

IoT: Estación Meteorológica

Lab 9 - Redes



Fredy Velasquez
Javier Valle
Angel Higueros

Simulación de un Sensor

```
PS C:\Users\sebas\Documents\GitHub\Lab9-Redes> & C:/Users/sebas/AppData/Local/Programs/Python/Pyt
Mensaje enviado: {'temperature': 63.92967152816598, 'humidity': 55, 'wind_direction': 'NE'}
Mensaje enviado: {'temperature': 68.71533928611385, 'humidity': 76, 'wind_direction': 'SW'}
Mensaje enviado: {'temperature': 66.84746807131255, 'humidity': 37, 'wind_direction': 'N'}
Mensaje enviado: {'temperature': 46.03271456874827, 'humidity': 38, 'wind_direction': 'S'}
Mensaje enviado: {'temperature': 14.236637090162226, 'humidity': 39, 'wind_direction': 'SE'}
Mensaje enviado: {'temperature': 47.82822658030636, 'humidity': 88, 'wind_direction': 'SW'}
Mensaje enviado: {'temperature': 27.91729340441552, 'humidity': 23, 'wind_direction': 'S'}
Mensaje enviado: {'temperature': 41.594424839144764, 'humidity': 88, 'wind_direction': 'S'}
Mensaje enviado: {'temperature': 41.73595805623646, 'humidity': 56, 'wind_direction': 'E'}
Mensaje enviado: {'temperature': 41.43907510755261, 'humidity': 37, 'wind_direction': 'NW'}
```



¿A qué capa pertenece JSON/SOAP según el Modelo OSI y porque?

Pertenecen a la capa de aplicación del modelo OSI, se utilizan para la representación de datos a nivel de aplicación, facilitando la compatibilidad entre sistemas

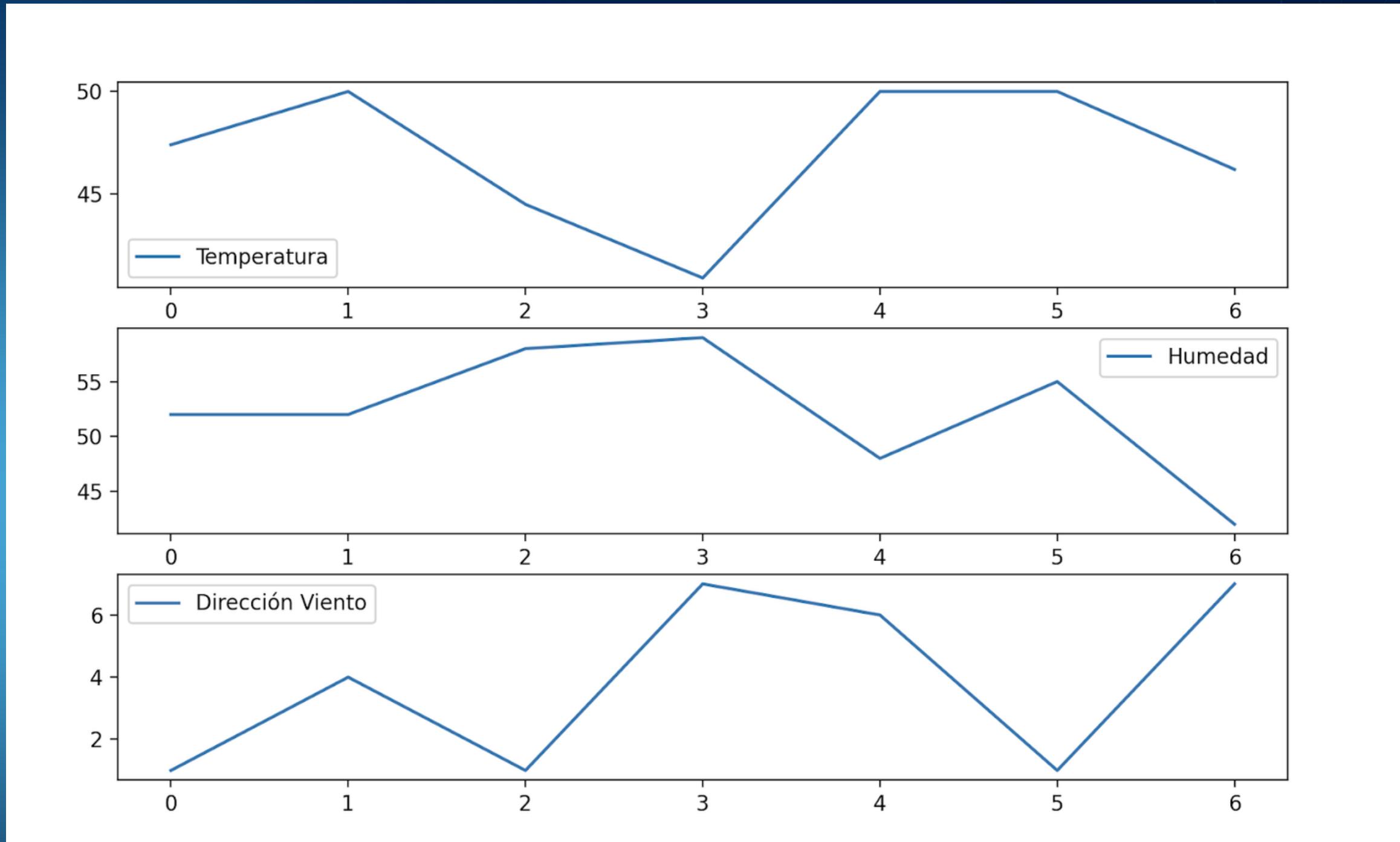
¿Qué beneficios tiene utilizar un formato como JSON/SOAP?

Existen muchos beneficios de estos formatos, como la facilidad y compatibilidad con las aplicaciones ya que es un estandar. Tambien el soporte web, adaptabilidad en el formato, entre otros.

Envío de Datos al Server Edge

```
PS C:\Users\sebas\Documents\GitHub\Lab9-Redes> & C:/Users/sebas/AppData/Local/Programs/Python/Python311/python  
Datos a enviar: {'temperatura': 28.386289996465052, 'humedad': 67.89276566942634, 'direccion_viento': 'NE'}  
Respuesta del servidor: 1  
Datos a enviar: {'temperatura': 80.86073415968343, 'humedad': 45.34874101538333, 'direccion_viento': 'NE'}  
Respuesta del servidor: 1  
Datos a enviar: {'temperatura': 66.49268958696993, 'humedad': 36.41058140068258, 'direccion_viento': 'SE'}  
Respuesta del servidor: 1  
Datos a enviar: {'temperatura': 60.45457078286988, 'humedad': 62.53791149338329, 'direccion_viento': 'S'}  
Respuesta del servidor: 1  
Datos a enviar: {'temperatura': 47.728724362122485, 'humedad': 35.49710449380646, 'direccion_viento': 'SW'}
```

Consumir y Desplegar Datos Meteorológicos



¿Qué ventajas y desventajas considera que tiene este acercamiento basado en Pub/Sub de Kafka?

Entre sus ventajas esta la escalabilidad que nos ofrece Kafka, manejo de errores, persistencia de datos y un sistema modular que nos permite tener flexibilidad en el desarrollo.



Las desventajas que tiene es su configuración previa, ya que puede llegar a ser algo compleja y tambien puede llegar a tener un overhead en recursos por sus funciones de persistencia de datos.

¿Para qué aplicaciones tiene sentido usar Kafka? ¿Para cuáles no?

Se recomendaria usarlo en aplicaciones que requieran un procesamiento y analisis de datos en tiempo real como streaming de datos. Se desaconseja usarlo en aplicaciones no muy robustas que no requieran gran estructura para el manejo de datos ya que implementar Kafka solo aumentaria su complejidad sin ningun proposito.

IoT en Entornos con Restricciones

```
PS C:\Users\sebas\Documents\GitHub\Lab9-Redes> & C:/Users/sebas/AppData/Local/Programs/Python/Python3.7.5100417164001 32 SE
14 7 3
01100101001011 0100000 011
mensaje original: {'temperature': 50.75100417164001, 'humidity': 32, 'wind_direction': 'SE'}
Mensaje enviado: 0110010100101101100000011
30.32860050981271 71 NE
14 7 3
00111100100000 1000111 001
mensaje original: {'temperature': 30.32860050981271, 'humidity': 71, 'wind_direction': 'NE'}
Mensaje enviado: 001111001000001000111001
37.47108813944401 79 N
14 7 3
01001010101111 1001111 000
mensaje original: {'temperature': 37.47108813944401, 'humidity': 79, 'wind_direction': 'N'}
Mensaje enviado: 01001010101111001111000
PS C:\Users\sebas\Documents\GitHub\Lab9-Redes>
```

¿Qué complejidades introduce el tener un payload restringido (pequeño)?

Agrega limitaciones en la representación de datos, especialmente en valores numéricos, lo que requiere estrategias de codificación eficientes

¿Cómo podemos hacer que el valor de temperatura quepa en 14 bits?

Utilizar técnicas de compresión o representación más eficientes, como la escala y desplazamiento, para reducir la cantidad de bits necesarios para representar la temperatura.

¿Qué sucedería si ahora la humedad también es tipo float con un decimal?

Se requeriría más espacio

¿Qué decisiones tendríamos que tomar en ese caso?

Se debería de ajustar la precisión de la humedad, aceptar cierta pérdida de precisión, o implementar técnicas de compresión para representar valores decimales en menos bits

¿Qué parámetros o herramientas de Kafka podrían ayudarnos si las restricciones fueran aún más fuertes?

Kafka ofrece varias herramientas para abordar diferentes restricciones, por ejemplo con la compresión de mensajes se puede usar gzip o snappy, para el batching de mensajes se puede aumentar el tamaño de los lotes (batch size) y el tiempo de espera (linger.ms) en el productor para agrupar mensajes antes de enviarlos, entre otras funciones.

