



UNIVERSIDAD TECNOLÓGICA NACIONAL
Facultad Regional Bahía Blanca

Proyecto Final Capstone

Año: 2025

Integrantes:

- Pignotti Joaquin
- Sinigaglia Franco

Tecnicatura Universitaria en Programación

Cátedra “Arquitectura y sistemas operativos”

Profesor: Mateo Menvielle

INTRODUCCIÓN:

Este proyecto tiene como objetivo desarrollar un sistema dockerizado que permita consumir y visualizar datos provenientes de un sensor. Dicho sensor proporciona información en tiempo real sobre humedad, temperatura, presión atmosférica, hora de medición y el dispositivo que realiza la medición.

Para lograrlo, se utilizará Node-RED como plataforma para el manejo del flujo de datos y visualización. Node-RED se ejecutará dentro de un contenedor gestionado con Docker, lo que facilita la portabilidad y el aislamiento del entorno.

El sistema realizará consultas periódicas al sensor, extraerá los datos más recientes y los mostrará en tiempo real mediante widgets gráficos permitiendo un monitoreo sencillo y visual del entorno medido.

DIAGRAMA DE BLOQUES:



TEORÍA:

Docker es una plataforma de código abierto que permite desarrollar, empaquetar y ejecutar aplicaciones en contenedores. Un contenedor es una unidad ligera y portátil que incluye el código de una aplicación, sus dependencias, bibliotecas y configuraciones necesarias para ejecutarla. Docker se usa para simplificar el desarrollo, la distribución y la ejecución de aplicaciones.

Una API REST es una Interfaz de Programación de Aplicaciones basada en Transferencia de Estado Representacional. Es un conjunto de endpoints (URLs) que permiten consultar, enviar, modificar o eliminar datos usando los métodos estándar de HTTP. Se utiliza como forma de permitir que diferentes sistemas se comuniquen entre sí a través de HTTP, siguiendo ciertos principios.

Un sistema IoT está compuesto por dispositivos conectados a Internet que recogen, envían y/o procesan datos del mundo físico. Su objetivo es automatizar, monitorear y tomar decisiones en tiempo real. Algunas de sus características principales son:

sensores y actuadores, capturan información del entorno y/o pueden actuar sobre su entorno (activar monitores, abrir válvulas, etc...).

Conectividad, usan Wi-Fi, bluetooth, ethernet y demás plataformas para comunicarse.

Procesamiento, interfaz de usuario, control y visualización a través de un dashboard interactivo, apps o webs, automatización.

Toman decisiones con lógica programada o IA y escalabilidad, pueden ampliarse con más nodos o dispositivos fácilmente.

Dependiendo del sensor y aplicación, los datos pueden incluir:

Ambientales: temperatura, humedad, presión, calidad del aire

Temporales: hora y fecha de la medición

Geográficos: ubicación (GPS)

Eléctricos: consumo energético, voltaje, corriente

Mecánicos: movimiento, vibración, aceleración

Identificación: qué dispositivo o sensor envió el dato

Eventos: detección de presencia, apertura de puertas, alarmas

La plataforma "Node-RED" cumple un papel clave en una arquitectura IoT al actuar como un orquestador de datos y procesos. Su función principal es facilitar la recolección, procesamiento, visualización y envío de datos de forma visual y sencilla.

DESARROLLO:

Este proyecto consiste en la implementación de un sistema de adquisición, procesamiento y visualización de datos de sensores ambientales utilizando Node-RED, ejecutado en un entorno Docker. El objetivo principal es mostrar en tiempo real los valores de temperatura, humedad y presión generados por un sensor remoto que publica los datos en una página web.

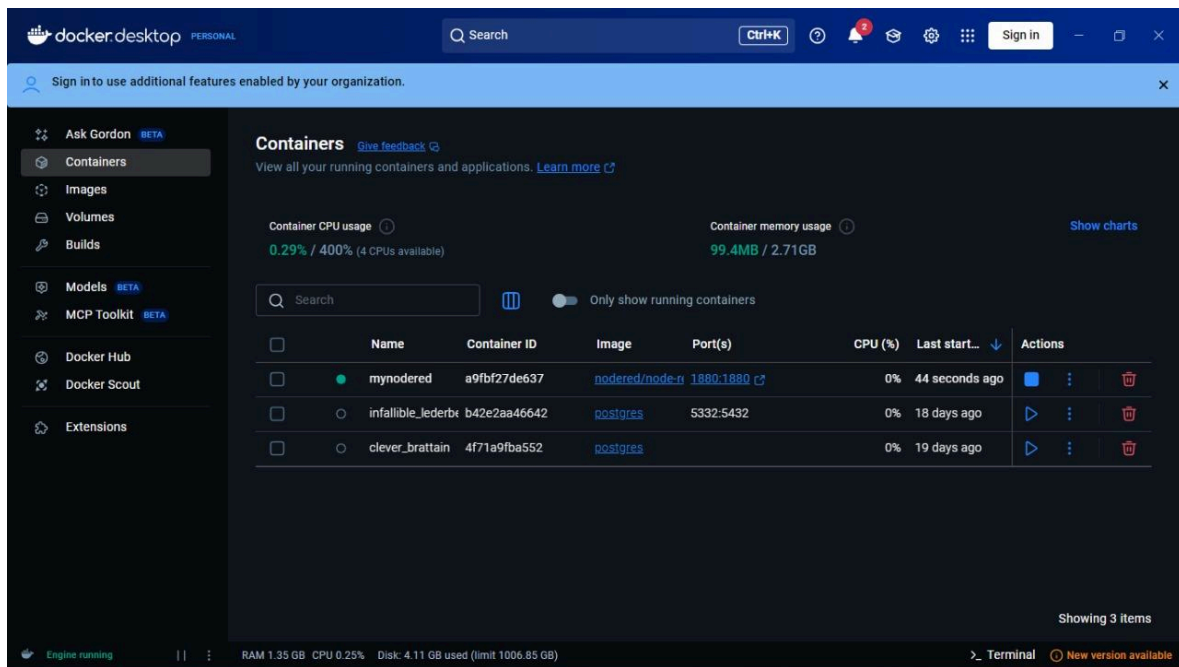
La primera etapa fue la configuración del entorno. Para ello, se creó un contenedor Docker utilizando la imagen oficial `nodered/node-red`, que permite ejecutar Node-RED de manera aislada y portable. El contenedor se inició con el nombre `mynodered`, redireccionando el puerto 1880 para acceder a la interfaz gráfica desde el navegador mediante `https://localhost:1880`. Esta configuración ofrece una solución liviana y eficiente para el desarrollo de flujos sin necesidad de instalar Node-RED directamente en el sistema operativo.

Una vez accedido al entorno de Node-RED, se diseñó un flujo que ejecuta una lectura periódica cada 30 segundos a una URL externa. Esta URL devuelve una respuesta en formato JSON con los últimos datos captados por el sensor. Para lograrlo, se utilizaron nodos como `inject`, que dispara el ciclo de lectura, seguido por `http request`, el cual realiza la petición web. El nodo `json` transforma la respuesta en un objeto interpretable por Node-RED, y un `function` personalizado permite extraer el último dato disponible, asegurando que se muestran las mediciones más recientes.

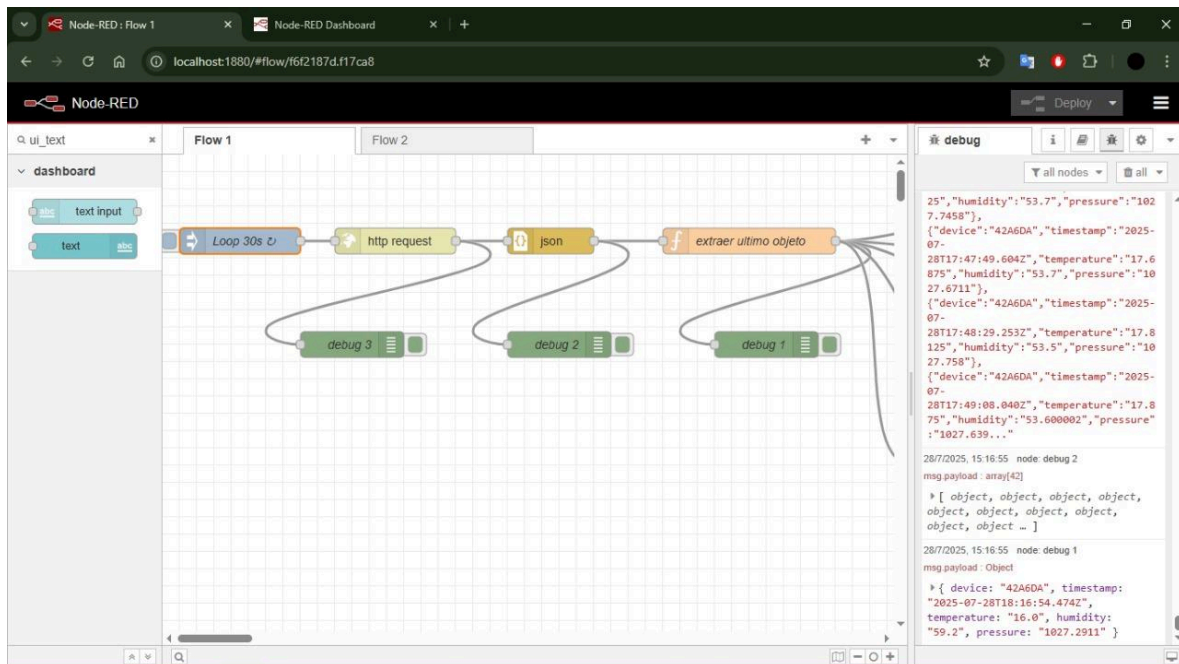
Los datos que se reciben típicamente incluyen el identificador del dispositivo, un `timestamp` con la fecha y hora de la lectura, y los valores para temperatura, humedad y presión. A partir de este objeto, se separan los distintos elementos mediante nodos `function` específicos, cada uno diseñado para formatear y preparar una de las variables. El `timestamp`, por ejemplo, es transformado a un formato local legible, mientras que las variables físicas se convierten a valores con decimales precisos, ajustados según el rango esperado del sensor.

En paralelo, se desarrolló una interfaz de usuario mediante el dashboard de Node-RED. En ella se presentan los valores procesados utilizando widgets de tipo `gauge` para mostrar la temperatura, humedad y presión de forma visual y dinámica, además de nodos `text` para mostrar el momento de la última lectura y el identificador del sensor. La UI está organizada en secciones temáticas que agrupan los datos para facilitar la lectura: una sección para los sensores, otra para la fecha y hora, y una tercera para los dispositivos. La actualización automática cada 30 segundos permite observar en tiempo real los cambios en el entorno medido.

CAPTURAS DEL DESARROLLO:



En esta captura vemos al contenedor de Node-Red en docker con su respectivo ID e imagen.

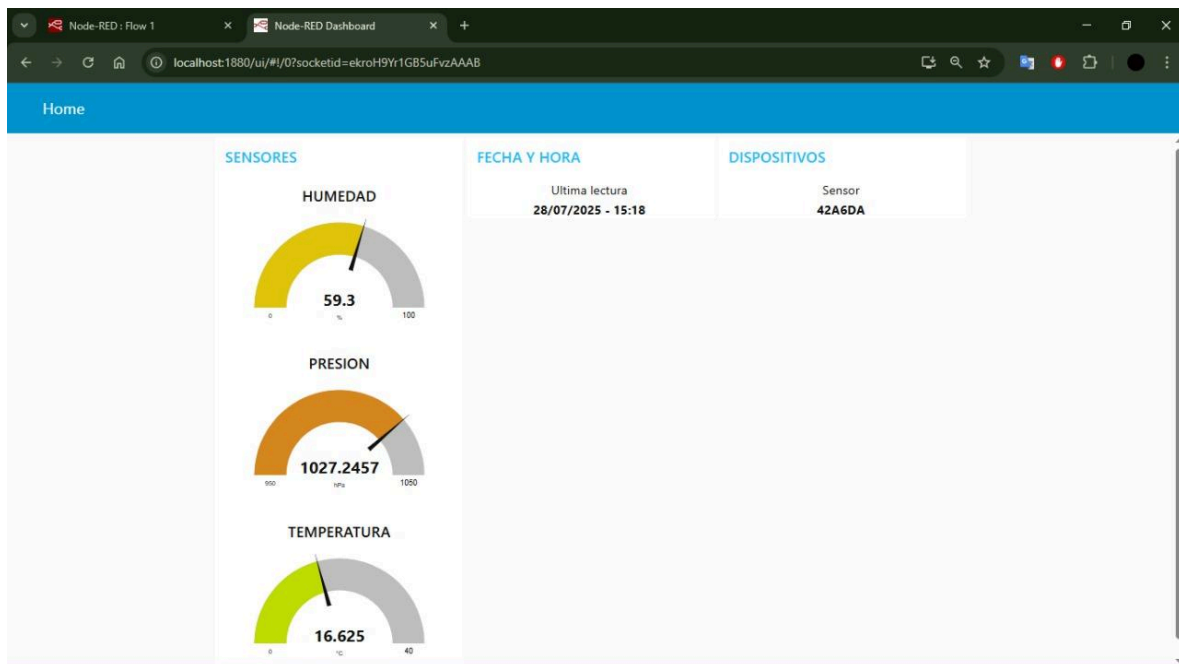


Acá se puede notar la primera parte del flow de Node-Red, en donde se consumen los datos, se crea el json y se accede a la medición encontrada en la última posición del arreglo.

CONCLUSIÓN:

Este proyecto nos permitió abarcar nuestros conocimientos sobre la virtualización, desarrollo y visualización en tiempo real dentro del sistema de IoT. Utilizando node-red dockerizado de forma aislada se facilitó el desarrollo y la portabilidad del entorno sin afectar al sistema operativo principal (windows).

El trabajo integró herramientas como docker y node-red. y nos permitió desarrollar una solución visual. Este tipo de práctica nos resultó muy útil para entender cómo se comunican los sistemas distribuidos, como se consumen APIs externas y como se presentan datos a tiempo real. Habilidades que, estamos seguros, serán fundamentales para el desarrollo de software actual y futuro.



En la imagen podemos ver como todo el flujo de Node-Red desemboca en la visualización de medición de la humedad, presion y temperatura del sensor, asi como la fecha y el numero de serie