

## **Entregable 3 - Ampliación**



### **Sistemas Industriales**

Francisco Javier Pérez Martínez

16 de enero de 2022

# Índice

<b>1. Descripción</b>	<b>3</b>
<b>2. API OpenWeather</b>	<b>3</b>
2.1. Dashboard node-red . . . . .	3
2.2. Dashboard Ubidots . . . . .	4
<b>3. Notificación de los datos capturados</b>	<b>6</b>
3.1. Email (SMTP) . . . . .	6
3.2. Bot de Telegram . . . . .	7
3.3. HiveMQ - Open source MQTT . . . . .	8
<b>Referencias</b>	<b>9</b>

## 1. Descripción

Para el presente entregable, correspondiente a la ampliación del bloque 1 de la asignatura, se ha utilizado la herramienta de node-red para conectar y probar la lectura de datos de una API de OpenWeather relacionada con el tiempo atmosférico en la ciudad de Alicante. Una vez capturado estos datos, se ha empleado una plataforma de IoT para lanzar y visualizar los datos obtenidos en un dashboard, externo a node-red, el cual se presentó en una de las clases de teoría denominado Ubidots.

Además, se ha utilizado varias formas para notificar estos datos capturados. Siendo estos enviados por correo electrónico utilizando el protocolo SMTP, un Bot de Telegram y el empleo del protocolo MQTT para publicar estos datos en un broker de código abierto.

## 2. API OpenWeather

En base al documento del bloque 1 entregado, vimos como capturar datos desde una API de OpenWeather y visualizarlos en node-red. En esta sección, a partir de los datos capturados veremos otra forma de visualizar los datos capturados.



Figura 1: OpenWeather logo

### 2.1. Dashboard node-red

Lo más sencillo es visualizar los datos desde la propia herramienta de node-red utilizando los nodos de dashboard.

Ejemplo de temperatura y humedad en Alicante:

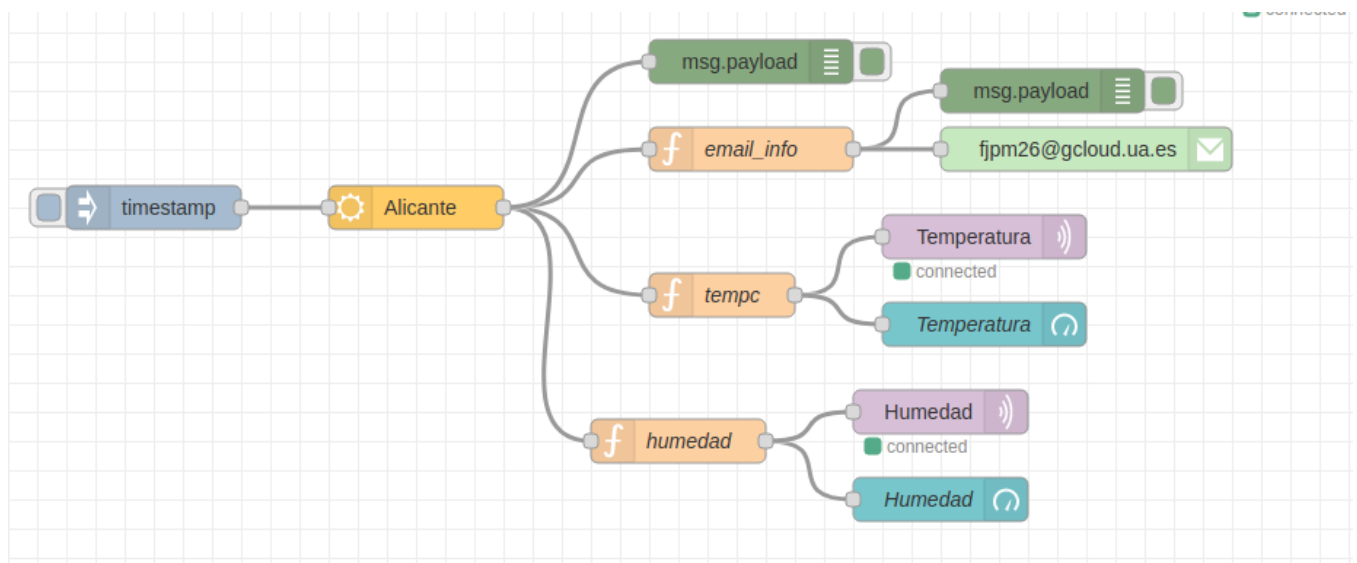


Figura 2: Nodos empleados

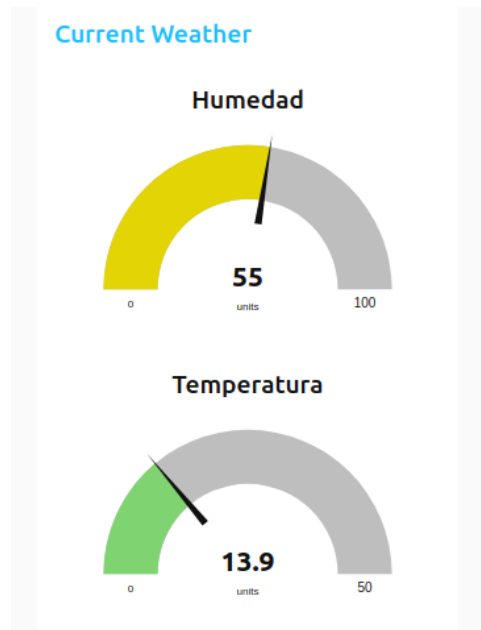


Figura 3: Dashboard node-red

## 2.2. Dashboard Ubidots

Existen otras plataformas para monitorear y visualizar aplicaciones de IoT, en este caso en particular, utilizaremos Ubidots como nueva opción de visualización de los datos, para así mostrar alguno de los datos capturados desde la API del tiempo atmosférico.



Figura 4: Ubidots logo

```
Información del tiempo actual : msg.payload : Object
▼ object
  id: 800
  weather: "Clear"
  detail: "cielo claro"
  icon: "01d"
  tempk: 287.05
  tempc: 13.9
  temp_maxc: 17.3
  temp_minc: 12.5
  humidity: 55
  pressure: 1027
  maxtemp: 290.5
  mintemp: 285.66
  windspeed: 1.34
  winddirection: 116
  location: "Alicante"
  sunrise: 1642317430
  sunset: 1642352740
  clouds: 0
  description: "The weather
  Alicante at coordinates:
  38.3452, -0.4815 is Clear (cielo
  claro)."
```

Figura 5: Datos capturados

Nodos ubidots:

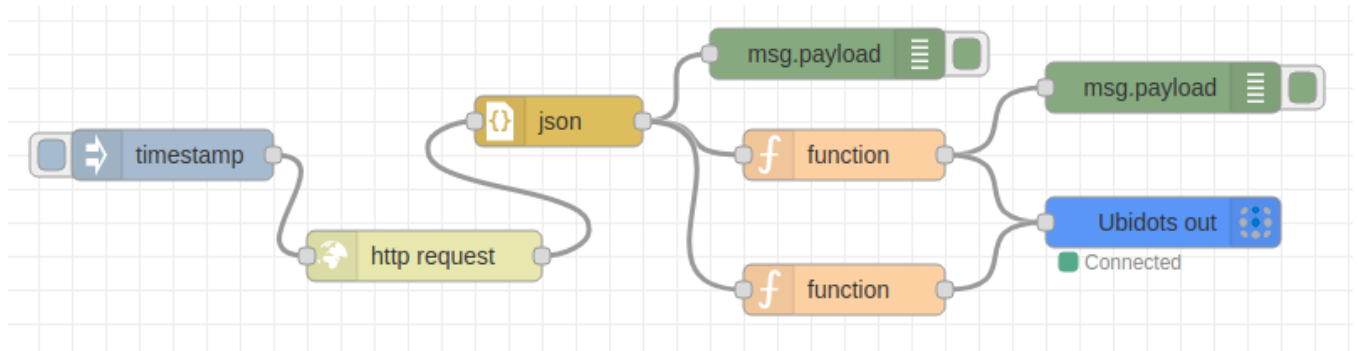


Figura 6: Datos capturados

Visualización de algunos de los datos en dashboards:

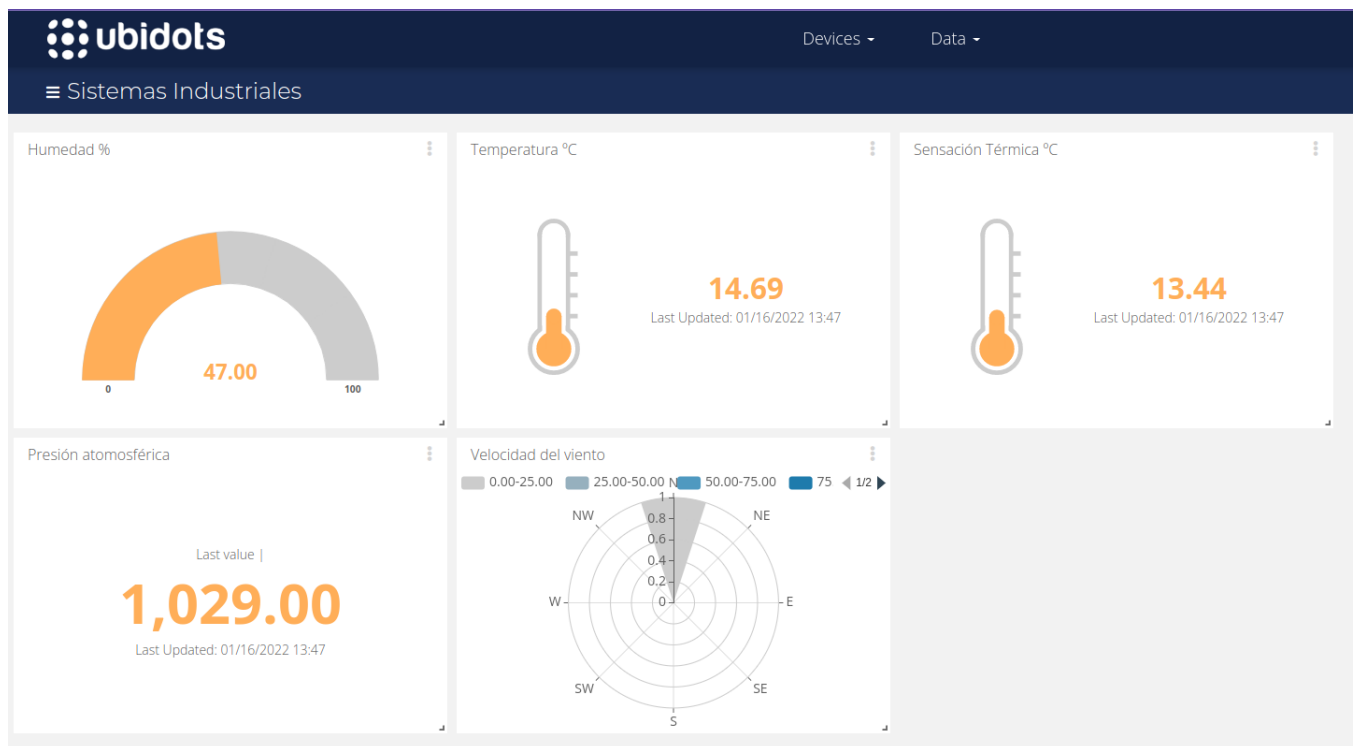


Figura 7: Dashboard Ubidots

### 3. Notificación de los datos capturados

En esta sección, veremos las diferentes formas que he utilizado para notificar los datos que se han capturado en la API mencionada.

#### 3.1. Email (SMTP)

En primer lugar, se ha empleado el protocolo SMTP, protocolo de red utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos. De esta forma, recibimos un email con la información requerida, en este caso información sobre el tiempo atmosférico, concretamente, la temperatura y el % de humedad.

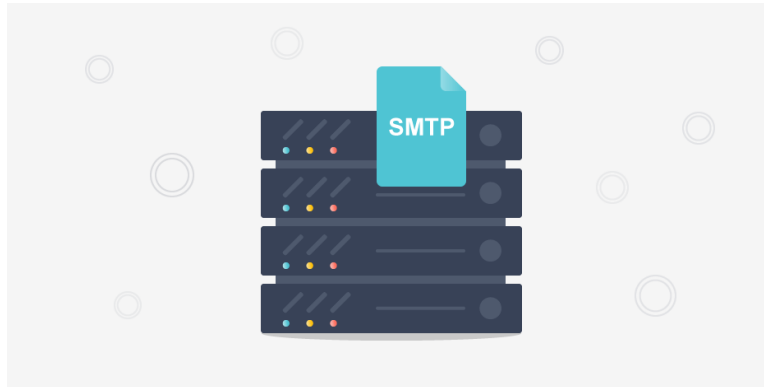


Figura 8: SMTP

Debemos configurar el nodo indicando el correo destinatario, el servidor smtp, el puerto y el usuario y contraseña que utilizaremos como emisor de la información.

El nodo en cuestión hace referencia a la figura 2 del presente documento.

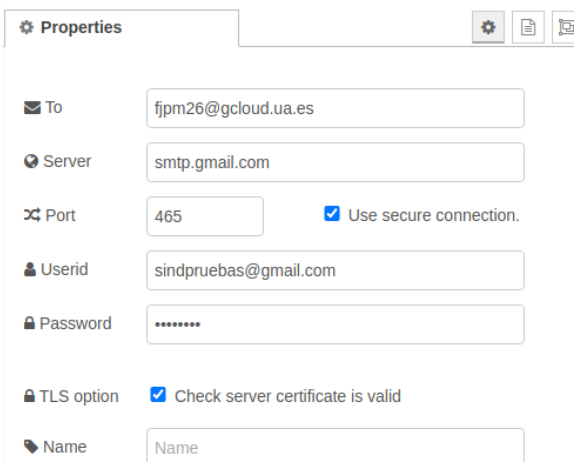
Una captura de pantalla de la interfaz de configuración de un nodo "email". El título es "Properties". Hay varios campos de entrada: "To" con el valor "fjpm26@gcloud.ua.es", "Server" con "smtp.gmail.com", "Port" con "465" y una casilla de verificación "Use secure connection." marcada. "Userid" es "sindpruebas@gmail.com" y "Password" está oculto con puntos. Hay una opción "TLS option" marcada "Check server certificate is valid" y un campo "Name" vacío.

Figura 9: Propiedades nodo email

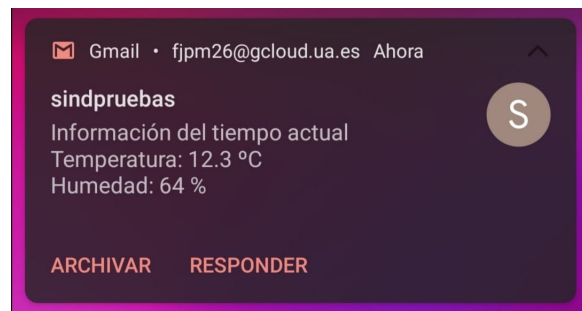


Figura 10: Ejemplo email recibido

### 3.2. Bot de Telegram

Otra forma de notificación utilizado es Telegram, se ha empleado un bot de telegram en el que según el comando introducido te responde con los datos capturados desde la API.

Nodos utilizados en node-red:

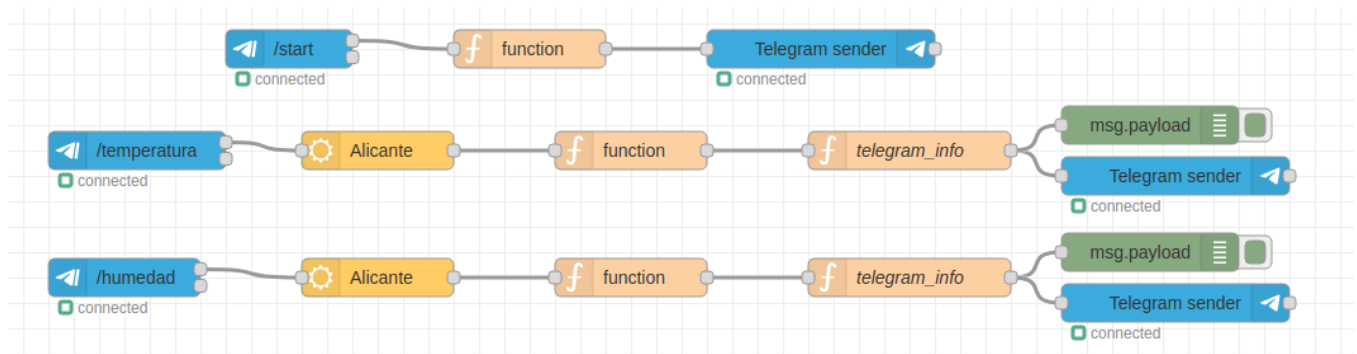


Figura 11: Nodos Telegram

Ejemplo de uso del bot:

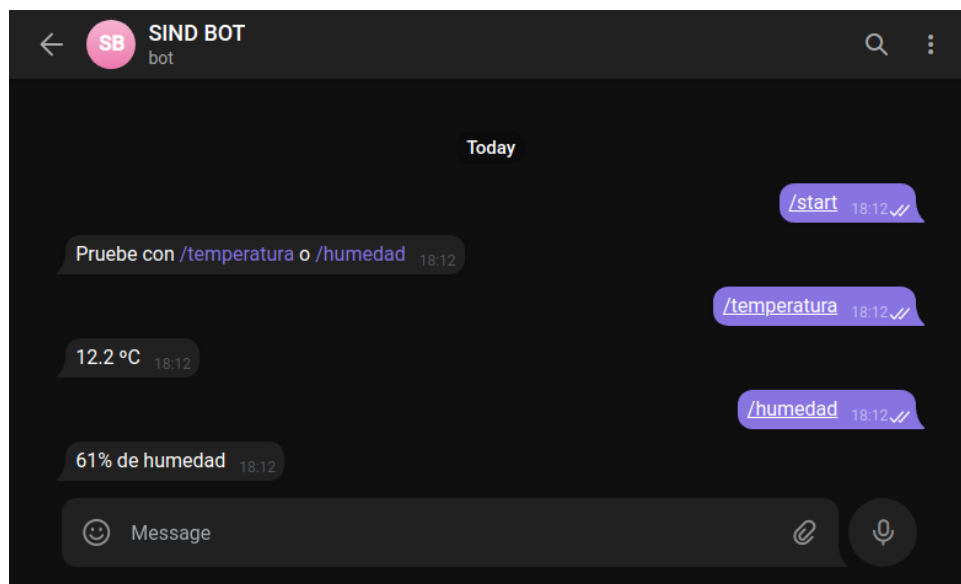


Figura 12: Ejemplo de uso bot telegram

Como hemos visto, según el comando introducido el bot te responde con la información deseada, en este caso solo se han implementado 2 comandos para mostrar el funcionamiento básico, pero podríamos introducir muchos más.

### 3.3. HiveMQ - Open source MQTT

Por último, a partir de los datos de temperatura y humedad capturados, éstos se han publicado en el broker MQTT de código abierto, HiveMQ. Para ello, introducimos el servidor con el puerto y el Topic. En la figura 2 del documento se pueden identificar los nodos mqtt out.



Figura 13: HiveMQ Logo



Figura 14: Propiedades nodo MQTT

Comprobamos que se haya publicado correctamente, suscribiéndonos al broker:

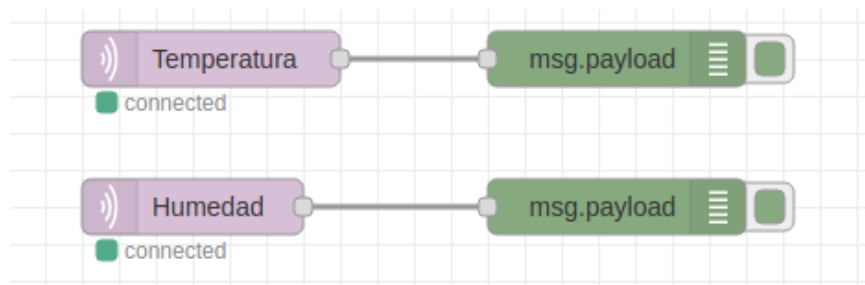


Figura 15: Nodos MQTT

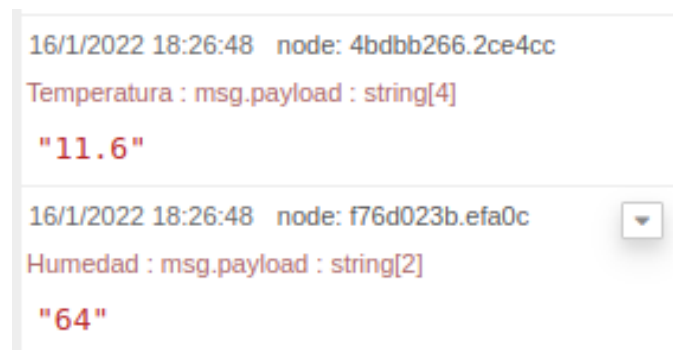


Figura 16: Mensaje debug mqtt



## Referencias

- [1] Documentation. *Nodered.Org*. 2021. URL: <https://nodered.org/docs/>.
- [2] Weather API. *Openweathermap.org*. 2021. URL: <https://openweathermap.org/api>.
- [3] Ubidots.com. *IoT platform*. 2021. URL: <https://ubidots.com/>.
- [4] HiveMQ. *Mqtt-Dashboard.Com*. 2021. URL: <http://www.mqtt-dashboard.com/index.html>.
- [5] Nodered.Org. *Node-red-contrib-telegrambo*. 2021. URL: <https://flows.nodered.org/node/node-red-contrib-telegrambot>.