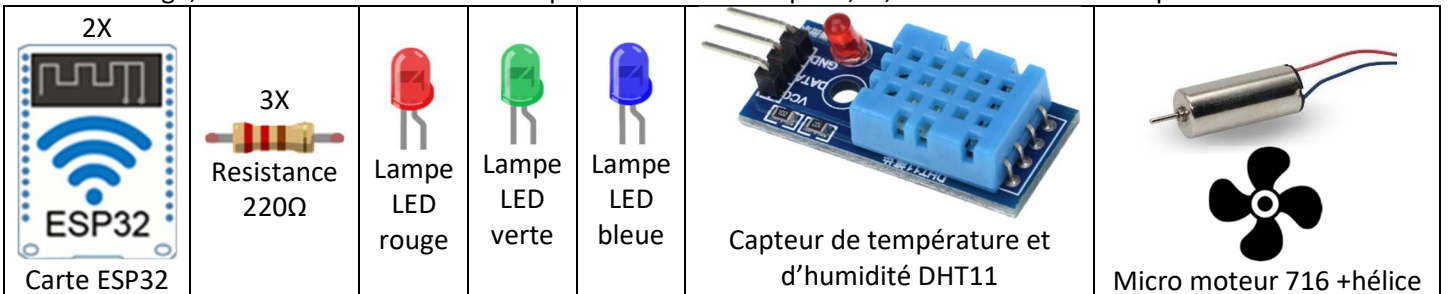
# MQTT

Broker	Topic(sujets)	Publish/Subscribe	Exemple :
Le broker(courtier/intermédiaire) est principalement chargé de recevoir tous les messages, de les filtrer, de décider qui les intéresse et de publier le message à tous les clients abonnés.	Le Topic permet de spécifier où vous souhaitez publier le message.	(Publier / s'abonner) un objet peut publier un message sur un sujet, ou il peut être abonné à un sujet particulier pour recevoir des messages	-Le client1 MQTT publie sur un sujet(Topic). -Le client2 MQTT est abonné au même sujet que le client1 MQTT. -Ainsi, le client2 MQTT reçoit le message.
	<b>Messages</b> Les messages (commandes/données) sont les informations que vous souhaitez échanger entre vos appareils.		

Principe :



**Etape 1: Montage :** En utilisant les composants ci-dessous , réaliser un montage permettant d'allumer les LEDs et d'actionner le moteur suivant la température capturée. La capture de la température doit se faire sur le pin4. Les LED rouge, verte et bleue à brancher respectivement sur les pin 5,18,19 et le moteur sur le pin 15

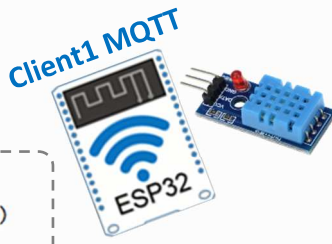


## Etape 2: Programmation de la carte ESP32 (Client1 MQTT)

Soit le programme suivant (main.py) à téléverser à la carte ESP32.

```

1 from machine import Pin
2 import network
3 from mqtt import MQTTClient
4 import time
5 import dht
6 import gc
7 gc.collect()
8 ssid = 'NOM_WIFI'
9 password = 'MOT_DE_PASSE_WIFI'
10 station = network.WLAN(network.STA_IF)
11 station.active(True)
12 station.connect(ssid, password)
13 while station.isconnected() == False:
14     pass
15 print('connexion établie')
16 print(station.ifconfig())
17
18 def settimeout(duration):
19     pass
20 client = MQTTClient("demo", server="farmer.cloudmqtt.com",
21                    user="fizwwsid", password="RpBWoH3wZFy2", port=10193)
22 client.settimeout = settimeout
23 client.connect()
24
25 sensor = dht.DHT11(Pin(4))
26
27 while True:
28     sensor.measure()
29     temp = sensor.temperature()
30     print("Envoyer la température=", temp)
31     client.publish("maison/temperature", str(temp))
32     time.sleep(5)
    
```



1. Dans la partie ① :
  - Modifier le nom du WiFi et le mot de passe pour permettre à la carte ESP32 de se connecter à Internet.
2. Dans la partie ② :
  - Déterminer le **MQTT broker** et son type (local/distant) .....
3. Dans la partie ③ :
  - Donner les numéros de lignes permettant la capture puis le stockage de la température dans une variable....., Donner le nom de cette dernière..
  - Donner le numéro de la ligne permettant de **publier** la température comme **message** sur le topic ...., en déduire le nom du **Topic**.....

Connexion au MQTT broker  
Cloudmqtt.com

Configuration du pin4 comme pin de lecture de température à partir de DHT11

- While True : permet de répéter indéfiniment un bloc d'instructions
- Lecture de la température et enregistrement dans la variable temp
- Publier la température(temp) sur le Broker dans le Topic **maison/temperature** tout le 5 secondes

## Etape 3: Programmation de la carte ESP32 (Client2 MQTT)

On veut compléter le programme suivant (main.py) pour le téléverser finalement à la carte ESP32. (Client2 MQTT)

```

1 import network
2 from mqtt import MQTTClient
3 import machine
4 import time
5 from machine import Pin
6 import gc
7 gc.collect()
8
9 ssid = 'NOM_WIFI'
10 password = 'MOT_DE_PASSE_WIFI'
11 def settimeout(duration):
12     pass
13 station = network.WLAN(network.STA_IF)
14 station.active(True)
15 station.connect(ssid, password)
16 while station.isconnected() == False:
17     pass
18 print('Connection successful')
19 print(station.ifconfig())
20 print("Connecté au Wifi\n")
    
```

```

21 topic_sub = b'maison/temperature'
22 mserver='farmer.cloudmqtt.com'
23
24 rouge=Pin(5,Pin.OUT)
25 vert=Pin(18,Pin.OUT)
26 bleu=Pin(19,Pin.OUT)
27 moteur=Pin(15,Pin.OUT)
28 def sub_cb(topic, msg):
29     print((topic, msg))
30     temp=int(msg.decode())
31     ..... |
32
33 client = MQTTClient("demo", server="farmer.cloudmqtt.com",
34                    user="fizwwsid", password="RpBWoH3wZFy2", port=10193)
35 client.set_callback(sub_cb)
36 client.connect()
37 client.subscribe(topic_sub)
38 print('Connecté au %s MQTT broker, abonné au %s topic' % (mserver, topic_sub))
39
40 while True:
41     new_message = client.check_msg()
    
```

1. Après avoir être connecté au MQTT broker, donner le numéro de la ligne permettant de **s'abonner au Topic** ....
2. Une fois abonné, on peut recevoir les **messages** du Topic (la température **temp**),
3. Compléter le programme par les conditions et les traitements nécessaires pour allumer les LEDs et actionner le moteur suivant cette température. (Voir ④)

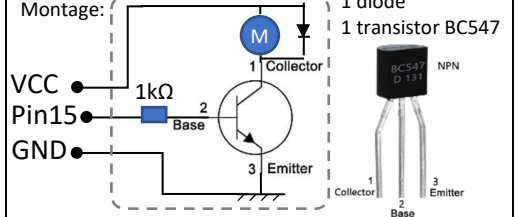
Remarque: On doit aussi téléverser la bibliothèque mqtt.py téléchargeable au lien :

<https://raw.githubusercontent.com/pycom/pycom-libraries/master/examples/mqtt/mqtt.py>

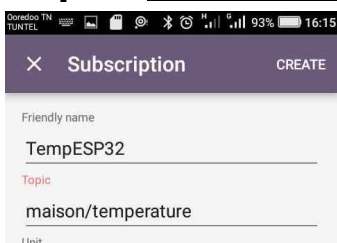
On propose l'algorithme suivant :  
**Si temp < 14 alors** allumer LED Bleue  
**Sinon si temp < 26 alors**  
 Allumer LED Verte  
**Sinon**  
 Allumer LED Rouge  
 Actionner le moteur  
**Finsi**

Traduire l'algorithme précédent en MicroPython :

**NB :** Il vaut mieux connecter le moteur au pin 15 avec le montage suivant :



## Etape 4: Installation d'une application mobile : (Client3 MQTT)



Installer l'application mobile **IoT MQTT Dashboard** à partir du **Play Store** pour se connecter au broker et s'abonner au Topic pour recevoir la température capturée sur son smartphone. **NB :** Pour se connecter au broker, utiliser les paramètres suivants : `server="farmer.cloudmqtt.com"`, `port=10193`, `username="fizwwsid"`, `password="RpBWoH3wZFy2"`

