

IoT DE COSTE REDUCIDO EN EL IES



INDICE

- INTRODUCCIÓN
- CARACTERÍSTICAS PRINCIPALES DEL HARDWARE
- ENTORNOS DE DESARROLLO
- MATERIAL NECESARIO
- RECURSOS Y COMUNIDAD
- EJEMPLOS DE PROYECTOS INTERESANTES
- CONCLUSIONES Y PREGUNTAS

INTRODUCCIÓN

- MOTIVACIÓN PARA LOS ESTUDIANTES
- EJEMPLOS DEL MUNDO REAL
- TECNOLOGIAS EMERGENTES
- BAJO COSTE
- DESARROLLO DE PROYECTOS EN LOS IES

#Imagina #Conecta #Programa #Crea

INTRODUCCIÓN AL HARDWARE

- El ESP32 es un pequeño "cerebro electrónico" capaz de conectarse a internet, recibir señales de sensores, controlar luces, motores, etc. Es ideal para aprender a programar y construir proyectos reales.
- Es una de las mejores soluciones para proyectos DIY de Internet de las cosas y proyectos de hogares inteligentes.
- Cuesta unos 6€.



CARACTERISTICAS PRINCIPALES

- Es una serie de microcontroladores en un chip desarrollado por
- Bajo coste
- Bajo consumo (podemos ponerlo en modo ahorro de energía)
- Capacidades de conectividad wireless (Wifi y Bluetooth)
- Dual-core (2 microprocesadores Xtensa 32-bit LX6)
- Múltiples pines para conectar sensores y actuadores
- Compatible con el lenguaje de programación de Arduino (C++)
- Compatible con micropython
- Incluso se puede programar por bloques



Placas de desarrollo

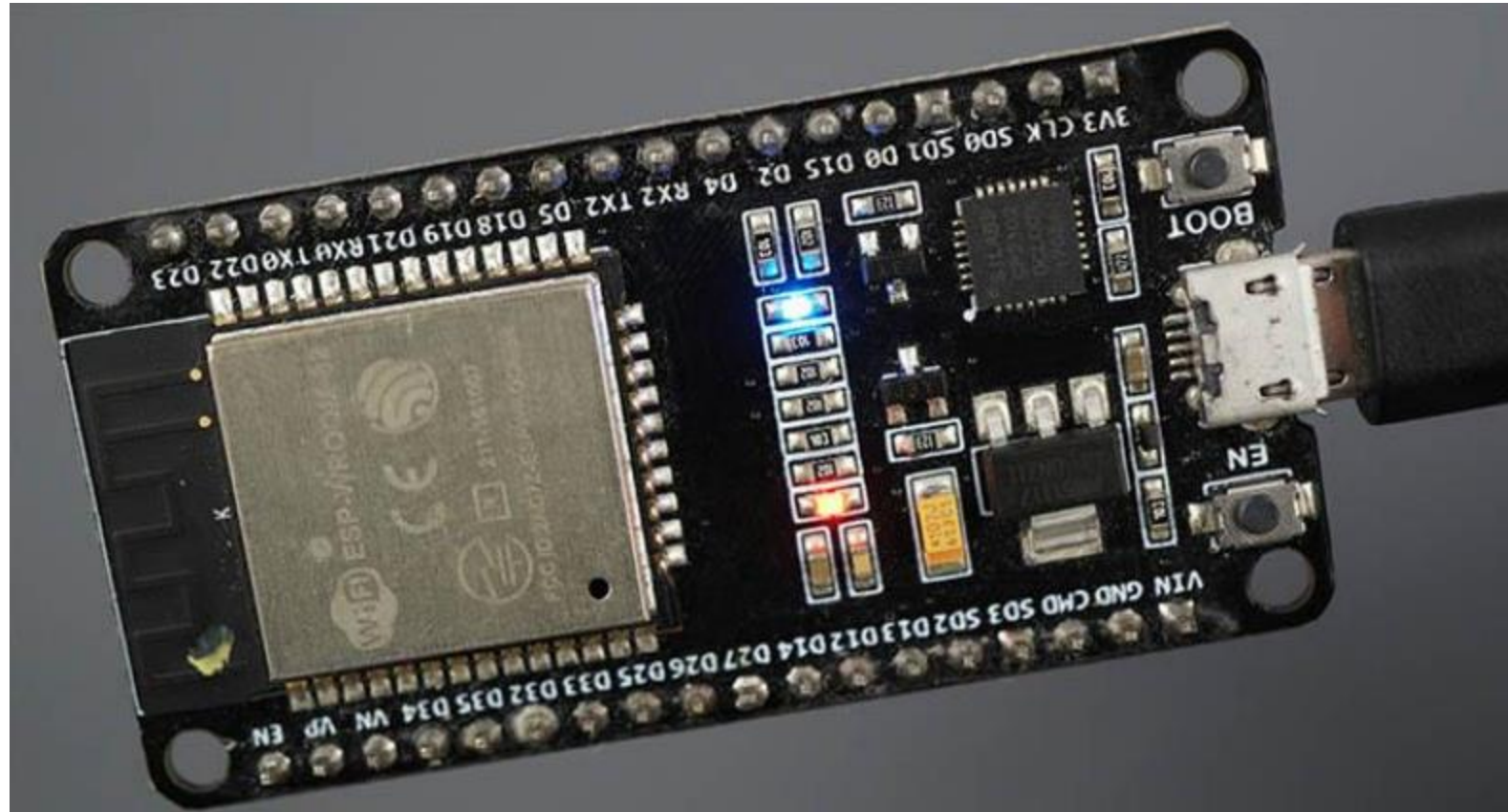
Vienen con todos los circuitos necesarios para alimentar y programar el chip, conectarlo al PC, pines para conectar periféricos, LED de encendido y control integrados, una antena para señal wifi y bluetooth además de otras funciones útiles.



ESP32 DEVKIT V1 DOIT (POPULAR)



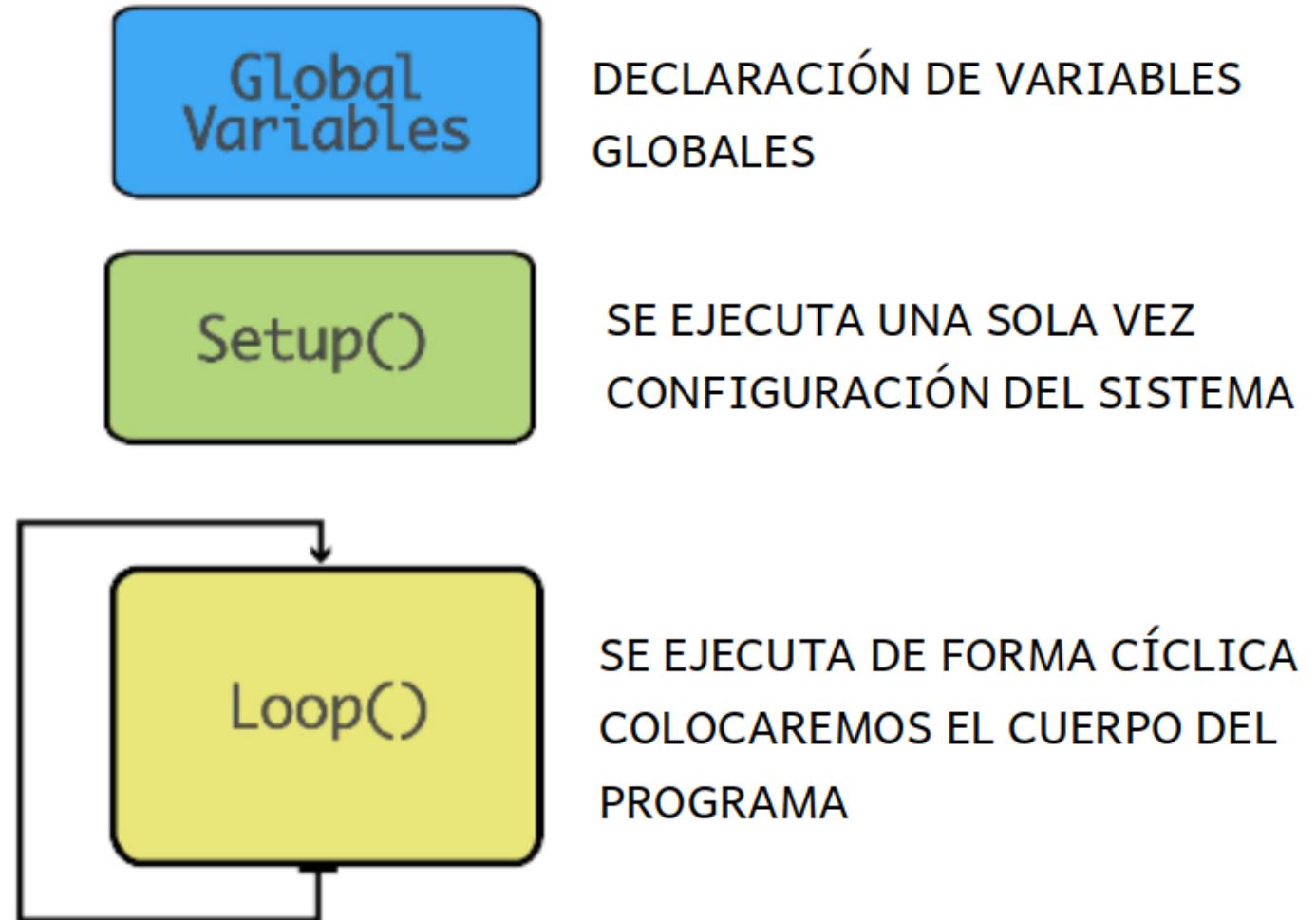
ESP32 DEVKIT V1 DOIT



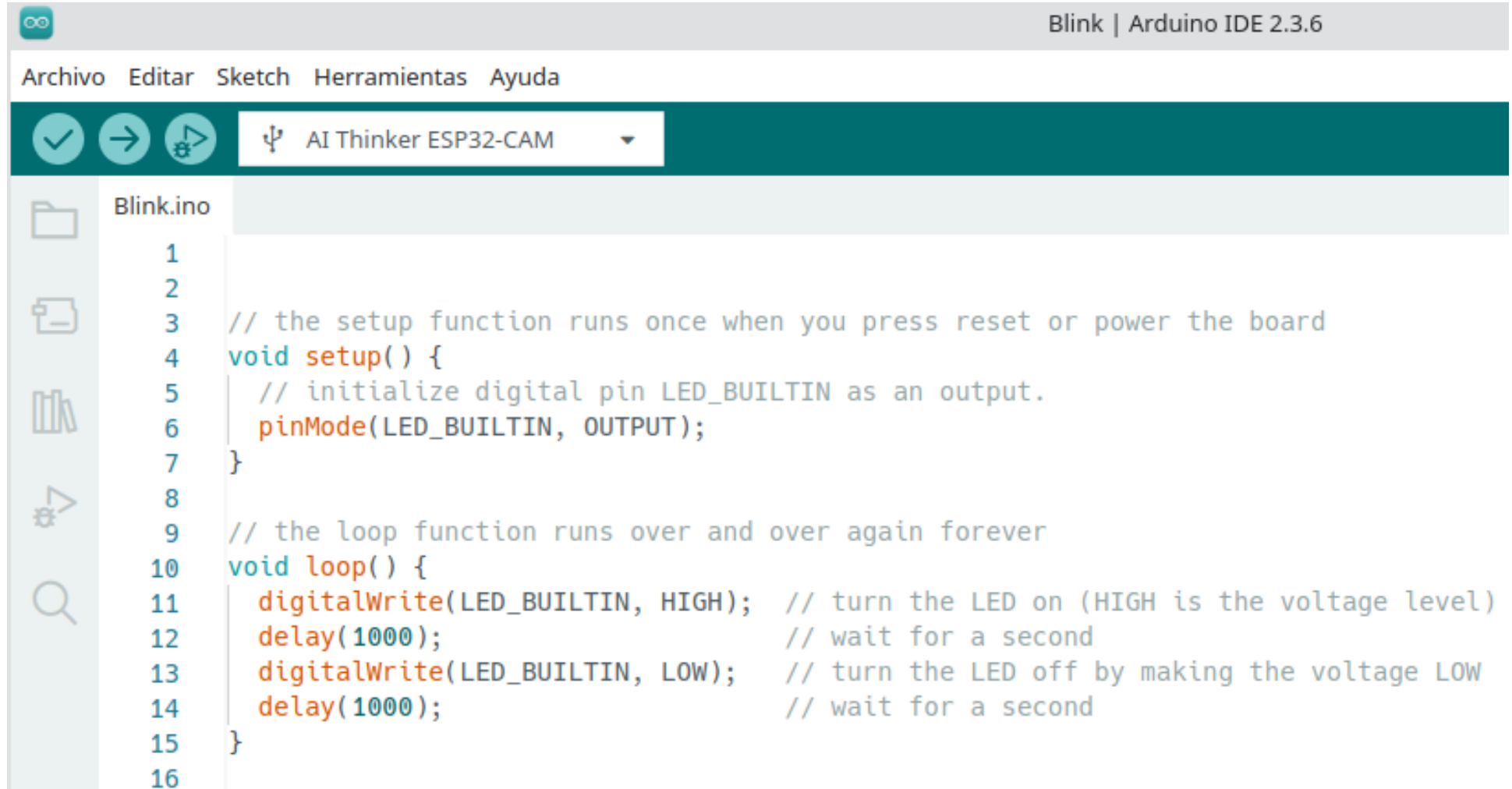
ENTORNOS DE DESARROLLO

- ARDUINO IDE
- PLATFORM IO + VSC
- ARDUINO BLOCKS

ESTRUCTURA



Arduino IDE (instalando librerías)



```
1
2
3 // the setup function runs once when you press reset or power the board
4 void setup() {
5     // initialize digital pin LED_BUILTIN as an output.
6     pinMode(LED_BUILTIN, OUTPUT);
7 }
8
9 // the loop function runs over and over again forever
10 void loop() {
11     digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
12     delay(1000); // wait for a second
13     digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
14     delay(1000); // wait for a second
15 }
16
```

VSC + Platform IO

The screenshot shows the Visual Studio Code interface with the extension marketplace open. The search bar contains the text "platform". The "PlatformIO IDE" extension is selected and highlighted in blue. The main panel displays the extension's details, including its logo, name, publisher (PlatformIO), download count (5.418.118), and rating (5 stars from 3796 reviews). The "Instalar" button is visible, along with a checkbox for "Actualización automática". The left sidebar shows a list of other extensions, including "OCaml Platf...", "Power Platfo...", "Uno Platform", and "Huawei Cloud ...". The bottom right panel shows the "Más información" section with details like "Identific...", "Versión", "Publicado", and "Última versión".

Archivo Editar Selección Ver Ir ... Buscar

EXTENSIONES: MARKETPLACE

platform

PlatformIO IDE 5.4M ★ 5
Your Gateway to Embedde...
PlatformIO Instalar

OCaml Platf... 228K ★ 4.5
Official OCaml language e...
OCaml Labs Instalar

Power Platfo... 215K ★ 4
Tooling to create Power Pl...
Microsoft Instalar

Uno Platform 33K ★ 5
Uno Platform Projects XA...
Uno Platform Instalar

Huawei Cloud ... 29K ★ 4
Huawei Cloud Toolkit VSC...
Huawei Cloud Instalar

PlatformIO IDE
PlatformIO platformio.org | 5.418.118 | ★★★★★ (3796)
Your Gateway to Embedded Software Development Excellence: CMSIS, ESP-IDF, FreeRTOS...
Instalar Actualización automática

DETALLES CARACTERÍSTICAS REGISTRO DE CAMBIOS DEPENDENCIAS

PlatformIO IDE for VSCode

PlatformIO: Your Gateway to Embedded Software Development Excellence.

Unlock the true potential of embedded software development with PlatformIO's collaborative ecosystem, embracing declarative principles, test-driven methodologies, and modern toolchains for unrivaled success.

Más información

Identific...	platformio.platformi ide
Versión	3.3.4
Publicado	2017-05-28, 0:02:51
Última versión	2025-01-11, 16:02:25

Arduino Blocks (I)

arduino blocks | Buscar proyectos | Proyectos ▾ | Recursos ▾ | jf.lopezsanchezmon@edu.gva.es ▾ | Cerrar sesión

Bloques ▾ | Información | Archivos | Sensor DHT22 - humedad y temperatura | Subir | Consola |

- Lógica
- Control
- Matemáticas
- Texto
- Variables
- Listas
- Funciones
- ESP
- Tiempo
- Entrada/Salida
- Sensores
- Actuadores
- Motor
- Periféricos
- Visualización
- Comunicaciones
- Tarjeta SD
- EEPROM

Inicializar

- Iniciar Baudios 115200

Bucle

- Establecer temperatura = DHT-22 Temperatura °C Pin 14
- Establecer humedad = DHT-22 Humedad % Pin 14
- + si temperatura < 27
 - hacer
 - Enviar "Relé ON" Salto de línea
 - Relé Pin 2 Estado ON
 - sino
 - Enviar "Relé Off" Salto de línea
 - Relé Pin 2 Estado OFF

Arduino Blocks (II)

The image shows a screenshot of an Arduino Blocks programming environment. The code is organized into two main sections: 'Inicializar' (Initialize) and 'Bucle' (Loop).

Inicializar






- Starts with an 'Iniciar Baudios' block set to 115200.

Bucle

- Contains two 'Establecer' (Set) blocks for the DHT-22 sensor. The first sets 'temperatura' (temperature) to 'Temperatura °C' on Pin 26 (D2). The second sets 'humedad' (humidity) to 'Humedad %' on Pin 26 (D2).
- Followed by five 'Enviar' (Send) blocks:
 - 1. Sends the string " Temperatura actual (C): " with a line skip option.
 - 2. Sends the variable 'temperatura' with a checked line skip option.
 - 3. Sends the string " Humedad actual (%): " with a line skip option.
 - 4. Sends the variable 'humedad' with a checked line skip option.
 - 5. Sends a string of dashes "-----" with a checked line skip option.
- Ends with an 'Esperar' (Wait) block for 5000 milliseconds.

MATERIAL NECESARIO . . .

- PLACA ESP32
- Cable USB.
- Protoboard.
- Algunos sensores, actuadores y cables.
- PC con **el software necesario . . .**
- En Lliurex zero center...

-  Arduino IDE: integrated development environment
-  LliureX-Robotics: library allows connection to Arduino and Lego
-  Scratch for Arduino
-  Visualino: Visual programming environment for Arduino
-  mLink: mBlock web driver
-  Arduino Blocks Connector: Arduino driver for Arduino Blocks

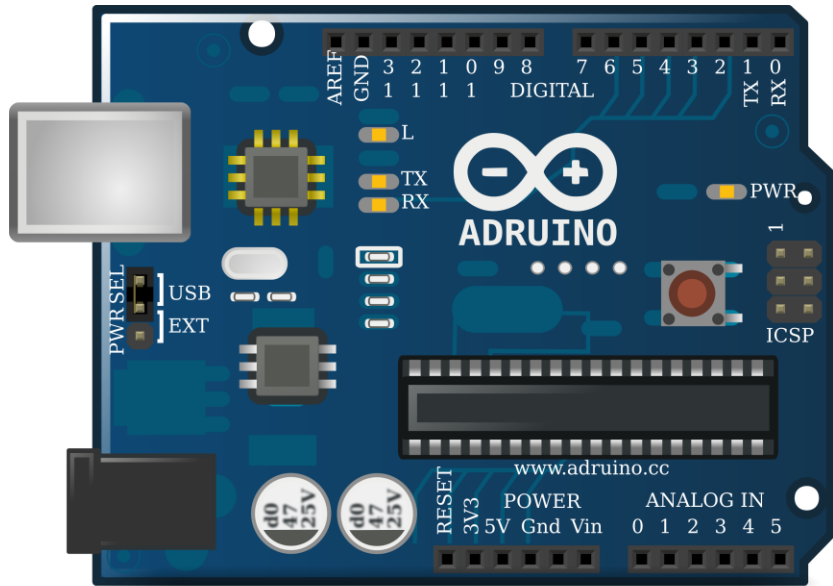
MATERIAL NECESARIO (KIT Básico)



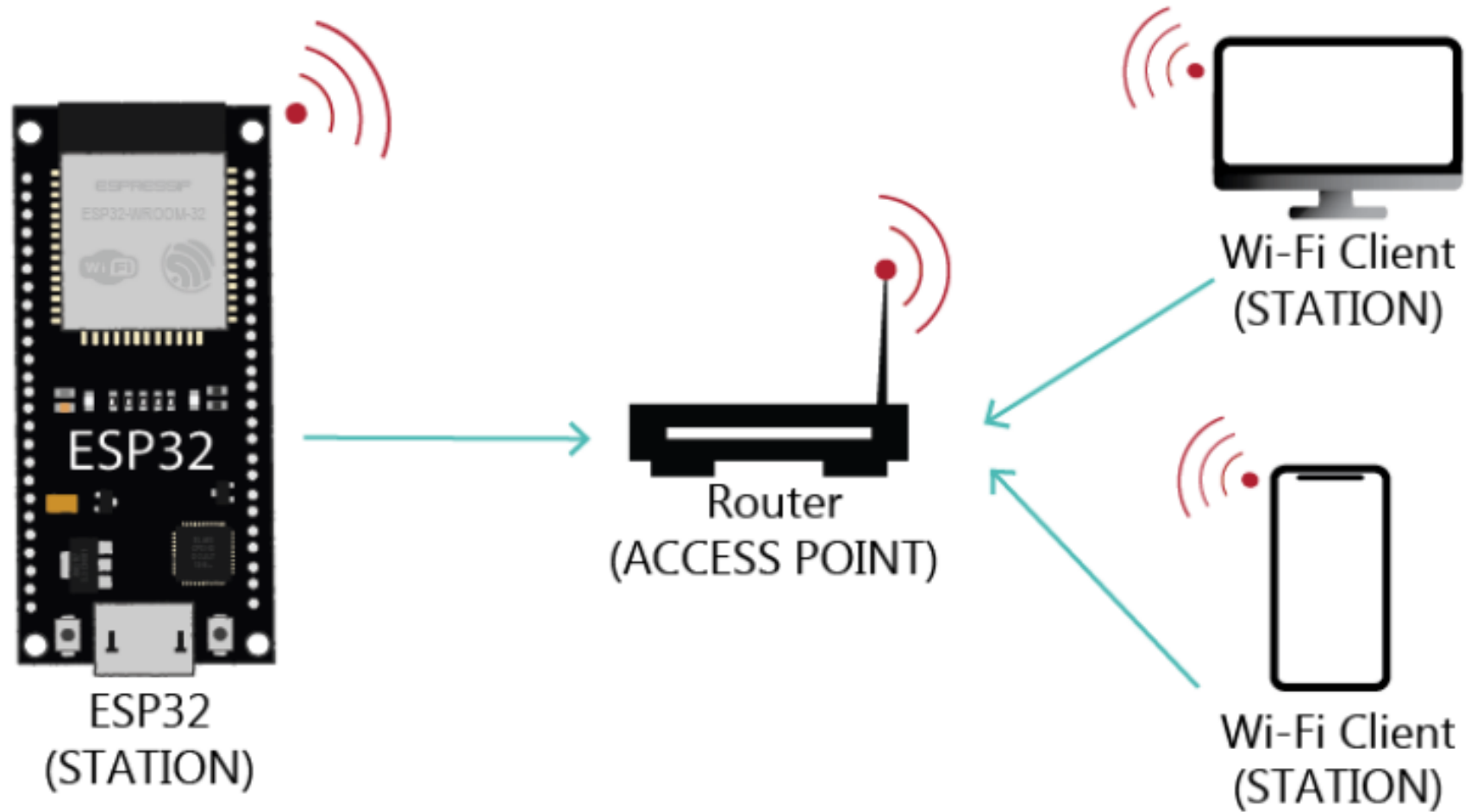
RECURSOS Y COMUNIDAD

- [INSTRUCTABLES](#)
- [RANDOM NERD TUTORIALS](#)
- [FOROS DE ESPRESSIF](#)

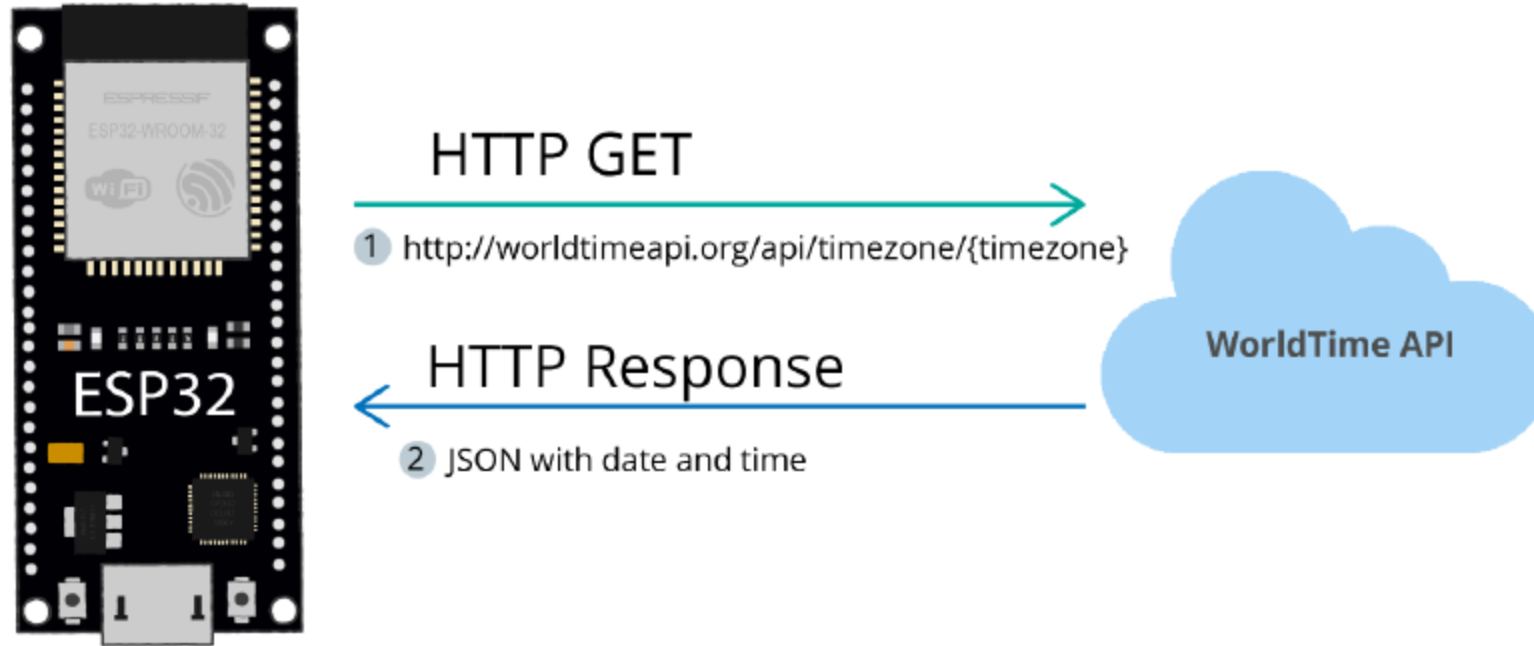
MI EVOLUCIÓN ...



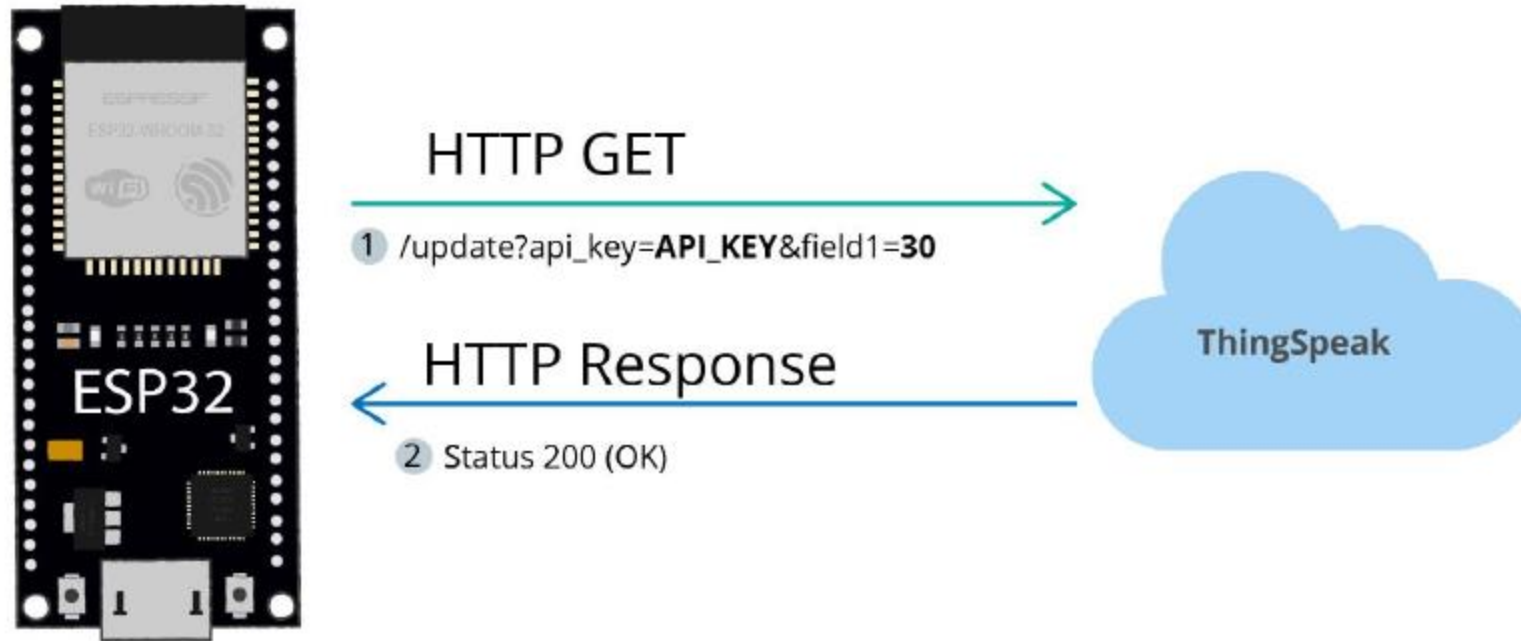
Modo WIFI ESTACION



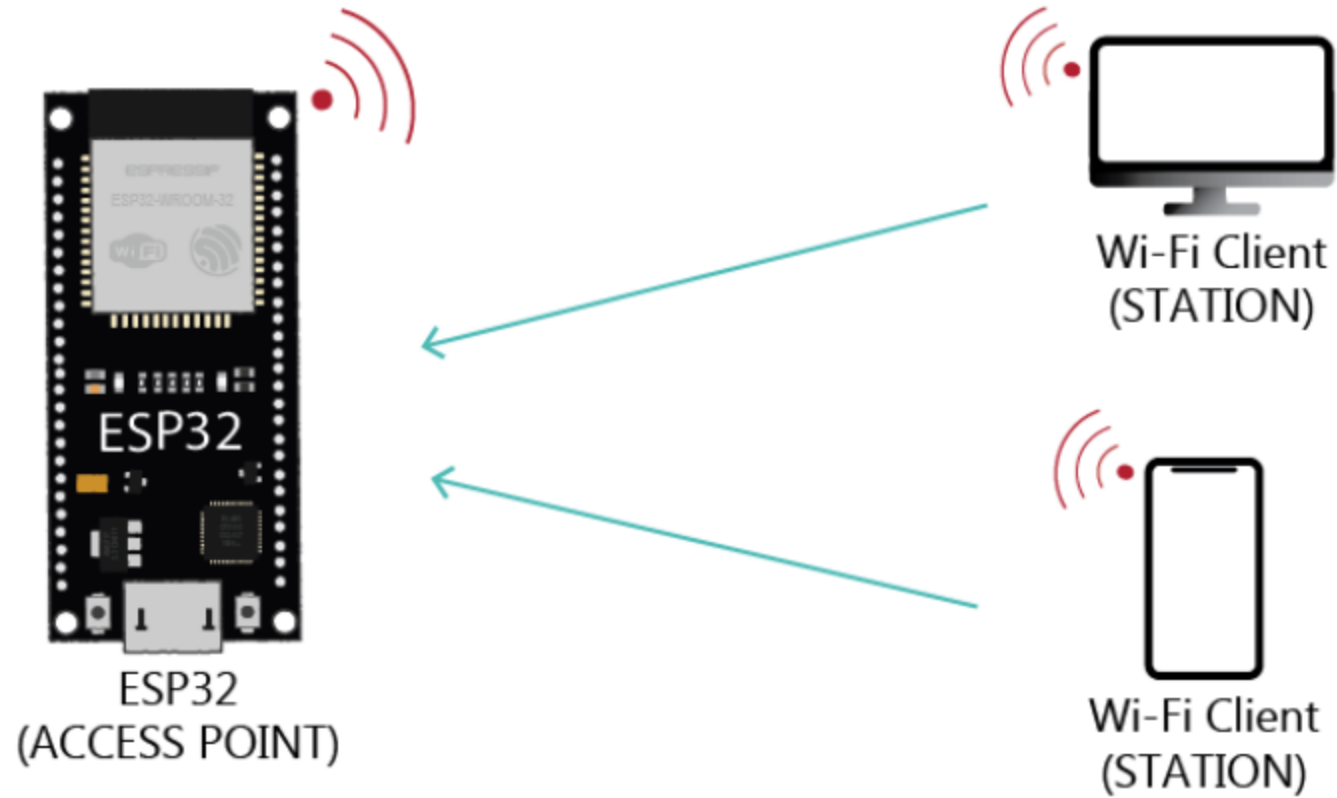
Conexión mediante APIs



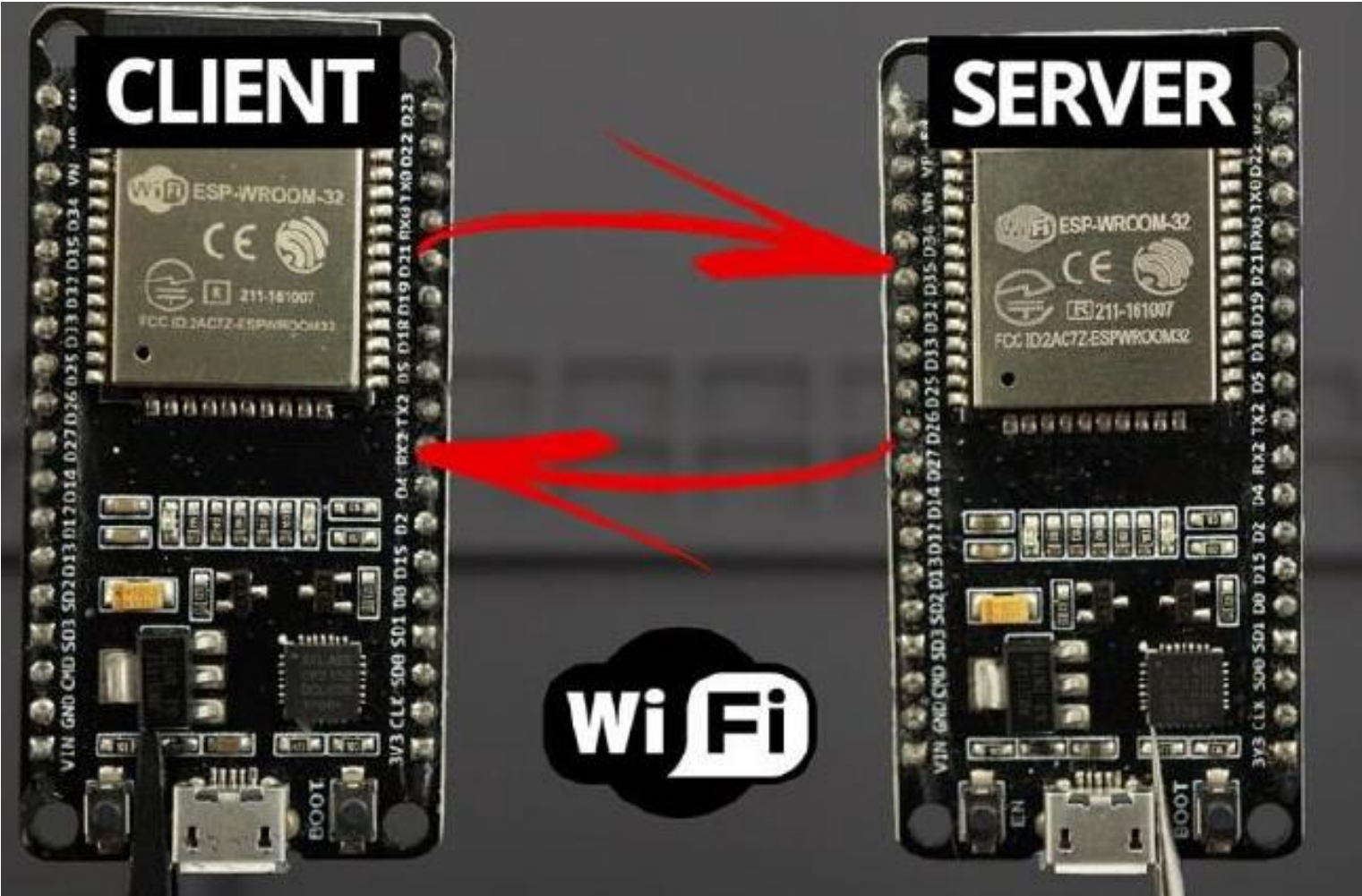
Conexión mediante APIs



Modo WIFI AP



CONECTIVIDAD ENTRE DOS ESP32



ESP32 SERVER

Access Point



HTTP GET: 192.168.4.1/temperature

22.16 °C

HTTP GET: 192.168.4.1/humidity

73.56 %

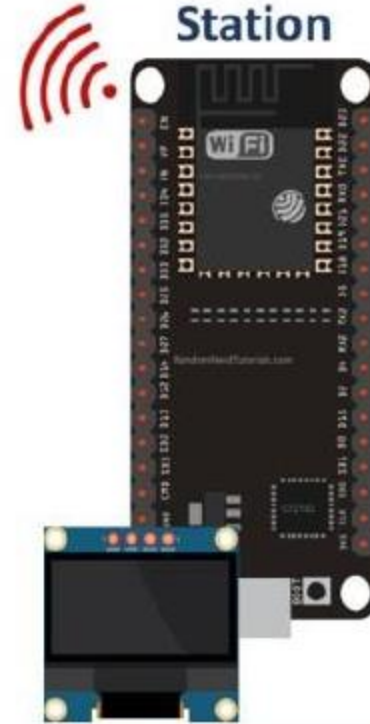
HTTP GET: 192.168.4.1/pressure

1017.79 hPa

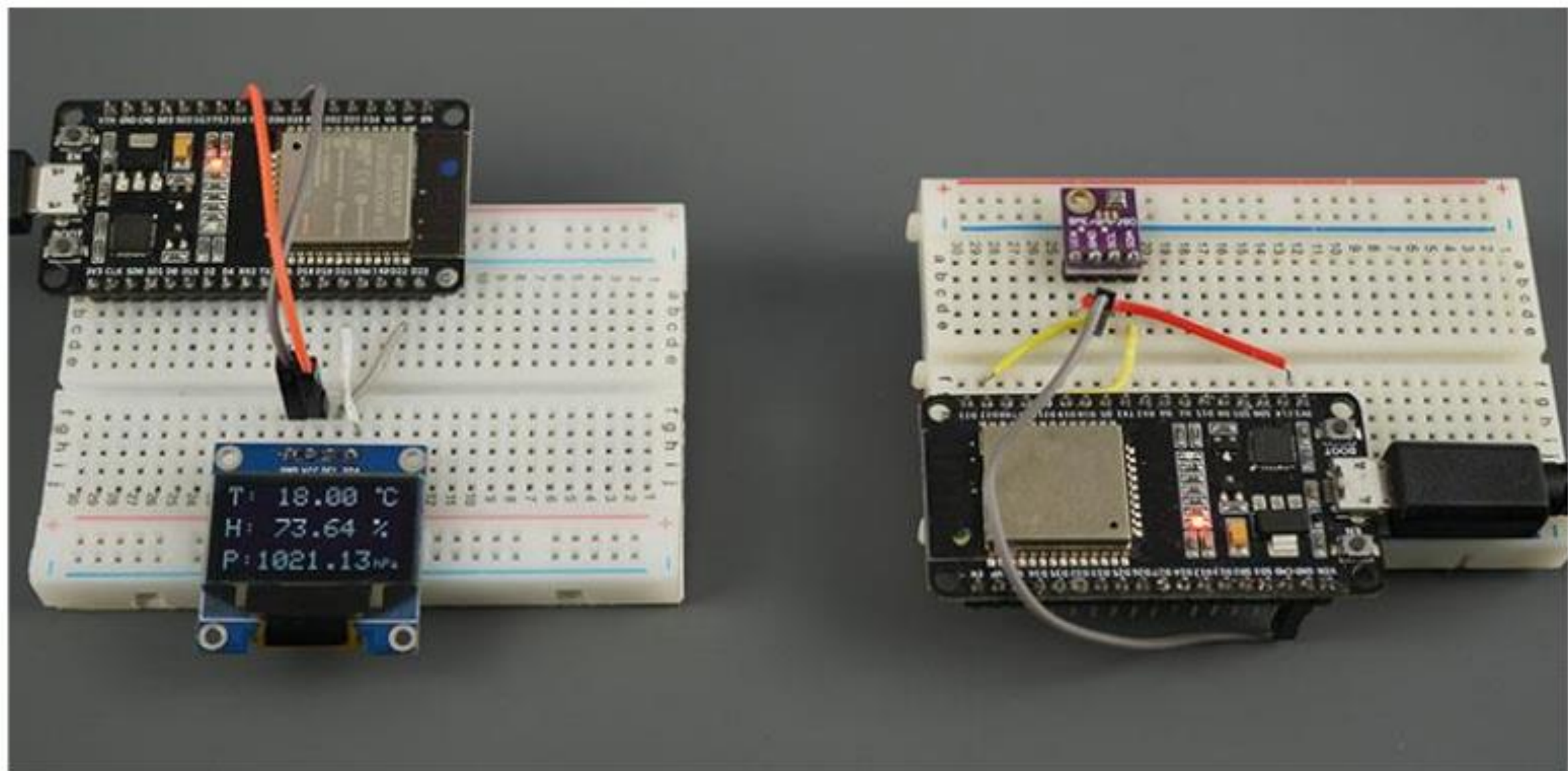
SSID: ESP32-Access-Point
Password: 123456789
IP Address: 192.168.4.1

ESP32 CLIENT

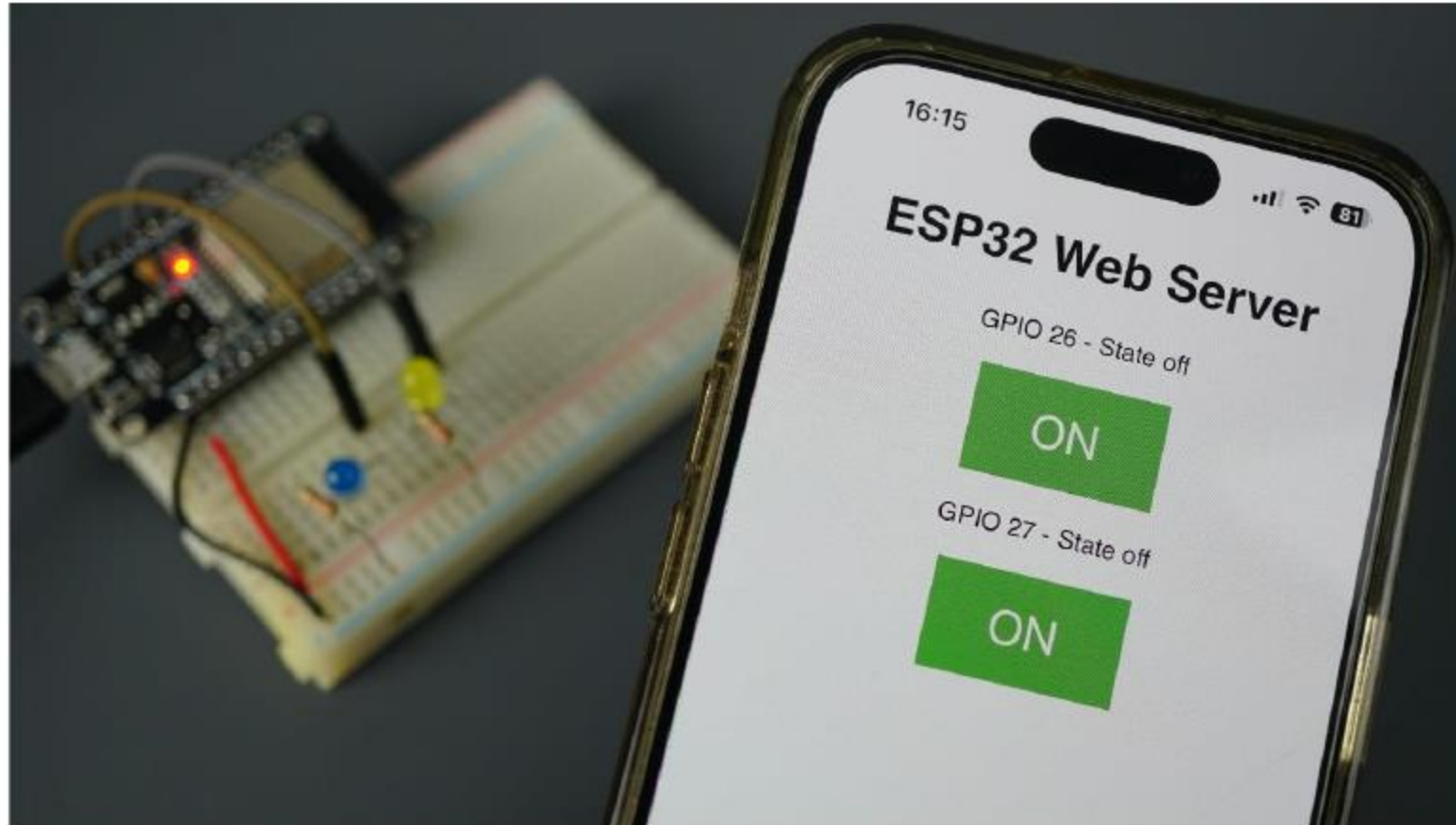
Station



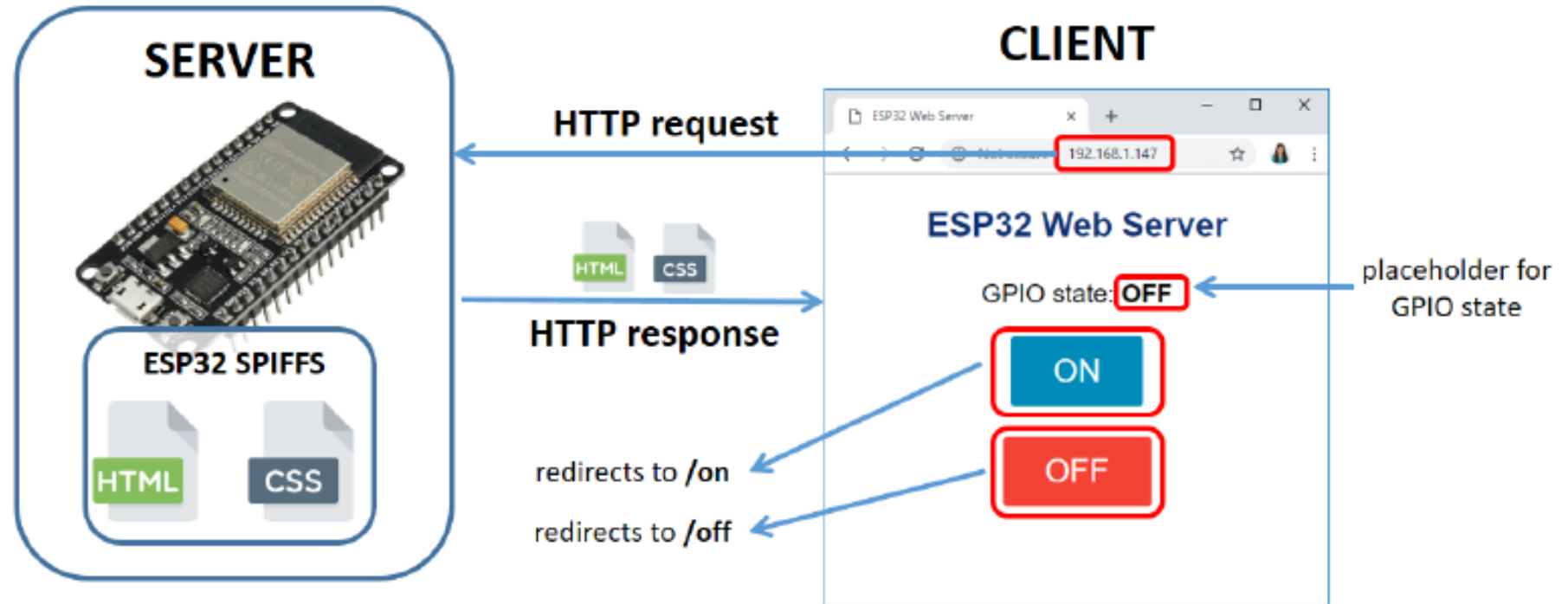
Station connected to
ESP32-Access-Point
wireless network



ESP32 COMO SERVIDOR WEB



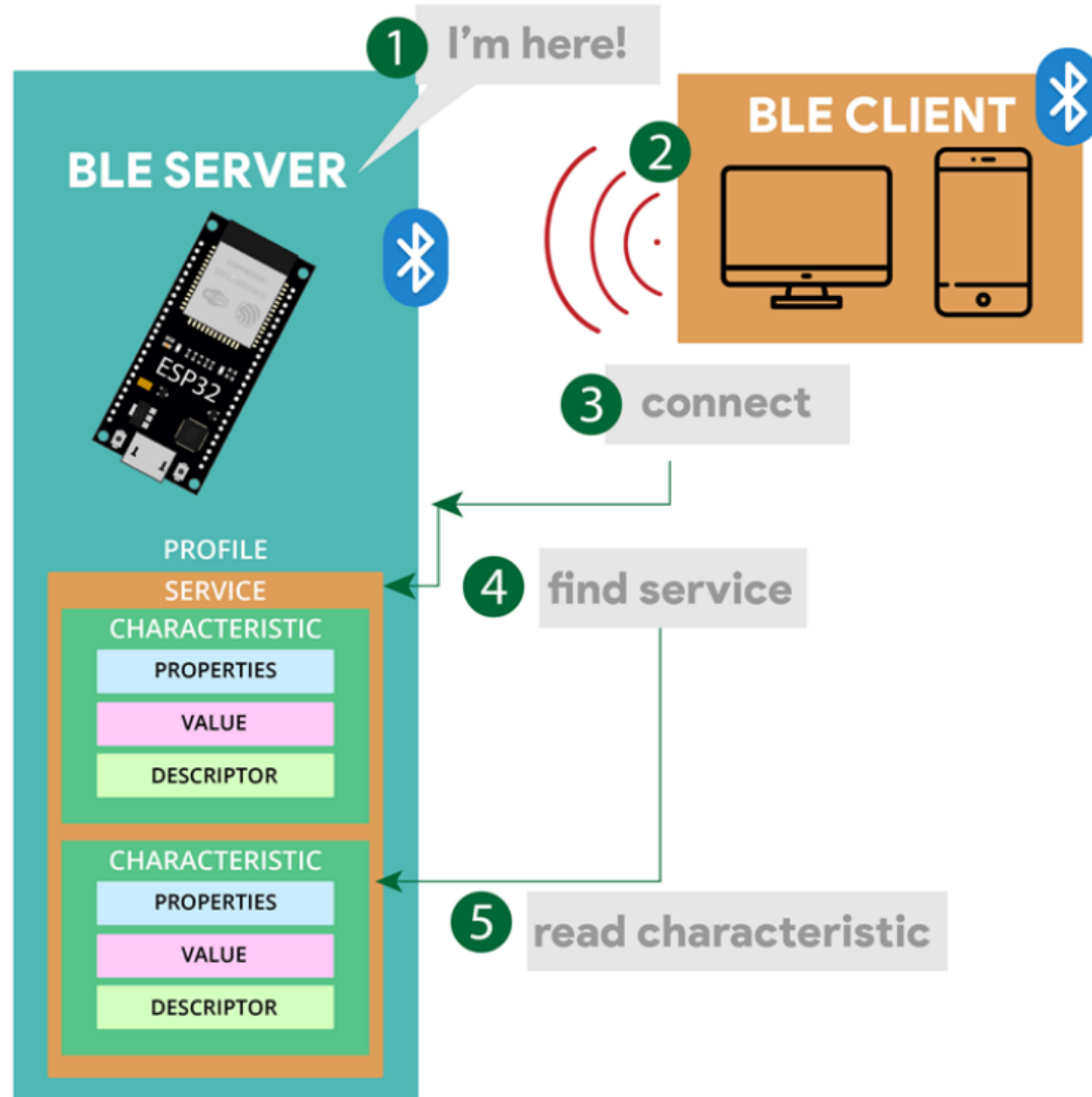
ESP32 COMO SERVIDOR WEB



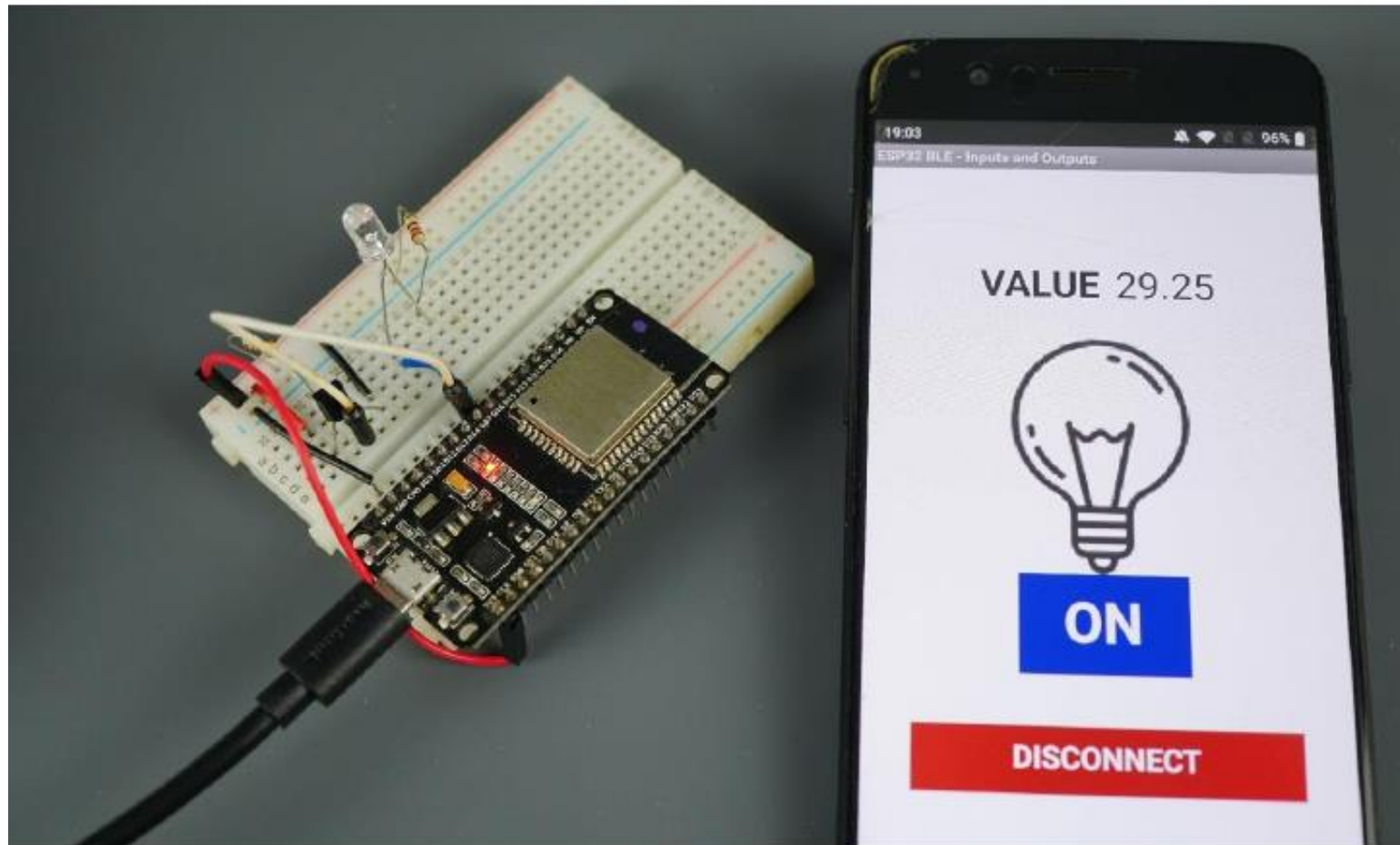
ESP32 COMO SERVIDOR WEB



ESP32 CONECTIVIDAD BLUETOOTH



ESP32 CONECTIVIDAD BLUETOOTH



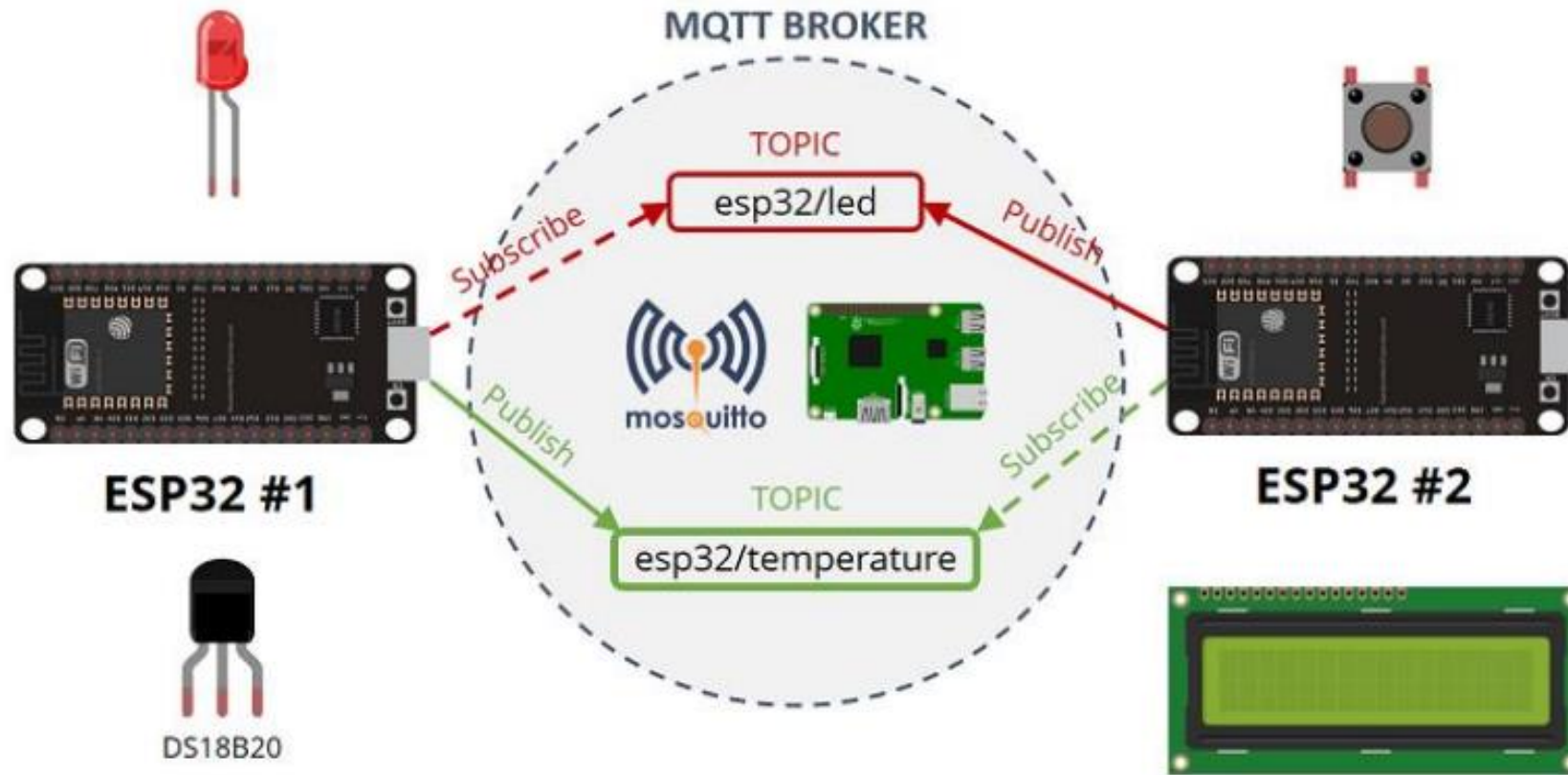
PROTOCOLLO MQTT



ESP32



PROTOCOLLO MQTT



NODE RED (FLUJOS)

The screenshot displays the Node-RED web interface. On the left, a sidebar lists various nodes under the 'common' category, including 'inject', 'debug', 'complete', 'catch', 'status', 'link in', 'link call', and 'link out'. The main workspace shows a flow named 'Flow 1' containing two nodes: an 'inject' node with the payload 'Hello' and a 'msg.payload' node. A red box highlights the inject node, with a red arrow pointing to it and the text 'Triggers the inject node'. On the right, a 'debug' console is open, showing three log entries, each with a timestamp, node ID, and the payload 'Hello'. The 'debug' console title and the 'Hello' payloads are highlighted with red boxes.

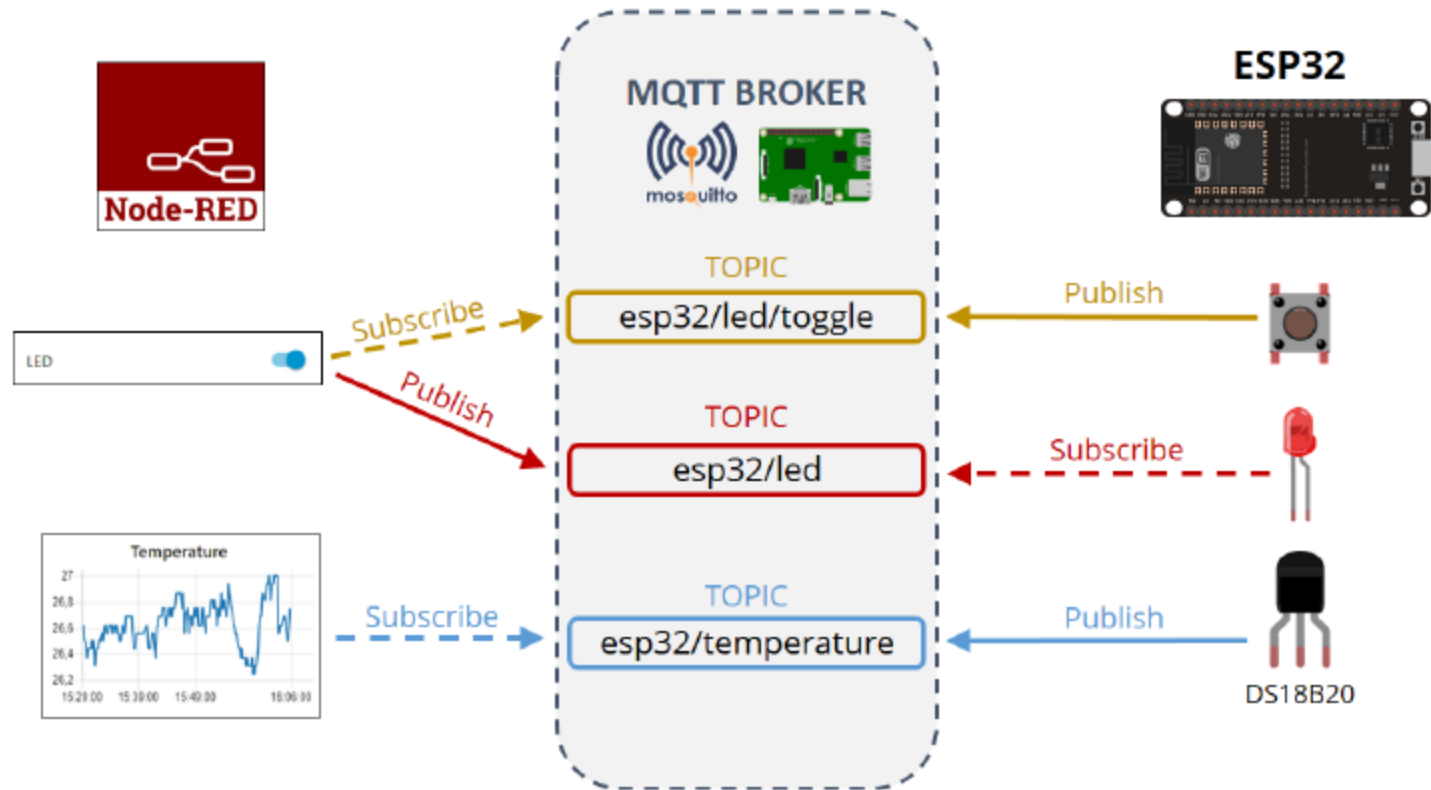
```
graph LR; inject[inject] --> msg_payload[msg.payload];
```

6/11/2022, 7:26:42 PM node: de95cdf163ee9711
msg.payload : string[5]
"Hello"

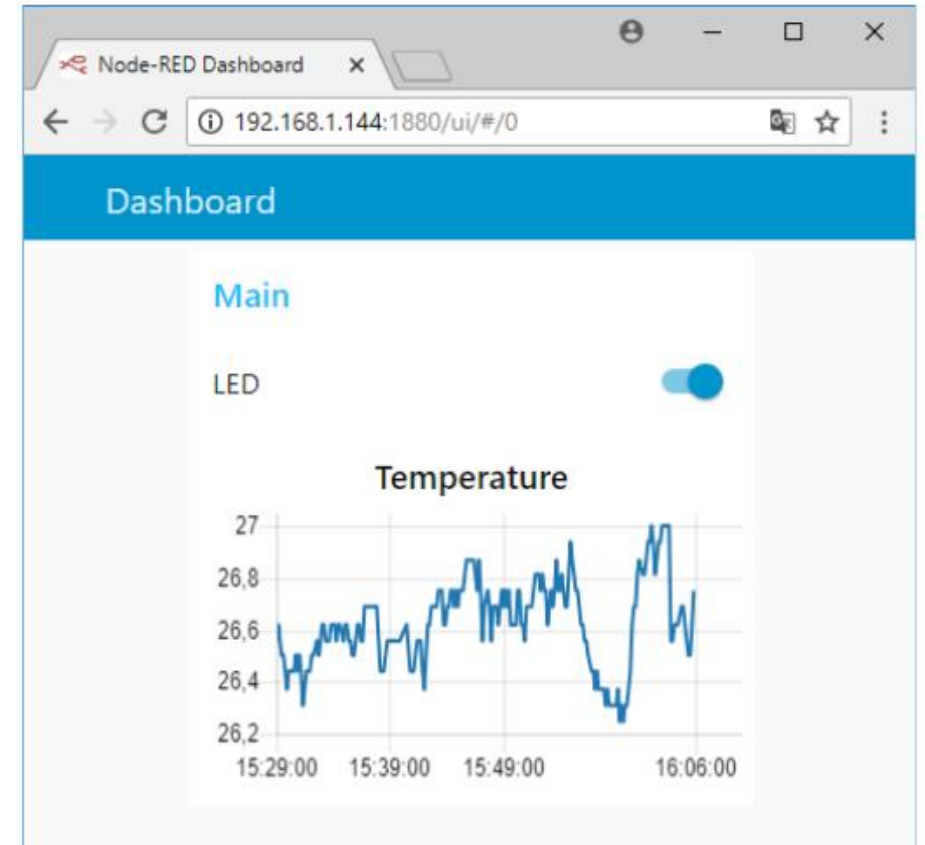
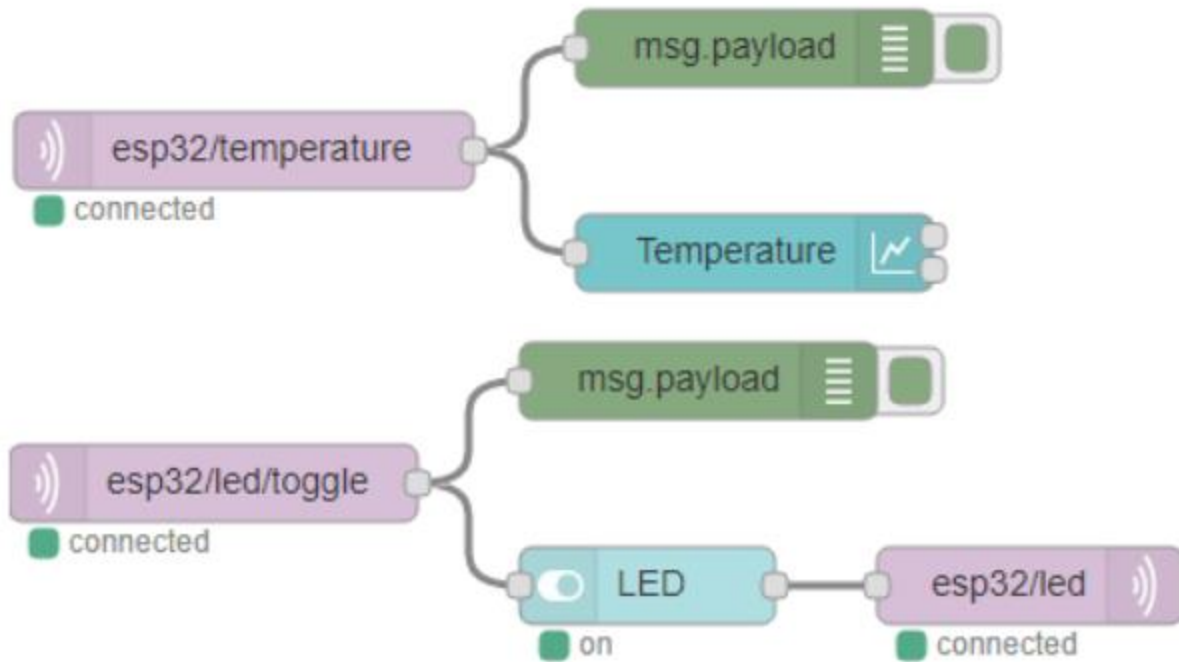
6/11/2022, 7:26:44 PM node: de95cdf163ee9711
msg.payload : string[5]
"Hello"

6/11/2022, 7:26:55 PM node: de95cdf163ee9711
msg.payload : string[5]
"Hello"

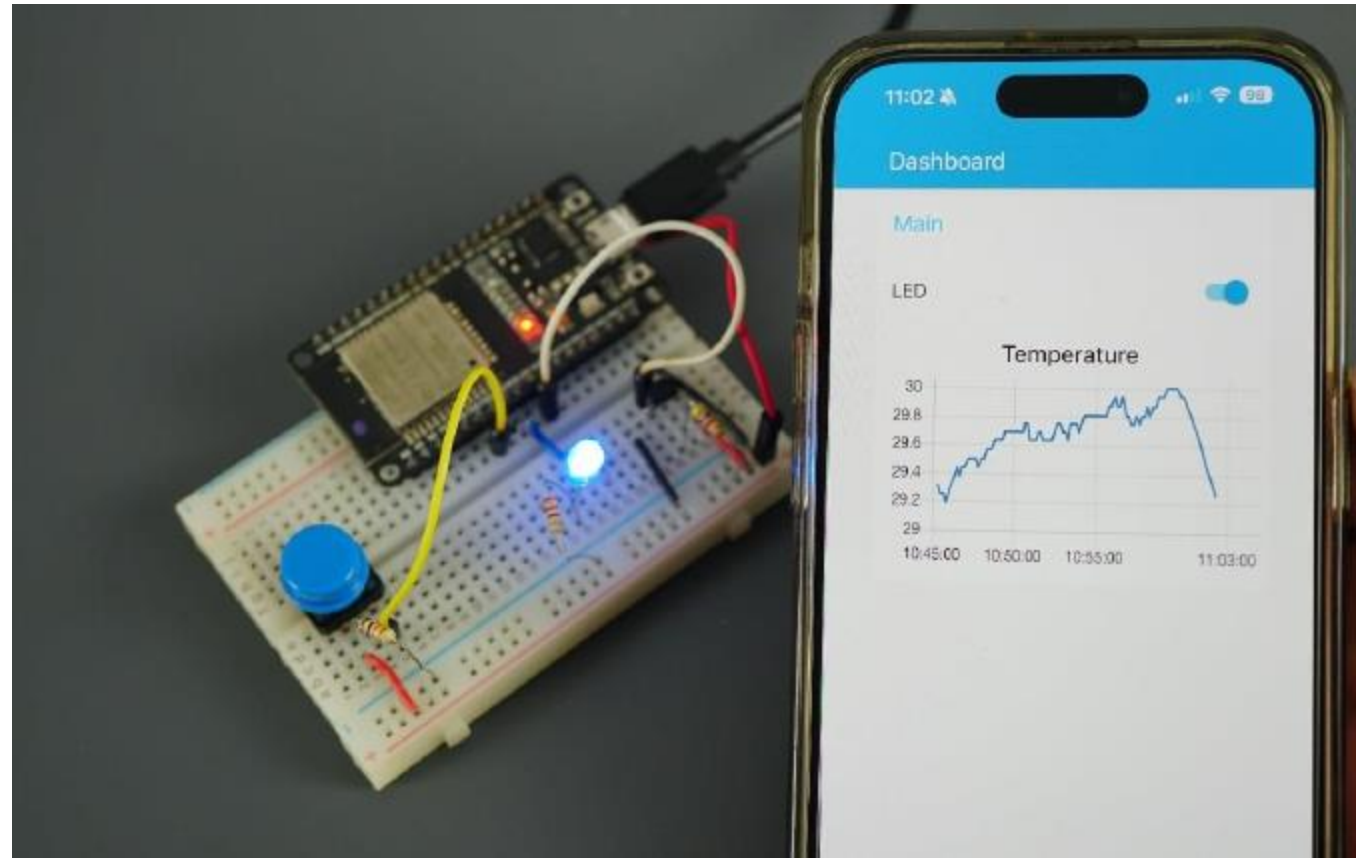
NODE RED (FLUJOS)



NODE RED (FLUJOS)



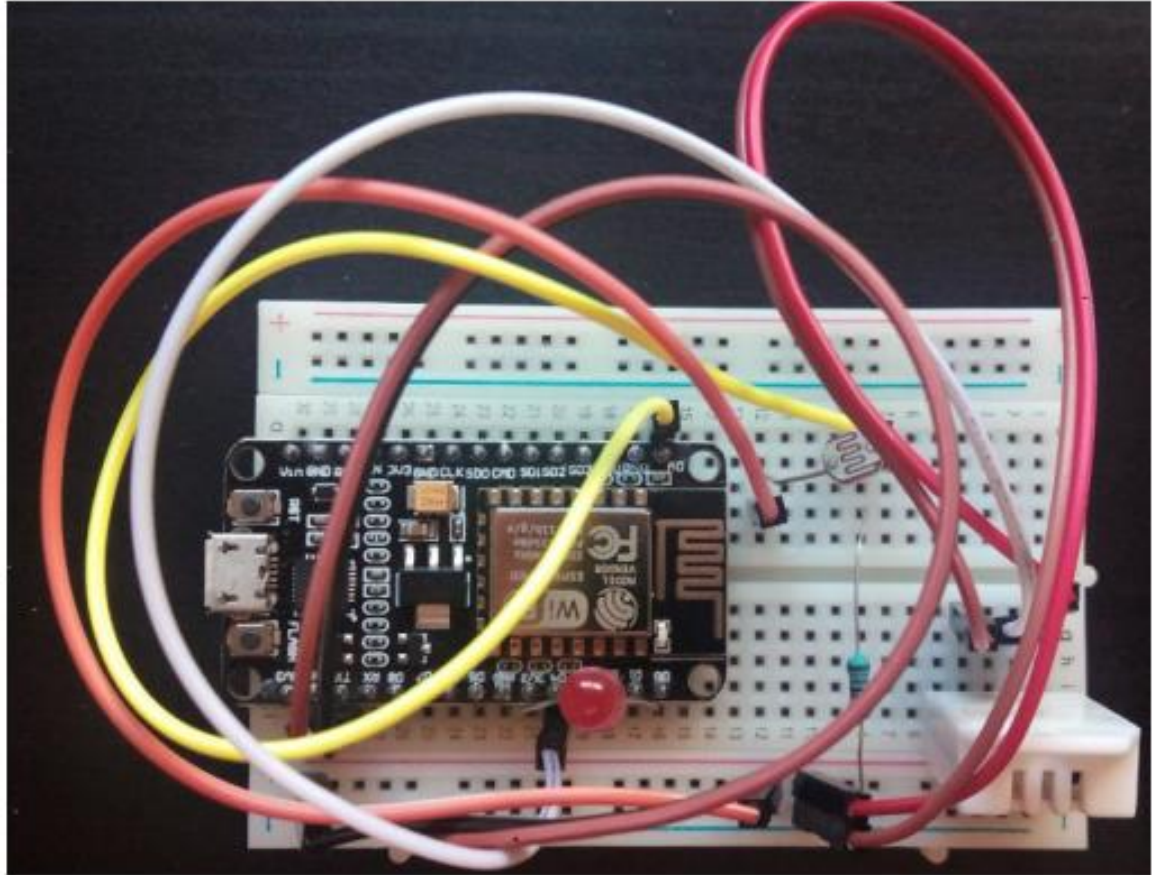
NODE RED (FLUJOS)



PROYECTOS INTERESANTES

LUZ, TEMPERATURA Y HUMEDAD

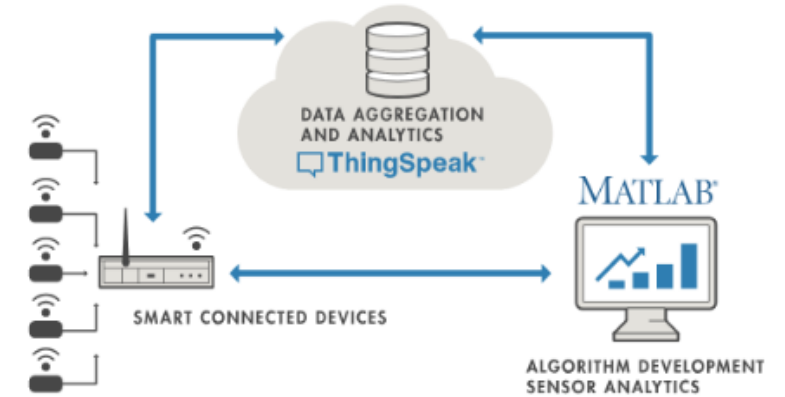
LDR, LED, DHT22



ENVIO DE DATOS A THINGSPEAK

```
Configuración Inicial (Setup)  
Declara temperatura type float Valor 0  
Declara humedad type float Valor 0  
Desconectar  
Imprime con salto de línea "Conectando a Internet ..."  
Espera Ms 3000  
Conectar Red ssid "testIOT" password "testIOT#2018"  
repetir mientras no Está conectado?  
hacer Espera Ms 300  
Imprime con salto de línea "!!!Ya estoy conectado!!!"
```

```
Bucle principal (Main loop)  
Estado del LED integrado ALTO  
establecer humedad a DHT humidity sensor  
PIN# D4  
Type: DHT 22  
Unit of measure Celsius  
establecer temperatura a DHT temperature sensor  
PIN# D4  
Type: DHT 22  
Unit of measure Celsius  
Imprime en la misma línea humedad  
Imprime en la misma línea "  
Imprime con salto de línea temperatura  
si Conectar Sitio Web "api.thingspeak.com" Puerto 80  
hacer Enviar a thingspeak Clave " Campo "Humedad" Value humedad  
si Conectar Sitio Web "api.thingspeak.com" Puerto 80  
hacer Enviar a thingspeak Clave " Campo "Temperatura" Value temperatura  
Espera Ms 5000
```





ThingSpeak for IoT Projects

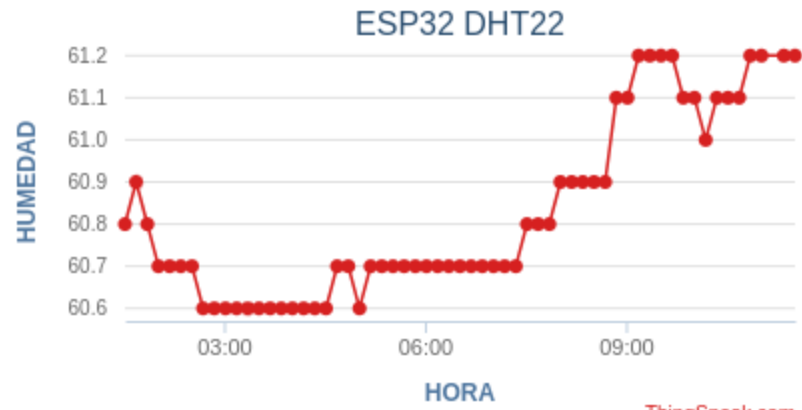
Data collection in the cloud with advanced data analysis using MATLAB

Get Started For Free

Learn More

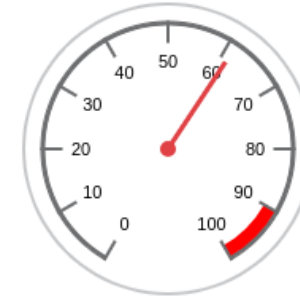


Field 3 Chart



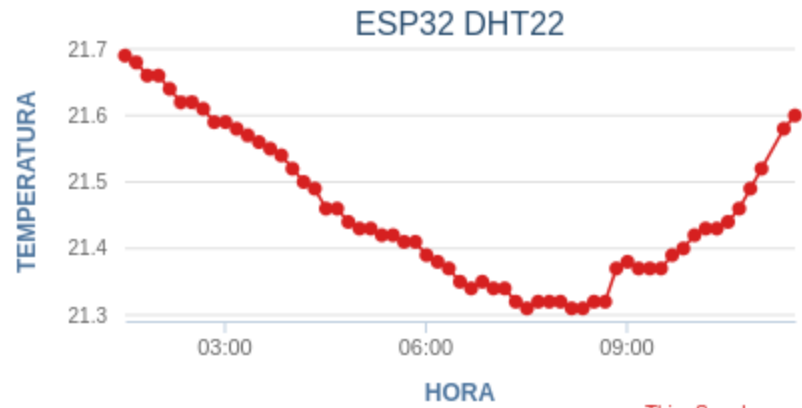
ThingSpeak.com

HUMEDAD



61.2

Field 5 Chart



ThingSpeak.com



Export

Download all of this Channel's feeds in CSV format.

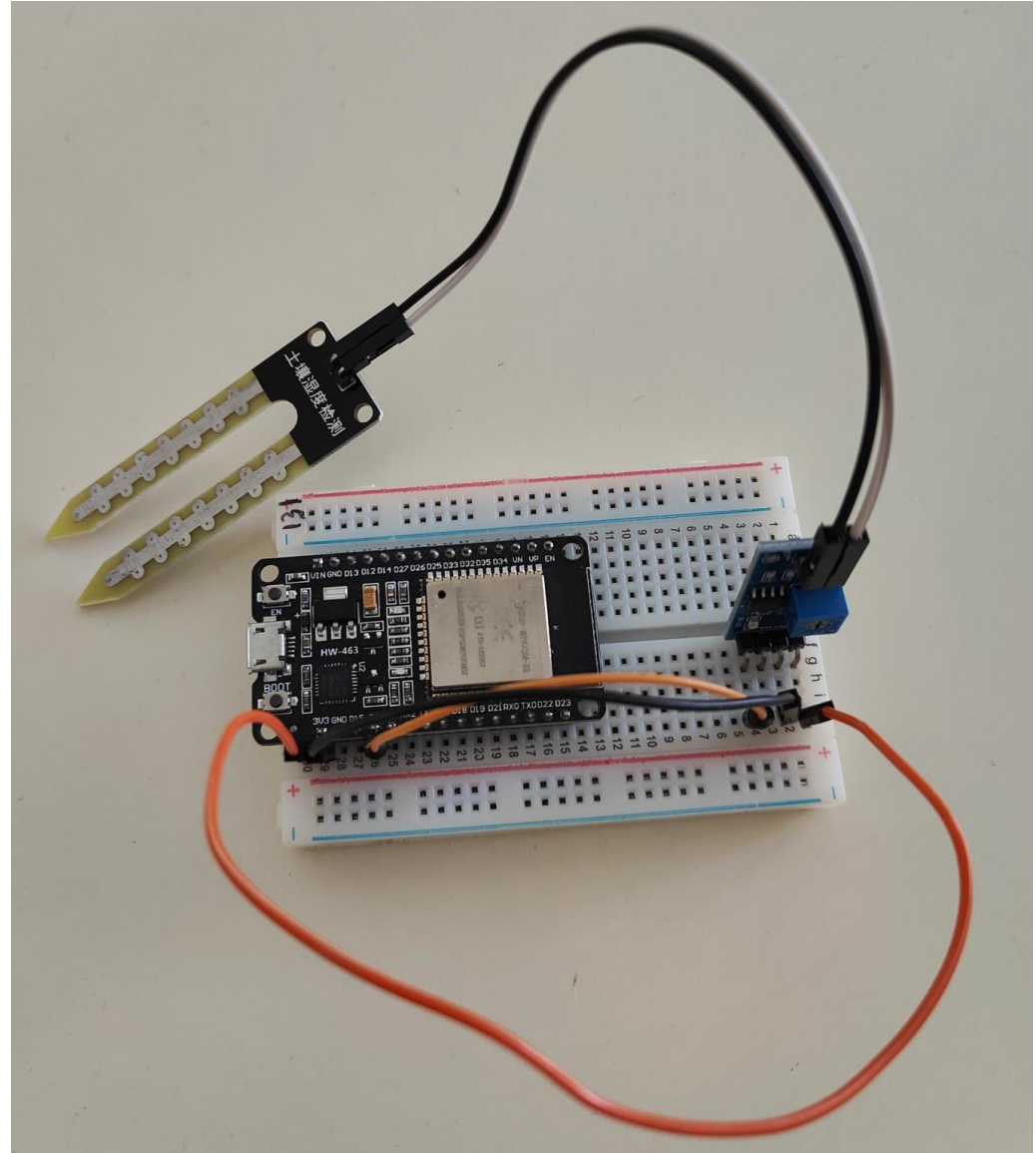
Time Zone

(GMT+01:00) Madrid

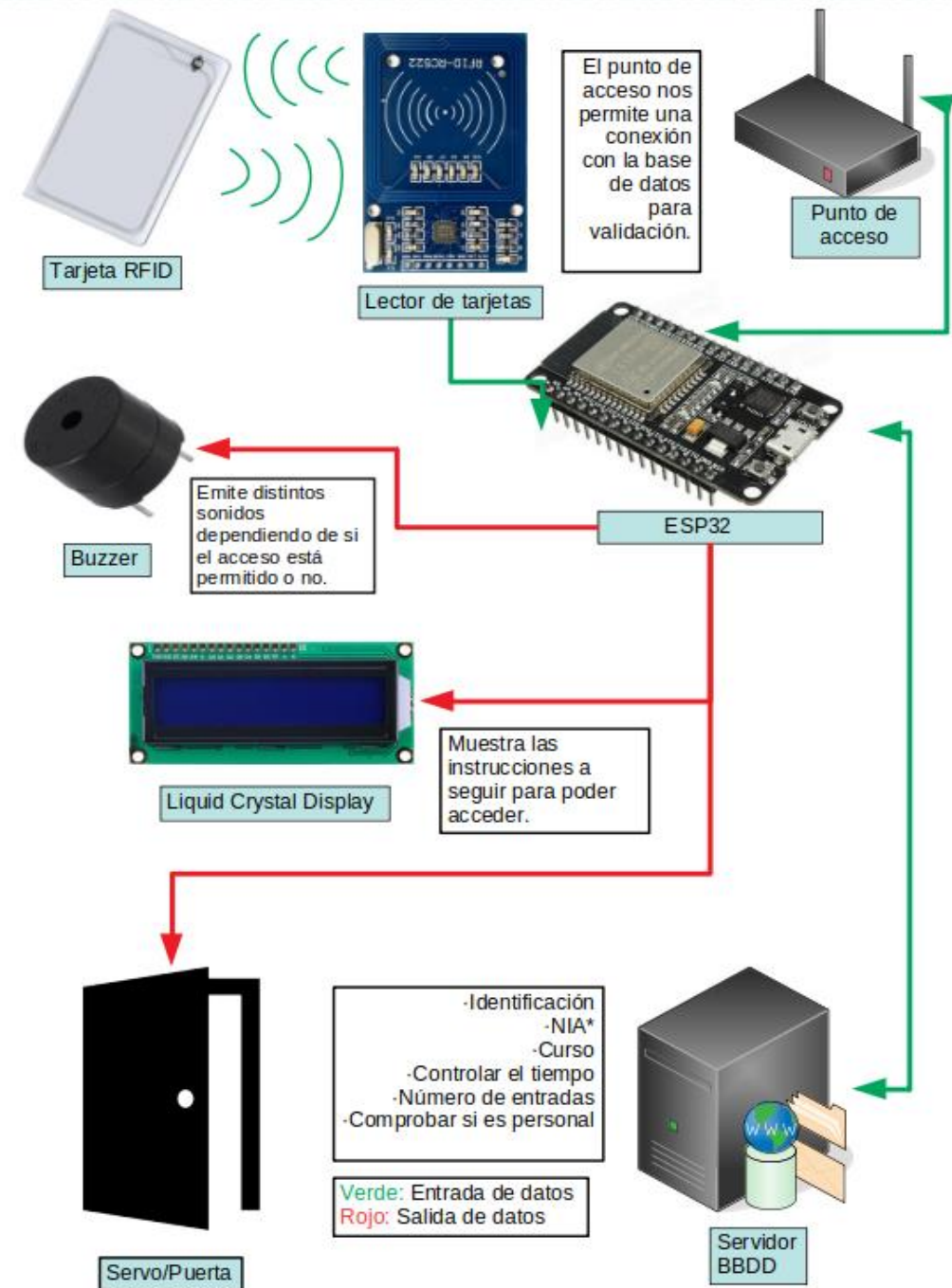
Download

	A	B	C	D
1	created_at	entry_id	field1	field2
2	2024-01-16T18:04:55+01:00	1	66.2	19.06
3	2024-01-16T18:14:55+01:00	2	65.6	19.31
4	2024-01-16T18:24:55+01:00	3	65.1	19.5
5	2024-01-16T18:34:55+01:00	4	64.8	19.5
6	2024-01-16T18:44:56+01:00	5	64.7	19.81
7	2024-01-16T18:54:56+01:00	6	64.4	19.81
8	2024-01-16T19:04:55+01:00	7	63.9	19.81
9	2024-01-16T19:14:56+01:00	8	63.6	19.87
10	2024-01-16T19:24:56+01:00	9	62.8	20.18
11	2024-01-16T19:34:56+01:00	10	61.9	20.37
12	2024-01-16T19:44:55+01:00	11	61	20.37
13	2024-01-16T19:54:56+01:00	12	60.6	20.12
14	2024-01-16T20:04:56+01:00	13	60.6	19.81
15	2024-01-16T20:14:56+01:00	14	60.8	19.43
16	2024-01-16T20:24:56+01:00	15	60.9	19.68
17	2024-01-16T20:34:56+01:00	16	60.7	19.75
18	2024-01-16T20:44:56+01:00	17	60.4	19.62

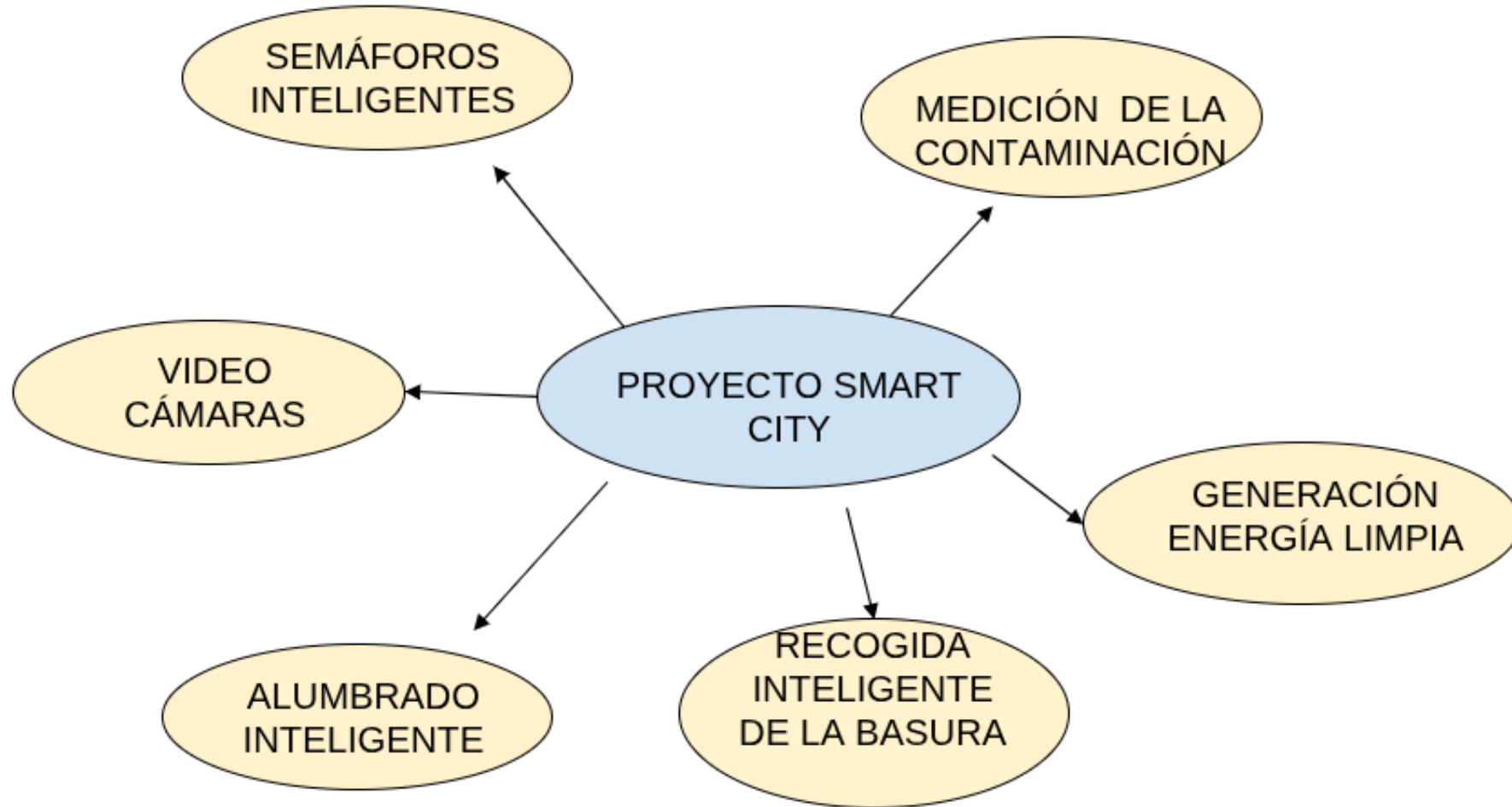
HUMEDAD TIERRA

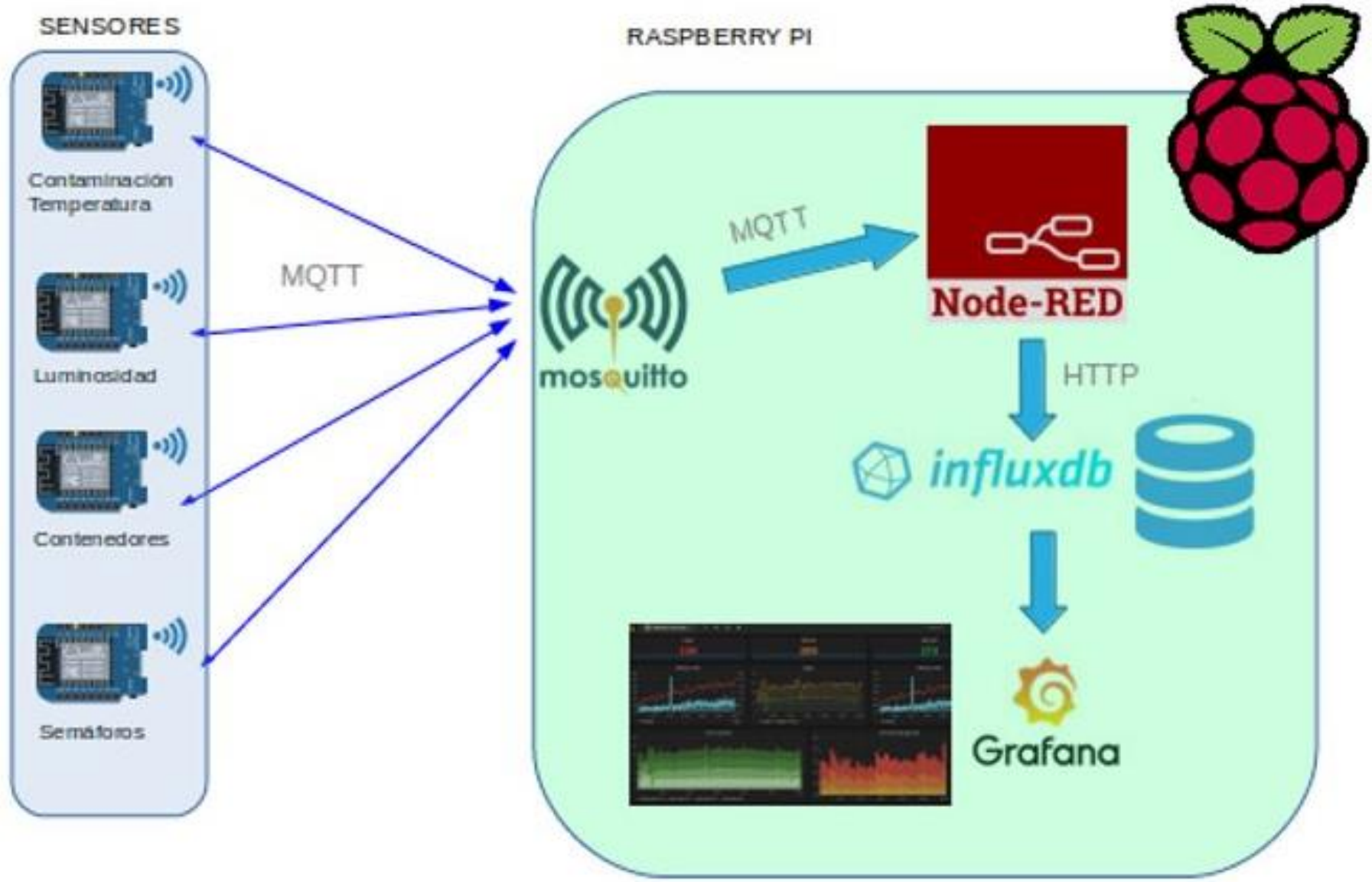


BIBLIOTECA



SMART CITY Y BIGDATA

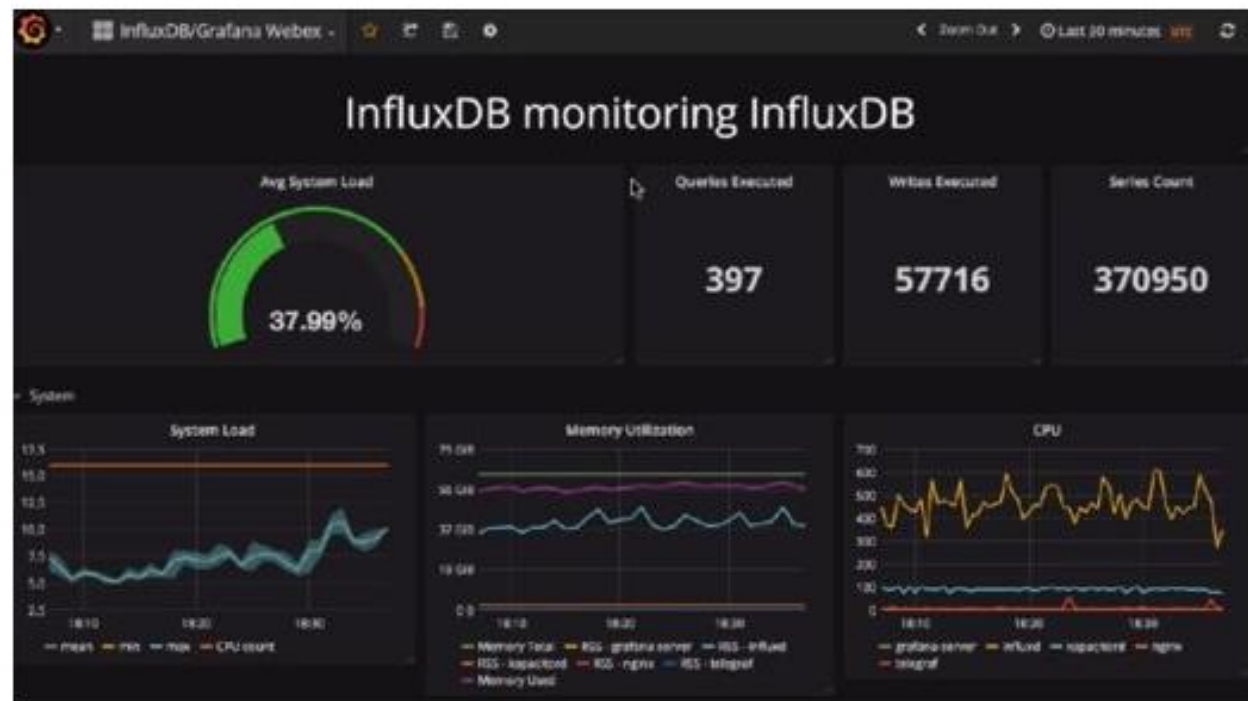


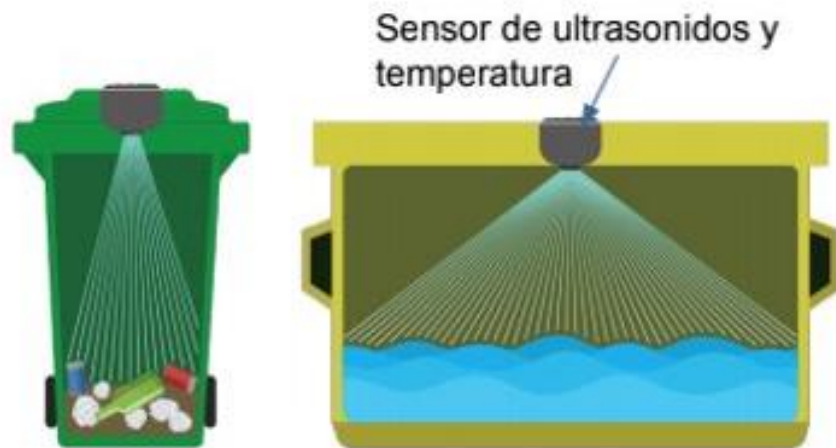


MQTT



GRAFANA





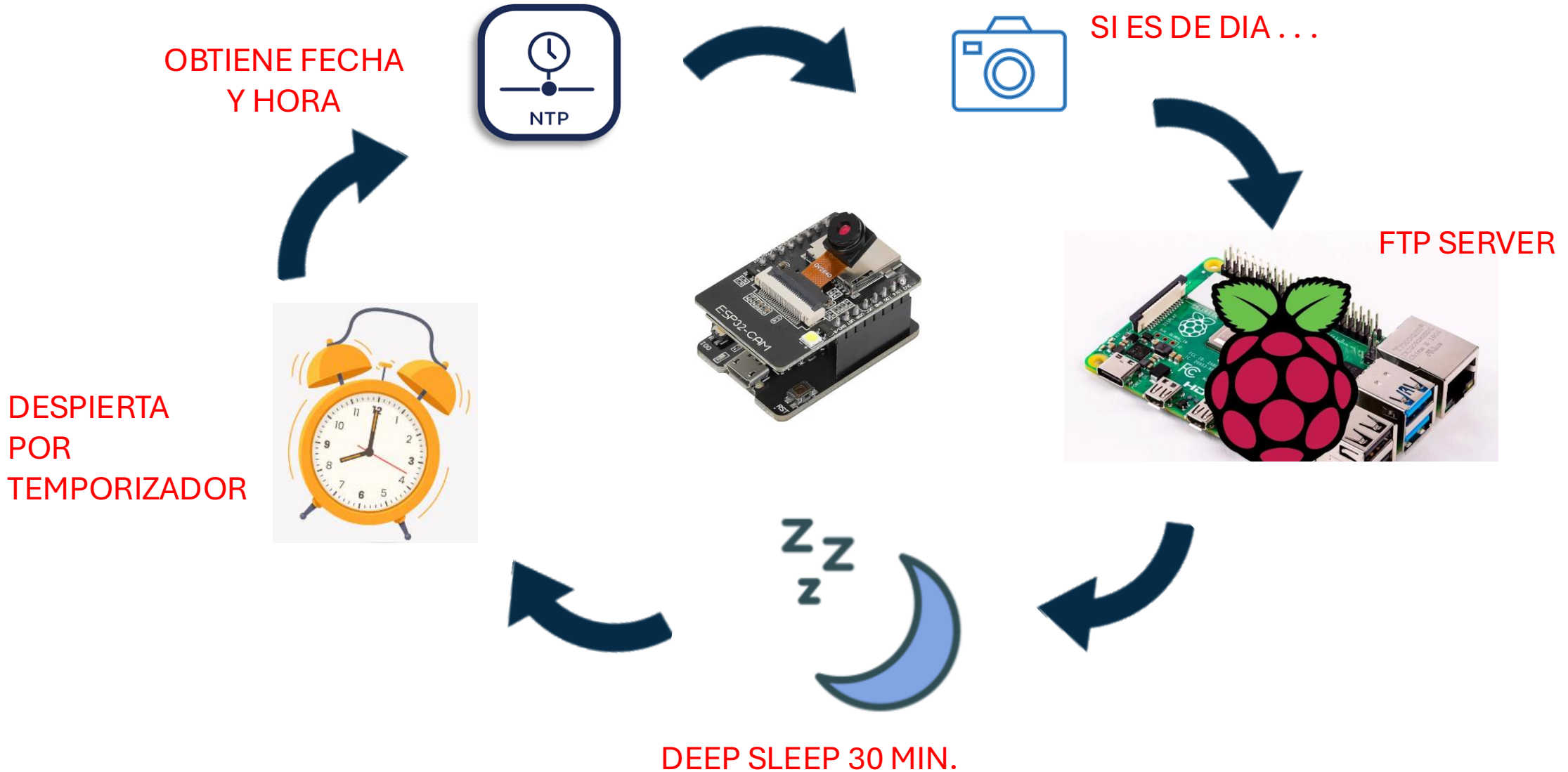
Sensor HC-SR04

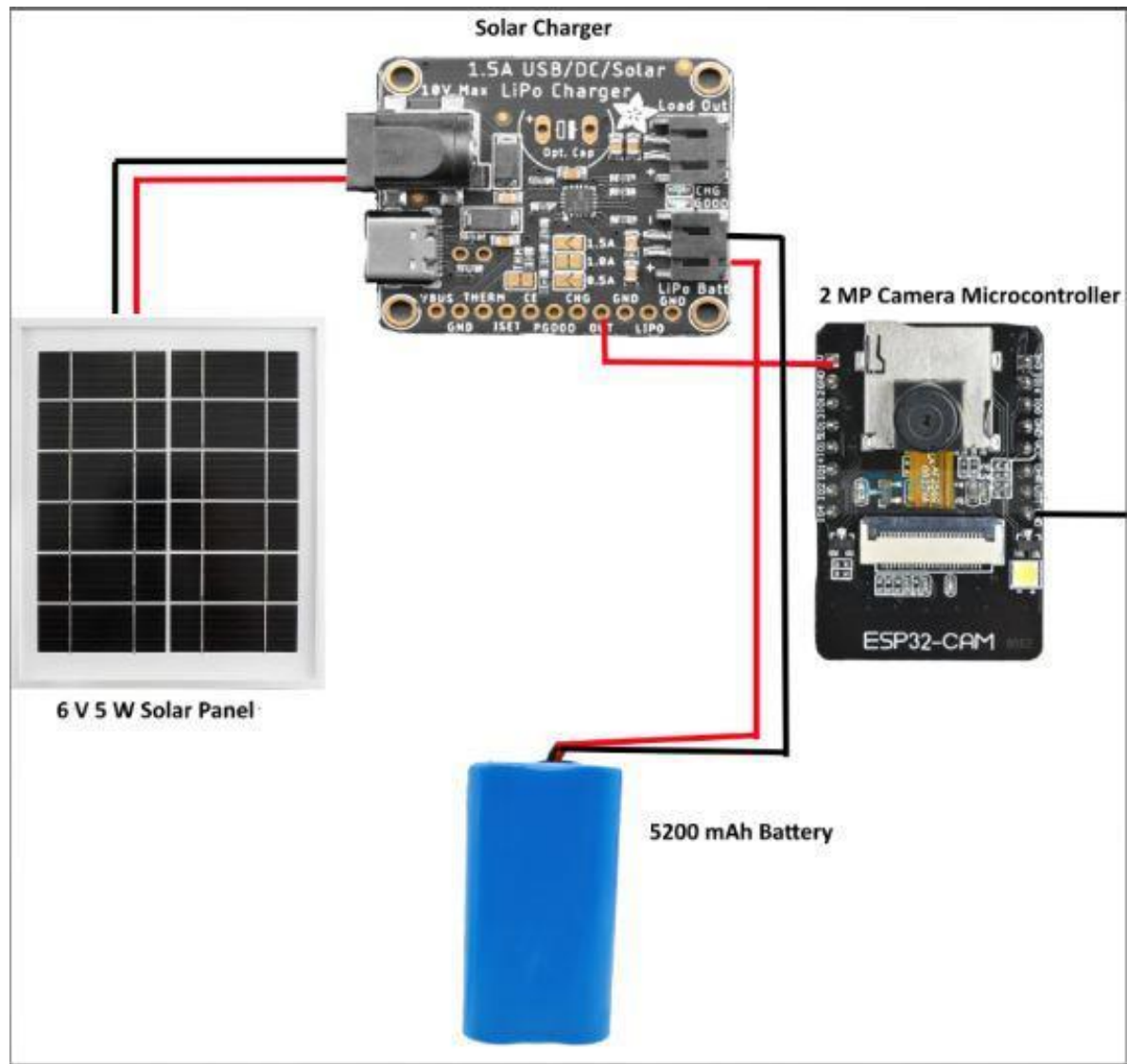


Sensor DHT22

CÁLCULO DE LA RUTA IDÓNEA PARA AHORRAR TIEMPO Y COMBUSTIBLE

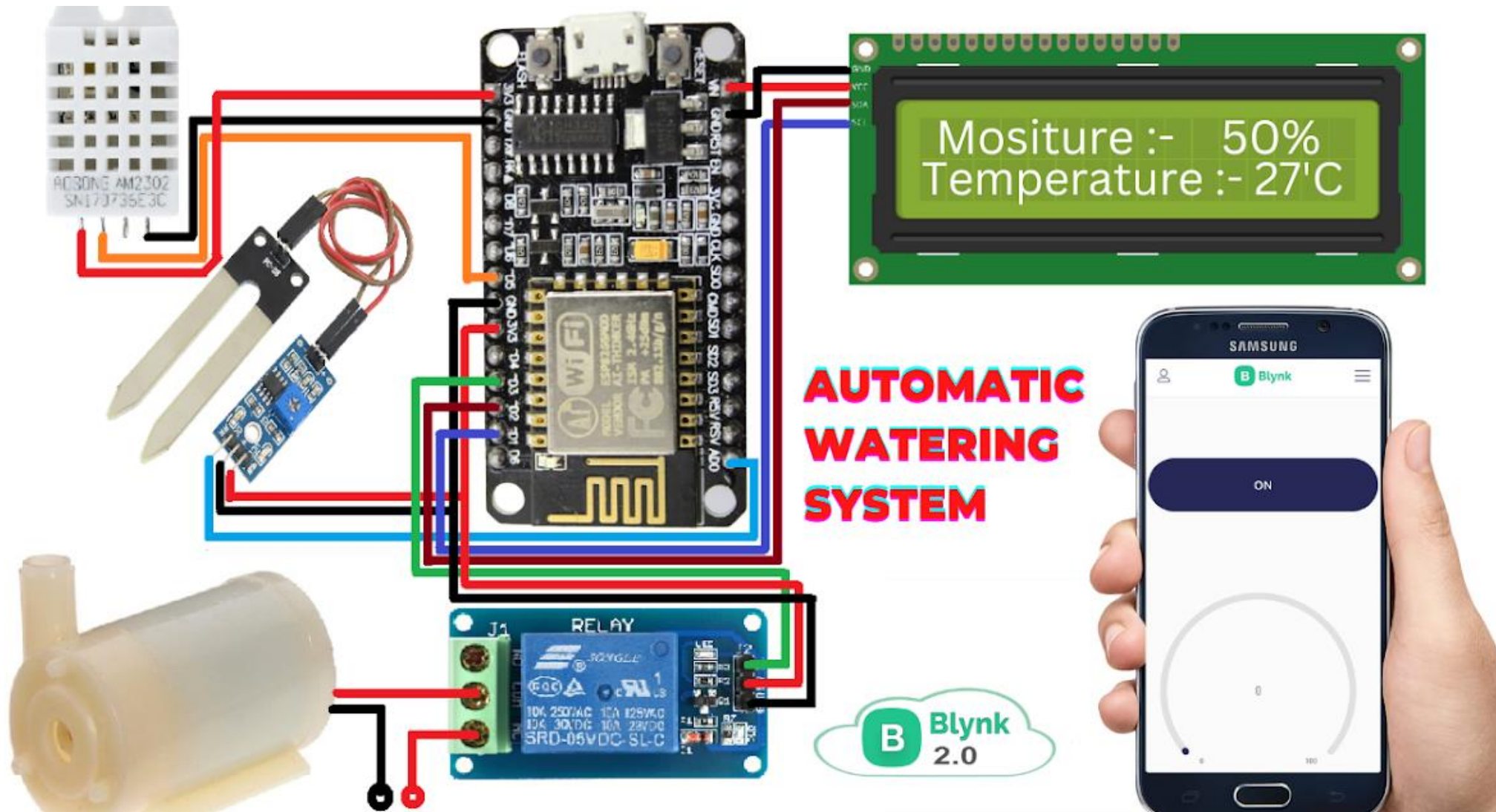
PIIE HUERTO ESCOLAR (time-lapse)







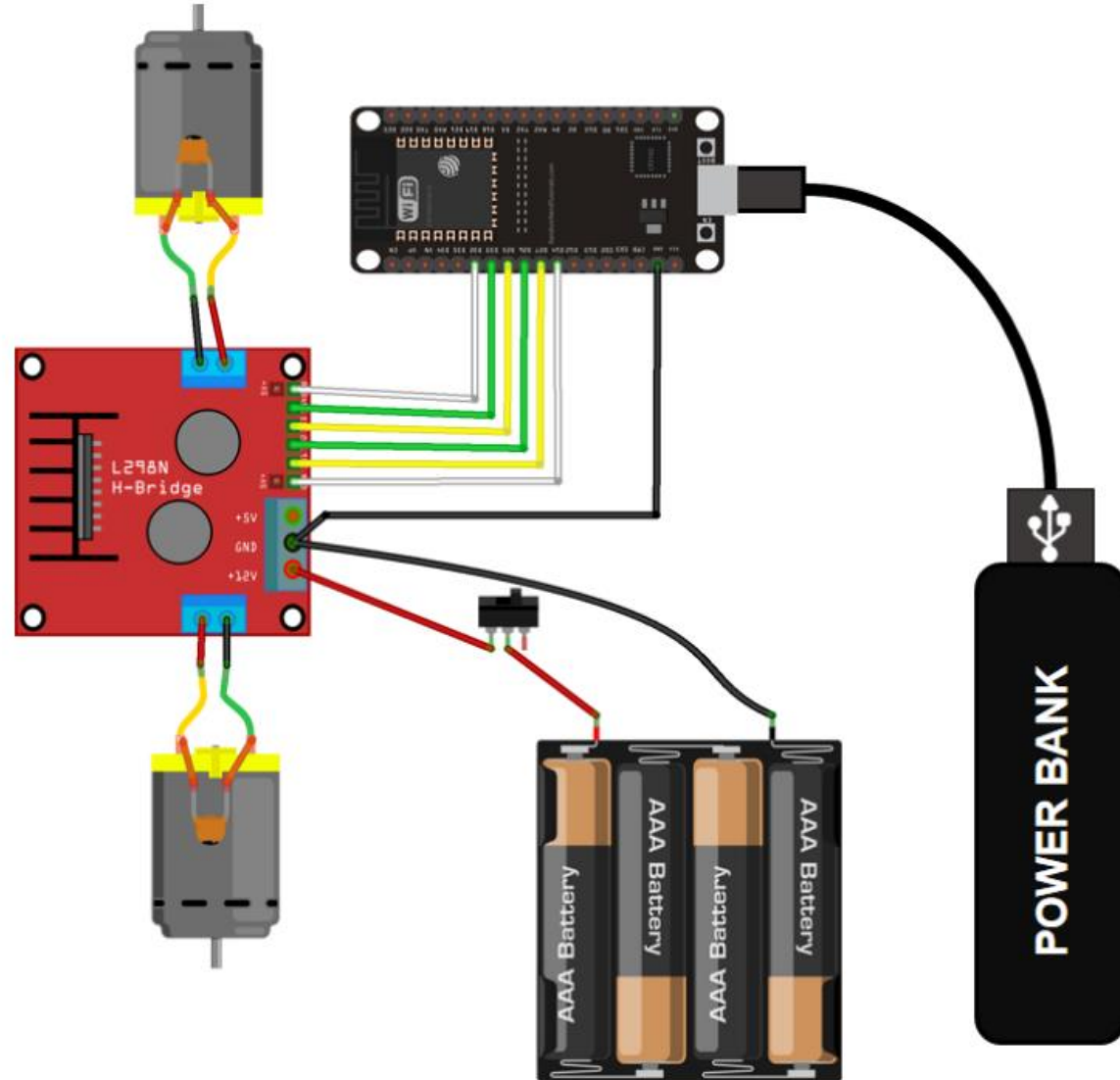
PIIE HUERTO ESCOLAR (RIEGO)



ROBOT CONTROLADO POR WIFI



ROBOT CONTROLADO POR WIFI



CONCLUSIONES Y PREGUNTAS

¡ Muchas
Gracias !