

EVIDENCIAS PRACTICA NODE RED

Integrantes: JOSE DAVID FLOREZ RAMOS - 2174241
DIEGO ANDRES GARCIA DIAZ - 2195533

Fecha: 08/11/2024

Actividad 1:

Pegar aquí los pantallazos de los resultados que evidencien la recepción de mensajes en el nodo debug en Node-RED:

Inicialmente se configuró el **servidor y puerto** en la Propiedad del bloque de “MQTT Out Node”, además de también agregar el nombre del **tópico (prac3_Extra/nodered)** en la respectiva propiedad del bloque:

The image shows two side-by-side configuration panels from Node-RED. The left panel is titled "Edit mqtt broker node" and the right panel is titled "Edit mqtt out node". Both panels have "Properties" tabs and "Delete", "Cancel", and "Update/Done" buttons.

Edit mqtt broker node (Left):

- Properties:** Name: Server EMQX, Connection: broker.emqx.io, Port: 1883, Protocol: MQTT V5, Client ID: Leave blank for auto generated, Keep Alive: 60, Session: Use clean start.
- Buttons:** Enabled, 2.

Edit mqtt out node (Right):

- Properties:** Server: Server EMQX, Topic: prac3_Extra/nodered, QoS: 0, Retain: none, User Properties: none, Response topic: none, Content Type: none, Expiry (secs): none, Name: Publicador.
- Tip:** Leave topic, qos or retain blank if you want to set them via msg properties.
- Buttons:** Enabled.

También se configuraron las propiedades del bloque “MQTT In Node”:

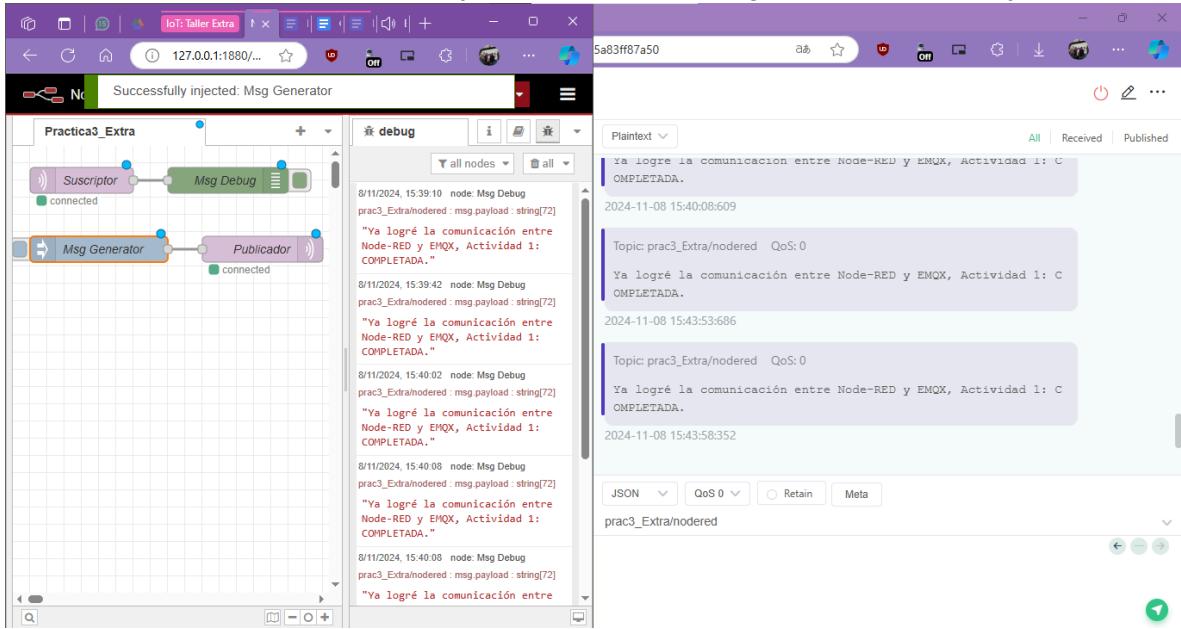
The image shows the configuration panel for the "Edit mqtt in node".

Properties:

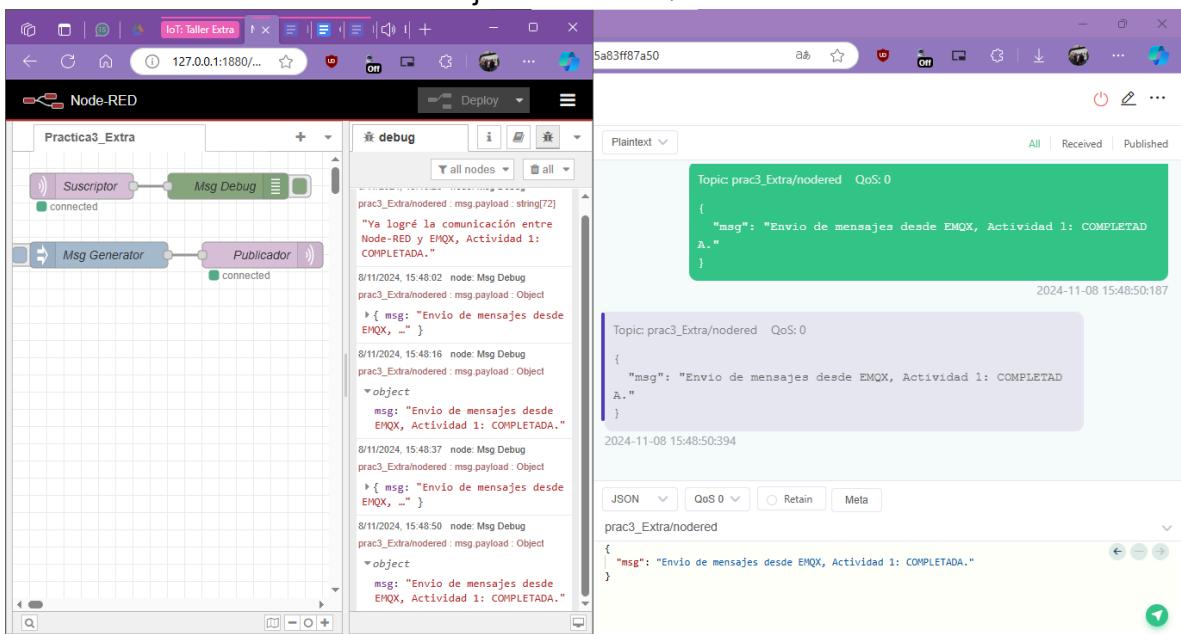
- Server:** Server EMQX
- Action:** Subscribe to single topic
- Topic:** prac3_Extra/nodered
- QoS:** 0
- Flags:** Do not receive messages published by this client (unchecked), Keep retain flag of original publish (checked).
- Retained message handling:** Send retained messages
- Output:** auto-detect (parsed JSON object, string or buffer)
- Name:** Suscriptor

Buttons: Enabled.

Se hace la prueba enviando mensajes desde Node-RED gracias al bloque “Inject”:



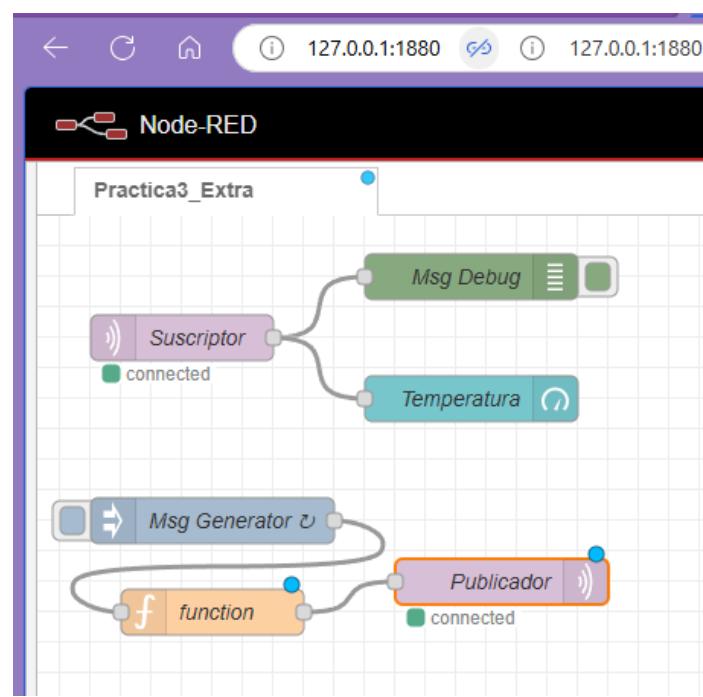
Finalmente se hace envío de mensajes desde EMQX:



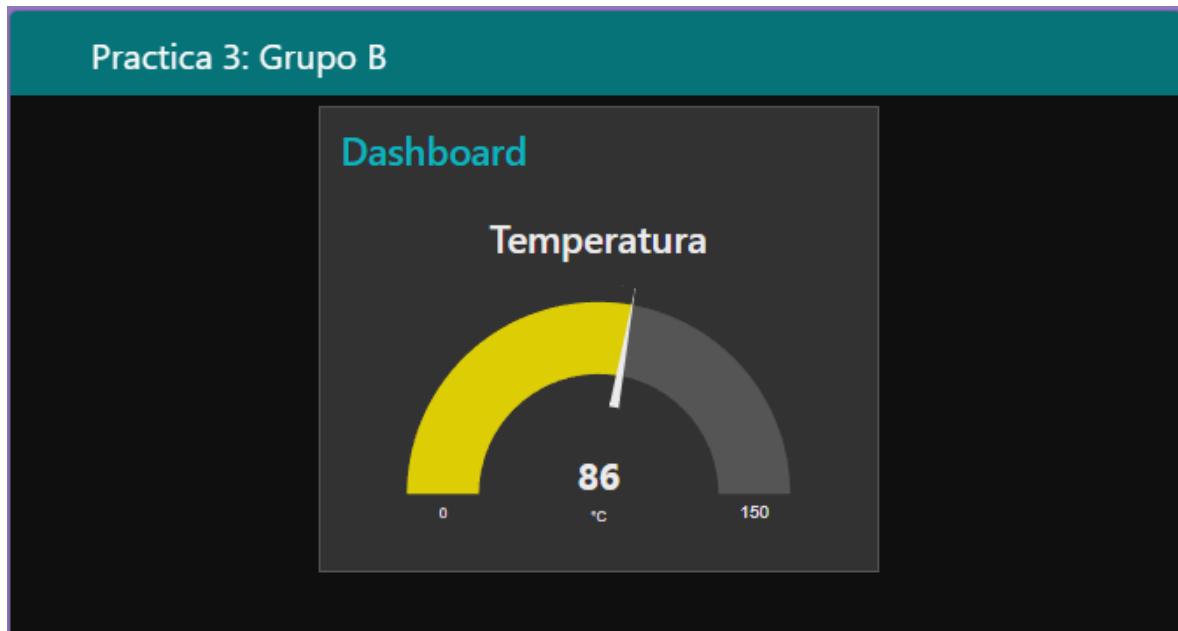
Actividad 2:

Pegar aquí el pantallazo del flujograma en Node-RED y la visualización del dashboard:

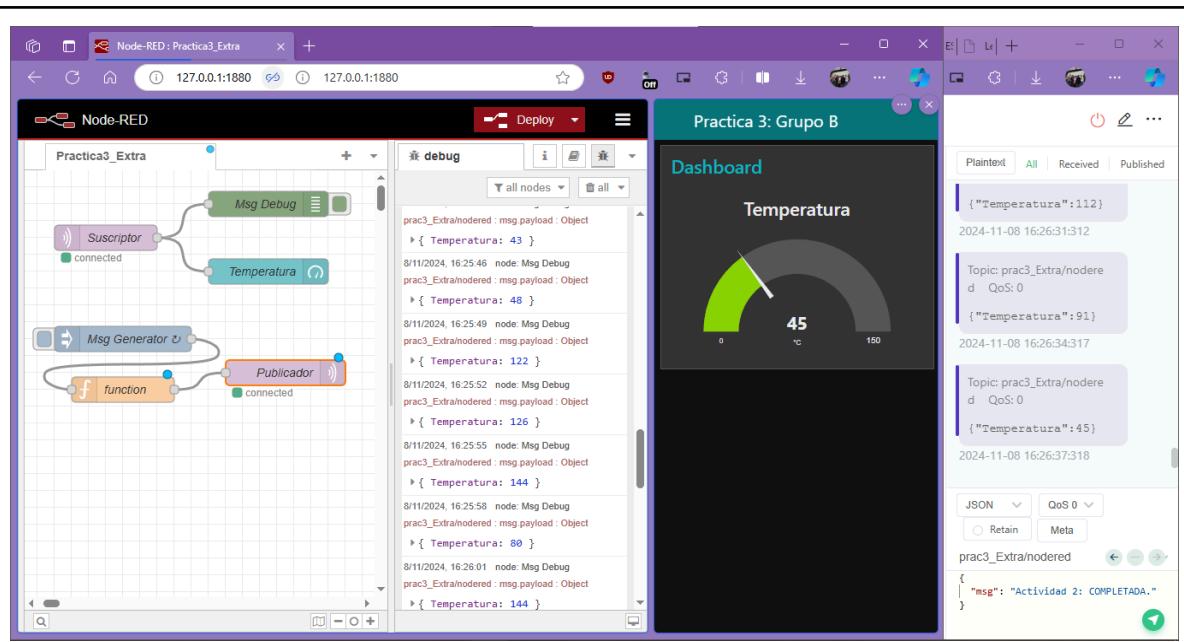
A continuación se muestra el flujograma en Node-RED:



A continuación se muestra la visualización del **Dashboard**:



Por último se muestra todo en conjunto: Node-RED, mensajes vistos desde Node-RED, el dashboard y los mensajes vistos desde EMQX:



Actividad 3:

Pegar aquí el pantallazo de la Fig. 28 de la práctica donde debe conectarse a una ciudad diferente a la de la práctica y mostrar el dashboard para medir la temperatura de esa ciudad:

Por ubicación para Bucaramanga, Santander:

```

msg.payload : Object
  object
    id: 741
    weather: "Fog"
    detail: "bruma"
    icon: "50n"
    tempk: 294.88
    tempc: 21.7
    temp_maxc: 21.7
    temp_min: 21.7
    humidity: 94
    pressure: 1010
    maxtemp: 294.88
    mintemp: 294.88
    windspeed: 2.57
    winddirection: 320
    location: "Bucaramanga"
    sunrise: 1731062485
    sunset: 1731105070
    clouds: 75
    description: "The weather in Bucaramanga at coordinates: 7.1254, -73.1198 is Fog (bruma)."
  
```

Por coordenadas para Carmen de Carupa, Cundinamarca:

Edit openweathermap node

Delete Cancel Done

Properties

API Key:
 Language: Spanish
 Current weather for: Coordinates
 Location: Coordinates
 Latitude: 5.3476413
 Longitude: -73.8991016
 Name: Uso-Nodo-RED

Tip: you may want to manually [search for the location](#) first to check it can be found. Often two character country codes work best.

Note: The API Key may not immediately ready to use after creation. As FAQ said: [Your API key will be activated automatically between 10 and 60 minutes](#).

Enabled

debug

```

object
id: 500
weather: "Rain"
detail: "lluvia ligera"
icon: "10n"
tempk: 283.02
tempc: 9.8
temp_maxc: 9.8
temp_min: 9.8
humidity: 98
pressure: 1014
maxtemp: 283.02
mintemp: 283.02
windspeed: 0.48
winddirection: 176
location: "Carmen de Carupa"
sunrise: 1731062544
sunset: 1731105387
clouds: 98
rain: 0.12
description: "The weather in Carmen de Carupa at coordinates: 5.3486, -73.9017 is Rain (lluvia ligera)."
  
```

Flujograma y datos al generar una marca de tiempo:

127.0.0.1:1880/#flow/65aae15e1aeb4582 Deploy

Node-RED

Practica3_Actividad3

debug

```

object
id: 500
weather: "Rain"
detail: "lluvia ligera"
icon: "10n"
tempk: 283.02
tempc: 9.8
temp_maxc: 9.8
temp_min: 9.8
humidity: 98
pressure: 1014
maxtemp: 283.02
mintemp: 283.02
windspeed: 0.48
winddirection: 176
location: "Carmen de Carupa"
sunrise: 1731062544
sunset: 1731105387
clouds: 98
rain: 0.12
description: "The weather in Carmen de Carupa at coordinates: 5.3486, -73.9017 is Rain (lluvia ligera)."
  
```

Por último se muestra un flujograma con algunas modificaciones y también se muestra la interfaz gráfica realizada:

The screenshot shows the Node-RED interface with a flow titled "Practica3_Actividad3". The flow consists of three parallel sections, each processing data from a "Uso-Nodo-RED" node. Each section includes a "Marca Tiempo" node, a "Uso-Nodo-RED" node, a "Debug" node, and a "Descripción" node. The output of each "Descripción" node connects to a "Publicador" node. The "Publicador" nodes publish data to topics like "prac3_Extra/nodered". A "debug" tab on the right shows the published messages, which include city names (Bucaramanga, Carmen de Carupa, Tunja), weather details (Cloudy, nubes), icons (Cloud icon), and temperatures (281.64, 283.03, 293.88). To the right, a dashboard titled "Practica 3 - Grupo B: Actividad 3" displays three cards: "Ciudad 1: Bucaramanga" with a temperature of 20.7°C, "Ciudad 2: Carmen de Carupa" with 8.4°C, and "Ciudad 3: Tunja" with 9.8°C. Each card includes a brief description of the weather at its respective coordinates.

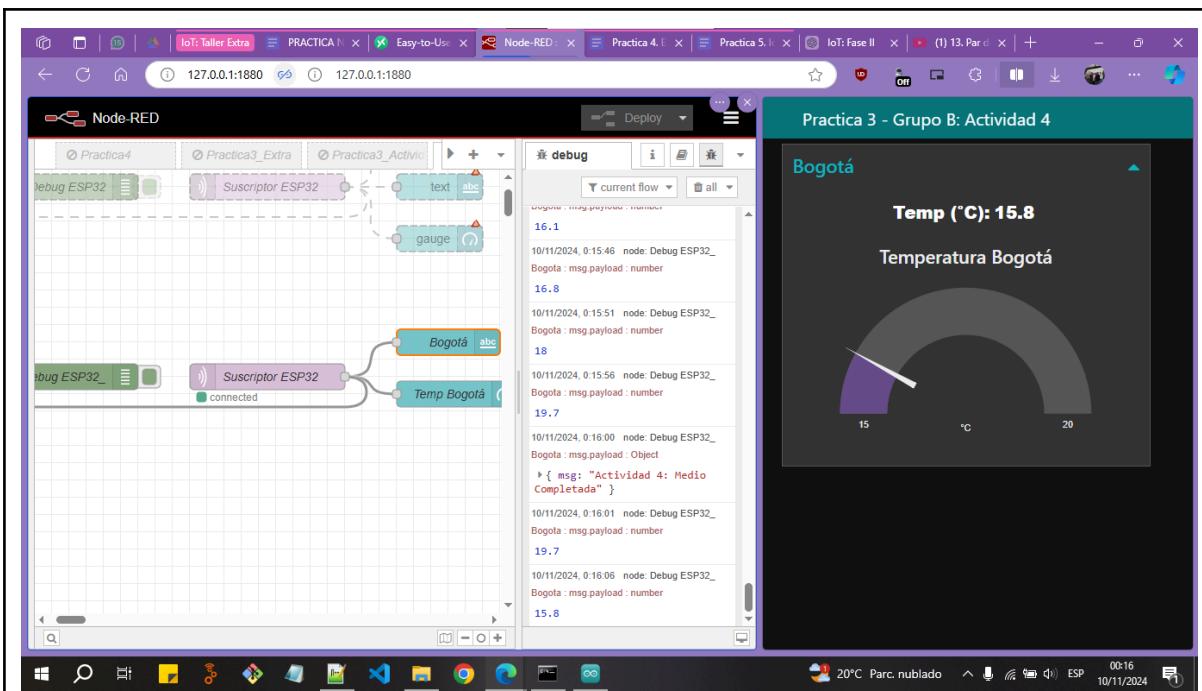
También se muestra la conexión a EMQX:

This screenshot shows the Node-RED interface connected to an MQTT broker. The flow "Practica3_E" receives data from a "Suscriptor" node and processes it through a "Marca Tiempo" node and a "Uso-Nodo-RED" node. The output is published to the topic "prac3_Extra/nodered". On the right, the MQTT interface shows the published message in plain text and JSON format. The message content includes weather details for Bucaramanga (Cloudy, nubes, icon 04n, tempk 293.88) and a completion message ("Actividad 3: COMPLETADA."). Below the message list, a green box indicates the message was received by the topic "Topic: practica3_Extra/nodered QoS: 1".

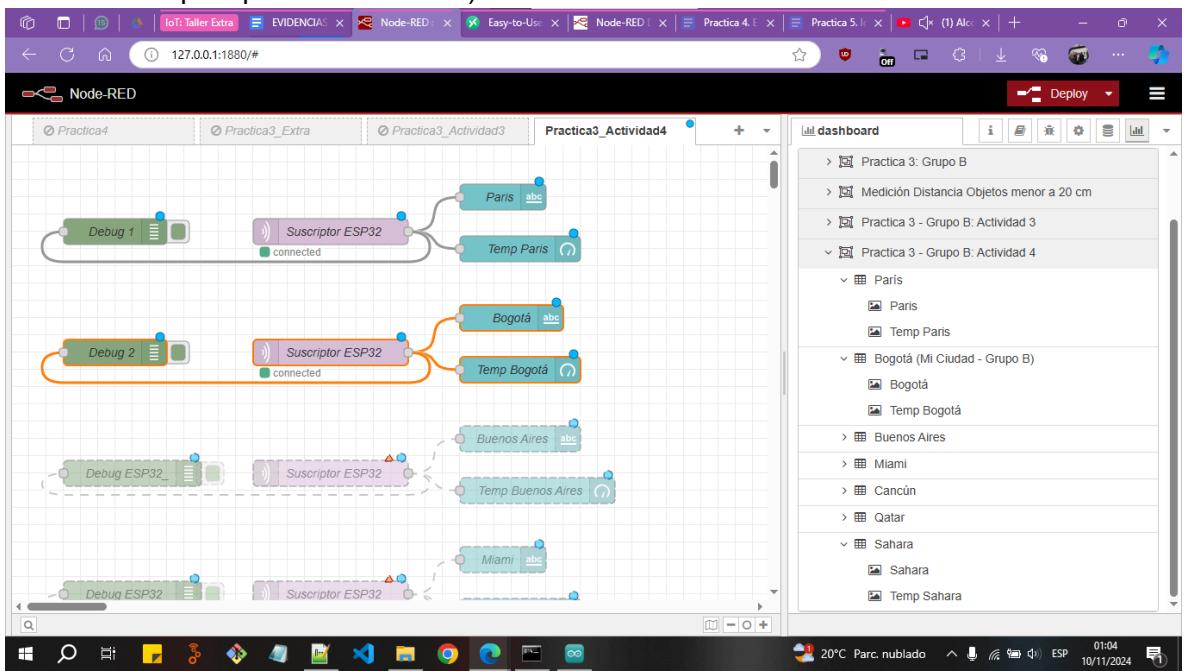
Actividad 4: RETO

Pegar aquí el flujo gráfico propuesto en node red para la recepción de los datos y el dashboard para la visualización de las 7 ciudades señalando cuál es su ciudad:

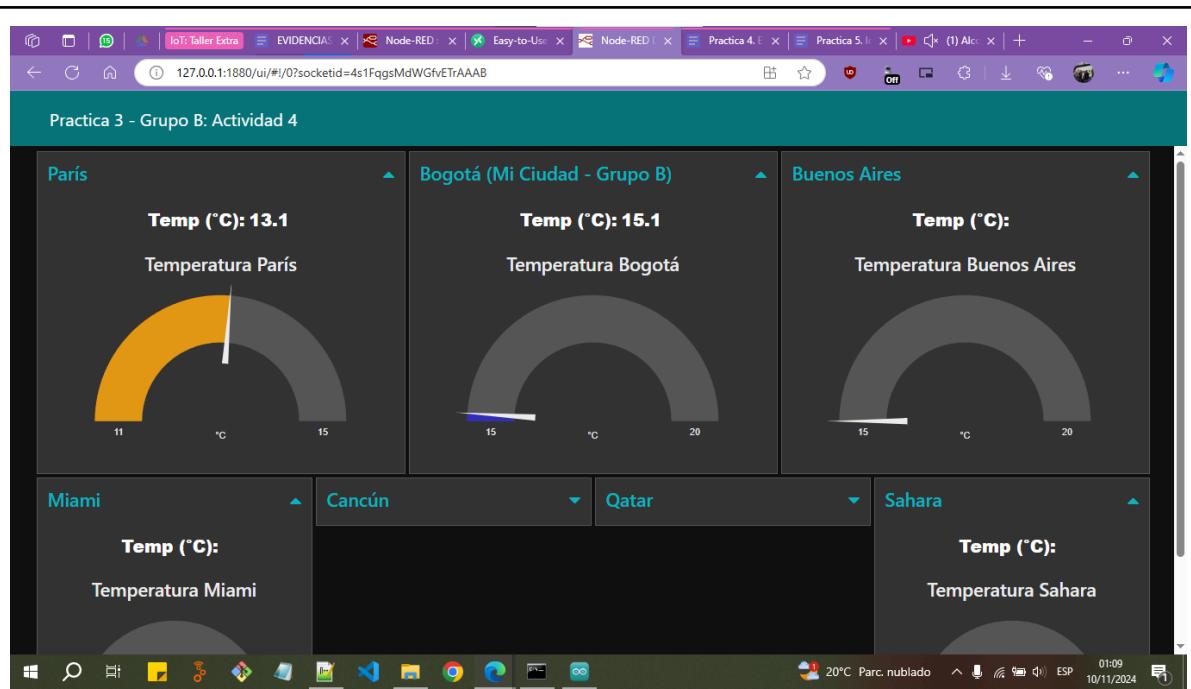
Se tiene el código cargado en la ESP32 y ya se logra observar el dato generado en la interfaz gráfica realizada por el momento:



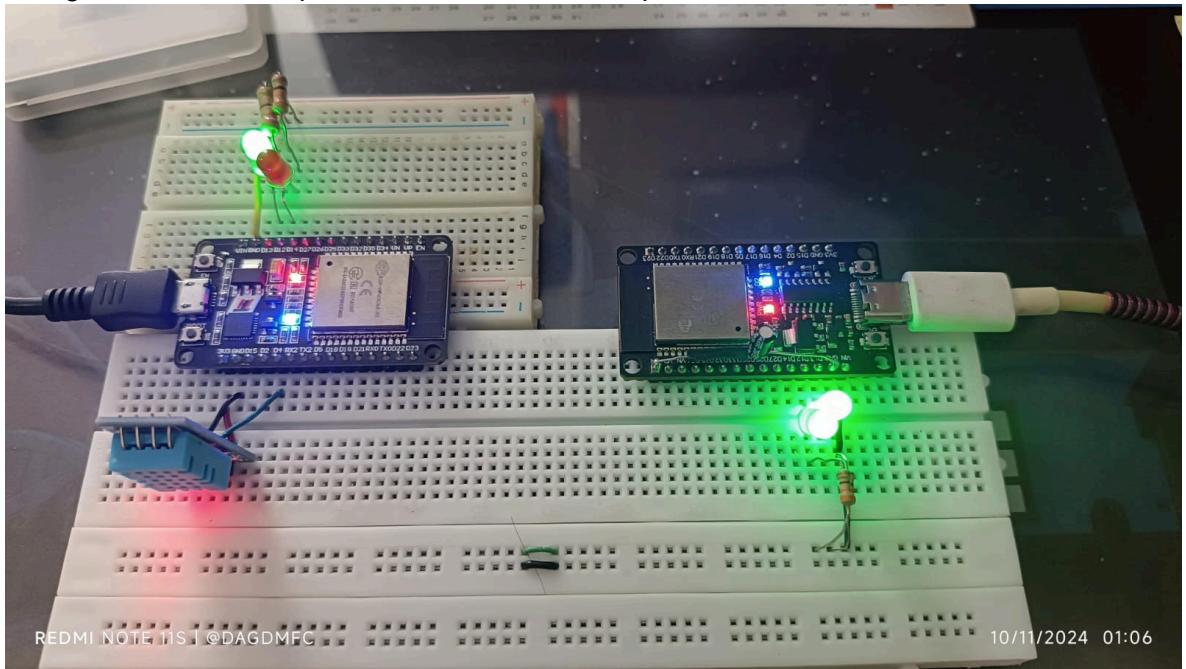
Luego, se procede a hacer lo mismo para las 6 ciudades restantes (se debe configurar el nombre del tópico para cada ciudad):



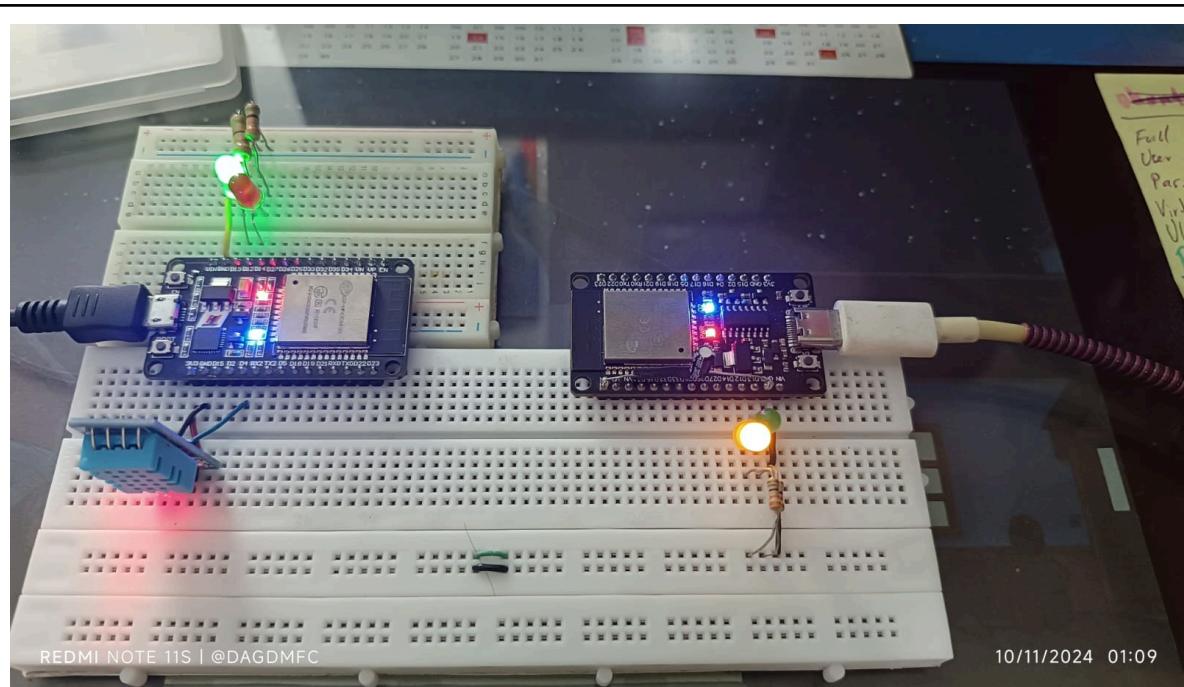
En este caso se hizo para las dos primeras ciudades la interfaz gráfica, haciendo uso de 2 ESP32, pero igualmente ya esta para observar la temperatura de cada una de las 7 ciudades:



Acá se observan dos ESP32, la de la parte izquierda es para la ciudad de Bogotá y la de la derecha es para la ciudad de París, los led color verde indican la correcta conexión al Broker, los leds azules indican la conexión correcta al Wi-Fi. En la ESP32 de la izquierda también se observa un sensor de temperatura (DHT11), el cual podría ser reemplazado en lugar de la función que simula un valor de temperatura:



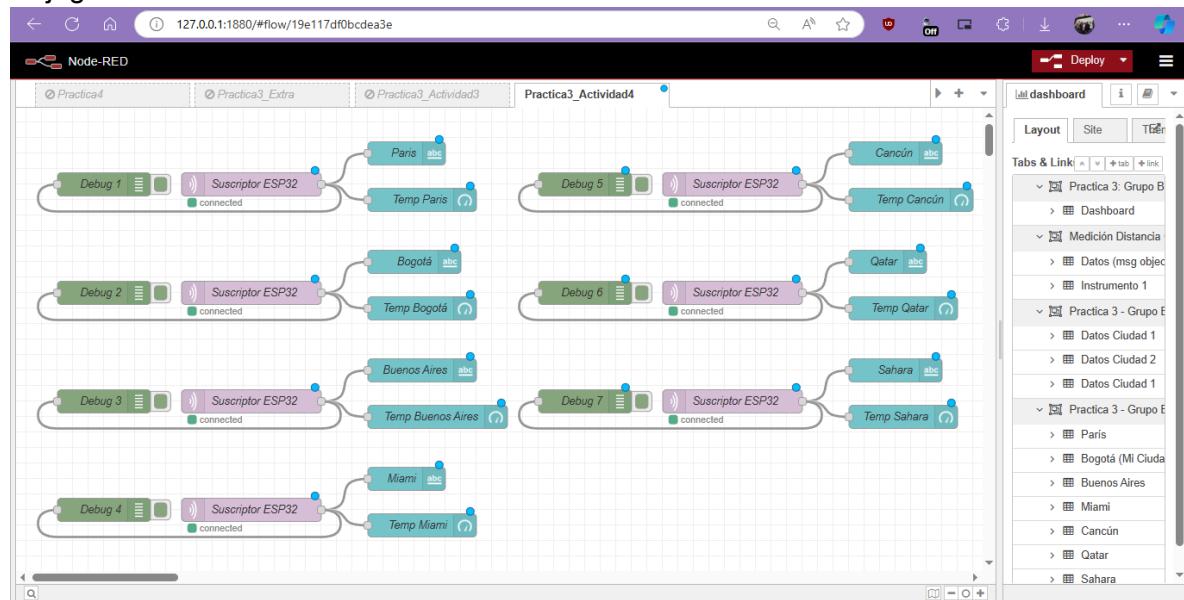
En este caso se observa lo mismo, solo que con la diferencia de que en la ESP32 de la ciudad de París (ESP32 de la derecha) por medio de led indicador amarillo, se observa que esta fallando la conexión al Broker:



12/11/2024

Los demás compañeros de clase terminaron de configurar los códigos para cada una de las 7 ciudades y luego se logró observar varios datos cambiando en la interfaz gráfica creada en Node-RED, pero algo raro ya que en un momento no hacia la correcta conexión al broker y no publicaba los datos:

Flujograma:



Interfaz Gráfica:

