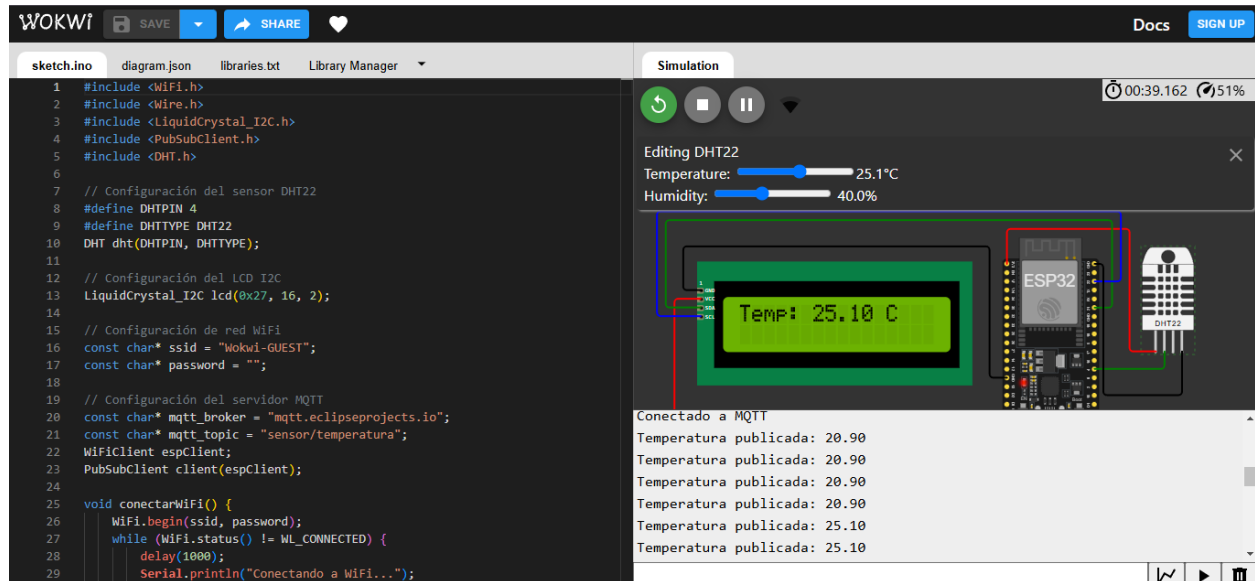


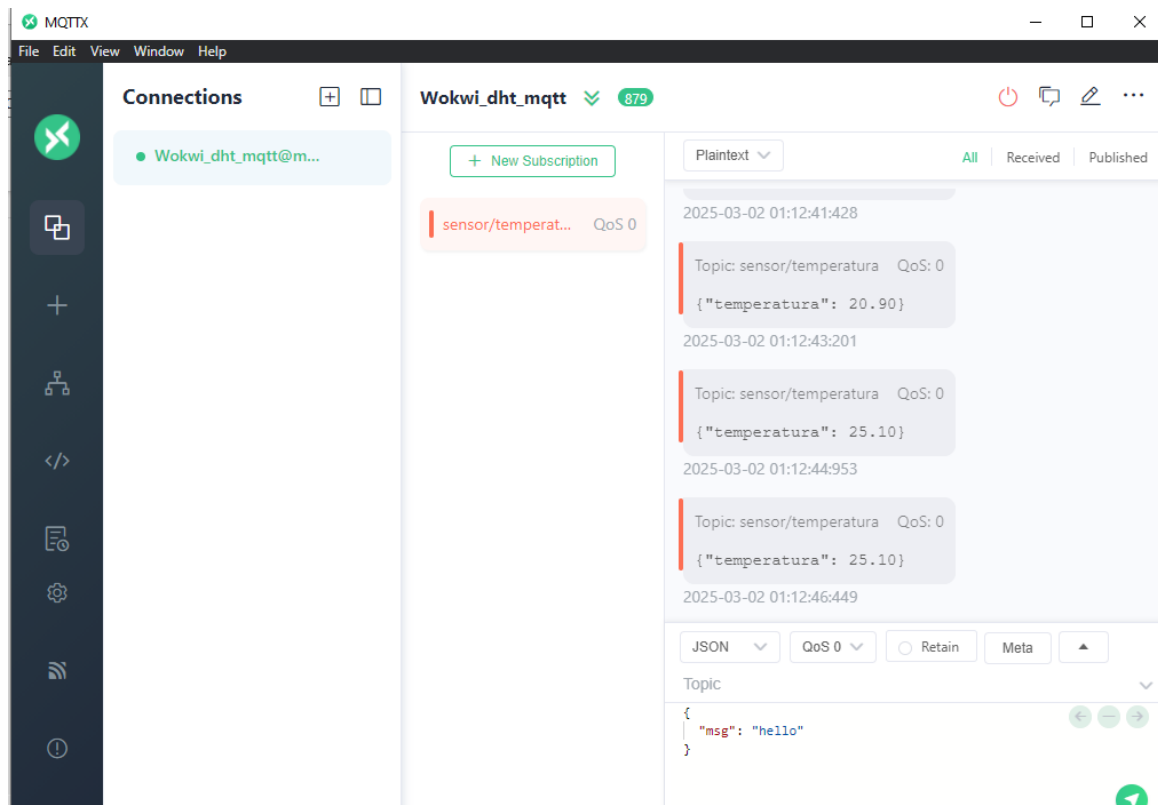
## PRUEBAS AUTOMATIZADAS UTILIZANDO LA LIBRERÍA UNITTEST CON DATOS SIMULADOS DEL WOKWI

En el programa Wokwi vemos como se van registrando los datos de la temperatura los cuales pueden ser modificados manualmente.

<https://wokwi.com/projects/422727683740780545>



Vemos que estos datos llegan correctamente al programa MQTTX:



Si nos dirigimos al Python vemos como los datos llegan correctamente desde bróker:

```
pruebas_sensor_realtime.py > ...
1  import unittest
2  import paho.mqtt.client as mqtt
3  import json
4  import time
5
6  BROKER = "mqtt.eclipseprojects.io"
7  TOPIC = "sensor/temperatura"
8
9  temperatura_actual = None
10
11 def on_message(client, userdata, msg):
12     global temperatura_actual
13     try:
14         payload = json.loads(msg.payload.decode())
15         temperatura_actual = payload.get("temperatura")
16
17
18 cls.client = mqtt.Client()
19 Esperando primeros mensajes... (5 segundos)
20 Esperando una temperatura válida...
21 Mensaje recibido: {'temperatura': 20.9}
22 Temperatura recibida: 20.9
23 Mensaje recibido: {'temperatura': 20.9}
24 Temperatura recibida: 20.9
25 Mensaje recibido: {'temperatura': 25.1}
26 Temperatura recibida: 25.1
27 Mensaje recibido: {'temperatura': 25.1}
28 .
29 -----
30 Ran 1 test in 18.360s
31
32 OK
33 PS C:\Users\Nancy\OneDrive\Escritorio\Clases Python>
```

Al finalizar nos aparece un mensaje de “OK”, lo que significa que las pruebas han concluido correctamente, es decir, no se detectaron temperaturas fuera del rango establecido.

Esto tambien aparece si lo analizamos desde el cmd.

```
C:\Windows\System32\cmd.exe
C:\Users\Nancy\OneDrive\Escritorio\Clases Python>python -m unittest pruebas_sensor_realtime
C:\Users\Nancy\OneDrive\Escritorio\Clases Python\pruebas_sensor_realtime.py:39: DeprecationWarning: Callback API version
1 is deprecated, update to latest version
  cls.client = mqtt.Client()
Esperando primeros mensajes... (5 segundos)
Esperando una temperatura válida...
Mensaje recibido: {'temperatura': 20.9}
Temperatura recibida: 20.9
Mensaje recibido: {'temperatura': 20.9}
Temperatura recibida: 20.9
Mensaje recibido: {'temperatura': 25.1}
Temperatura recibida: 25.1
Mensaje recibido: {'temperatura': 25.1}
.
-----
Ran 1 test in 17.968s
OK
```

### RESULTADOS DE LAS PRUEBAS REALES PARA LA GRAFICA DE CONTROL:

Es el mismo procedimiento que las pruebas simuladas solo que ya no se conecta con el Wokwi, en este caso, se esperan los datos del sensor DHT11, teniendo en cuenta que previamente ya conectamos la PC a la placa ESP32.

```
self.client = mqtt.Client()  
Conectado al broker broker.emqx.io, suscrito al t3pico temperatura/sensor
```

Estos datos indican las primeras temperaturas registradas:

```
Topic: temperatura/sensor  QoS: 0  
{"temperatura": 29.30}  
2025-03-02 09:02:11:569
```

```
Topic: temperatura/sensor  QoS: 0  
{"temperatura": 29.30}  
2025-03-02 09:02:13:668
```

```
Conectando a WiFi.....  
WiFi conectado.  
Conectando al broker MQTT...  
Conectado a MQTT.  
Temperatura publicada: {"temperatura": 29.30}  
Temperatura publicada: {"temperatura": 29.30}
```

## Grafica de control:

La gráfica de control muestra la evolución de la temperatura en el tiempo con base en límites de control estadísticos. En este caso, se observa que la temperatura se mantienen dentro de los límites establecidos, lo que indica que el sistema está operando de manera estable y sin anomalías significativas.

### Elementos de la Gráfica

1. Línea Azul - Temperatura: Representa los valores de temperatura registrados a lo largo del tiempo. Se observa una ligera disminución seguida de un aumento.
2. Línea Verde Punteada - Media: Representa el valor promedio de la temperatura durante el periodo analizado.
3. Línea Roja Punteada - Límite Superior de Control (LSC): Es el umbral superior (+3 desviaciones estándar) que indica la máxima variación aceptable antes de considerar que existe una anomalía de alta temperatura.
4. Línea Azul Punteada - Límite Inferior de Control (LIC): Es el umbral inferior (-3 desviaciones estándar), que marca la temperatura mínima aceptable antes de considerarse una anomalía de temperatura baja.

Los límites de control se establecen utilizando el criterio de Six Sigma, que se basa en la desviación estándar ( $\sigma$ ) de los datos. Se calculan de la siguiente manera:

- **Límite Superior de Control (LSC):**

$$LSC = \mu + 3\sigma$$

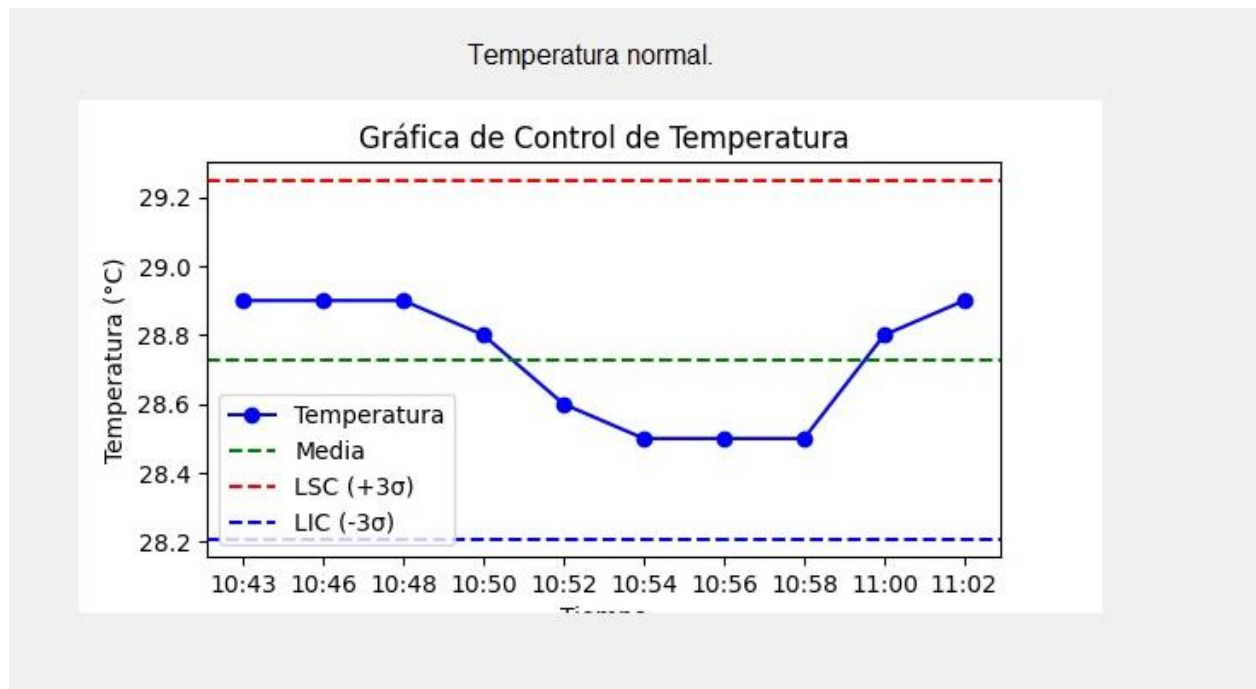
- **Límite Inferior de Control (LIC):**

$$LIC = \mu - 3\sigma$$

Donde:

- $\mu$  es la media de las temperaturas registradas.
- $\sigma$  es la desviación estándar de los datos.

Si se sigue corriendo el programa, se obtendrán mas datos de temperatura, de los cuales, los primeros 10 datos determinan los limites y los siguientes son analizados para determinar si se detecta alguna anomalía o si esta en el rango correcto como es el caso de la siguiente gráfica:



La gráfica indica que la temperatura se mantiene dentro de un rango aceptable sin sobrepasar los límites de control, lo que sugiere que el proceso es estable y no hay fallas en el sistema de monitoreo de temperatura.