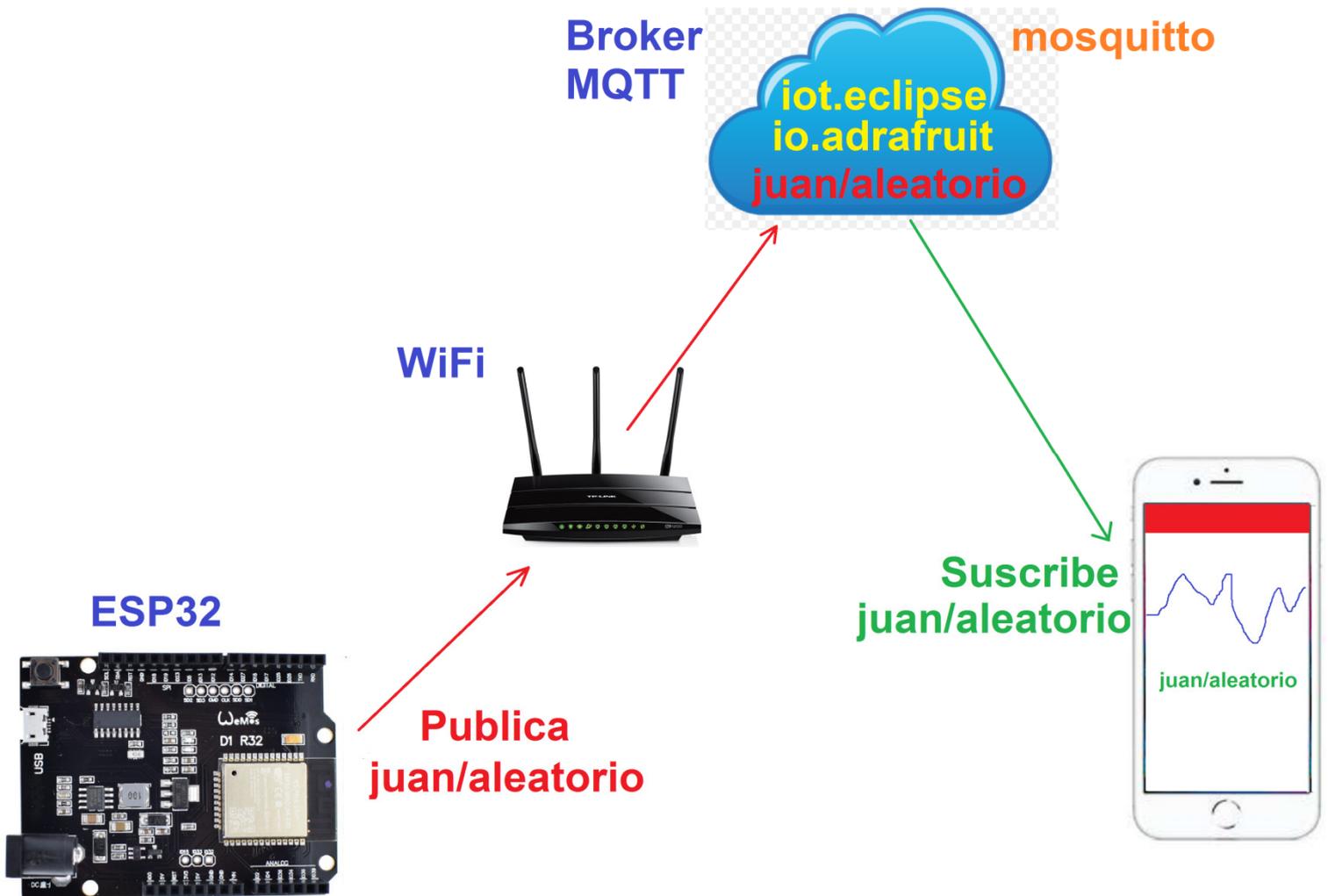


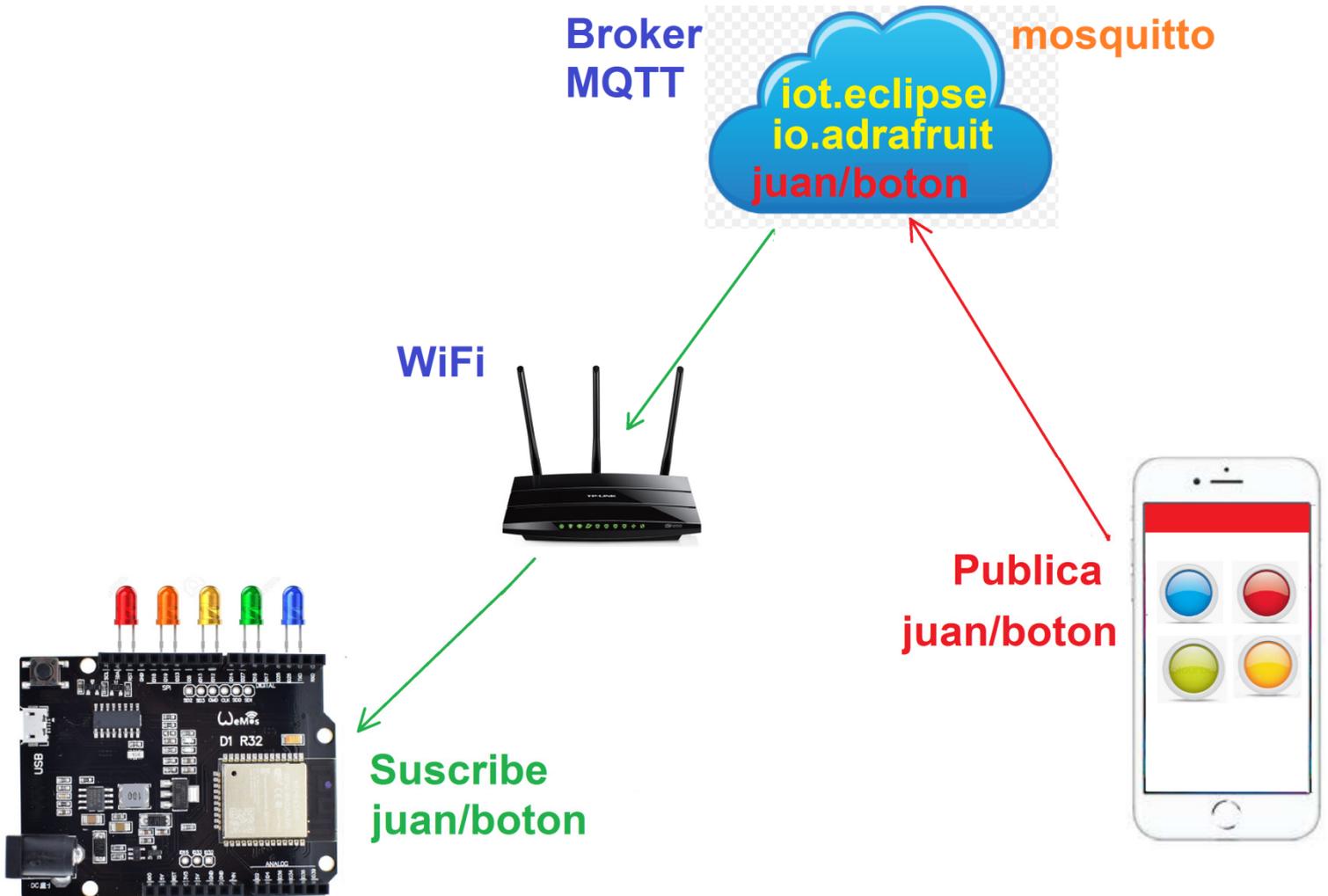
# Publicar desde Arduino a la aplicación del móvil.

- Arduino envía por WiFi un valor cada cierto tiempo al Broker MQTT.
- La aplicación del móvil está **suscrita** a ese valor y lo recibe.



# Publicar desde la aplicación del móvil al Arduino.

- La aplicación del móvil envía el estado de los botones al Broker MQTT.
- El Arduino está suscrito a esa variable y lo recibe.



## - IOT con el ESP8266-01+ módulo

### ESP8266-01 + Módulo adaptador



- Es muy barato se puede encontrar por 1€ en Internet.
- Se puede acoplar fácilmente al Arduino UNO.
- Puede funcionar con los bloques de la web:

<http://www.arduinoblocks.com>

- Desde el Arduino podemos **Publicar** valores que los recibirá mediante **Suscripción**, una aplicación en el móvil.

The image shows a Scratch-style code editor with two main sections: 'Inicializar' and 'Bucle'.  
In the 'Inicializar' section, there is a block titled 'Iniciar (Esp8266 WiFi)' with the following settings: Rx: 1-TX, Tx: 0-RX, Baudios: 115200, WiFi red: Nombre\_de\_la\_Red\_WiFi, clave: Clave\_WiFi, Broker: iot.eclipse.org, Puerto: 1883, Cliente Id: AB\_juan, Usuario: (checkbox), Clave: (checkbox), and Establecer aleatorio = 0.  
The 'Bucle' section contains a loop that runs every 5000 ms. Inside the loop, it sets 'aleatorio' to a random integer between 1 and 90, and then publishes the topic 'juan/aleatorio' with the value of 'aleatorio'.

- También podemos enviar información (**Publicar**) desde el móvil al Arduino (**Suscribir**).

The image shows a Scratch-style code editor with two main sections: 'Inicializar' and 'Bucle'.  
In the 'Inicializar' section, there is a block titled 'Iniciar (Esp8266 WiFi)' with the following settings: Rx: 1-TX, Tx: 0-RX, Baudios: 115200, WiFi red: Nombre\_de\_la\_Red\_WiFi, clave: Clave\_WiFi, Broker: io.eclipse.org, Puerto: 1883, Cliente Id: AB\_juan, Usuario: (checkbox), Clave: (checkbox), and Suscribir Tema: 'juan/led4' > poner\_led4.  
The 'Bucle' section contains a loop with two conditional blocks. The first block is 'si poner\_led4 == 1' followed by 'hacer Escribir digital Pin 4 On'. The second block is 'si poner\_led4 == 0' followed by 'hacer Escribir digital Pin 4 Off'.

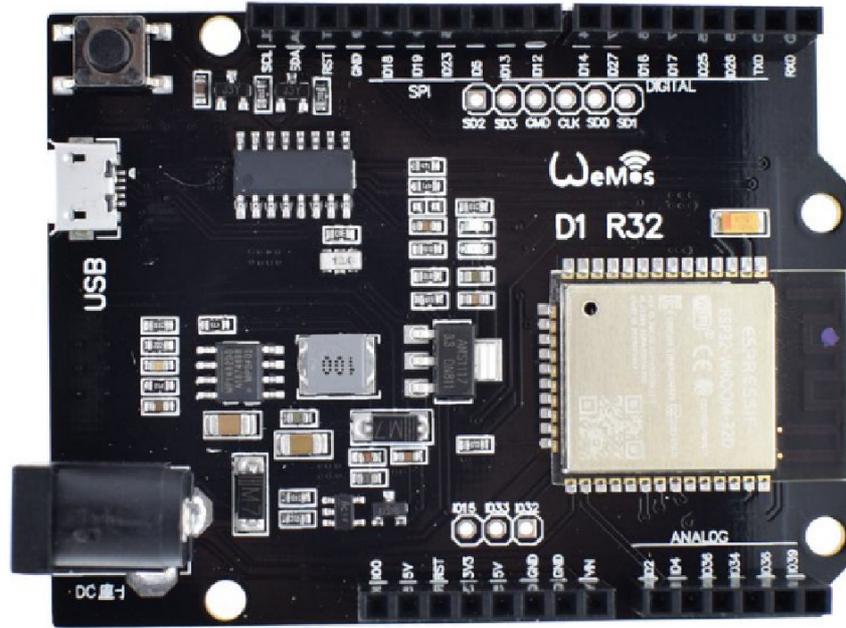
## - IOT con el ESP8266 D1 R2



- Se puede encontrar por 5 € en Internet.
- Tiene WiFi.
- Solo una entrada analógica.
- Funciona con las librerías de Arduino.
- Funciona con Guadalinex.
- Se puede programar con el IDE de Arduino.

```
esp8266 Arduino 1.8.2
Archivo Editar Programa Herramientas Ayuda
esp8266_thinspeak_3b $
1 #include <ESP8266WiFi.h> // Para el ESP8266
2 WiFiClient WIFI_CLIENT;
3 #include <PubSubClient.h>
4 PubSubClient MQTT_CLIENT;
5
6 // Nombre y contraseña de tu red WiFi.
7 const char* ssid = "Nombre_de_Red_WiFi";
8 const char* password = "Contraseña_WiFi";
9
10 void setup() {
11   Serial.begin(115200);
12   Serial.println();
13   Serial.print("Conectando con ");
14   Serial.println(ssid);
15
16   WiFi.begin(ssid, password);
17
18   while (WiFi.status() != WL_CONNECTED) {
19     delay(500);
20     Serial.print(" ");
21   }
Guardado.
```

## - IOT con el ESP32



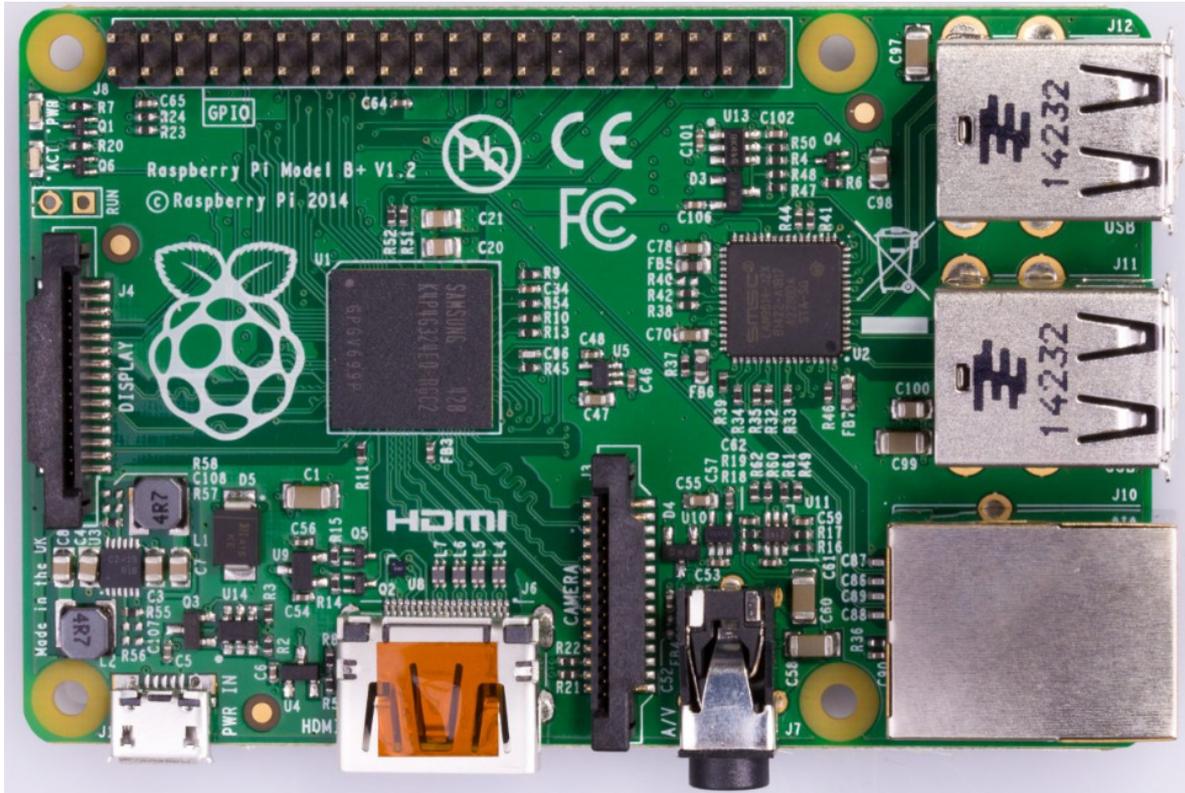
- Se puede encontrar por 6 € en Internet.
- Tiene WiFi.
- Tiene Bluetooth clásico y BLE.
- Muchas entradas analógicas.
- OTA. RTOS. DAC. Sensores de contacto.
- Se puede programar con el IDE de Arduino.

```
esp8266 Arduino 1.8.2
Archivo Editar Programa Herramientas Ayuda
esp8266_thinspeak_3b $
1 #include <ESP8266WiFi.h> // Para el ESP8266
2 WiFiClient WIFI_CLIENT;
3 #include <PubSubClient.h>
4 PubSubClient MQTT_CLIENT;
5
6 // Nombre y contraseña de tu red WiFi.
7 const char* ssid = "Nombre_de_Red_WiFi";
8 const char* password = "Contraseña_WiFi";
9
10 void setup() {
11   Serial.begin(115200);
12   Serial.println();
13   Serial.print("Conectando con ");
14   Serial.println(ssid);
15
16   WiFi.begin(ssid, password);
17
18   while (WiFi.status() != WL_CONNECTED) {
19     delay(500);
20     Serial.print(".");

```

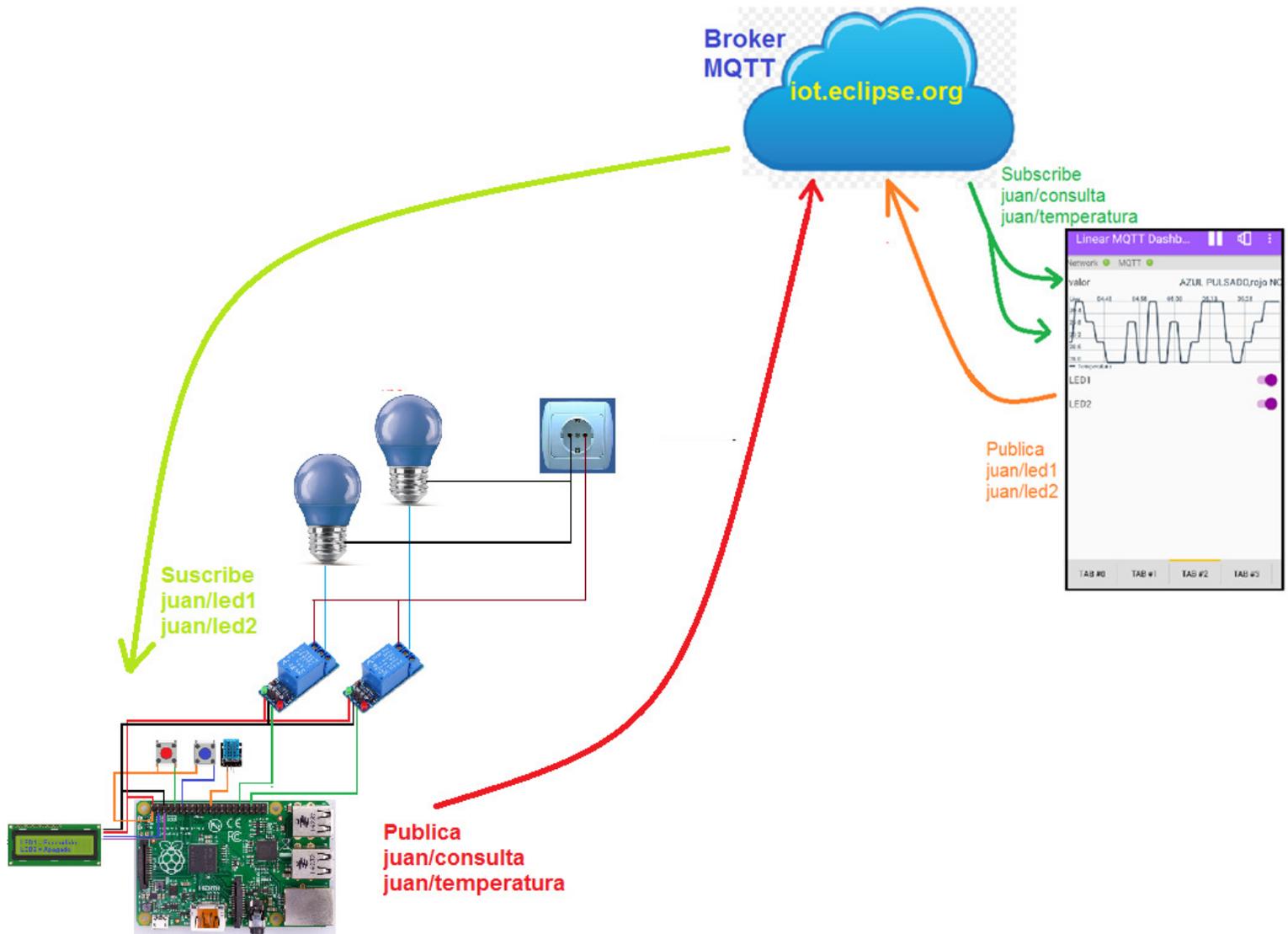
Guardado.

## - IOT con el Raspberry Pi



- Se puede encontrar por 36 €
- Mini ordenador con Linux.
- Conector para monitor HDMI
- Ratón y teclado USB (o BT)
- WiFi y Bluetooth.
- I2C.
- Se puede programar con Python.
- Podemos instalar el Broker MQTT Mosquitto.

# IOT con Raspberry Pi



## 1.- Publicación de valores del Raspberry Pi y suscripción en una aplicación.

- Raspberry Pi con el módulo DHT-11, sensor de temperatura-humedad, enviará esos valores mediante WiFi al Broker MQTT [iot.eclipse.org](http://iot.eclipse.org).
- También dispondremos de dos pulsadores, sus estados también se enviará al Broker.

## 2.- Suscripción en una aplicación de Android.

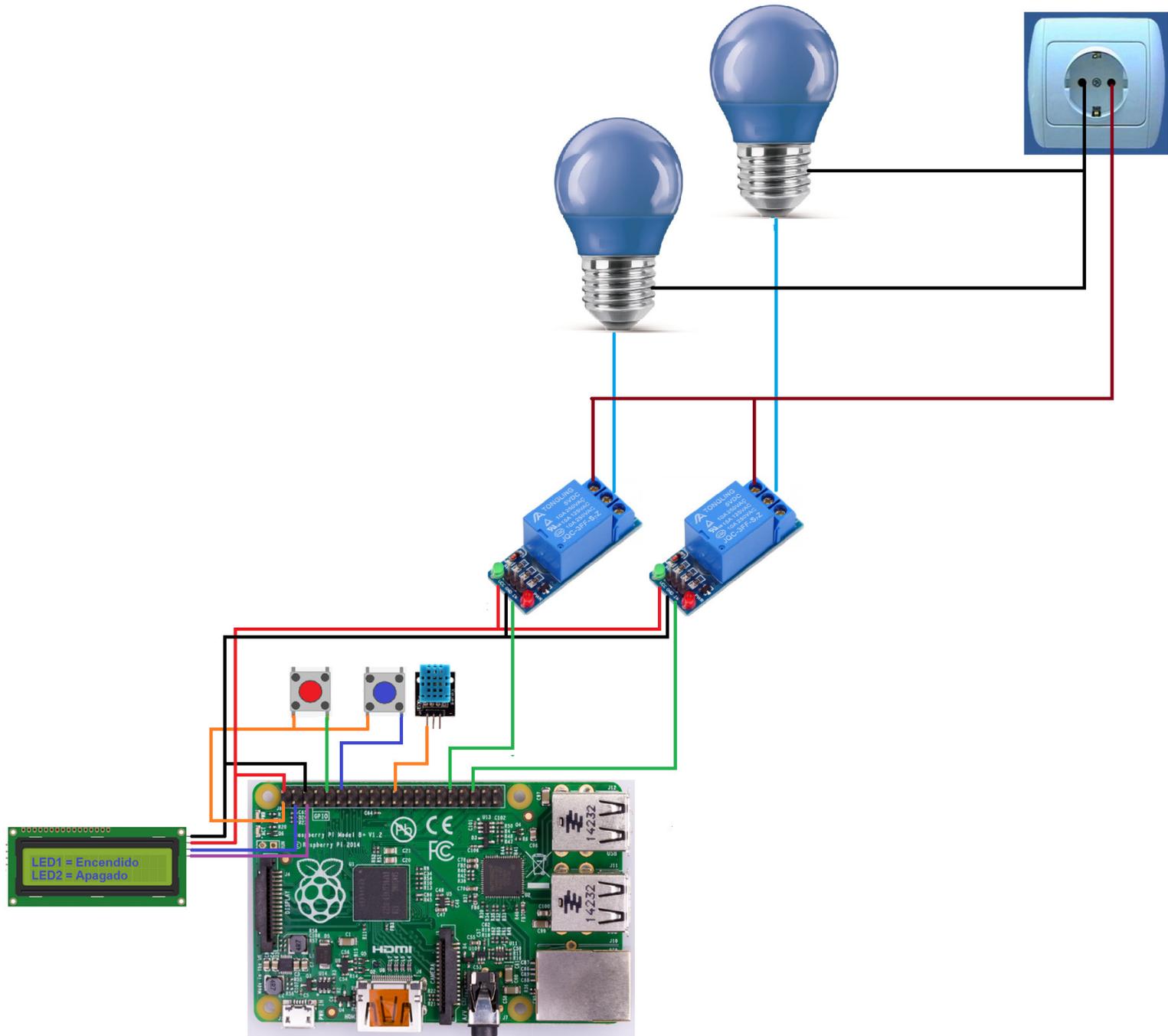
- Instalaremos la aplicación de Android Linear MQTT Dashboard.
- La aplicación se suscribirá a los valores publicados por nuestro Raspberry Pi en [iot.eclipse.org](http://iot.eclipse.org).
- Esos valores se mostrarán en nuestra aplicación mediante gráficas y textos.

## 3.- Publicación de valores en una aplicación y suscripción en la Raspberry Pi.

- La aplicación Linear MQTT Dashboard, tiene configurado dos Botones, al pulsarlos se publicará su estado en el broker [iot.eclipse.org](http://iot.eclipse.org).

## 4.- Suscripción del Raspberry Pi a la aplicación.

- Raspberry Pi estará suscrito al Broker y captará el estado de esos botones, según sus estados, encenderá o apagará dos lámparas. Además mostrará un mensaje en una pantalla LCD con bus I2C conectada al Raspberry Pi.



18:59 📶 🔋 41%

### Linear MQTT Dashboard SAVE

Connection settings to the MQTT server  
Server ('tcp://', 'ssl://' or 'tls://' prefixes available)

tcp://iot.eclipse.org

Port  
1883

Username

Password

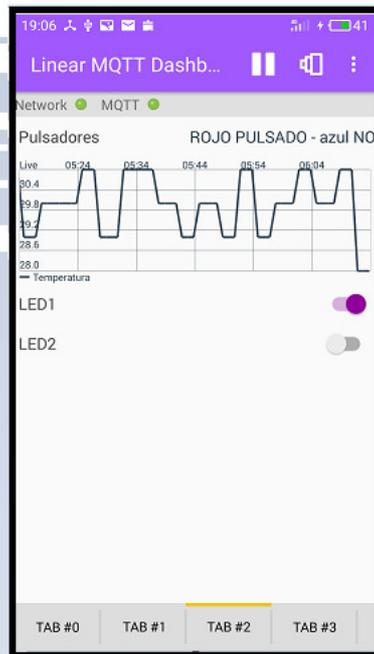
Push notifications subscribe topic ?  
juan/notifica

Keep connection in the background for receiving push notifications

Other settings  
 Application server mode ?

Server topic

View



18:44 📶 🔋 42%

### Linear MQTT Dashboard SAVE

Widget type Value ▼ Any ▼

Name Pulsadores

Sub.topic juan/consulta ⬆

Pub.topic (optional)

OnReceive() ?

OnShow() ?

18:44 📶 🔋 42%

### Linear MQTT Dashboard SAVE

Widget type Graph ▼ Live ▼

Name Temperatura

Sub.topic juan/temperatura ⬆

Name 2

Topic 2 ⬆

Name 3

Topic 3 ⬆

Name 4

Topic 4 ⬆

OnReceive() ?

18:44 📶 🔋 42%

### Linear MQTT Dashboard SAVE

Widget type Switch ▼

Name LED1

Sub.topic

Pub.topic (optional) juan/led1

On value a      Off value (optional) b

OnReceive() ?

18:44 📶 🔋 42%

### Linear MQTT Dashboard SAVE

Widget type Switch ▼

Name LED2

Sub.topic

Pub.topic (optional) juan/led2

On value c      Off value (optional) d

OnReceive() ?