

E.T.S. de Ingeniería Industrial,
Informática y de Telecomunicación

Monitorización de la instalación fotovoltaica del campus universitario de Pamplona: Arrosadia.



Grado en Ingeniería Eléctrica y Electrónica

Trabajo Fin de Grado

Autor: Garatea Alcázar, Fermín

Tutores: Berrueta Irigoyen, Alberto

Catalán Ros, Leyre

Pamplona, 07 de Septiembre de 2023

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

RESUMEN

Este trabajo final de grado se centra en la exploración y la aplicación de las tecnologías actuales para realizar la monitorización de un sistema fotovoltaico en la Universidad Pública de Navarra.

Más concretamente, este TFG pondrá un primer foco de atención en la monitorización de la instalación fotovoltaica de la UPNA, utilizando tecnologías innovadoras como Node-RED, InfluxDB y Grafana. El objetivo principal de este TFG es proporcionar una herramienta de monitorización que permita una visión detallada del comportamiento de los sistemas fotovoltaicos.

Por otro lado, también se estudiará la posibilidad de que alguien sin conocimientos previos sobre las tecnologías empleadas pueda llevar a cabo el desarrollo e implementación de la monitorización propuesta mediante la herramienta ChatGPT.

Por último, se explorará la viabilidad de exportar los datos de la monitorización para su uso en prácticas y actividades académicas en distintas carreras de la UPNA.

Los resultados de este trabajo han permitido ahondar en el potencial de las herramientas actuales a la hora de realizar monitorizaciones. Y han permitido dar un primer enfoque al desarrollo de dicha monitorización con los sistemas informáticos, lo que puede facilitar el diseño de aplicaciones o sistemas de gestión de futuros proyectos universitarios.

Palabras clave: Monitorización, sistema fotovoltaico, Node-RED, InfluxDB, Grafana, Tecnologías renovables, ChatGPT, educación energética.

ABSTRACT

This Bachelor's Thesis focuses on the exploration and application of current technologies to carry out the monitoring of a photovoltaic system at the Public University of Navarre.

More specifically, this bachelor's project will initially concentrate on monitoring the photovoltaic installation at UPNA, employing innovative technologies like Node-RED, InfluxDB, and Grafana. The primary goal of this project is to provide a monitoring tool that enables a detailed view of the behavior of photovoltaic systems.

On the other hand, the possibility of someone without prior knowledge of the employed technologies being able to carry out the development and implementation of the proposed monitoring using the ChatGPT tool will also be studied.

Finally, the feasibility of exporting the monitoring data for use in practical and academic activities across various disciplines at UPNA will be explored.

The results of this work have allowed for a deeper understanding of the potential of current technologies in conducting monitoring and have provided an initial approach to the development of such monitoring with current computer systems, which could facilitate the design of applications or management systems for future university projects.

Keywords: Monitoring, photovoltaic system, Node-RED, InfluxDB, Grafana, Renewable technologies, ChatGPT, energy education.

ÍNDICE

Resumen	2
Abstract.....	3
Índice	4
1. Introducción.....	6
2. Objetivos.....	8
3. Fundamentos teóricos.....	10
3.1. Capas del modelo TCP/IP.....	10
3.2. Protocolo MODBUS	11
3.3. Modbus – Serie.....	11
3.4. Modbus – TCP.....	12
4. Entorno de trabajo	13
4.1. Metodología	13
4.1.1. Elementos para la realización del proyecto	14
4.2. Montaje del servidor	15
4.3. Propuestas del entorno de desarrollo	15
4.4. Objetivos fundamentales	19
5. Configuración del entorno de trabajo	20
5.1. Creación de token en InfluxDB	20
5.2. Configuración de Node-RED	22
5.2.1. Procedimiento para volcar los datos del inversor a InfluxDB	22
5.2.2. Procedimiento para volcar los datos del inversor a un fichero CSV	32
5.3. Configuración de InfluxDB	38
5.4. Configuración de Grafana.....	43
5.5. Configuración de Apache2	56
5.6. Configuración de Python y Crontab	58
6. Montaje del entorno web y publicación de datos.....	59
7. Comparativa de ChatGPT.....	63
8. Comparativa practicas UPNA.....	67
9. Conclusiones.....	69
10. Líneas Futuras.....	71
11. Bibliografía y referencias	73
12. Anexos	76

12.1. Configuración VMware Workstation Pro 17	77
12.2. Instalación Ubuntu	83
12.3. Ubuntu server.....	92
12.4. Instalar ssh.....	98
12.5. Utilización de Putty a través de SSH	100
12.6. Network tools	103
12.7. Instalación Node-RED	104
12.8. Configuración Node-RED	107
12.9. Instalación InfluxDB	111
12.10. Error de InfluxDB	115
12.11. Instalar Grafana	116
12.12. Error en la instalación de Grafana	119
12.13. Instalación de Apache2	120
12.14. Instalación de Python	122
12.15. Datos del inversor Ingeteam.....	123
12.16. Conversación con ChatGPT.....	130

1. INTRODUCCIÓN

Actualmente la Universidad Pública de Navarra está demostrando un firme compromiso con la promoción y divulgación de las energías renovables, con el fin de tener un futuro energético más limpio. Esto se puede evidenciar a través de las siguientes iniciativas:

- El proyecto “Energía sostenible. Sin malos humos” [1] constó de una obra de divulgación sin fines lucrativos, en el cual se pretende arrojar un punto de vista imparcial sobre la energía sostenible, explicando el potencial que pueden alcanzar las energías renovables. Se hace una reflexión sobre el tipo de consumo energético del país, explorando a su vez la reducción del consumo energético individual. En la última sección se desarrollan los cálculos sobre los datos de las dos cuestiones anteriormente mencionadas. Dicha obra se aborda dando un enfoque imparcial a fin de que cada lector pueda sacar sus propias conclusiones.
- A su vez se promovió en 2022 un proyecto llamado “Navarra sin malos humos: Plan urgente de descarbonización 2027” [2], en el cual se planteó el reto de diseñar un plan de descarbonización para tener una sociedad menos dependiente de los combustibles fósiles, estableciendo un límite temporal en 2027.
- También en colaboración con la Mancomunidad de Pamplona, se realizó en las instalaciones de la universidad una línea de transporte urbano comarcal 100% eléctrico [3], dicha línea garantizaba una energía totalmente renovable, a fin de evitar las emisiones de dióxido de carbono y poder realizar dichos 360.000 km anuales de forma sostenible. Cumpliendo con los objetivos de la directiva 2012/27/UE relativa a la eficiencia energética y el Plan Energético de Navarra Horizonte 2030.

Con el objetivo de realizar este último proyecto, se realizó una instalación fotovoltaica en el campus de la Arrosadía, donde se encontraba una microrred compuesta de un campo fotovoltaico ubicado en el tejado del Aulario, que puede observarse en la Ilustración 1, y una sección de pruebas en la parte superior del edificio de los Pinos.



Ilustración 1 - Instalación fotovoltaica del Aulario.

La instalación del Aulario estaba destinada al estudio de la carga de una villavesa eléctrica apoyada por un sistema fotovoltaico y una batería de litio. Por ello, se dispone de unas baterías que se descargan en los momentos en los que la villavesa se conecta al punto de carga. Dicha instalación tiene la estructura mostrada en la Ilustración 2.

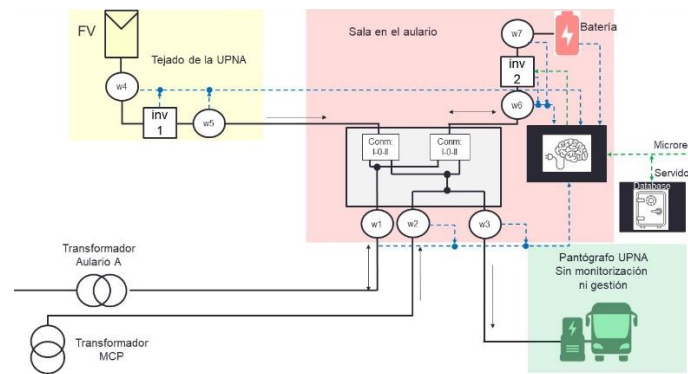


Ilustración 2 - Instalación microrred UPNA.

Dicha instalación fotovoltaica tiene una potencia estimada de 134,64 kWp. Los módulos están conectados a inversores con una potencia nominal de 100 kWn. Además, existe un sistema de cableado que conecta la instalación fotovoltaica a la red interna de consumo de energía eléctrica del edificio.

El sistema también cuenta con un equipo de acumulación de energía eléctrica, formado por dos baterías de segunda vida con una capacidad total de 84 kWh. Estas baterías están conectadas a un inversor con una capacidad de 100 kWn, lo que permite almacenar la energía producida.

A su vez, se ha puesto en marcha el proyecto “UPNA Green: smart & sustainable campus”, que instalará distintos bloques fotovoltaicos a lo largo de los distintos edificios que componen el campus universitario, como se aprecia en la Ilustración 3. Entre los diversos edificios, en los cuales se instalarán dichos paneles solares destacan especialmente los edificios departamentales, véase: Tejos con una capacidad de 164 KWp, Pinos con 158,4 KWp, Olivos con 266 KWp, Acebos con 98,78 KWp, Madroños con 110 KWp, Magnolios con 110 KWp y Encinas con 110 KWp.

Ilustración 3 - Proyecto de instalación de generación fotovoltaica.

2. OBJETIVOS

En el contexto actual de creciente conciencia ambiental y búsqueda de fuentes de energía sostenibles, la instalación fotovoltaica representa una solución prometedora para la generación de energía eléctrica limpia y renovable. A medida que el mundo ha enfrentado desafíos energéticos y ambientales cada vez más apremiantes, la adopción de tecnologías de energía solar fotovoltaica ha ido ganando tracción. La creciente demanda de energía, la preocupación por la reducción de las emisiones de gases de efecto invernadero y la disminución de las reservas de combustibles fósiles han impulsado la transición hacia fuentes de energía más sostenibles.

En este contexto, la Universidad Pública de Navarra (UPNA) ha demostrado un firme compromiso con la adopción de energías renovables como parte de su visión de promover la sostenibilidad y la innovación. La universidad ha sido testigo de la creciente influencia de las políticas internacionales y de las tendencias energéticas globales que priorizan la reducción de la huella de carbono y el fomento de la generación de energía a partir de fuentes limpias.

A medida que la generación de energía eléctrica mediante sistemas fotovoltaicos ha madurado en términos de eficiencia y rentabilidad, la UPNA ha reconocido la oportunidad de integrar esta tecnología en sus instalaciones como un paso estratégico. Con el objetivo de reducir su impacto ambiental y fomentar la investigación y la educación en energías renovables la universidad ha optado por implementar diversos grupos de fotovoltaica distribuidos en distintos edificios de su campus. Esta decisión no solo refleja su compromiso con la innovación tecnológica y la responsabilidad ambiental, sino que también sirve como un ejemplo inspirador para la comunidad académica y la sociedad en general.

El presente Trabajo de Fin de Grado (TFG) se centra en el desarrollo de la monitorización de la instalación fotovoltaica de la UPNA, utilizando tecnologías innovadoras como Node-RED, InfluxDB y Grafana. Esta propuesta permitirá recopilar y analizar datos en tiempo real de los sistemas fotovoltaicos, lo que brindará a la universidad un control más eficiente y una mejor comprensión del rendimiento de su infraestructura energética.

El objetivo principal de este TFG es proporcionar una herramienta de monitorización que permita a los responsables de la UPNA tener una visión detallada del comportamiento de sus sistemas fotovoltaicos. Se pretende ofrecer información en tiempo real sobre el rendimiento de cada grupo de instalación fotovoltaica, detectar posibles fallos o degradaciones en su funcionamiento.

Adicionalmente, este proyecto tiene un objetivo secundario relevante, que consiste en evaluar la posibilidad de que alguien sin conocimientos previos sobre las tecnologías empleadas pueda llevar a cabo el desarrollo e implementación de la monitorización propuesta. Con este fin, se realizará una comparación entre el uso de ChatGPT (un modelo de lenguaje basado en inteligencia artificial) y la metodología tradicional en la resolución de problemas técnicos asociados a este tipo de proyectos.

Por último, como una característica adicional del TFG, se explorará la viabilidad de exportar los datos de la monitorización para su uso en prácticas y actividades académicas en distintas carreras de la UPNA. De esta manera, no solo se maximizará el beneficio del sistema de monitorización para la gestión interna de la universidad, sino que también se potenciará su utilidad como herramienta educativa para fomentar el conocimiento y la comprensión de las energías renovables entre los estudiantes.

Por ello y a fin de cumplir con los objetivos mencionados hasta el momento, se ha decidido estructurar la memoria de la siguiente forma:

En el Capítulo 3, “Fundamentos teóricos”, se realizará una pequeña introducción teórica a modo de funcionamiento de las comunicaciones Modbus, así como su variante TCP, a fin de entender cómo se comunica la planta fotovoltaica con el servidor.

En el Capítulo 4, “Entorno de trabajo”, se adentrará la metodología utilizada a lo largo del proyecto, ofreciendo también una visión panorámica del sistema a implantar. Definiendo los objetivos a seguir y explorando las distintas opciones que se presentaban a la hora de realizar el proyecto, detallando el porqué de las herramientas seleccionadas.

La “Configuración del entorno de trabajo”, se explorará en el Capítulo 5, donde se detallan las configuraciones de las herramientas a utilizar, teniendo como objetivo la explicación de los puntos fundamentales a fin de llevar los datos del inversor hasta el entorno web.

En el Capítulo 6, “Montaje del entorno web y publicación de datos”, se hará una explicación de la estructura de la página web y como se ha desarrollado en conjunto con ChatGPT, haciendo énfasis en los puntos fundamentales de exportación de variables y gráficos del inversor al entorno web.

En el Capítulo 7 “Comparativa de ChatGPT”, se expondrá la utilización de la herramienta ChatGPT en la ejecución de este proyecto, así como su posible desempeño en un escenario en el que no se contará con conocimientos previos. El propósito es evaluar si fuera viable llevar a cabo la implementación de esta instalación partiendo desde cero.

En el Capítulo 8 “Comparativa practicas UPNA”, se hará una sugerencia con el fin de ampliar las prácticas de energías renovables de la universidad, para utilizar la monitorización realizada en este proyecto.

Finalmente, en el Capítulo 9 “Conclusiones”, se presentan las deducciones obtenidas a lo largo de este estudio, resumiendo los logros, las lecciones aprendidas y las posibles líneas futuras para la mejora y expansión del sistema.

3. FUNDAMENTOS TEÓRICOS

3.1. Capas del modelo TCP/IP

El Modelo TCP/IP (Transmission Control Protocol/Internet Protocol) es un modelo de referencia para la arquitectura de redes que se utiliza ampliamente en Internet y en la mayoría de las redes de ordenadores. El modelo TCP/IP se compone de cuatro capas principales, tal y como se muestra en la Ilustración 4.

1. Capa de Aplicación: Esta capa es la interfaz con las aplicaciones del usuario final. Incluye protocolos como HTTP, SMT, FTP, etc.
2. Capa de Transporte: La capa de transporte se encarga de la entrega de datos de manera confiable y ordenada. Incluye los protocolos TCP y UDP (User Datagram Protocol). TCP es orientado a la conexión y garantiza la entrega confiable de datos, mientras que UDP es no orientado a la conexión y se utiliza cuando la velocidad y la eficiencia son más importantes que la confiabilidad.
3. Capa de Internet: En esta capa, los datos se dividen en paquetes y se enrutan a través de la red utilizando direcciones IP. Aquí es donde se encuentra el Protocolo IP (IPv4 o IPv6), que permite el direccionamiento y el enrutamiento de los paquetes.
4. Capa de Acceso a la Red: Esta es la capa más baja del modelo TCP/IP y está relacionada con la forma en que los datos se transmiten físicamente a través de la red. Incluye protocolos como Ethernet, Wi-Fi, PPP, entre otros.

El Modelo TCP/IP se basa en la arquitectura cliente-servidor, donde los clientes (como navegadores web o aplicaciones) envían solicitudes a servidores, y estos últimos responden con los datos solicitados. Es un modelo flexible y escalable que ha demostrado ser efectivo para la comunicación en redes de gran tamaño como Internet.

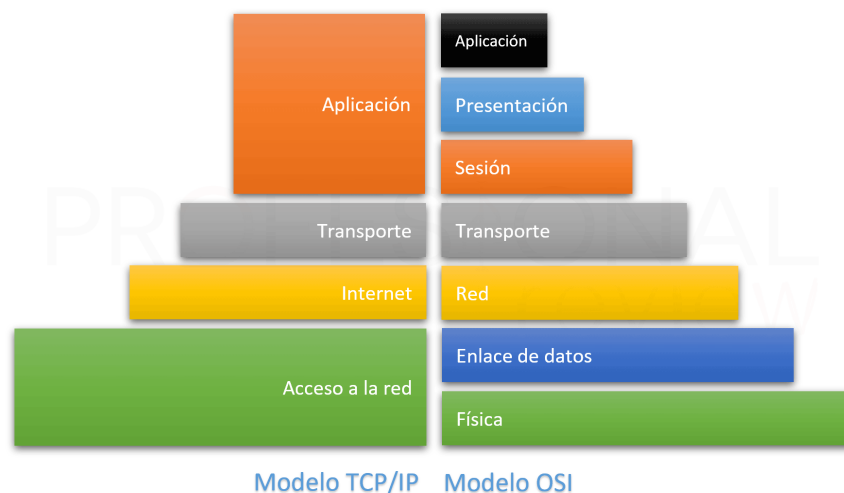


Ilustración 4 - Capas de modelo TCP y modelo OSI.

3.2. Protocolo MODBUS

Modbus es un protocolo de comunicación utilizado en sistemas de automatización industrial que permite la transferencia de datos entre dispositivos electrónicos. Fue desarrollado por Modicon (actualmente Schneider Electric) en 1979 y ha sido ampliamente adoptado en la industria debido a su simplicidad y facilidad de implementación.

El protocolo Modbus es un protocolo de capa de aplicación, lo que significa que su principal objetivo es establecer reglas y normas que permitan organizar e interpretar datos. Esto lo convierte en un sistema que facilita el envío y recepción de mensajes entre dispositivos, independientemente de la capa física utilizada para la comunicación.

La naturaleza de capa de aplicación del protocolo Modbus le otorga una gran versatilidad, lo que significa que los mensajes pueden ser transferidos a través de diversas redes mediante técnicas de encapsulación y desencapsulación.

3.3. Modbus – Serie

Modbus serie RTU (Remote Terminal Unit) es una de las variantes del protocolo Modbus que se utiliza para la comunicación serie entre dispositivos en sistemas industriales, esta variablemente es ampliamente utilizada debido a su simplicidad y eficiencia, especialmente en donde la comunicación serial es común.

Es un tipo de arquitectura Maestro/esclavo, donde un dispositivo maestro (como un PLC o PC) inicia la comunicación enviando solicitudes a uno o varios dispositivos esclavos. Cabe destacar que la máxima longitud recomendada para este tipo de comunicaciones son 1200 metros.

En la Ilustración 5, se puede observar un ejemplo de trama entre el Maestro y un esclavo, que mantienen una comunicación Modbus.

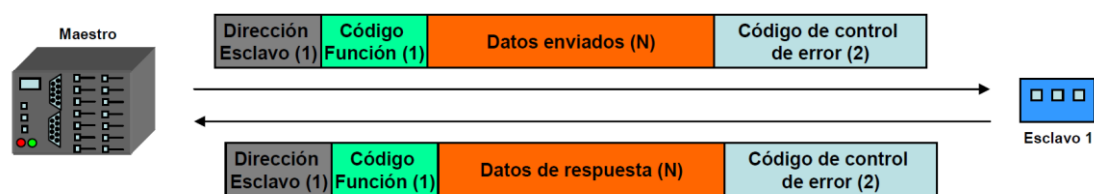


Ilustración 5 – Trama de comunicación Modbus serie RTU.

Dicha trama se divide en cuatro partes, las cuales contienen los siguientes datos:

1. Dirección (1 byte): Contiene con valores del 1 al 247, dejando libre el 0 para los mensajes de broadcast, y representa el Id del esclavo al que se hace referencia.
2. Código función (1 byte): Representa el tipo de mensaje enviado (Holding register, input register, Coils, etc).
3. Datos (Varios bytes): Dependiendo de la función (puede ir vacío si el tipo de función solo requiere asentimiento por el esclavo).

4. CRC (2 byte): Al final de la trama para tener una comprobación de errores. Se realiza haciendo un sumatorio en binario de los datos enviados y al resultado se le realiza un complemento A2.

3.4. Modbus – TCP

El Modbus – TCP (Transmission Control Protocol), es una variante del Modbus serie RTU la cual elimina la dirección de esclavo y el control de errores para adaptarlo al protocolo TCP/IP, tal y como se muestra en la Ilustración 6.

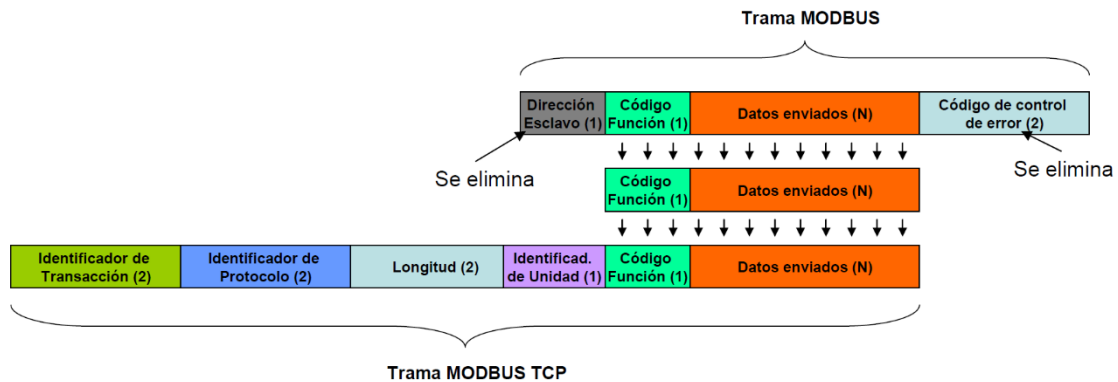


Ilustración 6 - Trama de Modbus TCP.

En la Ilustración 6, se puede observar un ejemplo de trama entre el Maestro y un esclavo, que mantienen una comunicación Modbus TCP. Dicha trama se divide en seis partes, las cuales contienen los siguientes datos:

1. Identificación de la transacción (2 bytes): Es un identificador único asignado a cada mensaje por el cliente (dispositivo que inicia la comunicación) y se utiliza para emparejar las respuestas del servidor con las solicitudes correspondientes. El servidor responderá con el mismo ID de transacción que se incluye en la solicitud.
2. Identificación de protocolo (2 bytes): Es siempre 0 en el caso de Modbus TCP, que indica el uso del protocolo Modbus TCP/IP.
3. Longitud (2 bytes): Es el número total de bytes en el paquete Modbus TCP, incluyendo el campo de unidad y el PDU (Protocol Data Unit).
4. Identificación de unidad (1 byte): Es el identificador del dispositivo esclavo que debe responder a la solicitud. En el caso de Modbus TCP, el valor típico para el campo de unidad es 0.
5. Código función (1 byte): Representa el tipo de mensaje enviado (Holding register, input register, Coils, etc).
6. Datos (Varios bytes): Dependiendo de la función (puede ir vacío si el tipo de función solo requiere asentimiento por el esclavo).

4. ENTORNO DE TRABAJO

En esta sección del proyecto se explicará el software desarrollado e implementado, siendo por tanto la estructura base del mismo.

4.1. Metodología

Todo software o sistema informático tiene lo que se denomina ciclo de vida del software, es decir, las distintas etapas en las que pasa un software o sistema van desde su concepción inicial, su desarrollo e implementación, uso diario y mantenimiento hasta su obsolescencia o sustitución. Dichas etapas están diseñadas para organizar y guiar el proceso de forma sistemática y eficiente. Suelen tener las siguientes partes:

1. Requisitos: En esta etapa se identifican y documentan las necesidades del software. Esto implica comprender las funciones que se deben cumplir, así como las restricciones técnicas y funcionales.
2. Diseño: En esta etapa se hace un diseño detallado del software basado en los requisitos establecidos. Se definen la arquitectura del software, la estructura de datos, la interfaz de usuario y otros aspectos técnicos.
3. Implementación: Se realiza la codificación real del software utilizando el diseño previamente elaborado.
4. Pruebas: Se llevan a cabo pruebas exhaustivas del software para identificar y corregir errores y asegurarse de que todo funcione según lo previsto.
5. Despliegue: Una vez que el software ha pasado las pruebas pertinentes, se lanza y se pone a disposición de los usuarios finales, puede ser de tipo gradual o completo dependiendo del tipo de sistema, la preferencia del equipo de desarrollo y la magnitud del proyecto.
6. Mantenimiento: Después del despliegue, el software necesitará un mantenimiento continuo (para corregir errores que puedan surgir en el entorno de producción) o actualizaciones (para añadir parches de seguridad o funcionalidades adicionales).
7. Retiro: Con el paso del tiempo el software puede volverse obsoleto o ser reemplazado por una versión más avanzada. Esta etapa suele concluir con una migración de datos a una nueva plataforma o su cierre completo.

Pero todo ciclo de vida no siempre es lineal ni rígido, sino que, dependiendo del tipo de software o sistema a implementar, algunas secciones del ciclo de vida serán más fundamentales que otras. Por ello y atañéndose al proyecto que se tiene entre manos, se podría decir que las etapas que se seguirán serán: Requisitos, Implementación, Pruebas y Despliegue.

Partiendo de esta base y con el fin de explicar de forma más clara y precisa este proceso, se ha hecho una fusión de dos metodologías [4]:

- Modelo de desarrollo en espiral: Es un enfoque de desarrollo de software que combina elementos del modelo en cascada con iteraciones controladas. En lugar de avanzar linealmente a través de las etapas, se realizan ciclos repetidos de

planificación, análisis de riesgos, desarrollo, evaluación e implementación. Con cada ciclo o giro, implica la identificación y mitigación de riesgos con el fin de avanzar a la siguiente fase.

- **Modelo evolutivo:** Está basado en la idea de construcción de una versión inicial y básica del software que cumpla con los requisitos esenciales y partiendo de dicha base, realizar una iteración y evolución a través de los ciclos de desarrollo adicionales, con el fin de agregar gradualmente más funcionalidades y mejoras. Con cada iteración se agrega valor al software y permite la utilización del software a etapas tempranas, lo que facilita una retroalimentación con los usuarios finales y la adaptación continua del proyecto.

El motivo de esta decisión fue que la Universidad Pública de Navarra estaba en medio de un proyecto de construcción del campo fotovoltaico para el campus. En primera instancia se tiene una pequeña instalación en el edificio del Aulario, dicho proyecto se divide en secciones: La renovación de las cubiertas de los tejados de los edificios departamentales, posteriormente la instalación de los soportes de los paneles fotovoltaicos y para finalizar la instalación de los paneles solares, en los que de forma estimada, se hará un edificio por semana. Por lo que el avance de los datos será de forma progresiva, y por ello también su implementación.

El proyecto de monitorización ha sido ejecutado de manera focalizada en la instalación fotovoltaica del edificio del Aulario, debido a que las labores de construcción en los demás edificios del campus todavía se encuentran en proceso. Una vez estas etapas se encuentren concluidas, la metodología implementada en el proyecto actual demostrará su capacidad para ser extrapolada al proyecto en su totalidad. Esto permitirá adoptar un enfoque flexible y eficaz en la supervisión de la instalación fotovoltaica.

4.1.1. Elementos para la realización del proyecto

En el proceso de monitorización y representación de la instalación fotovoltaica, hay varios elementos esenciales a fin de brindar un sistema eficiente y funcional, desde la captura de datos del inversor hasta su presentación visual en un entorno accesible e intuitivo, cada paso contribuye a la comprensión de los valores obtenidos.

Para poder realizar dicho proceso, es necesario obtener los datos generados por los inversores, que es el encargado de convertir la energía generada por los paneles fotovoltaicos (DC) a alimentación de red (AC). Dichos datos se enviarán mediante un protocolo de comunicación vía ethernet, para leerlos se deberá tener un ordenador conectado a la misma red que el inversor.

En este caso, la UPNA tiene una subred destinada únicamente para generación de energía fotovoltaica. Dicha subred está aislada del resto de la red universitaria y tiene actualmente la dirección 172.18.126.1 con máscara 255.255.254.0. Esto quiere decir que se ha creado una división lógica en la red universitaria a fin de aislar esta sección del resto de la universidad, y al aplicarle la máscara 255.255.254.0 permite tener hasta 510 equipos en la subred.

4.2. Montaje del servidor

Ya que la finalidad del proyecto es hacer público los datos de los inversores en una página web, será necesario la utilización de un servidor con el fin de hacer el montaje del sistema y la puesta en marcha del proyecto. Por ello cabe destacar que cada tipo de servidor tiene sus ventajas e inconvenientes, algunos de los sistemas más comunes son Linux, Windows Server y FreeBSD.

Entre estas, Linux es ampliamente considerado como una de las mejores opciones debido a que es de código abierto, puede ser modificado y distribuido libremente y confiere una gran estabilidad y flexibilidad al sistema debido a su diseño modular. También cuenta con una gran comunidad de desarrolladores y abundante documentación para montar y resolver cualquier conflicto con los sistemas.

A su vez, Linux es capaz de funcionar en hardware más antiguo y utiliza menos recursos en comparación con los otros sistemas operativos, por lo que se tendría un uso más eficiente de la infraestructura.

4.3. Propuestas del entorno de desarrollo

Debido a que el protocolo Modbus fue desarrollado en 1979, se ha llegado casi al decalustro de su creación, por lo que se han desarrollado infinidad de herramientas de gestión y captura de datos que abren un amplio abanico de posibilidades a favor.

Algunos de los programas más comunes para capturar datos Modbus disponibles serían:

- Modbus Poll: Es una aplicación diseñada para facilitar la comunicación con dispositivos y equipos que utilizan Modbus TCP. Dicho software permite leer y escribir registros, lo que proporciona la capacidad de supervisar y controlar dispositivos. Además, permite realizar consultas y analizar los datos capturados de forma más eficiente. Sin embargo, es un software de pago único y las opciones de exportación de datos y comunicación con otras aplicaciones se queda un poco limitada.
- LabVIEW: Es una plataforma de desarrollo utilizada en entornos de ingeniería y control. Puede integrarse con dispositivos Modbus TCP para crear aplicaciones y sistemas de adquisición de datos. Su ventaja principal es su potente plataforma de desarrollo con la capacidad de comunicarse con otras plataformas y dispositivos como Arduino. Además cuenta con una amplia comunidad de usuarios. Sin embargo, es un software de pago y requiere inversión en licencias.
- Python con bibliotecas Modbus: Utilizar Python con bibliotecas como "pymodbus" permite implementar captura y análisis de datos Modbus TCP de manera personalizada. Su ventaja más destacada es la flexibilidad total que ofrece, al ser gratuito y de código abierto. Es adecuado para desarrolladores con conocimientos de programación y desarrollo, que deseen una solución altamente personalizada.
- Node-RED: Node-RED es una plataforma de programación visual que permite crear flujos de datos y automatizaciones. Puede integrarse con dispositivos

Modbus TCP a través de módulos disponibles. Su principal ventaja radica en la flexibilidad para crear flujos personalizados, siendo gratuito y de código abierto. No obstante, configurar y personalizar flujos puede requerir tiempo y conocimientos.

En este contexto, las dos opciones más viables para la implementación del proyecto serían "Python con bibliotecas Modbus" y "Node-RED" debido a su gran flexibilidad tanto en la gestión de datos como en la comunicación con otros programas. Sin embargo, dada su facilidad de uso y rápida implementación, la elección se inclina hacia Node-RED ya que su entorno de programación visual simplifica la comprensión del código para futuros administradores, permitiendo diseñar y ajustar flujos de trabajo de manera eficiente sin la necesidad de escribir código desde cero. Esto garantiza una implementación efectiva y una gestión sencilla a lo largo del tiempo.

A su vez, Node-RED proporciona una amplia gama de nodos y bibliotecas, que facilitan la exportación de los datos fuera del entorno de trabajo, entre ellos destacan los nodos que comunican con bases de datos, en los que entre ellos se encuentran:

- Base de datos relacional (SQL): Utiliza bases de datos SQL como MySQL, PostgreSQL o SQLite para almacenar datos en tablas estructuradas. Node-RED ofrece nodos específicos para interactuar con estas bases, permitiendo ejecutar consultas y almacenar información de manera organizada.
- Base de datos NoSQL: Las bases de datos NoSQL, como MongoDB, son flexibles para almacenar datos sin una estructura rígida. Node-RED tiene nodos que ayudan a conectarse a bases de datos NoSQL y guardar información de forma más adaptable.
- Plataformas en la Nube: Aprovecha servicios en la nube como Amazon DynamoDB, Google Cloud Firestore o Microsoft Azure Cosmos DB. Se puede usar nodos de Node-RED para comunicarse con estas plataformas y almacenar los datos de manera remota.
- Historiadores industriales: Si se trabaja en un entorno industrial, se considera usar historiadores industriales como InfluxDB. Estas bases de datos están diseñadas para almacenar y analizar datos de series temporales, como los datos Modbus.

Entre las opciones mencionadas anteriormente, las dos que mejor se ajustan a este proyecto serían los "historiadores industriales" y "Base de datos relacionales". Sin embargo, al considerar la opción de las bases de datos relacionales, se puede observar que conlleva un estudio de Entidad-Relación para poder hacer una estructura de datos coherente y funcional. Además para sacarle todo el potencial a dicha herramienta haría falta un conocimiento previo del lenguaje SQL, por lo que para simplificar el proyecto se ha decidido hacer uso de los "Historiadores industriales".

Dentro de los "Historiadores industriales" se encuentra la aplicación InfluxDB, que además de contar con una interfaz sencilla que facilita la curva de aprendizaje de la herramienta, cuenta con una comunidad activa y sólida, respaldada mediante foros y

una amplia variedad de recursos como “InfluxDB University” y herramientas vía API. Además, dicha herramienta permite la gestión de datos, uso de Chronograf y un lenguaje tipo JSON.

La estructura principal utilizada por InfluxDB para almacenar datos se llama "shard". Cada shard contiene un rango de tiempo específico y almacena los datos en segmentos comprimidos para ese intervalo. Esta división en shards permite la distribución y administración eficiente de los datos en el tiempo.

Dentro de cada shard, los datos se organizan en "series". Una serie es una secuencia de puntos de datos que comparten la misma "medición" y conjunto de "etiquetas". La medición es el nombre del conjunto de datos, y las etiquetas son pares clave-valor que proporcionan metadatos para filtrar y agrupar los datos.

A su vez existen los “buckets”, son contenedores lógicos en las que se agrupan las series temporales, cada bucket puede almacenar diversas series temporales y su característica principal es que definen la política de retención temporal de los datos antes de ser eliminados.

En resumen, InfluxDB almacena los datos en una estructura basada en "shards", donde cada shard abarca un intervalo de tiempo específico y almacena datos en segmentos comprimidos. Dentro de cada shard, los datos se organizan en "series" que comparten una misma "medición" y conjunto de "etiquetas". Estas series se agrupan en "buckets", que son contenedores lógicos con políticas de retención definidas.

Para poder acceder a estos datos desde fuera de InfluxDB se pueden realizar consultas HTTP (mediante Curl, Axios o Request) o se pueden utilizar solicitudes mediante APIs, las opciones más utilizadas actualmente son:

- Grafana: Es una popular plataforma de visualización y monitoreo que se integra muy bien con InfluxDB. Permite la creación de paneles interactivos y gráficos personalizados de los datos almacenados.
- Telegraf: Es una herramienta de código abierto desarrollada por InfluxData, que funciona como consultor de datos de una forma eficiente y ordenada, tiene un tipo de arquitectura modular que se basa en “input plugin” y “output plugins”, con el fin de enviar los datos a otros sistemas.
- JSON API: Se pueden utilizar librerías JavaScript como D3.js, las cuales permiten crear gráficos directamente en la web, después de haber obtenido los datos mediante una consulta. Esta opción sería la más recomendada respecto a la seguridad de los servidores y la base de datos, ya que permite captar y limitar el tipo de consultas que se pueden realizar.
- Google charts: Es una biblioteca de visualización de datos gratuita de Google, que tiene como finalidad la generación de gráficos y visualizaciones a partir de los datos en InfluxDB.

Otras de las posibilidades que se mostraron a lo largo de la realización del proyecto, fue la API-REST realizada mediante SpringBoot por Carlos Alonso (Servicio Informático de la

Universidad Pública de Navarra), dicha API permitía acceder a los datos de InfluxDB localizada en el servidor del laboratorio y enviar una respuesta tipo JSON con los datos deseados, dicha API proporciona la misma flexibilidad que Node-RED junto a MQTT [5] a la hora de ofrecer los datos e importarlos en una página web junto con el añadido en seguridad que permite filtrar el tipo de comandos SQL, por lo que se tendría un plus de seguridad a la hora de realizar el proyecto. Por el contrario Node-RED junto a MQTT se veía en una clara desventaja debido a que el servidor web se alojaba dentro de la propia Node-RED, pero no ofrecía tanta flexibilidad de configuración y podría afectar al rendimiento o gestión de los recursos en caso de que se realizasen múltiples solicitudes o la plataforma escalase con el tiempo.

Aun habiendo visto que el uso de JSON API sería lo más recomendable para mantener la integridad de la base de datos y evitar problemas informáticos en la UPNA, debido al margen temporal en el que se trabaja y debido a que implicaría una mayor elaboración del entorno web en el que se mostrarán los datos, se ha optado por hacer uso de Grafana, ya que tiene una API interna con la que InfluxDB trabaja con una gran fluidez.

A su vez, Carlos Alonso hizo una representación de los valores obtenidos mediante "Plotly". Dicha biblioteca proporciona herramientas para crear gráficos interactivos a través de lenguajes como Python, R o JavaScript. Por lo que brinda una gran flexibilidad de personalización que conlleva un tiempo de desarrollo concreto.

Por el contrario, Grafana destaca por su gran variedad de representaciones visuales de datos en diversas formas, así como una gran flexibilidad de personalización y capacidad de consolidar diferentes fuentes en una sola interfaz. Para ello suele utilizar bases de datos SQL o los datos importados desde InfluxDB. También ofrece la posibilidad de exportar dichas representaciones visuales con formato "iframe", el cual con solo una línea de código permite la integración en un entorno web de forma sencilla y sin dedicarle una gran cantidad de tiempo.

Para la construcción del entorno web, existe una variedad de lenguajes de programación que se ajustan a los criterios del proyecto, entre los más destacados se encuentran PHP y HTML. En este caso se optará por este último, dado que este lenguaje no solo permite interactuar de manera sencilla con JavaScript, sino que también posibilita la utilización de CSS para dar una capa de personalización accesible y amigable.

En relación con el alojamiento de la página web y considerando la elección de Ubuntu como plataforma de desarrollo, se ha tomado la decisión de emplear el conocido servidor web Apache2. Este es un servidor web de código abierto ampliamente utilizado que permite alojar y servir contenido web en internet. Ofrece la función de VirtualHost, una característica que posibilita alojar múltiples sitios web en una misma máquina física, compartiendo los recursos del servidor, pero manteniendo los sitios web aislados y accesibles mediante nombres de dominio o puertos únicos. Esto permite una administración eficientemente de varios sitios en una sola instancia.

4.4. Objetivos fundamentales

Si bien la finalidad del proyecto y sus pasos a seguir están claramente definidos, las ideas básicas e inamovibles para su desarrollo son:

- Adquisición de datos de los inversores mediante protocolo Modbus TCP.
- Filtrado general de los datos del inversor teniendo en cuenta el mapa de registros de los mismos mencionados en el Anexo: Datos del inversor Ingeteam.
- Almacenamiento de la información tratada y evaluada, creando a ser posible sistemas de respaldo.
- Publicación de la información en un entorno web, con el fin de que sea intuitivo y sencillo.

5. CONFIGURACIÓN DEL ENTORNO DE TRABAJO

5.1. Creación de token en InfluxDB

Para poder comenzar con este proyecto, después de haber realizado la correcta instalación del entorno de trabajo, el primer paso a seguir sería la creación del token de InfluxDB. Una token API, a su vez conocida como token de autenticación, es una cadena de caracteres alfanuméricos, que se utiliza para verificar la identidad de un usuario o una aplicación. Es decir, actúa como una credencial digital que da acceso seguro y controlado a los servicios asociados a la API.

Para ello, se deberá abrir un navegador web y conocer la dirección IP del servidor y conocer el puerto específico para acceder a la plataforma, que en este caso será el puerto 8086. Una vez que se disponga de esta información, se abrirá el navegador y se ingresará la dirección IP junto al puerto correspondiente. Se podrá observar entonces un portal de “get started”, por lo que se clicará el botón azul, que se puede apreciar en la Ilustración 7.

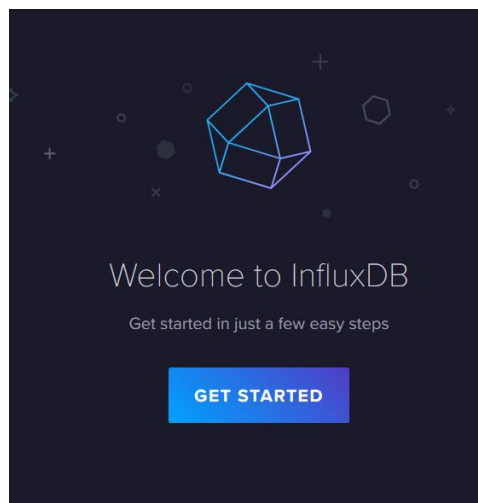
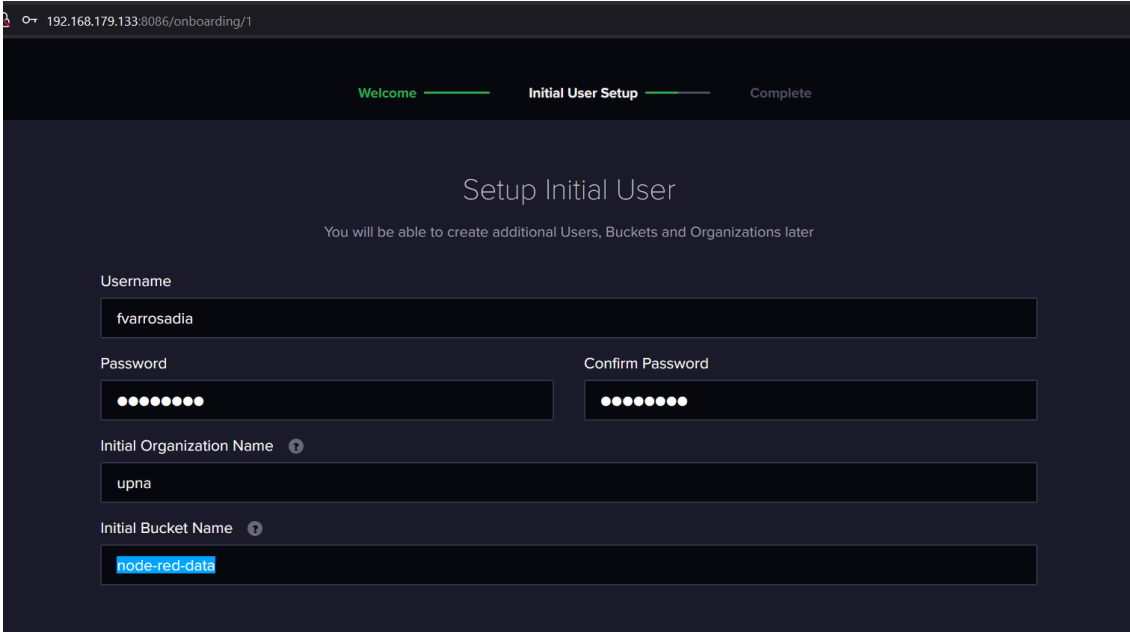


Ilustración 7 – Get started de InfluxDB.

Una vez accedido al “get started”, se observa un formulario a rellenar que tiene como objetivo la creación de un usuario, en este caso se utilizará “fvarrosadia” con la contraseña “**Qwerty1**.” A su vez, se pide el nombre de la organización, dicho nombre tiene la función de estructurar y categorizar los buckets dentro de la base de datos con el fin de facilitar su posterior tratamiento y organización. Por otro lado, solicita también un nombre de bucket, tal y como se muestra en la Ilustración 8. Este último es un contenedor de datos temporales, es decir, secuencias indexadas por tiempo donde se almacenan los datos, lo que comúnmente se denomina base de datos, pero con la particularidad de que esta no funciona a base de tablas conectadas por tokens. Se trata de una sola tabla en la que las columnas representarían las distintas variables, mientras

que las filas estarían dedicadas a unidades de tiempo. En esta configuración, se logra una conexión fluida entre los valores de las variables, dispuestos en una secuencia temporal.

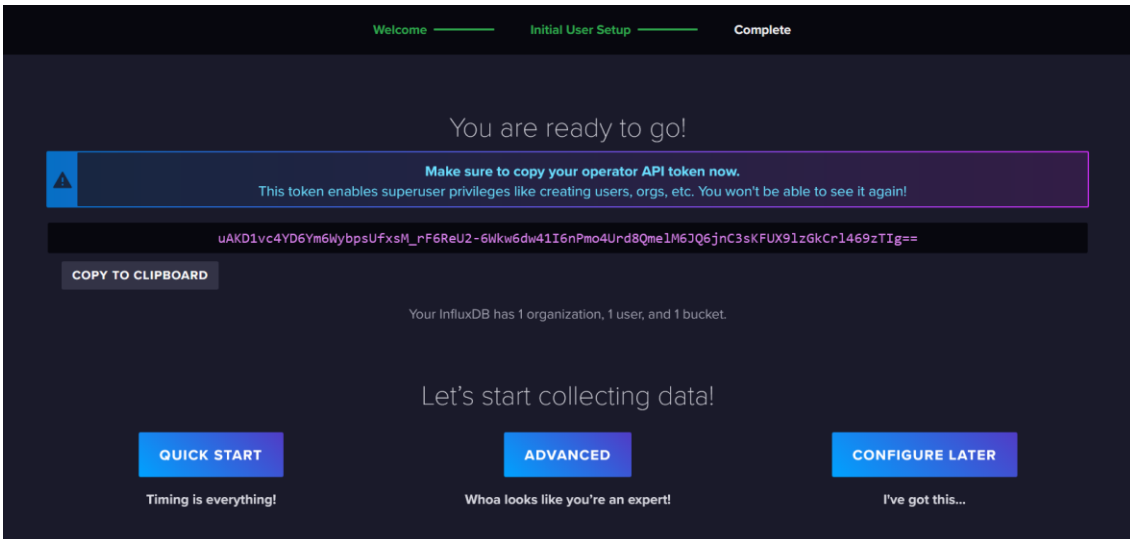


The screenshot shows the 'Setup Initial User' page in InfluxDB. At the top, there are three progress indicators: 'Welcome' (completed), 'Initial User Setup' (active), and 'Complete'. The main heading is 'Setup Initial User' with a subtext: 'You will be able to create additional Users, Buckets and Organizations later'. The form contains the following fields:

- Username:** Input field containing 'fvarrosadia'.
- Password:** Input field with masked characters (dots).
- Confirm Password:** Input field with masked characters (dots).
- Initial Organization Name:** Input field containing 'upna'.
- Initial Bucket Name:** Input field containing 'node-red-data'.

Ilustración 8 – Creación de usuario en InfluxDB.

Una vez completada la creación de usuario, se mostrará en pantalla la token API del recurso recién creado, tal y como se muestra en la Ilustración 9. Por lo que se clicará a “Quick start” para llegar al menú principal de InfluxDB y parar este proceso por el momento.



The screenshot shows the 'You are ready to go!' screen in InfluxDB. At the top, there are three progress indicators: 'Welcome' (completed), 'Initial User Setup' (active), and 'Complete'. A warning message states: 'Make sure to copy your operator API token now. This token enables superuser privileges like creating users, orgs, etc. You won't be able to see it again!'. Below the message is a text box containing the API token: 'uAKD1vc4YD6Ym6WypbsUFxsM_rF6ReU2-6Wkw6dw41I6nPmo4Urd8Qme1M6JQ6jnC3sKFUX91zGkCr1469zTIg=='. A 'COPY TO CLIPBOARD' button is located below the token. Below the token, it says 'Your InfluxDB has 1 organization, 1 user, and 1 bucket.' The main heading is 'Let's start collecting data!' with three buttons: 'QUICK START' (with subtext 'Timing is everything!'), 'ADVANCED' (with subtext 'Whoa looks like you're an expert!'), and 'CONFIGURE LATER' (with subtext 'I've got this...').

Ilustración 9 – API token de InfluxDB.

5.2. Configuración de Node-RED

5.2.1. Procedimiento para volcar los datos del inversor a InfluxDB

Para llevar a cabo la configuración de Node-RED [6], se seguirá un proceso sencillo a través de un navegador web. En primer lugar, será necesario conocer la dirección IP del servidor y conocer el puerto específico para acceder a la plataforma, que en este caso será el puerto 1880. Una vez que se disponga de esta información, se abrirá el navegador y se ingresará la dirección IP junto al puerto correspondiente.

Tras ingresar a la interfaz de Node-RED, se encontrará con una pantalla de bienvenida y se le ofrecerá la opción de seguir un pequeño tutorial introductorio. Este tutorial es muy útil para aquellos que no estén familiarizados con Node-RED, ya que brinda una visión general de la plataforma y de cómo utilizarla, tal y como se muestra en la Ilustración 10.

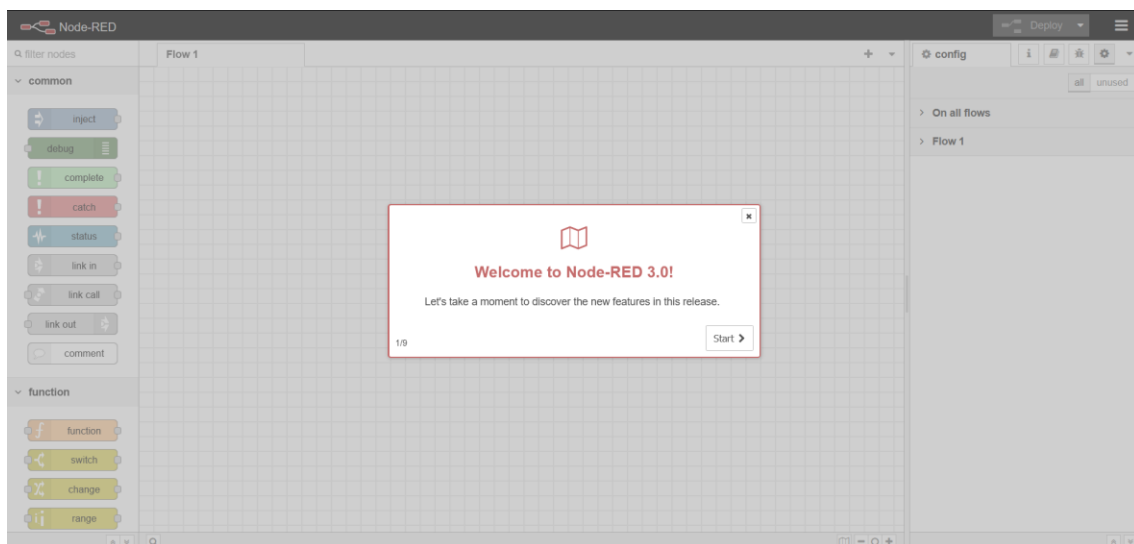


Ilustración 10 – Primer acceso a Node Red.

Una vez finalizado el tutorial, el usuario se encontrará con un lienzo en blanco denominado "Flow", tal y como se muestra en la Ilustración 11. Este lienzo es el área principal donde se desarrollarán los flujos de trabajo, es decir, la secuencia de acciones y eventos que se configurarán mediante la interconexión de nodos.

Los nodos son elementos esenciales en Node-RED, y representan una variedad de funciones, desde entradas y salidas hasta operaciones lógicas y de procesamiento de datos. Estos nodos se encuentran disponibles en la barra lateral izquierda de la interfaz, y se presentan de manera gráfica e intuitiva, lo que facilita su selección y colocación en el lienzo de trabajo.

El enfoque de programación por bloques que proporciona Node-RED resulta especialmente beneficioso para aquellos usuarios que no tienen experiencia previa en programación o desarrollo de software. Al conectar los nodos entre sí, se establecen las relaciones necesarias para crear flujos de trabajo eficientes y funcionales.

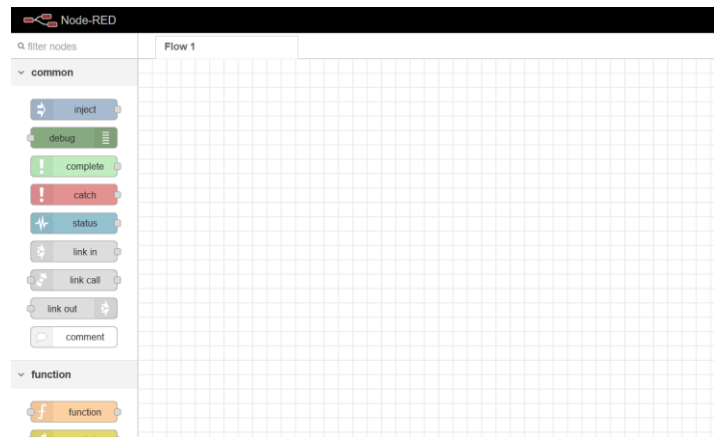


Ilustración 11 – Flow de trabajo de Node-RED.

No obstante, antes de adentrarse en la configuración y creación de flujos, es crucial seguir las indicaciones presentadas en el Anexo: Configuración . Este anexo proporciona una guía detallada sobre cómo configurar los diferentes paquetes y dependencias necesarias para el correcto funcionamiento del proyecto. Al seguir estas instrucciones, se asegura la preparación adecuada del entorno para el trabajo con Node-RED y se minimizan posibles errores o incompatibilidades en el proceso.

Una vez realizada la instalación y configuración de los paquetes requeridos, se podrá proceder a diseñar y desarrollar el proyecto.

El primer paso consistirá en añadir el nodo Modbus-read, este nodo es fundamental, ya que posibilita la lectura de datos utilizando el protocolo Modbus a través del protocolo TCP/IP. Para llevar a cabo esta tarea, se requerirá que el servidor esté conectado a la misma red o subred que el inversor fotovoltaico y que se conozca la dirección IP del mismo, así se podrán capturar los datos. Para agregar este nodo, simplemente se seleccionará de la barra lateral de nodos y se arrastrará al Flow, con el fin de conseguir lo mostrado en la Ilustración 12.

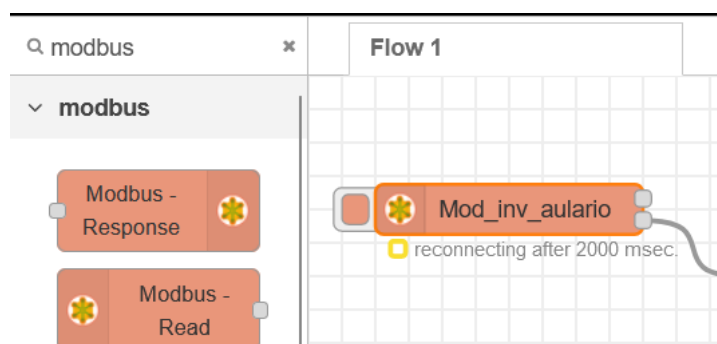


Ilustración 12 – Inserción de Modbus – Read al Flow.

Una vez que el nodo Modbus-read ha sido ubicado en el flujo, se procederá a configurarlo. Para ello, se hará doble clic sobre el nodo y se desplegará una ventana emergente que nos permitirá establecer los parámetros necesarios, tal y como se puede observar en la Ilustración 13.

Ilustración 13 – Configuración del nodo Modbus en Node-RED.

En dicha ventana, se encontrarán los siguientes elementos:

- **Nombre**: Aquí se podrá asignar un nombre descriptivo al nodo.
- **Topic**: Establece una etiqueta al nodo. Debido a que el Flow será sencillo y no tendrá muchos elementos, se puede obviar el hecho de poner etiquetas.
- **Unit-id**: Especifica el identificador de unidad Modbus con el que se comunicará el nodo para realizar la lectura de datos. En este caso como es la primera etiqueta, se pondrá un “1”.
- **FC (Function Code)**: Determina el tipo de función Modbus que se utilizará para acceder a los datos. En este caso, debido a que se están leyendo los registros del inversor se tendría que seleccionar la función “04 – Read input registers”, se pueden observar más códigos de registros en la Tabla 1 – Códigos de funciones tipo Modbus.

Tabla 1 – Códigos de funciones tipo Modbus.

Code	1/16-bit	Description	I/O Range
01	1-bit	Read coils	00001 - 10000
02	1-bit	Read contacts	10001 - 20000
05	1-bit	Write a single coil	00001 - 10000
15	1-bit	Write multiple coils	00001 - 10000
03	16-bit	Read holding registers	40001 - 50000
04	16-bit	Read input registers	30001 - 40000
06	16-bit	Write single register	40001 - 50000
16	16-bit	Write multiple registers	40001 - 50000
22	16-bit	Mask write register	40001 - 50000
23	16-bit	Read/write multiple registers	40001 - 50000
24	16-bit	Read FIFO queue	40001 - 50000

- **Address:** Indica la dirección de registro Modbus desde donde se obtendrán los datos que se desean leer. En este caso se escribirá "0", ya que con ello se comenzará a obtener los datos desde el primer registro del inversor, y posteriormente se hará una filtración de los datos necesarios.
- **Quantity:** Determina la cantidad de registros que se leerán a partir de la dirección especificada. Se escribirá al valor máximo de registros que proporciona el inversor, por lo que con ello se obtendrán todos los datos.
- **Poll rate:** Frecuencia con la que el nodo realizará la lectura de datos. En este caso se han seleccionado 100 ms, con el fin de tener los datos más actualizados posibles, pero dando un margen a completar los cálculos posteriores que se harán con los datos obtenidos.
- **Server:** Corresponde a la configuración del servidor Modbus que se utilizará para establecer la comunicación.

Si se selecciona el botón del lápiz situado a la derecha de Server en la configuración de Modbus-read, aparecerá una ventana emergente en la cual se podrá especificar los parámetros necesarios del inversor a configurar y se podrán observar los siguientes parámetros, tal y como se muestra en la Ilustración 14.

Edit Modbus-Read node > **Edit modbus-client node**

Delete Cancel Update

Properties

Settings Queues Optionals

Name Inv_Bajocubierta

Type TCP

Host 172.18.126.7

Port 502

TCP Type DEFAULT

Unit-Id 1

Timeout (ms) 1000

Reconnect on timeout

Reconnect timeout (ms) 2000

Enabled **1 node uses this config** On all flows

Ilustración 14 – Configuración del Inversor en Modbus-read en Node-RED.

- **Nombre:** Nombre descriptivo del inversor.
- **Tipo:** Aquí se especificará el tipo de protocolo Modbus, en este caso será modo TCP.
- **Host:** Dirección IP del inversor fotovoltaico. La dirección IP del inversor fotovoltaico del aulario será la 172.18.126.7.
- **Puerto:** Número de puerto que se utilizará para la comunicación. Es importante asegurarse de que el puerto utilizado sea el adecuado para la comunicación específica del inversor, en este caso es el puerto 502.
- **TCP Type:** En caso de utilizar el protocolo Modbus TCP, se seleccionará el tipo. En este caso será el “DEFAULT”.
- **Unit-id:** Identificador de unidad Modbus, el cual permitirá identificar el dispositivo dentro de la red y acceder a sus datos de manera adecuada.
- **Timeout(ms):** Indica el tiempo máximo de espera para recibir una respuesta del inversor. Se dejará por defecto, debido a que es un buen tiempo de timeout.
- **Reconnect timeout:** Tiempo que se esperará antes de intentar una nueva reconexión. Se dejará por defecto, debido a que es un buen tiempo de reconexión.

El segundo paso consistirá en añadir el nodo Buffer-parser, al igual que en la Ilustración 15. Este nodo juega un papel importante, ya que permite analizar y procesar datos binarios recibidos, convirtiéndolos en un formato legible y manejable. A su vez, permite dividir el mensaje en variables para su mejor manejo.

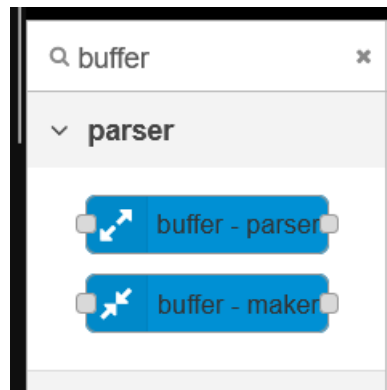


Ilustración 15 – Nodo buffer-parser en Node-RED.

Al agregar el nodo Buffer-parser al "flow", al hacer doble clic sobre el nodo, se abrirá una ventana emergente que permitirá establecer los parámetros de configuración, como en la Ilustración 16.

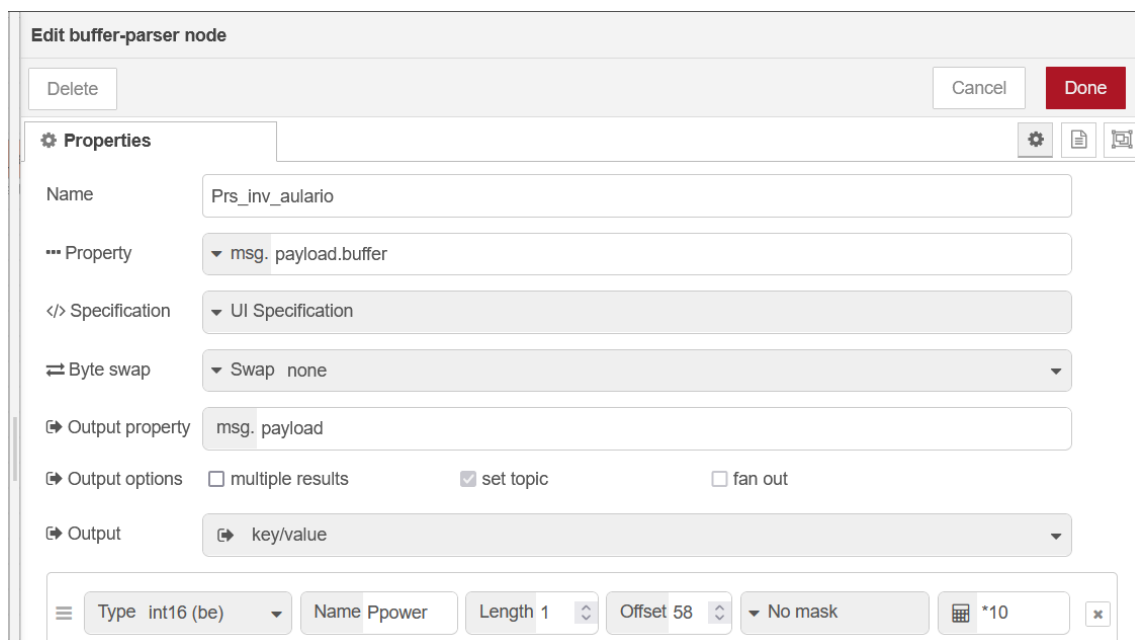


Ilustración 16 – Configuración del nodo Buffer-parser del inversor en Node-RED.

Nombre: Nombre descriptivo al nodo Buffer-parser.

Property: Aquí se definirá la propiedad específica que se desee extraer o analizar del dato binario recibido. En este caso el tipo de mensaje será de “payload.buffer”.

Output.property: Este parámetro determinará el nombre de la propiedad que se creará en la salida del nodo Buffer-parser, en este caso es tipo “payload”

Output: En esta opción se establecerá el formato de salida de los datos procesados. En este caso será de tipo “key/value”.

Posteriormente, se presentará una tabla que permitirá desglosar el mensaje tipo Modbus en variables para su posterior tratamiento. Para realizar esta tarea de manera efectiva, será necesario consultar el Anexo: el cual proporcionará la información necesaria sobre la ubicación de los datos de interés.

Por ejemplo, si se desea obtener el valor de voltaje de salida del inversor, será primordial conocer el modelo de inversor con el que se está trabajando, en este caso, el modelo sería 3PLAY 100 TL. Una vez identificado el modelo, se buscará en la tabla de registros del inversor para seleccionar los valores que se deseen representar.

Es esencial tener en cuenta que cada valor ocupará dos bytes en el mensaje, por lo que, si se desea obtener el registro nº33, será necesario escribir el offset 66. Además, en la tabla de registros, se indicará el tipo de valor, que en este caso será Unit16. Además, será necesario ajustar el valor para su correcta representación, aunque en este caso no hay que aplicar ningún factor de corrección.

Los valores seleccionados para su representación serán los mostrados en la Tabla 2 – Tabla de registros a mostrar del inversor.

Tabla 2 – Tabla de registros a mostrar del inversor.

Tipo	Nombre	Longitud	Offset	Mascara	Adecuación
int16 (be)	Ppower	1	58	No mask	*10
int16 (be)	Qpower	1	60	No mask	*10
int16 (be)	Spower	1	56	No mask	*10
int16 (be)	InVolt	1	66	No mask	0
int16 (be)	InCorr	1	64	No mask	/100
int16 (be)	CosPhi	1	62	No mask	/1000

Para comprender la información presentada en la Tabla 2, es esencial familiarizarse con los siguientes campos:

- **Ppower**: Este campo se refiere a la potencia activa generada por el inversor y se mide en vatios (W).
- **Qpower**: Se trata de la potencia reactiva con la que trabaja el inversor y se mide en voltamperios reactivos (VAR).
- **Spower**: Se refiere a la potencia aparente producida por el inversor. La potencia aparente es la combinación de la potencia activa y la potencia reactiva, y se mide en voltamperios (VA).
- **InVolt**: Este campo representa el voltaje en el que está operando actualmente el inversor y se mide en voltios (V). El voltaje es la diferencia de potencial eléctrico entre dos puntos.
- **InCorr**: Se refiere a la corriente con la que está operando en ese momento el inversor y se mide en amperios (A). La corriente es el flujo de carga eléctrica a través de un conductor.
- **CosPhi**: Este campo corresponde al factor de potencia del inversor. El factor de potencia es la relación entre la potencia activa y la potencia aparente en un circuito eléctrico. No tiene unidades, ya que es una relación adimensional.

El siguiente nodo a incrustar en el Flow, sería el “Influxdb out”, como se aprecia en la Ilustración 17, dicho nodo es el encargado de enviar los datos desde Node-RED a la aplicación de InfluxDB.

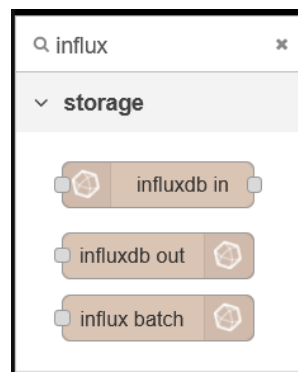


Ilustración 17 – Nodo Influxdb out.

Por lo que una vez arrastrado al Flow, se le clicará dos veces y mostrará una ventana emergente de configuración, dicha configuración será similar a la mostrada en la Ilustración 18.

Edit influxdb out node

Delete Cancel Done

Properties

Name inv_aulario

Servidor [v2.0] Database.fv

Organización upna

Bucket node-red-data

Medición flow

Precisión Milliseconds (ms)

Ilustración 18 – Configuración del nodo Influxdb out.

En dichos campos es importante reflejar la configuración previa hecha en InfluxDB para su correcto funcionamiento, concretamente en la Ilustración 8, teniendo en cuenta que el tipo de “Medición” será “Flow”, ya que se refiere a la recolección de datos en series temporales. A su vez, se tendrá que configurar el “Servidor”, por lo que se clicará en el lápiz situado a la derecha. Por último, se le dará un nombre al bucket contenedor de datos, dicho bucket se creará posteriormente en Configuración de InfluxDB. La configuración deberá quedar similar a la mostrada en la Ilustración 19.

Edit influxdb out node > Edit influxdb node

Delete Cancel Update

Properties

Name Database.fv

Versión 2.0

URL http://localhost:8086

Token ●●●●●●●●

Verificar el certificado del servidor

Ilustración 19 – Configuración del servidor en Influxdb out.

En esta sección es bastante importante seleccionar la versión instalada de InfluxDB, debido a que, según la versión escogida, la asociación con la aplicación se realiza con distintos procesos. En este caso, se descargó la versión 2.7 de InfluxDB, tal y como se muestra en el Anexo: Ilustración 154, por lo que los parámetros a configurar quedarían:

- **Name:** Este campo permite una mejor identificación de la conexión con el servidor, por lo que sería aconsejable poner un nombre identificativo. En este caso “Database.fv”.
- **Versión:** Como ya se ha mencionado previamente, la versión de InfluxDB sería la 2.7, por lo que se seleccionará “2.0” (ya que incluye las posteriores).
- **URL:** En este campo es necesario poner la dirección IP del servidor que aloja la aplicación, junto al puerto de la misma. En este caso al estar en la misma máquina, se puede poner “http://localhost:8086”.
- **Token:** Este campo es de crucial ya que es el encargado de iniciar sesión en la aplicación para su correcto funcionamiento. Dicha token se ha obtenido anteriormente en la Ilustración 9, por lo que se tendría que copiar y pegar dicha cadena de caracteres.
- Por último, la opción de “verificación de servidor” serviría para comprobar que el servidor al que se está conectado es el legítimo y no una copia, por lo que no está de más marcar dicha opción.

En resumen y hasta el momento, si se han realizado los pasos anteriormente mencionados, se habrá establecido la conexión Modbus entre la aplicación Node-RED y el inversor. Además, se habrá implementado el filtrado de los datos del inversor mediante el buffer parser, y dicha información se habrá volcado a la aplicación InfluxDB. Por lo que se debería tener un Flow, tal que quede igual que en la Ilustración 20.

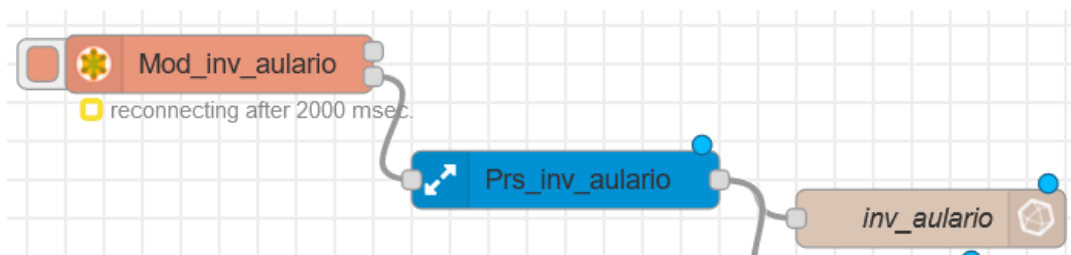


Ilustración 20 – Conexión de los nodos Inversor a InfluxDB.

5.2.2. Procedimiento para volcar los datos del inversor a un fichero CSV

En esta segunda parte de la programación de Node-RED, se aprovechará el código realizado en la sección anterior “Procedimiento para volcar los datos del inversor a InfluxDB”, para ello se realizará una bifurcación posterior al Buffer-parser con el fin de tener los mismos datos y variables en InfluxDB como en el fichero CSV.

Para ello, el protocolo Modbus da como resultado un array con los datos filtrados. Pero para una mejor gestión y manipulación de los mismos se transformará dicho array a un objeto de JavaScript, ya que es un tipo de dato muy versátil y sencillo de usar, por eso se buscará el nodo “json” y se arrastrará al Flow, como se observa en la Ilustración 21.



Ilustración 21 – Búsqueda del nodo json en InfluxDB.

Para su configuración, simplemente se tiene que hacer doble clic sobre el nodo y se abrirá una ventana emergente, similar a la mostrada en la Ilustración 22.

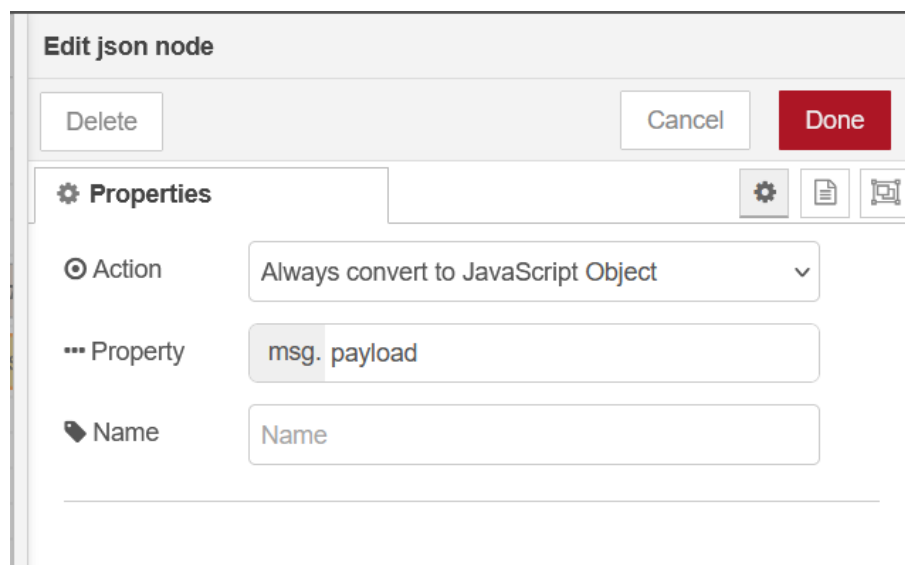


Ilustración 22 – Configuración del nodo json en Node-RED.

En esta ventana se tiene que marcar la acción que se quiera realizar, que sería “Always convert to JavaScript Object” y el tipo de propiedad será de “msg.payload”, debido a que cuando los nodos interactúan entre sí, es común que el resultado de una operación o la información relevante se almacene en este tipo de formato. Este atributo puede contener cualquier tipo de dato como números, cadenas de texto, objetos JSON, matrices, booleanos, etc. Su versatilidad radica en los flujos de trabajo.

Una vez transformado el array a tipo javascript, con el fin de registrarlos en el fichero CSV, es necesario hacer un proceso de reestructuración del mensaje, para ello se debe insertar el nodo “csv” en el Flow, dicha búsqueda se realizará al igual que en la Ilustración 23.



Ilustración 23 – Agregar el nodo CSV a Node-RED.

Una vez insertado el nodo en el Flow, se procede a su configuración haciendo doble clic sobre el mismo. Entonces se mostrará una ventana emergente como se puede apreciar en la Ilustración 24.

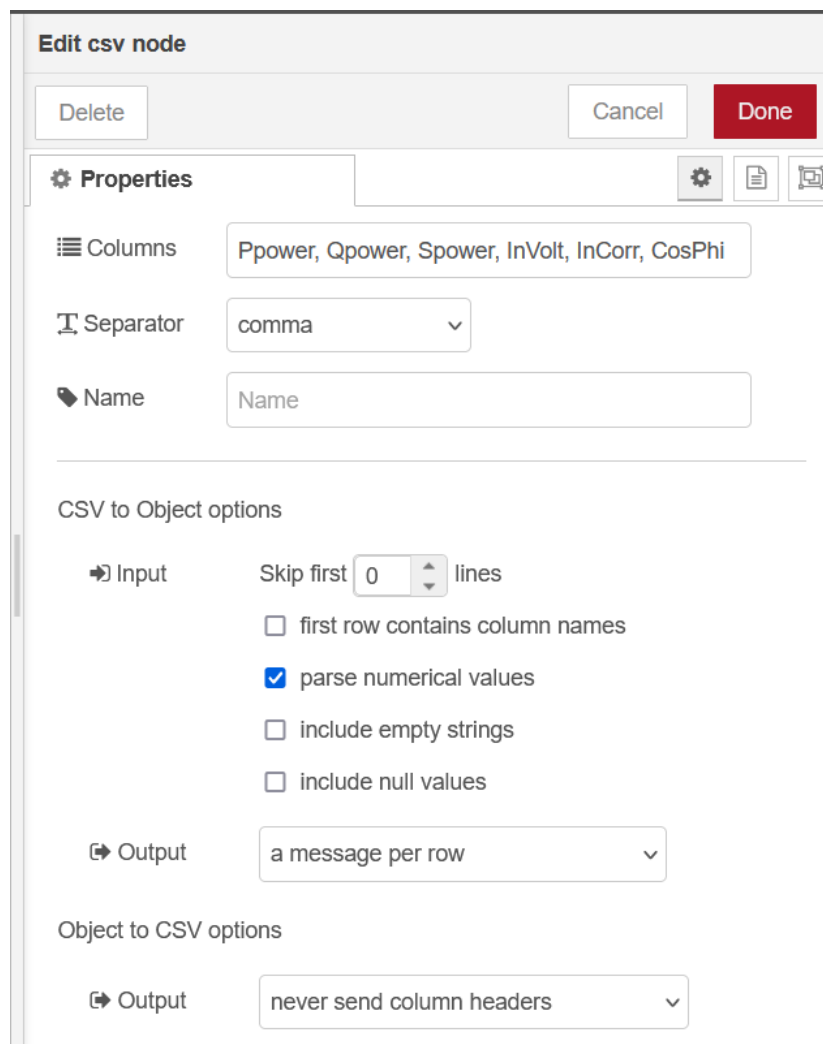


Ilustración 24 – Configuración del nodo CSV en Node-RED.

- Columns: En este apartado se deben meter las variables obtenidas en el Modbus separadas con coma, con el fin de saber que variables se quieren volcar en el fichero.
- Separator: Indica como se quieren separar las variables. En este caso se ha decidido hacer una separación mediante comas, debido a que es una manera muy sencilla de separar los valores posteriormente con código, y a su vez en caso de querer tratar los datos manualmente con un Excel, al hacer el volcado de datos es una de las separaciones más comunes en este tipo de ficheros.
- Input: Este apartado da distintas opciones de gestión de los valores, en este caso se ha decidido dejarlos por defecto debido a que ninguna de las opciones que se ofrecen beneficia a la gestión de los datos.
- Output: Se selecciona el tipo de división que hay entre mensajes, siguiendo los ejemplos de PVGIS, se ha decidido hacer un mensaje por fila, con el fin de que estén ordenados de forma cronológica descendente en el fichero.
- Object to csv options: Ofrece la opción de insertar el cabecero de las variables en el fichero de Excel, en este caso no se mandarán las cabeceras debido al tipo de tratamiento que se le va a dar al fichero CSV.

Con el fin de comprobar que todos los datos se han transformado y configurado correctamente hasta el momento, se añadirá un nodo de tipo “debug”, por lo que mostrará en la consola de Node-RED los resultados de las variables obtenidas y procesadas. Para ello se arrastrará el nodo “debug” al Flow, de acuerdo con lo que se refleja en la Ilustración 25.

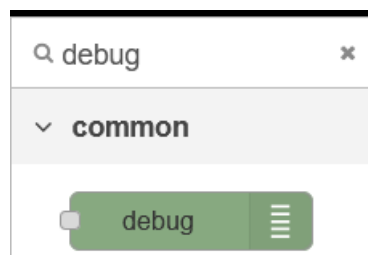


Ilustración 25 – Inserción del nodo Debug en Node-RED.

Para configurar dicho nodo simplemente se tiene que hacer doble clic y se mostrará una ventana emergente, en la cual se tendrá que poner que el tipo de output será de “msg.payload”, como se indica en la Ilustración 26.

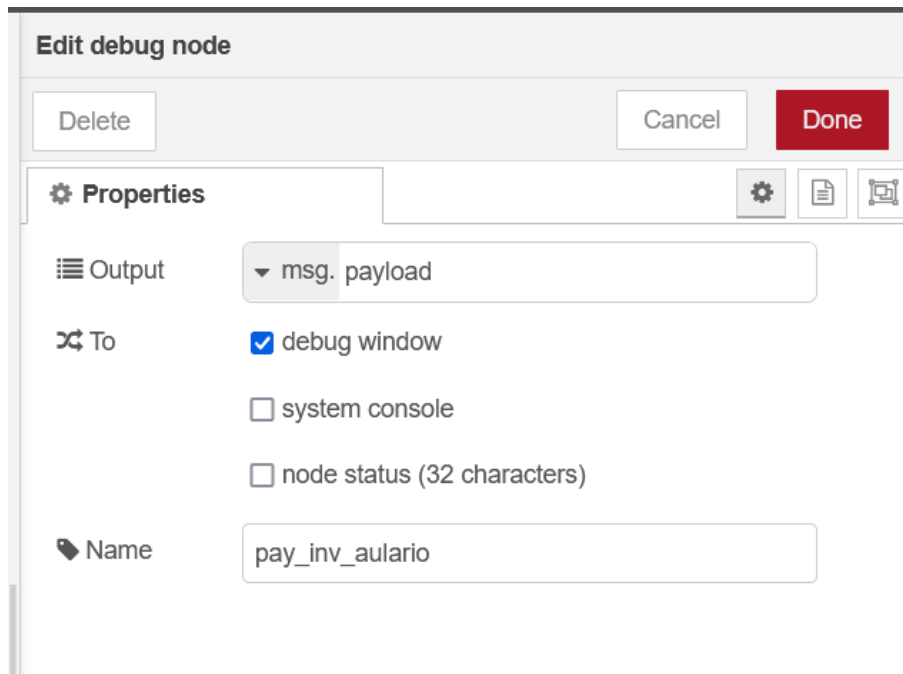


Ilustración 26 – Configuración del nodo debug en Node-RED.

Por último, se añadirá el nodo “write file” que es el encargado de coger todas las variables y datos procesados hasta el momento y escribirlos en el fichero CSV. Para ello, como se ha realizado anteriormente, se buscará el nodo pertinente y se arrastrará al Flow, como se refleja en la Ilustración 27.



Ilustración 27 – Inserción del nodo write file en Node-RED.

Para configurarlo, se tendrá que hacer doble clic sobre el nodo y se mostrará una ventana emergente, que se rellenará al igual que en la Ilustración 28.

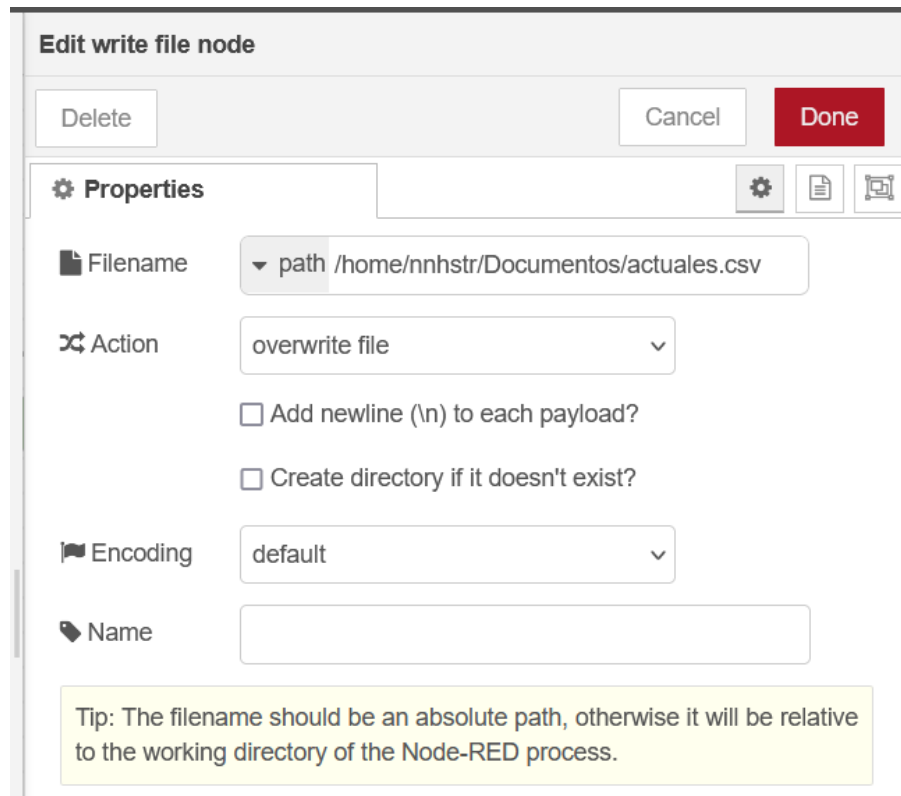


Ilustración 28 – Configuración del nodo write file en Node-RED.

Se pueden observar los siguientes campos:

- **Filename:** En esta selección se pondrá el tipo “path”, lo que quiere decir que se tendrá que introducir la localización del fichero en su totalidad, comenzando con la raíz, en este caso “/”. Dicha raíz es la predeterminada en Ubuntu. En este caso, al estar el fichero CSV localizado en la carpeta “Documentos” y con el nombre “actuales.csv”, la ruta completa sería:

`/home/%username%/Documentos/actuales.csv`

- **Action:** Es la acción a realizar por parte de Node-RED para grabar los datos en el fichero CSV. Debido a que lo que se quiere son los datos momentáneos, y no los datos históricos, se seleccionará la opción de “overwrite file”, que se encargará de sobrescribir sobre la misma fila los datos obtenidos de la comunicación Modbus. A su vez, en esta sección se puede elegir agregar un “\n” al final de cada inserción, lo que viene a ser introducir un salto de línea, comúnmente denominado como “intro”, en caso de querer hacer un almacenamiento de un histórico de datos es una opción bastante recomendable.

En resumen y hasta el momento, si se han realizado los pasos anteriormente mencionados, se habrá establecido la conexión Modbus entre la aplicación Node-RED y el inversor. A su vez se han transformado esos datos en tipo JavaScript y se han volcado en un fichero CSV. Por lo que se debería tener un Flow similar al mostrado en la Ilustración 29.

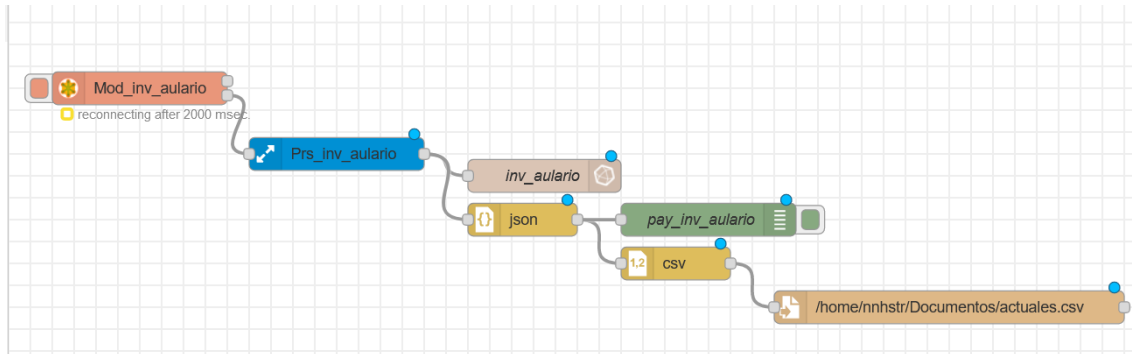


Ilustración 29 – Flow final de Node-RED.

Para poner en marcha la programación realizada hasta el momento, se deberá ir a la parte superior derecha de la aplicación de Node-RED y clicar la opción de “Deploy”. Con ello compilará y se pondrá en funcionamiento, dicho botón se puede apreciar en la Ilustración 30.

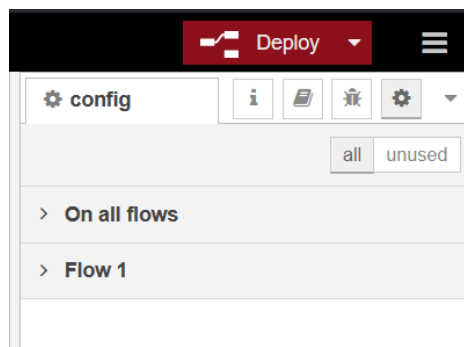


Ilustración 30 – Deploy de Node-RED.

5.3. Configuración de InfluxDB

Para llevar a cabo la configuración de InfluxDB, se seguirá un proceso sencillo a través de un navegador web. En primer lugar, será necesario conocer la dirección IP del servidor y conocer el puerto específico para acceder a la plataforma, que en este caso será el puerto 8086. Una vez que se disponga de esta información, se abrirá el navegador y se ingresará la dirección IP junto al puerto correspondiente.

Tras ingresar a la interfaz de InfluxDB, se encontrará con una pantalla de inicio de sesión, la cual se puede observar en la Ilustración 31, y se tendrá que introducir el usuario y la contraseña configurado en el apartado Ilustración 9.

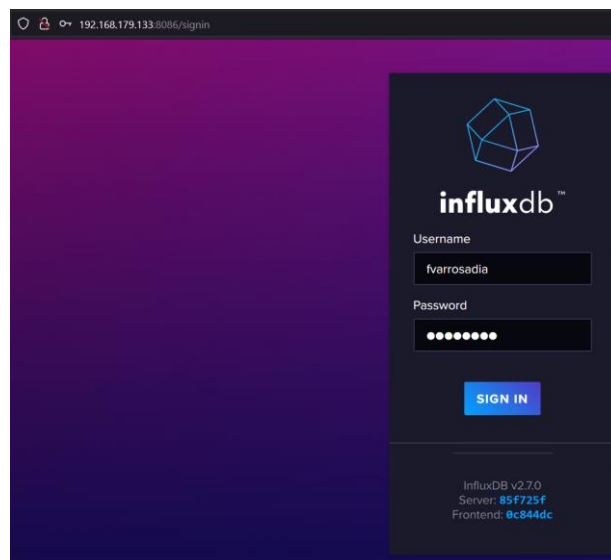


Ilustración 31 – Página de inicio de InfluxDB.

Una vez iniciada sesión, se podrá observar un panel de mando. El cual ofrece en la parte izquierda un menú desplegable, en la parte central una introducción a los plugins más comunes con InfluxDB y a la derecha una serie de enlaces de introducción a la plataforma al igual que en la Ilustración 32.

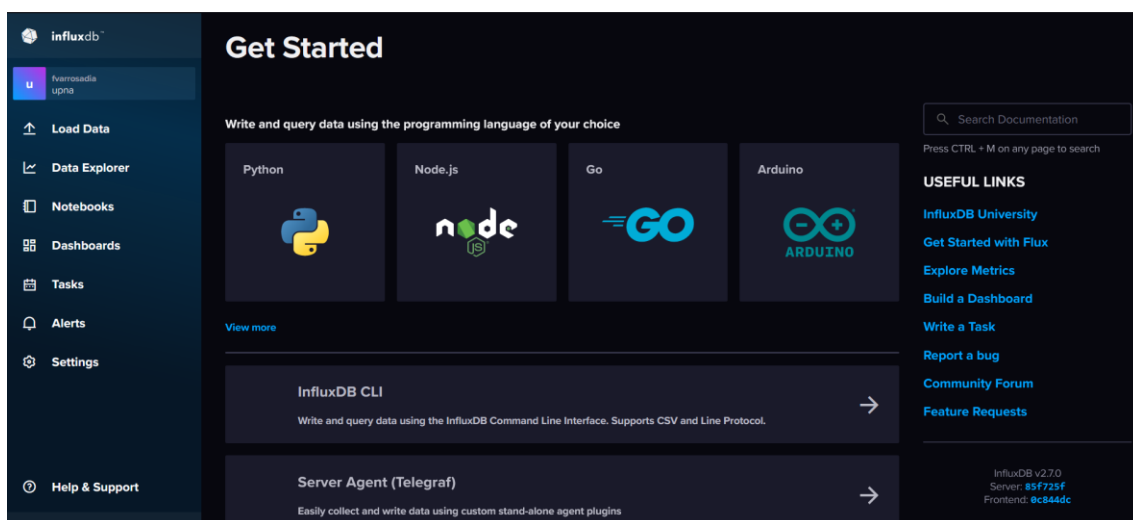


Ilustración 32 – Panel de mando de InfluxDB.

Para poder asociar Node-RED con InfluxDB, en primera instancia se deberá ir al menú desplegable de la izquierda del panel de mandos, y posteriormente seleccionar la opción de “Data Explorer”. Al realizar esta acción se mostrará una interfaz de consulta de la base de datos, para poder utilizarla más cómodamente es aconsejable ir a la parte superior derecha de la web y seleccionar “Switch to Old Data Explorer”, debido a que mostrará de una forma más visual y sencilla los apartados técnicos que se van a utilizar, y se mostrara al igual que en la Ilustración 33.

Una vez cambiada la interfaz, se irá a la parte inferior y se seleccionará el botón de “Create Bucket”, el cual permitirá crear el contenedor de datos importados desde Node-RED, tal y como se configuró en la Ilustración 18.

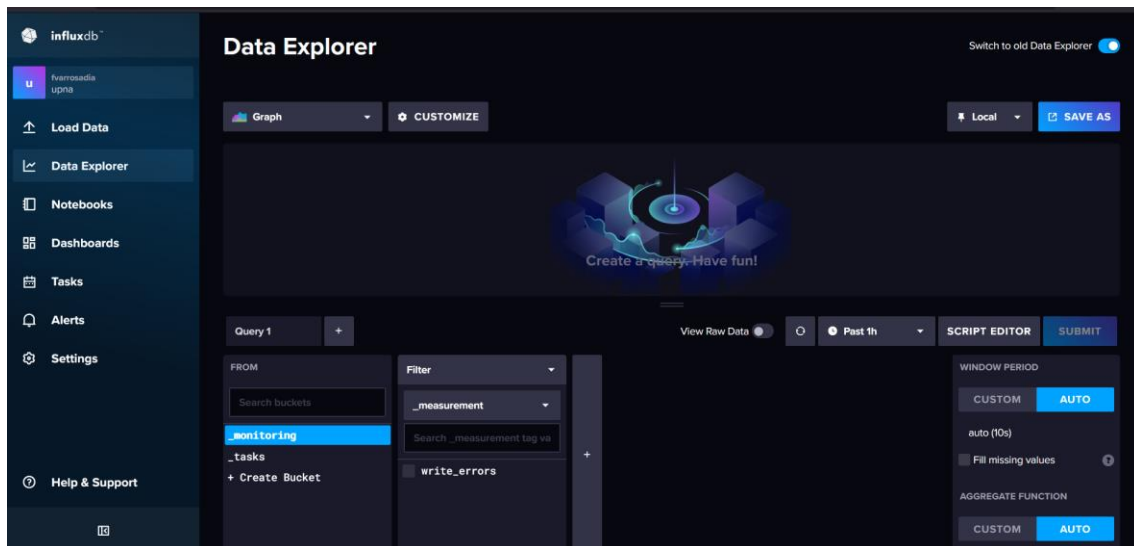


Ilustración 33 – Selección de dataexplorer en InfluxDB.

Es de vital importancia que cuando aparezca el desplegable de creación de contenedores, se introduzca el mismo nombre que se configuró en Node-RED, debido a que si no se presentarían problemas de sincronización entre ambas plataformas. También se ofrece la posibilidad de marcar un límite al registro temporal, en este caso y debido al interés de almacenar todos los datos posibles, se seleccionará la opción de nunca “never”, al igual que en la Ilustración 34.

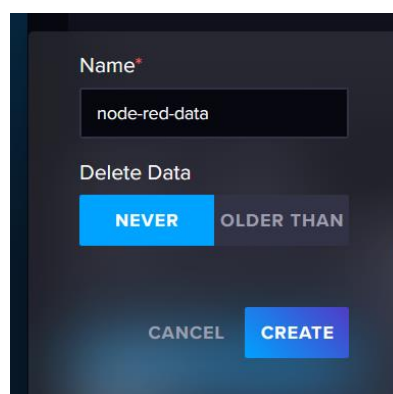


Ilustración 34 – Creación de un dataexplorer en InfluxDB.

Una vez creado el bucket de datos, la aparición de las variables anteriormente configuradas será instantánea. Por lo que para hacer una consulta se deberá seleccionar el bucket de datos “Node-RED-data”, el tipo de filtro “flow”, con el fin de que quede similar a la Ilustración 35 y posteriormente las variables con las que se quiera trabajar.

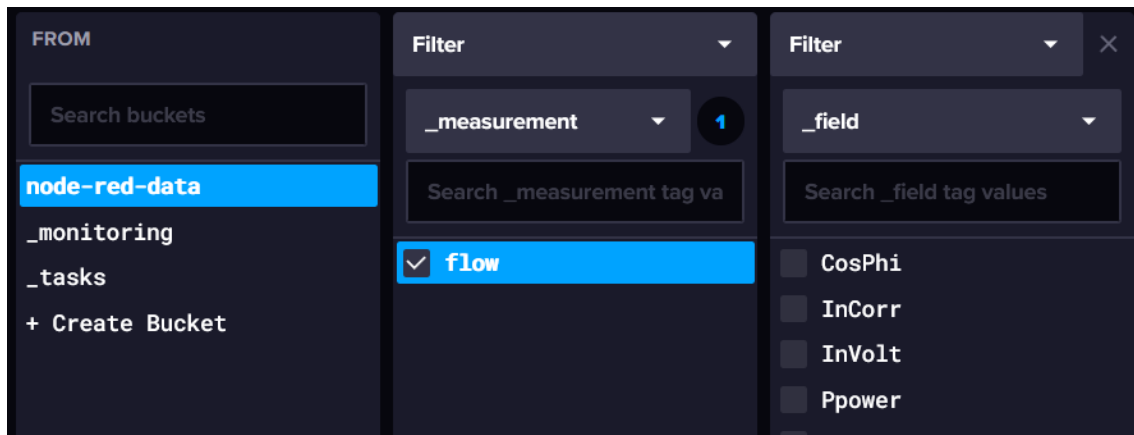


Ilustración 35 – Selección del Bucket de datos en InfluxDB.

Con el fin de realizar un proyecto sencillo y visual, se ha optado por hacer una representación de las potencias activa, reactiva y aparente. Por lo que se seleccionará “Ppower”, “Qpower” y “Spower”, como se puede observar en la Ilustración 36.

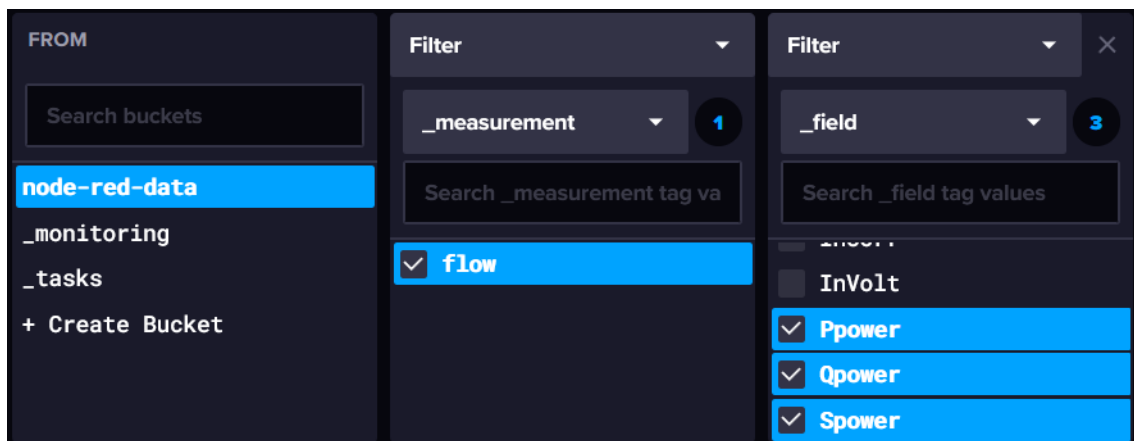


Ilustración 36 – Selección de las variables a representar en InfluxDB.

Una vez elegidas las variables a visualizar, estas se representarán en el gráfico central en forma de una línea temporal. Dada la naturaleza de estos datos, dicho gráfico se mantendrá en formato "graph". No obstante, es importante mencionar que InfluxDB brinda la flexibilidad de modificar el tipo de gráfico desde la esquina superior izquierda, como se puede observar en la Ilustración 37.

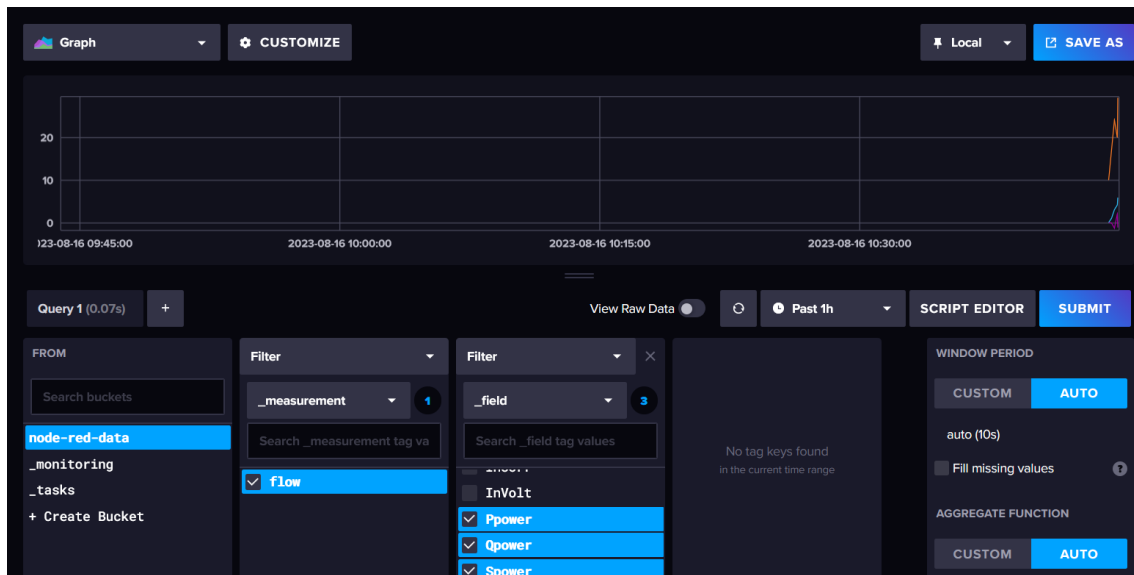


Ilustración 37 – Panel de visualización de las variables seleccionadas en InfluxDB.

Una vez confirmadas las que variables a representar sean las correctas y se visualizan sin problema alguno, se deberá clicar en el botón de “Script editor”, ubicado en la sección derecha a mitad de altura de la pantalla, como se puede apreciar en la Ilustración 38.

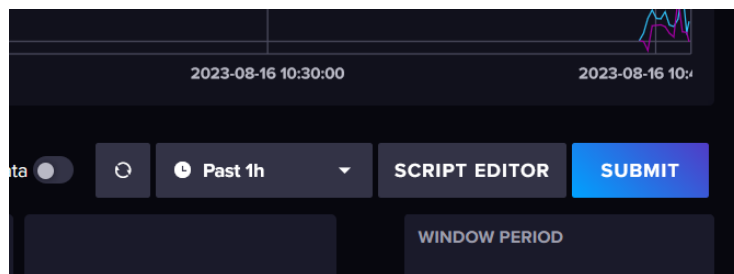


Ilustración 38 – Selección de Scripteditor en InfluxDB.

Esta acción provocará un cambio visual en el panel de administración, el cual hará desaparecer la selección de variables y mostrará un código Flux, por lo que se seleccionará dicho código y se copiará a un bloc de notas, dicho código tendrá la misma estructura que la Ilustración 39.

```

Query 1 (0.41s) + View Raw Data Past
1 from(bucket: "node-red-data")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r["_measurement"] == "flow")
4   |> filter(fn: (r) => r["_field"] == "Ppower" or r["_field"] == "Qpower" or r["_field"] == "Spower")
5   |> aggregateWindow(every: v.windowPeriod, fn: mean, createEmpty: false)
6   |> yield(name: "mean")

```

Ilustración 39 – Script de selección de variables de InfluxDB.

Por último, en caso de que se quiera guardar la selección de variables, se deberá clicar el botón de “Save as” situado en la parte superior derecha de la Ilustración 37. Dicha acción provocará una ventana emergente, que permitirá seleccionar el panel de Dashboards donde se quiere guardar el gráfico, como no se ha creado ninguno previamente, se optará por la opción de “Create a New Dashboard”, como se ve en la Ilustración 40.

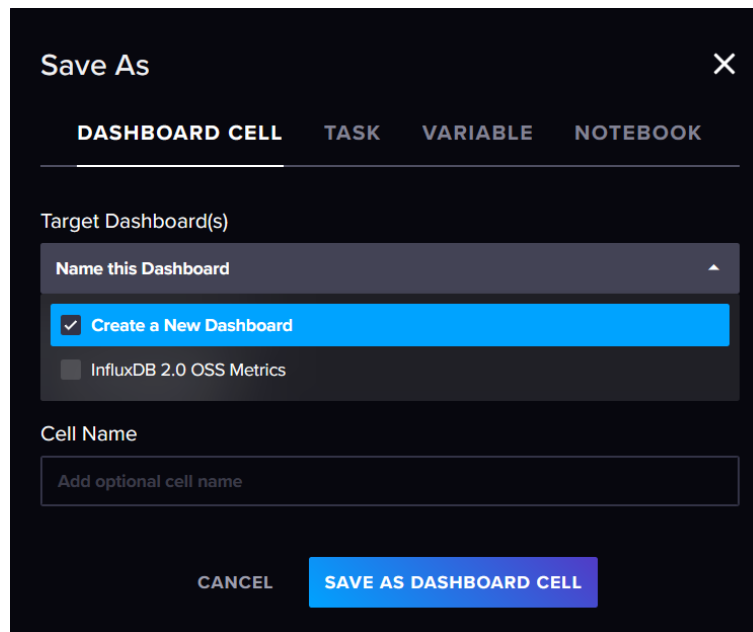


Ilustración 40 – Creación de un Dashboard en InfluxDB.

Al realizar esta acción, la ventana emergente cambiará, y permitirá la opción de escritura del nombre del Dashboard. Por lo que se escribirá un nombre identificativo y se seleccionará el botón de guardar “Save as Dashboard cell”, como se puede observar en la Ilustración 41.

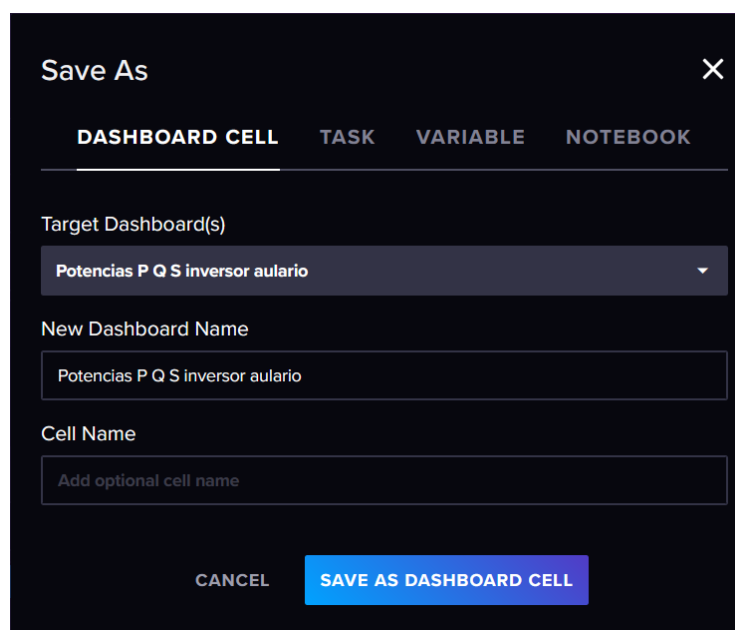


Ilustración 41 – Guardado del Dashboard en InfluxDB.

5.4. Configuración de Grafana

Con el fin de lograr una representación y gestión más eficiente de los resultados obtenidos con InfluxDB, se va a hacer uso de Grafana [7] [8]. Para llevar a cabo la configuración de Grafana, se seguirá un proceso sencillo a través de un navegador web. En primer lugar, será necesario conocer la dirección IP del servidor y conocer el puerto específico para acceder a la plataforma, que en este caso será el puerto 3000. Una vez que se disponga de esta información, se abrirá el navegador y se ingresará la dirección IP junto al puerto correspondiente.

Tras ingresar a la interfaz de Grafana, se encontrará con una pantalla de inicio de sesión al igual que en la Ilustración 42, debido a que está recién instalado y no se ha usado ningún comando de modificación de usuarios, el log in por defecto será:

- **Usuario:** admin
- **Password:** admin

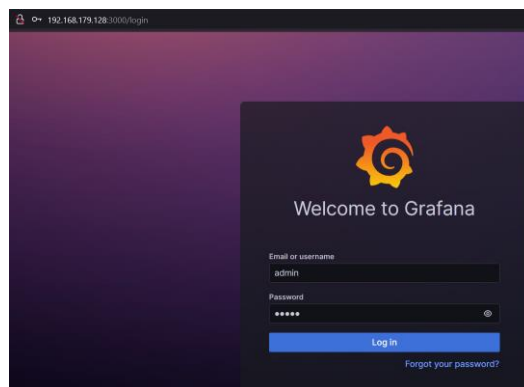


Ilustración 42 – Página de inicio de Grafana.

Después de haber iniciado sesión utilizando las credenciales por defecto, se tendrá la opción de cambiar la contraseña y establecer una más segura, al igual que en la Ilustración 43. En este caso, se utilizará “**Qwerty1.**” A su vez, Grafana ofrece la posibilidad de crear y modificar usuarios, pero debido a que es un entorno de pruebas se ha optado por dejarlo tal cual viene por defecto.

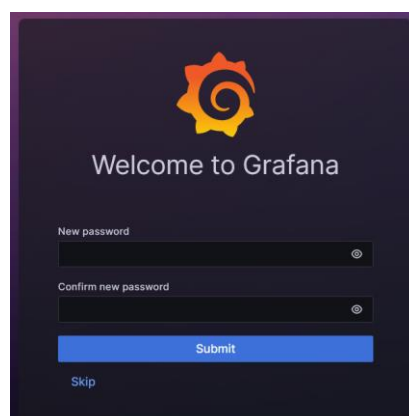


Ilustración 43 – Inicio de sesión de Grafana.

Una vez cambiada la contraseña por defecto, se mostrará el panel de mando de Grafana, al igual que en InfluxDB en la parte superior izquierda se tiene el botón hamburguesa, que permitirá desplegar el menú de opciones y el resto del panel son enlaces de interés y tutoriales básicos, como aparece en la Ilustración 44.

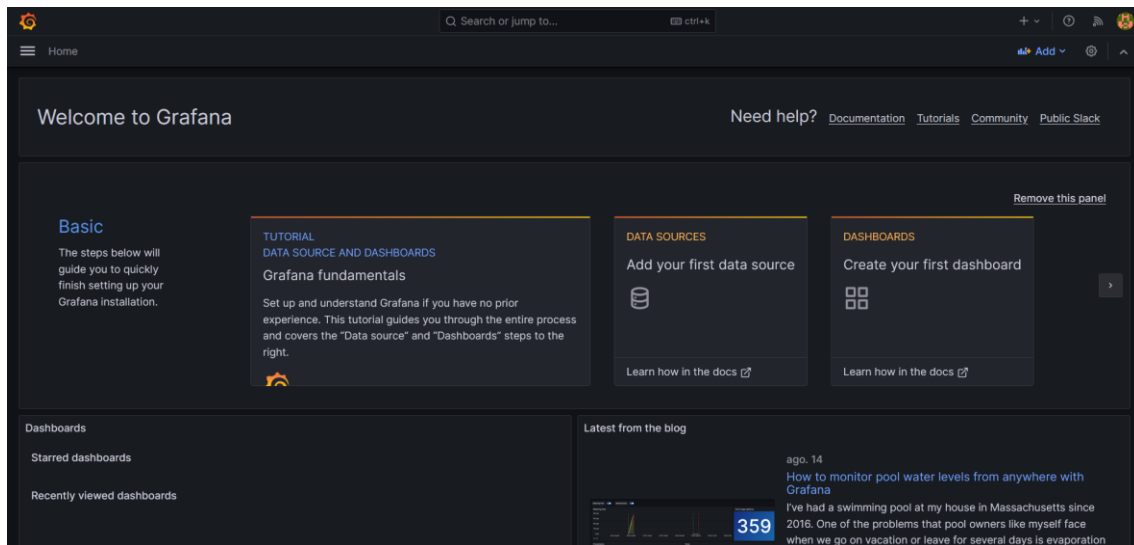


Ilustración 44 – Panel de mandos de Grafana.

Se accederá al icono de tipo "hamburguesa", ubicado en la esquina superior izquierda de la interfaz, con el fin de desplegar el menú de opciones. Una vez dentro de dicho menú, se elegirá el botón identificado como "Dashboards", en la sección mostrada en la Ilustración 45 – Menú de opciones de Grafana.. con el objetivo de crear la plantilla que albergará los gráficos. Este paso permitirá la configuración y diseño de los paneles de visualización y su posterior exportación a HTML.

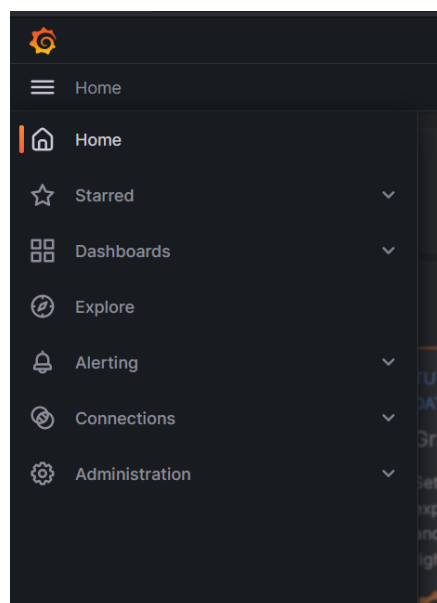


Ilustración 45 – Menú de opciones de Grafana.

Dentro de la sección “Dashboards” el siguiente paso consistirá en dirigirse a la parte derecha de la pantalla, donde se ubicará el botón "New", al igual que en la Ilustración 46, con el fin de crear las plantillas de los gráficos.

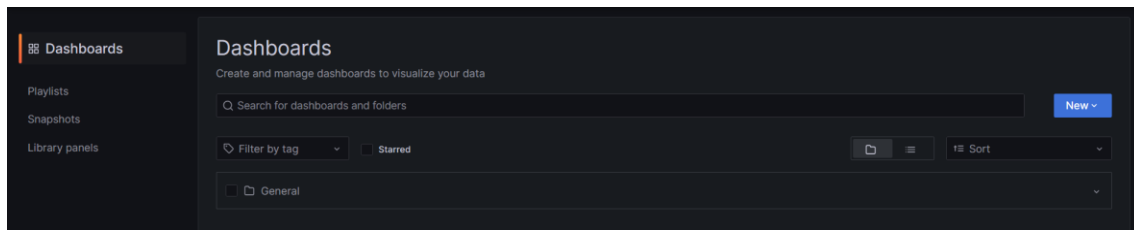


Ilustración 46 – Panel de Dashboards de Grafana.

Al hacerlo, se desplegará un pequeño menú, en el cual se seleccionará la opción de “New Dashboards”, al igual que en la Ilustración 47, debido a que no hay configuración previa ni plantillas creadas.

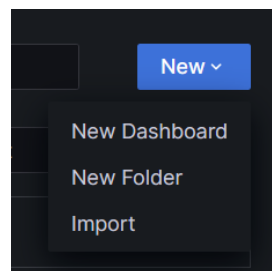


Ilustración 47 – Creación de un nuevo Dashboard en Grafana.

Dentro de la sección de creación de nuevos Dashboards, se pueden observar tres alternativas. En primer lugar se observa la opción “Start a new Dashboard by adding a visualizaton”, que implica la creación de un nuevo tablero de control mediante la incorporación de visualizaciones específicas en este caso importadas desde InfluxDB. La segunda alternativa, identificada como “add a row” ofrece la posibilidad de añadir una fila al diseño del gráfico, que permite una reestructuración más organizada de las visualizaciones y módulos de información o la importación de un panel entre distintos Dashboards.

En este caso se seleccionará la primera opción “Start your new dashbaoard by adding a visualization”, como se puede observar en la Ilustración 48 debido a que la creación del gráfico ha sido hecha completamente en InfluxDB.

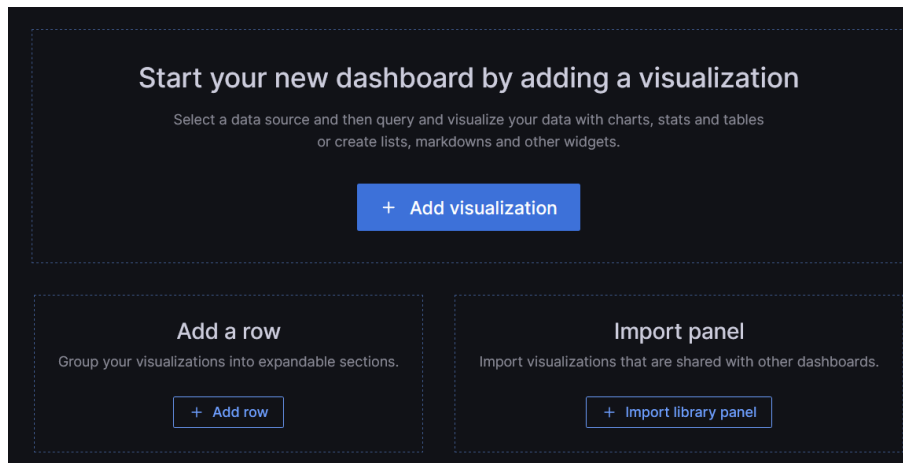


Ilustración 48 – Creación de una nueva visualización en Grafana.

Una vez seleccionada la opción mencionada, se desplegará un menú de adquisición de datos. Debido a que por el momento es el primer uso de la herramienta Grafana, la aplicación carece de asociación con otras aplicaciones, por lo que no se puede mostrar ningún dato. Para ello, se tendrá que asociar la aplicación de Grafana con la de InfluxDB seleccionando la opción de “Configure a new data source”, al igual que en la Ilustración 49.

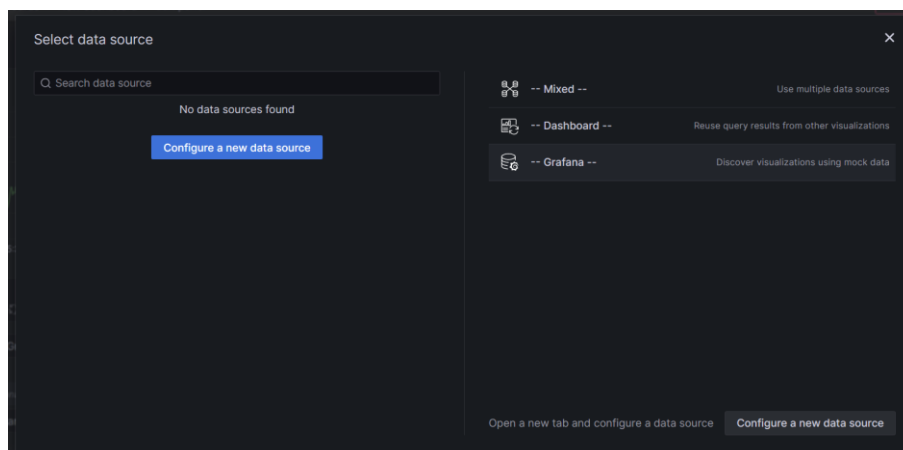


Ilustración 49 – Creación de un nuevo Datasource en Grafana.

Como es común en este tipo de casos, se muestran las aplicaciones más comunes en uso con Grafana, y entre ellas se puede observar InfluxDB, por lo que se seleccionará dicha opción, como en la Ilustración 50.

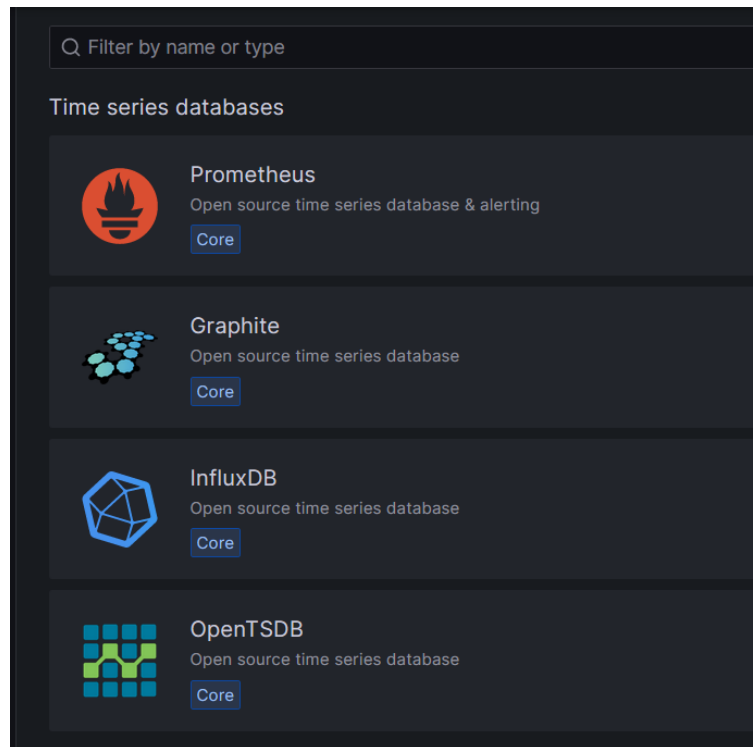
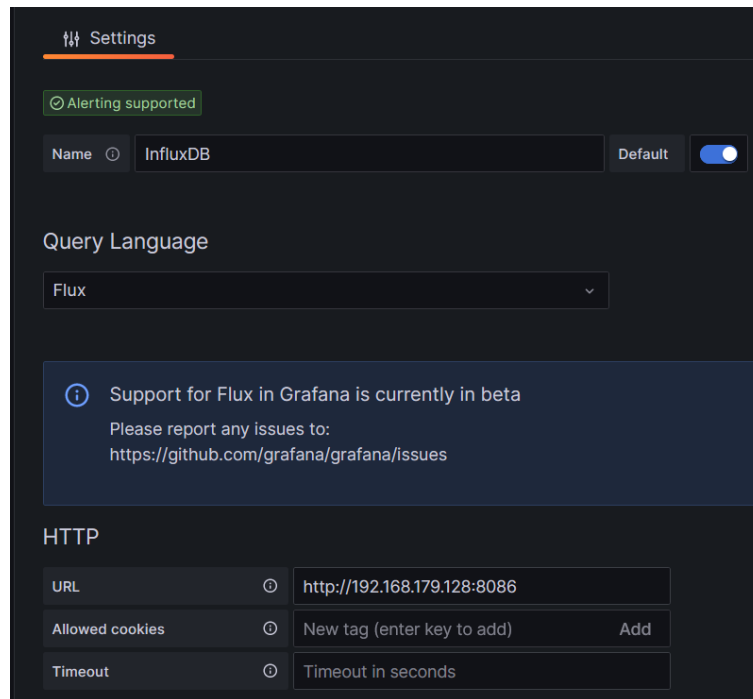


Ilustración 50 – Selección de InfluxDB como database en Grafana.

Para poder asociar Grafana con InfluxDB simplemente tendremos que rellenar el formulario que se muestra en la pantalla de la Ilustración 51. Se observan los siguientes campos:

- **Name:** Este campo se destina para asignar un nombre identificativo al enlace establecido entre Grafana e InfluxDB. En este caso se ha seleccionado "InfluxDB".
- **Query Language:** Para versiones de Grafana posteriores a la 2.0, se seleccionará el tipo de lenguaje de consulta "Flux". Sin embargo, en situaciones donde la versión sea anterior a esta, se debe optar por la alternativa correspondiente.
- **URL:** En esta sección, será necesaria la dirección completa de la aplicación InfluxDB. Para ello se introducirá la dirección IP de la máquina en la que se está trabajando, acompañada por el puerto 8086.



The image shows the 'Settings' page in Grafana for an InfluxDB data source. At the top, there is a 'Settings' header with a hamburger menu icon. Below it, a green status bar indicates 'Alerting supported'. The 'Name' field is set to 'InfluxDB' and has a 'Default' toggle switch which is turned on. Under the 'Query Language' section, a dropdown menu is set to 'Flux'. A blue information box contains the text: 'Support for Flux in Grafana is currently in beta. Please report any issues to: https://github.com/grafana/grafana/issues'. The 'HTTP' section contains three rows: 'URL' with the value 'http://192.168.179.128:8086', 'Allowed cookies' with a 'New tag (enter key to add)' field and an 'Add' button, and 'Timeout' with the value 'Timeout in seconds'.

Ilustración 51 – Configuración de InfluxDB en Grafana parte I.

La siguiente parte del formulario es más sencilla, debido a que como Grafana se va a utilizar con el fin de añadir los gráficos al HTML (que se desarrollará más adelante en la sección Montaje del entorno web y publicación de datos), se tendrá que desactivar cualquier tipo de autenticación, por ello en la sección de “Auth” se dejarán sin marcar las opciones que se muestran al igual que en la Ilustración 52.

Por último y con el fin de que se asocie con el bucket pertinente del cual se importaban los datos de Node-RED. Se tendrán que rellenar los campos “Organization” y “Token” según el paso de la Ilustración 9, en el cual se configuró InfluxDB en primera instancia, y posteriormente el nombre del bucket por defecto, el cual se originó en la sección Ilustración 9.

The image shows the configuration interface for InfluxDB in Grafana. It is divided into three main sections:

- Auth:** Contains several toggle switches. 'Basic auth' and 'Skip TLS Verify' are turned on. 'With Credentials', 'With CA Cert', and 'Forward OAuth Identity' are turned off.
- Custom HTTP Headers:** A section with a '+ Add header' button.
- InfluxDB Details:** A form with the following fields:
 - Organization: upna
 - Token: configured (with a 'Reset' button)
 - Default Bucket: node-red-data
 - Min time interval: 10s
 - Max series: 1000

Ilustración 52 – Configuración de InfluxDB en Grafana parte II.

Una vez rellenado el formulario se seleccionará “Save & Test”, que se observa en la Ilustración 53, con el fin de comprobar que los campos se han rellenado correctamente.

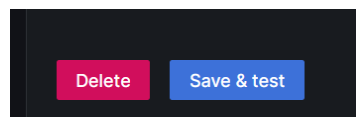


Ilustración 53 -Guardado del Datasource de InfluxDB en Grafana.

En caso afirmativo, se mostrará el siguiente mensaje en pantalla, como aparece en la Ilustración 54.

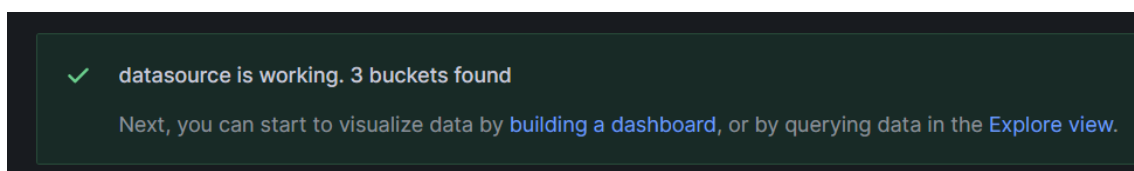


Ilustración 54 – Mensaje de funcionamiento de Datasource en Grafana.

Una vez creada la asociación con el “Data source” se volverá al paso de la Ilustración 49 Ilustración 49 – Creación de un nuevo Datasource en Grafana. y esta vez se seleccionará “InfluxDB”, al igual que en la Ilustración 55, que es la asociación creada en el paso anterior.

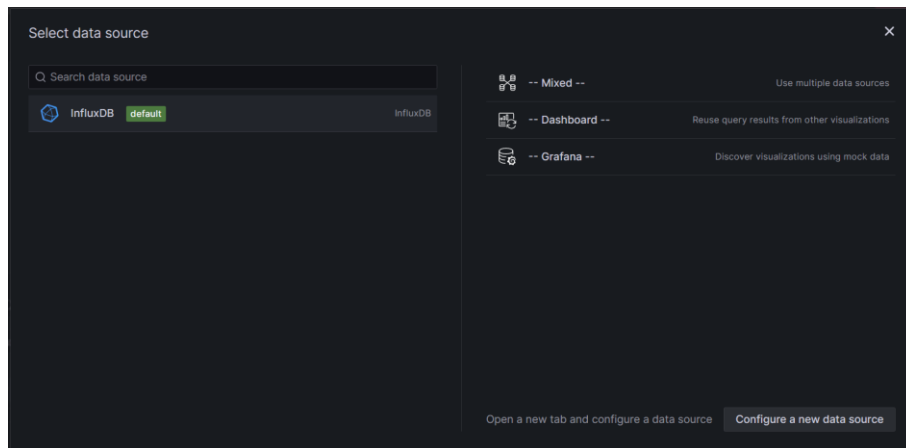


Ilustración 55 – Selección de Datasource en Grafana.

Una vez seleccionada dicha opción, se mostrará en pantalla un cuadro de control similar al de InfluxDB, en el que solamente se tendrá que pegar el código Flux obtenido en la Ilustración 39, en la sección inferior de la pantalla de la Ilustración 56.

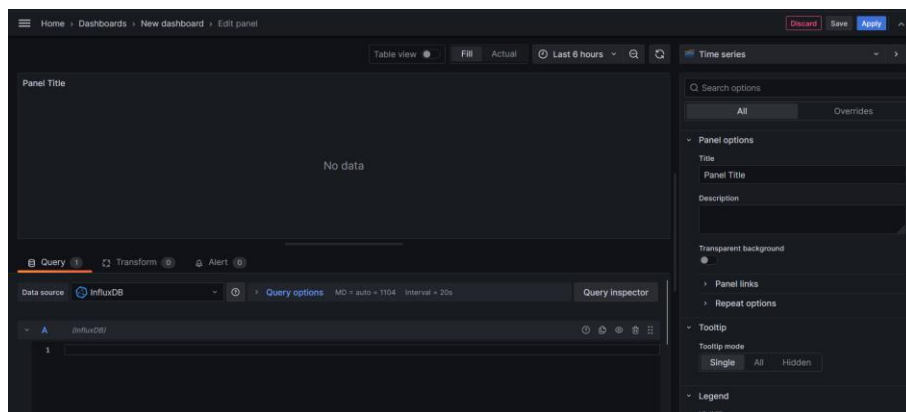


Ilustración 56 – Panel de control del Dashboard en Grafana.

En el momento en el que se pegue el código Flux, al igual que en la Ilustración 57 se comenzará a observar la línea temporal de la generación de potencia. Se aconseja observar la sección derecha de la pantalla con el fin de dejar el gráfico al gusto del usuario.

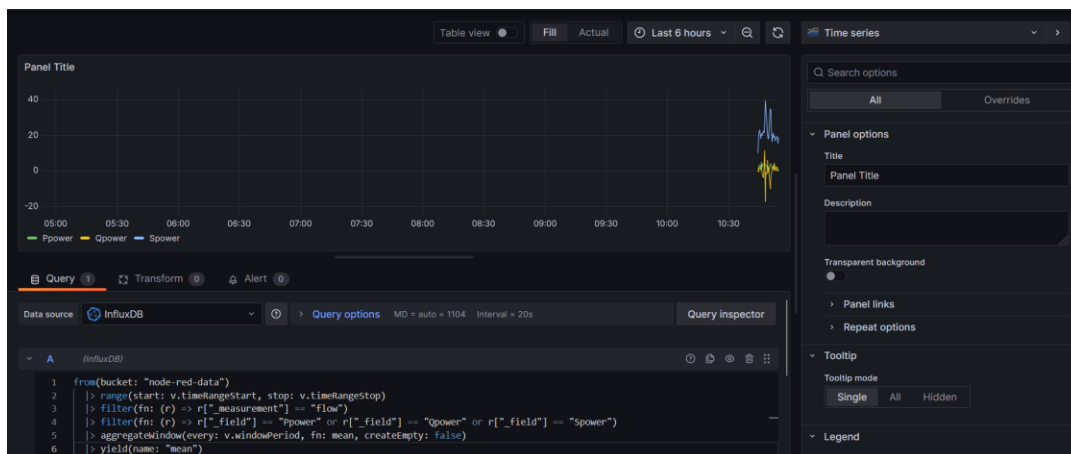


Ilustración 57 – Pegado del Script de InfluxDB en el Dashboard de Grafana.

Una vez completada la creación del gráfico en Grafana, se le dará a “Save” para guardar los cambios realizados, al igual que en la Ilustración 58.

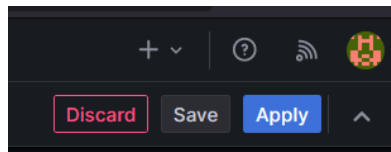


Ilustración 58 – Guardado de datos en el Dashboard de Grafana.

Esta acción devolverá a la pantalla de la Ilustración 48 y se tendrá que seleccionar el botón de guardado, de la Ilustración 59, para que guarde el Dashboard y se pueda comenzar a usar. En caso de no guardarlo se mostrarán errores a la hora de la publicación de los gráficos en la página HTML.

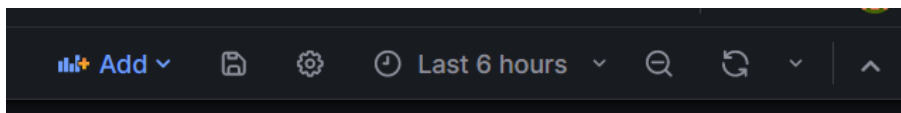


Ilustración 59 – Guardado del Dashboard completo de Grafana.

Al darle al botón de guardar, se mostrará una ventana emergente, mostrada en la Ilustración 60, en la cual se introducirá el nombre identificativo del Dashboard pertinente y se seleccionará “Save” para de guardarlo definitivamente.

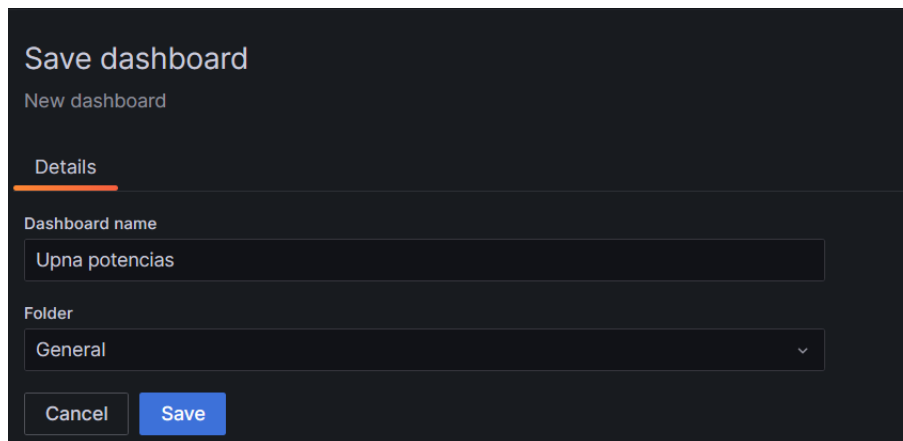


Ilustración 60 – Nombramiento del Dashboard panel de Grafana.

Una vez finalizado con este proceso, se podrá ir al gráfico recién creado y en la sección superior derecha seleccionar los tres puntos que indican la configuración del gráfico, como se puede observar en la Ilustración 61.

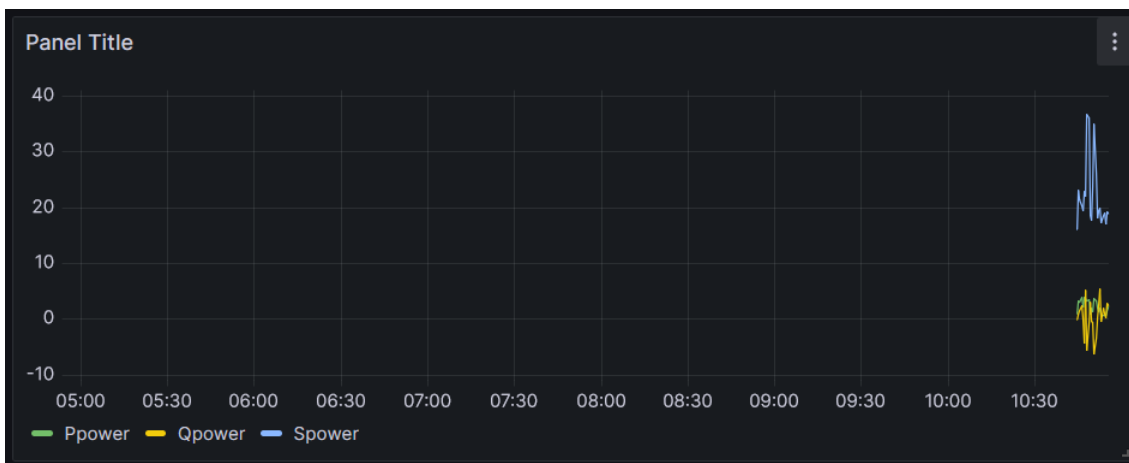


Ilustración 61 – Visualización del gráfico completo en Grafana.

Esta acción desplegará un menú de opciones, en la cual se seleccionará “Share” para compartir el gráfico, al igual que en la Ilustración 62.

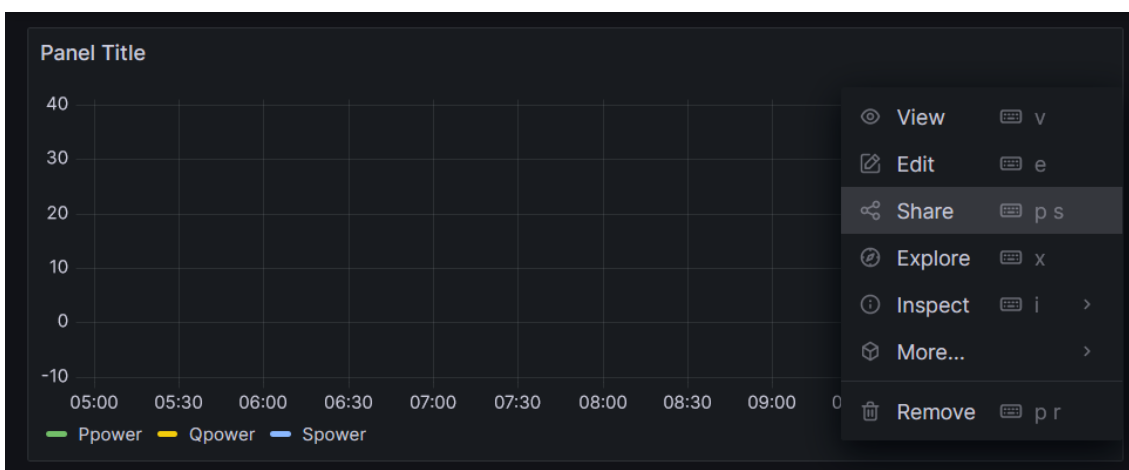


Ilustración 62 – Panel de opciones del Dashboard de Grafana.

Entre la lista de opciones de “Share” la que se utilizará en este proyecto es la opción de “Embed”. Esta sección proporcionará un fragmento de código HTML, que se puede apreciar en la Ilustración 63, que una vez pegado en la página web, hace una incrustación del gráfico.

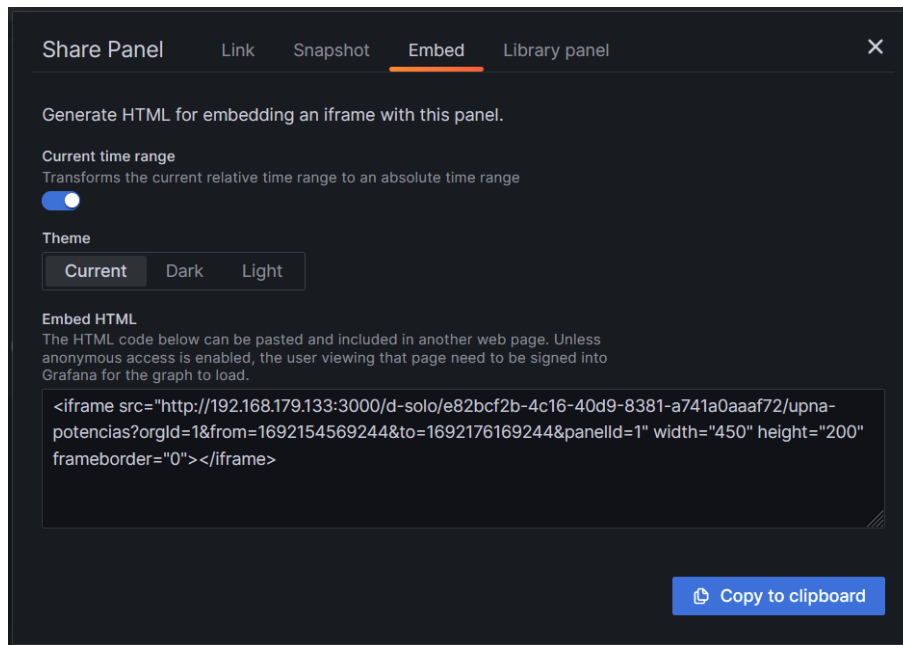
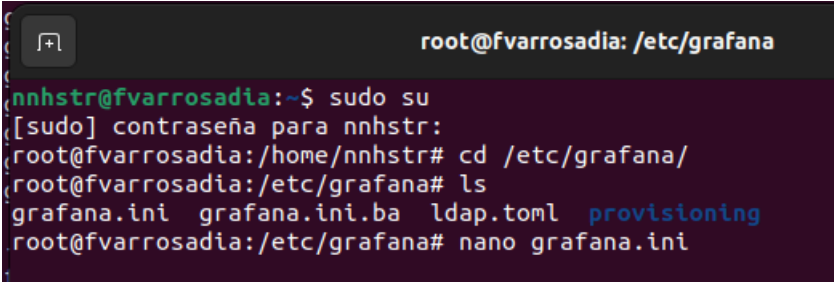


Ilustración 63 – Selección de embed del Dashboard de Grafana.

Por último y para que el código Embed funcione para cualquier página web, ya sea dentro o fuera del servidor, se tendrá que hacer una configuración adicional de Grafana. Para ello, se abrirá el Terminal de Ubuntu en el cual está instalado Grafana y se accederá al modo super usuario. Una vez se esté como administrador, se irá a la carpeta “/etc/grafana” y se editará el fichero “grafana.ini” el cual es el encargado de almacenar la configuración básica de la aplicación, como en la Ilustración 64. Para ello se usarán los siguientes comandos:

```
sudo su
cd /etc/grafana/
nano grafana.ini
```



```
root@fvarrosadia: /etc/grafana
nnhstr@fvarrosadia:~$ sudo su
[sudo] contraseña para nnhstr:
root@fvarrosadia:/home/nnhstr# cd /etc/grafana/
root@fvarrosadia:/etc/grafana# ls
grafana.ini  grafana.ini.ba  ldap.toml  provisioning
root@fvarrosadia:/etc/grafana# nano grafana.ini
```

Ilustración 64 – Consola de comandos del servidor de Ubuntu.

Una vez dentro del fichero de configuración se puede usar la combinación de teclas “control” + “w” con el fin de buscar la opción de “allow_embedding”. Una vez encontrada se quitará el punto y coma que le precede, y se cambiará el valor a “true”, como en la Ilustración 65. Dicha opción permitirá incrustar el código html en cualquier web y que se muestre sin problemas.

```
</>, <iframe>, <embed> or <object>. default is false.  
allow_embedding = true
```

Ilustración 65 – Ajuste de opción de embedding de Grafana en Ubuntu.

A su vez se cambiará la opción enabled de “auth.anonymous”, para que el gráfico pueda ser visto públicamente y sin necesidad de hacer uso de un usuario. Para ello y como se ha realizado en el paso anterior se quitará el punto y coma que le precede y posteriormente se cambiará a true, con el fin de que quede igual a la Ilustración 66.

```
#####  
[auth.anonymous]  
# enable anonymous access  
enabled = true
```

Ilustración 66 – Ajuste de opción de anonymous de Grafana en Ubuntu.

Una vez realizados estos cambios se presionará “control” + “x” y se le dará a “enter”, para guardar los cambios de la configuración. Por último y para que se apliquen estos cambios se hará un reinicio del servicio de Grafana, como se observa en la Ilustración 67 con los siguientes comandos:

```
systemctl stop grafana-server
```

```
systemctl start grafana-server
```

```
root@fvarrosadia:/etc/grafana# systemctl stop grafana-server  
root@fvarrosadia:/etc/grafana# systemctl start grafana-server  
root@fvarrosadia:/etc/grafana# systemctl status grafana-server  
● grafana-server.service - Grafana instance  
   Loaded: loaded (/lib/systemd/system/grafana-server.service;  
   Active: active (running) since Wed 2023-08-16 11:05:51 CEST;  
   Docs: http://docs.grafana.org
```

Ilustración 67 – Reinicio del servicio de grafana en Ubuntu.

Para comprobar que los cambios se han aplicado correctamente, se irá a la web de Grafana y en el menú hamburguesa se seleccionará la opción de “Administration”, al igual que la Ilustración 68.

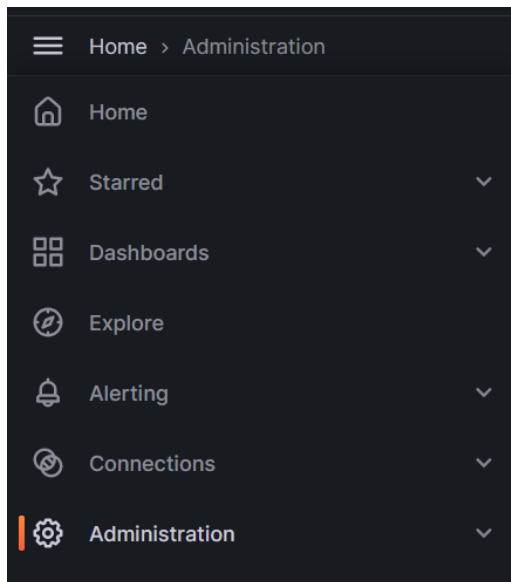


Ilustración 68 – Menú del panel de mandos de Grafana.

Dentro de “Administration” en la opción de “settings” aparecerá una lista con la configuración de Grafana, entre ella se podrá observar la opción de “auth.anonymous” que debería estar en “true”, y el “embedding” que deberá estar a su vez del mismo modo que la Ilustración 69.

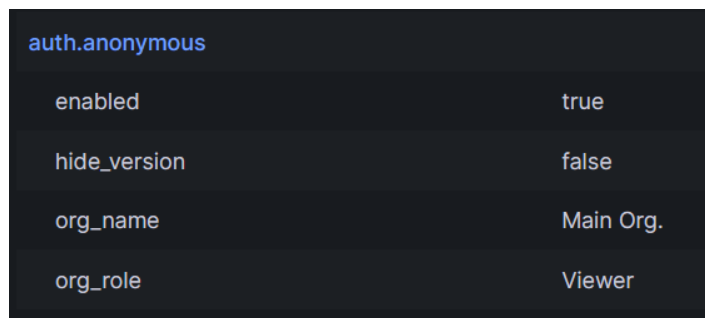


Ilustración 69 – Comprobación del settings en Grafana.

5.5. Configuración de Apache2

Como ya se ha explicado anteriormente al final de la sección Propuestas del entorno de desarrollo, Apache2 tiene la funcionalidad de VirtualHosting, por lo que se enfocará la configuración de este servicio como si se tuviesen varias páginas webs, por si en un futuro se añaden distintas webs evitar la configuración a posteriori.

Por ello, la ruta donde se localizará la página web será:

```
/var/www/html/fotovoltaica/*
```

En dicha carpeta, se localizarán los siguientes ficheros mostrados en la Ilustración 70.

```
C:.\n|   index.html\n|   styles.css\n|   \n+---csvs\n|   \n|   actuales.csv\n|   \n+---images\n|   \n|   imagen1.png\n|   imagen2.png\n|   imagen3.png\n|   imagen4.png\n|   imagen5.png\n|   imagen6.png\n|   \n+---tff\n|   \n|   GothamRounded-Bold.otf\n|   GothamRounded-BoldItalic.otf\n|   GothamRounded-Book.otf\n|   GothamRounded-BookItalic.otf\n|   GothamRounded-Light.otf\n|   GothamRounded-LightItalic.otf\n|   GothamRounded-Medium.otf\n|   GothamRounded-MediumItalic.otf\n|   GothamRoundedBold_21016.ttf\n|   GothamRoundedBook_21018.ttf\n|   GothamRoundedLight_21020.ttf\n|   GothamRoundedMedium_21022.ttf\n|   \n\\---webs\n|   \n|   campus.css\n|   campus.html
```

Ilustración 70 – Ficheros de la página web.

Para hacer una copia de todos estos ficheros al directorio mencionado, en primera instancia se llevarán todos los ficheros a una carpeta con nombre “fotovoltaica” localizada en el escritorio de Ubuntu.

Posteriormente, se creará la carpeta ‘fotovoltaica’ en el directorio predeterminado de Apache2, con el fin de hacer uso del VirtualHost, y posteriormente se aplicará el comando de copia para poder llevar los ficheros del escritorio a Apache2.

```
Mkdir /var/www/html/fotovoltaica
```

```
cp -r fotovoltaica/* /var/www/html/fotovoltaica/
```

Como se muestra en la Ilustración 71.


```
root@fvarrosadia:/home/fvarrosadia/Desktop# mkdir /var/www/html/fotovoltaica
root@fvarrosadia:/home/fvarrosadia/Desktop# cp -r fotovoltaica/* /var/www/html/fotovoltaica/
root@fvarrosadia:/home/fvarrosadia/Desktop#
```

Ilustración 71 – Copia de la página web en el directorio de Apache2.

Una vez copiados todos los ficheros, se abrirá el navegador web de conveniencia y se introducirá la dirección IP del servidor en cuestión junto con el puerto designado en la configuración, es decir el 80, y se mostrara la web de la Ilustración 72.



Ilustración 72 – Prueba de funcionamiento de Apache2.

Por último, se necesitará reemplazar el archivo CSV, que se encuentra en la ruta **“/var/www/html/fotovoltaica/csvs”**, por un enlace simbólico hacia el archivo de datos exportados de Node-RED, ubicado en **“.../Documentos/actuales.csv”**. Para llevar a cabo este proceso, se tendrá que ir a la carpeta Documentos y darle botón derecho “Abrir en una terminal” y se introducirá el comando de la Ilustración 73.

```
In actuales.csv /var/www/html/fotovoltaica/csvs/
```

```
In energia.csv /var/www/html/fotovoltaica/csvs/
```

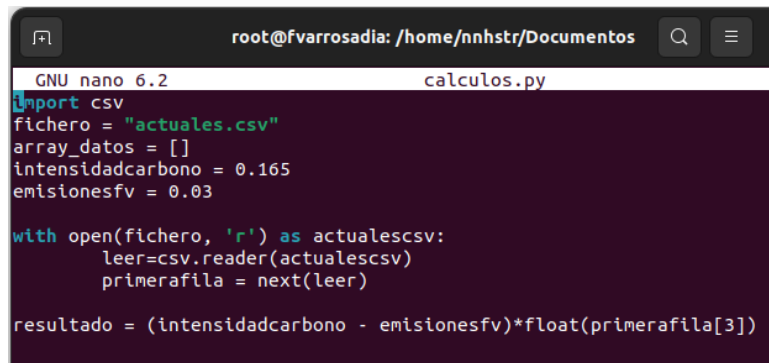
```
root@fvarrosadia:/home/nhstr/Documentos# ln actuales.csv /var/www/html/csvs/
root@fvarrosadia:/home/nhstr/Documentos#
```

Ilustración 73 - Enlace simbólico del fichero CSV de Node-RED.

La estructura y configuración de la página web se explicarán posteriormente en Montaje del entorno web y publicación de datos.

5.6. Configuración de Python y Crontab

Otro de los puntos a tratar en el proyecto es la capacidad de modificación de los datos recolectados por el inversor. Para ello, se creó un fichero tipo “*.py” en la ruta “\home\%username%\Documentos”. Y se hace uso de un par de comandos tal y como se muestra en la Ilustración 74.



```

root@fvarrosadia: /home/nnhstr/Documentos
GNU nano 6.2 calculos.py
import csv
fichero = "actuales.csv"
array_datos = []
intensidadcarbono = 0.165
emisionesfv = 0.03

with open(fichero, 'r') as actualescsv:
    leer=csv.reader(actualescsv)
    primerafila = next(leer)

resultado = (intensidadcarbono - emisionesfv)*float(primerafila[3])
  
```

Ilustración 74 - Código de Python sobre el fichero csv.

En dicho programa, se selecciona el fichero “actuales.csv” que es el encargado de gestionar los datos de Node-RED a tiempo real y se hace una lectura en la cual se selecciona la primera fila, debido a que es la más reciente. Posteriormente se selecciona la columna de “Energía producida” que ofrece la cantidad de energía producida por el campo fotovoltaico desde que se inició el inversor. Después de la selección de la variable hace un cálculo haciendo uso de la Ecuación 1 y se guarda el resultado en la variable “resultado”.

$$CO_{2\text{evitado}} = (0,165 - 0,03) \cdot (Wattimeter_n)$$

Ecuación 1 – Ecuación de CO₂ no emitido a la atmósfera.

Dicha ecuación está ofrecida por el Departamento de Energías Renovables de la UPNA y respaldada por la European Environmental Agency [9], y tiene como objetivo transformar la energía generada en un equivalente a dióxido de carbono evitado, teniendo como referencia la emisión media de CO₂ a la hora de generar la energía eléctrica en España (0,165). Pero hay que tener en cuenta las emisiones asociadas a la fabricación, generación y transporte de dichos paneles, por ello al coeficiente de generación se le resta un 0,03.

Posteriormente se hace un volcado de dicha variable en el fichero “energia.csv”, para guardar los datos de energía producida y la cantidad de dióxido de carbono evitado a la atmósfera, para poder tener un registro temporal y poder reflejarlo en la página web, ya sea como gráfico temporal o variable a tiempo real.

En este caso, se ha decidido implementar un registro de carácter diario. Para ello, el archivo “cálculos.py” será ejecutado cada día a las 00:00 horas. Posteriormente, se llevará a cabo una operación de resta entre las filas de “actuales.csv” y la posterior, consiguiendo así un conjunto de datos que refleje con precisión los valores diarios solicitados.

6. MONTAJE DEL ENTORNO WEB Y PUBLICACIÓN DE DATOS

Una vez que se ha establecido la configuración completa de Node-RED, InfluxDB y Grafana, se procederá a iniciar el proceso de implementación del entorno web, encargado de la publicación de los datos provenientes del Inversor. En este contexto, se ha optado por utilizar como base la plantilla del Dashboard proporcionada por Alberto Berrueta y Leyre Catalán, mostrada en la Ilustración 75. Esta elección se basa en el hecho de que en el pasado, la Universidad Pública de Navarra realizó una inversión con el propósito de desarrollar un portal de interfaz amigable y accesible. Dicho portal tiene como objetivo presentar de manera simplificada los aspectos más relevantes de la instalación fotovoltaica.

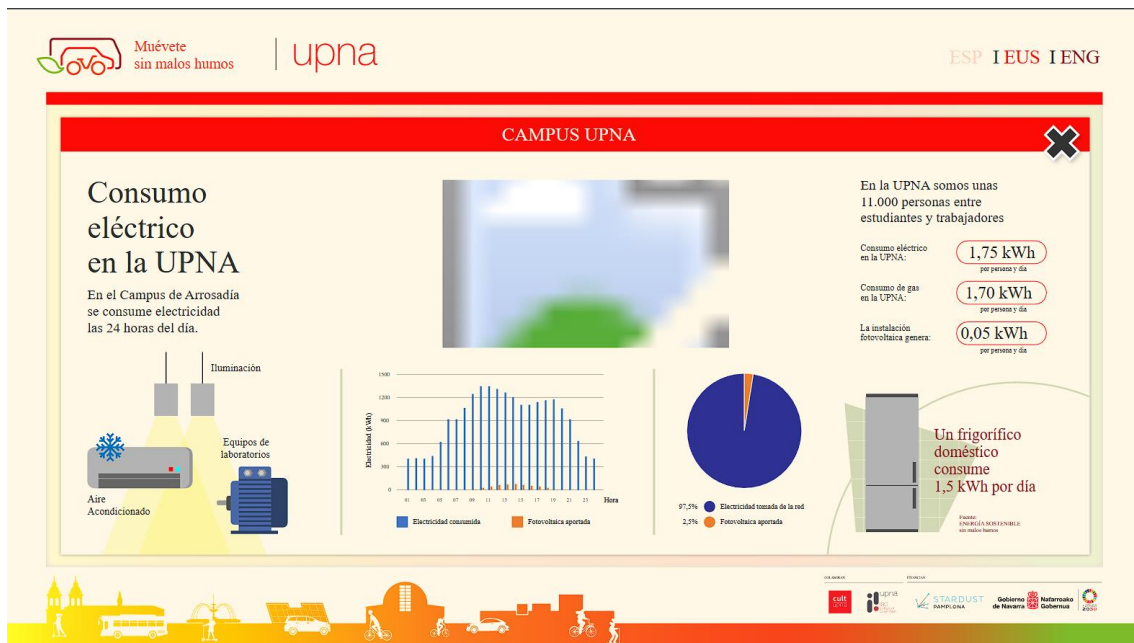


Ilustración 75 - Dashboard proporcionado por la UPNA.

El inconveniente asociado a esta plantilla radica en que fue originada mediante el uso de Illustrator. Como resultado, el código HTML puede tornarse en ciertas ocasiones complicado de manejar. Por esta razón, se ha tomado la decisión de elaborar una versión alternativa que conste de un archivo HTML y un archivo CSS con un diseño similar, pero con un código que resulte más legible para los programadores en el futuro.

Inicialmente, se contempló la opción de realizar esta creación empleando los conocimientos previos en HTML. Sin embargo, posteriormente se decidió evaluar la capacidad de ChatGPT para llevar a cabo el desarrollo de la página web y ver hasta qué punto se podía abarcar.

Con este propósito, se empleó la plantilla suministrada por los tutores, y se procedió a dividir el documento en secciones de mayor relevancia. A través de operaciones matemáticas sencillas, se calcularon las proporciones en porcentajes de cada uno de los elementos. Como resultado de este proceso, se obtuvieron tres secciones dispuestas en

forma vertical (izquierda, centro y derecha), siendo la sección central subdividida a su vez en dos segmentos distintos (superior e inferior).

Llegados a este punto se comenzó con un proceso iterativo de prueba y error, solicitando a ChatGPT el Código HTML y CSS de la web, de forma que ofreciera la plantilla que se esperaba. En dicho proceso se le solicitaba que las divisiones siguieran un esquema de colores para verificar que estuvieran correctamente realizadas, como se puede observar en la Ilustración 76.

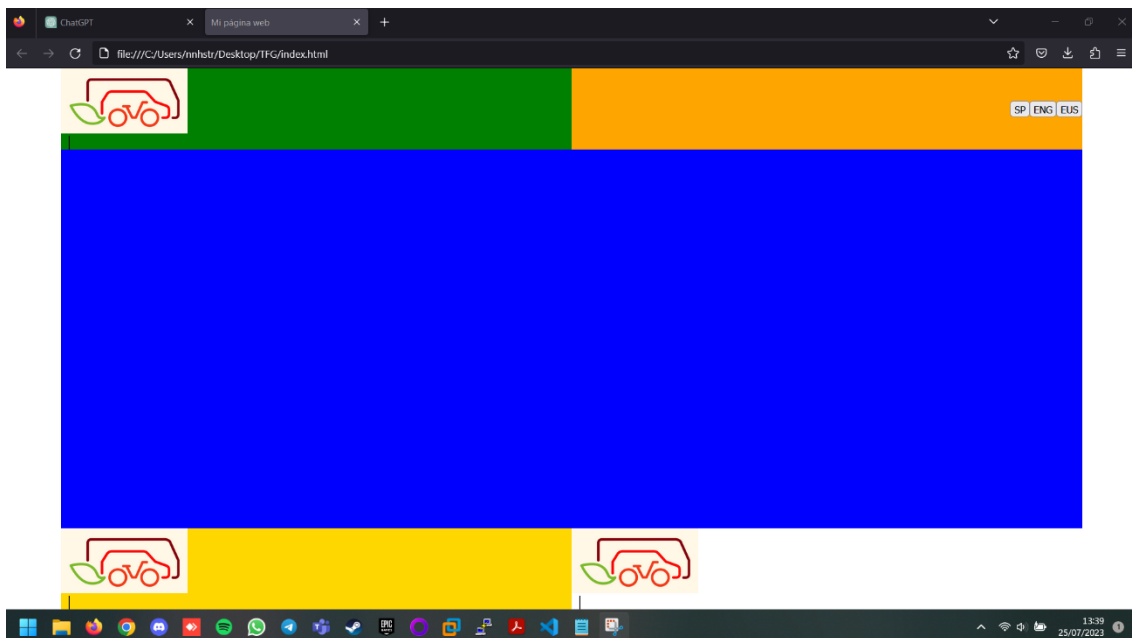


Ilustración 76 - Plantilla HTML conseguida con ChatGPT.

Una vez obtenida la plantilla base, con los conocimientos previos de HTML, se modificó un poco el código a fin de que visualmente fuera más atractivo para el usuario, se añadieron un par de divisiones y hubo un reformato del texto que aparecía en pantalla.

Posteriormente, se insertó el iframe obtenido con Grafana, y tras comprobar que funcionaba correctamente, al igual que en Ilustración 77. Se decidió hacer uso de JavaScript, para poder modificar la escala temporal de la gráfica, esto fue posible debido al formato que presentaba el iframe.



Ilustración 77 - Inserción del iframe en la web.

<http://192.168.179.128:3000/d-solo/b3cad421-cd75-4ad3-9ba8-0cee73011416/new-dashboard?orgId=1&from=1690478717655&to=1690479017655&panelId=3>

Se puede observar que las marcas de tiempo “from” y “to” especificadas en el enlace, están en formato Unix Epoch, que representa el número de segundos transcurridos desde el 1 de Enero de 1970, por lo tanto haciendo una sustitución de esos valores mediante código, se pueden añadir dos botones “Fecha inicio” y “Fecha fin”, como se puede observar en la Ilustración 78, con el fin de adaptar la escala. A su vez, se exploró la opción de hacer uso de los froms provenientes de HTML que permiten escalas temporales respecto a la hora actual “now” y operacionales temporales como “s” segundos y “h” horas.

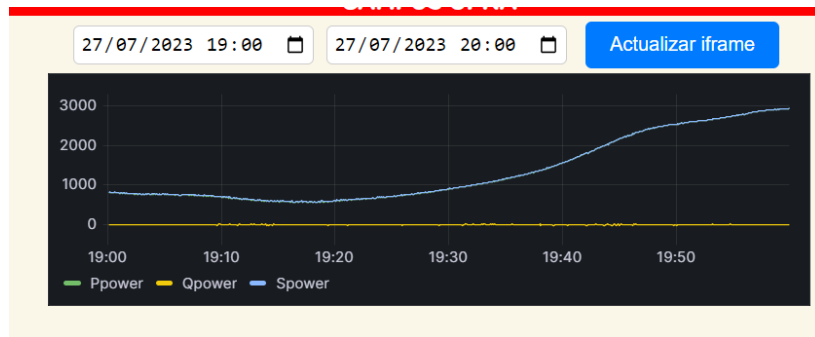


Ilustración 78 - Creación de franjas temporales del iframe.

Por último y a razón de tener una página un poco más dinámica, se ha optado por hacer uso de variables numéricas, que permitan la reformulación de variables y su posterior modificación a base de cálculo, por ejemplo de “Potencia generada” a “Toneladas de CO2 evitadas”. Para ello, se exploraron diversos métodos de exportación de variables, pero la que mejor resultados ofreció fue la realizada directamente desde Node-RED.

Para ello, se creó un documento CSV en la máquina Ubuntu, la cual tenía un enlace fuerte con un archivo CSV localizada en el servicio de Apache2. Por lo que los datos que se modificasen en el fichero de documentos se actualizarían instantáneamente en el servicio de Apache2. Por lo que haciendo uso de los ficheros CSV y un poco de JavaScript, se consiguió publicar la variable en formato JSON en la página web, de esta forma se consiguió que el valor de la variable se publicase en una tabla creada en la sección derecha de la web, como se parecía en la Ilustración 79.



Ilustración 79 - Datos obtenidos de Node-RED mediante CSV.

A su vez se hará uso de un fichero Python, para poder reformular los datos a mostrar en la web. En este caso se ha decidido transformar la cantidad de energía diaria producida por los paneles fotovoltaicos a razón de saber cuánta cantidad de CO₂ no se está emitiendo a la atmósfera gracias al uso del transporte eléctrico, para ello se hará uso de la Ecuación 1.

Quedando entonces una web tal y como se muestra en la Ilustración 80.



Ilustración 80 – Página web completada en castellano.

A su vez se le integraron tres botones de selección de idioma a fin de poder adaptarse al usuario correspondiente, tal y como se muestra en la Ilustración 81.



Ilustración 81 - Página web completada en euskera.

7. COMPARATIVA DE CHATGPT

ChatGPT es una tecnología de procesamiento de lenguaje desarrollada por OpenAI. Se basa en una arquitectura GPT (Generative Pre-trained Transformer) que es capaz de comprender y generar texto de manera relativamente coherente y contextual. Dicha herramienta se ha convertido en un recurso valioso en una amplia gama de aplicaciones y de asistencia a usuarios.

Con el fin de analizar el potencial de dicha herramienta, se va a realizar una comparación entre los puntos vistos anteriormente realizados por un alumno de ingeniería de la Universidad Pública de Navarra y lo que podría obtenerse mediante la herramienta ChatGPT sin poseer conocimiento previo sobre el tema. Para ello y con el fin de evitar complicaciones a la hora de estructurar el campo fotovoltaico, a la aplicación ChatGPT se le mencionará que únicamente se tienen 10 paneles fotovoltaicos instalados en el domicilio.

Dicha conversación se ha obtenido en la versión GPT 3.5 a día 14 de Agosto de 2023.

Nada más presentarle el proyecto a ChatGPT, muestra una lista con los pasos más importantes a seguir con la instalación, es decir, una estructura general previa a un proyecto. Por lo que se debe ser más específico con el material que se tiene y que es lo que se quiere hacer.

Por ello se le comenta a ChatGPT que se tiene un ordenador sin sistema operativo y que el fin del proyecto es crear la propia web. Por lo que ofrece la posibilidad de hacer una instalación limpia de sistema operativo, aconsejando hacer uso de la distribución Ubuntu, debido a que es muy amigable para principiantes. Tal y como se ha visto en “Montaje del servidor”. Cabe destacar que, aunque la recomendación está bien hecha, el motivo por el cual el sistema sería el óptimo está totalmente equivocado, ya que en algunos casos las distribuciones Linux basan la mayor parte de su configuración vía terminal, por lo que para alguien que no esté familiarizado con este tipo de interfaz puede ser una barrera a la hora de desarrollar el proyecto.

Posteriormente, se le comenta a ChatGPT que se quiere montar el servidor web, por lo que propone Apache2 como servicio. Hace una guía de instalación bastante buena, aunque sí que es cierto que es una respuesta parcial, obvia completamente la creación de VirtualHost y en primera instancia no comenta el directorio en el cual se encuentra la carpeta que muestra exactamente el puerto 80 de la máquina. Se desconoce si el motivo es debido a que se le ha informado que no se es experto en el tema o debido a que se requieren unas preguntas más detalladas para obtener más información sobre el servicio de Apache2.

El siguiente paso a realizar con ChatGPT es el intento de comunicarse con el inversor, es cierto que sus respuestas son correctas debido a que ofrece la posibilidad de conectarse vía Python y hacer un filtrado de datos, pero no es la mejor respuesta. Por lo que se decide volver a preguntarle haciendo énfasis en que se quiere hacer uso de una herramienta visual y atractiva para el usuario, en ese momento hace una lista de las

aplicaciones (Google Sheets, Node-RED, IoT Plataformas y Plataformas varias de monitoreo energético). Aun siendo ciertas sus respuestas se coaccionará el camino a seguir, siendo elegido el campo de Node-RED para realizar la comunicación.

Una vez escrita la opción escogida, se muestra una guía para su instalación. Por lo que una vez completada hará un resumen de las partes más importantes de Node-RED, así como del tipo de configuración que se debe seguir, es decir, que paquetes y bibliotecas se deben añadir para el correcto funcionamiento de la comunicación Modbus. Ya que ChatGPT no tiene información del inversor utilizado para este proyecto, menciona en repetidas ocasiones que se debe leer el manual o ponerse en contacto con el desarrollador del inversor, con el fin de saber qué tipo de protocolo se está utilizando para realizar la comunicación (HTTP, MQTT, Modbus, etc.).

En caso de que se le responda con el tipo de protocolo que se está utilizando, mostrará los pasos previos a la configuración del flow, es decir, guiará al usuario hasta el panel de "Manage Palette" con el fin de agregar los nodos apropiados de Modbus. Posteriormente describirá el procedimiento para agregar Nodos al Flow de forma bastante eficaz, aunque de forma redactada, por lo que dependiendo de la versión a la que tenga acceso ChatGPT y la versión instalada en el servidor puede diferir, lo que incurrirá a campos inexistentes o configuración mal detallada.

Por ejemplo, a la hora de hacer una configuración del servidor en el nodo "Modbus read", menciona que hay un campo denominado Length, el cual en la versión utilizada para este proyecto no existe. Y a su vez, a la hora de denominar el número de registro de comienzo, aunque sí que es cierto que por ejemplo sería el 40001, pero la forma de configurarlo correctamente en la versión actual de Node-RED sería en el campo de código de función especificar el "4000" y en el campo de dirección especificar el "1".

Una vez que se ha establecido y se ha mostrado en pantalla todos los valores del inversor, se le solicita a ChatGPT que cree un HTML y un CSS, el cual se usará para mostrar los datos del inversor. En esta sección, sí que es preferible tener un conocimiento previo sobre cómo funciona una página web y como suelen estar construidas, debido a que facilitará en el proceso de creación de la misma. Por lo que se le hace una descripción detallada de las distintas divisiones que va a tener la página y el uso de cada una de ellas. A su vez, se le solicitarán tres botones con el fin de poder cambiar de idioma y una sección para integrar una web dentro de la creada, con el fin de añadir los gráficos de Grafana.

El resultado de dicha consulta ha sido totalmente inesperado y gratificante, debido a que, si se consigue ser lo suficientemente preciso, claro y conciso se obtendrán los dos ficheros totalmente funcionales y con las especificaciones requeridas. Para ver el potencial de la herramienta, se hicieron modificaciones al fichero HTML proporcionado por ChatGPT con diversos errores en el código que hacía que partes no funcionasen o que los divs se solapasen unos sobre otros. Posteriormente se enviaba el código completo de la página y el error que se apreciaba en pantalla. La respuesta

proporcionada en la mayoría de los casos corregía el error o modificaba el fichero HTML con el fin de alcanzar el resultado esperado.

Por ejemplo, en la Ilustración 82 - Un intento de ChatGPT., se puede observar que aun habiendo solicitado lo siguiente, no lo ha realizado de forma correcta:

Quiero que me hagas una página web en HTML y CSS con tres secciones: superior, central e inferior. La sección superior e inferior ocuparán el 15% de la altura, mientras que la sección central abarcará el 70%.

Después, en la sección superior, deseo que la dividas en dos partes. En la parte izquierda, por favor escribe "Foto", y en la parte derecha crea tres botones: "SP", "ENG" y "EUS".

Luego, en la sección central, habrá una división superior y otra inferior. En la parte superior incorporarás un iframe de Google, y en la parte inferior mostrarás "texto".

A su vez, quiero que colores las secciones según el arcoíris.

En la primera instrucción se le ha solicitado que haga una web en HTML y CSS, al no haberle especificado que haga dos ficheros separados, ha proporcionado un código único, que aun siendo funcional, no es la mejor opción de trabajo. A su vez se observa que la parte superior e inferior no tienen el mismo porcentaje, esto es debido a que no se le ha mencionado que quite los "paddings", es decir, el espacio entre un elemento y su borde, por lo que aunque sí que mantuvo las alturas (15%, 70% y 15%), no quedó proporcionado correctamente.



Ilustración 82 - Un intento de ChatGPT.

La segunda instrucción la completó satisfactoriamente, es decir, creó el texto a la izquierda y tres botones a la derecha. Si se observa Ilustración 83, se puede comprobar que también se han realizado correctamente las divisiones izquierda y derecha.

La tercera instrucción no la ha realizado correctamente, esto ha sido debido a que la sección central-inferior e inferior tenían el mismo nombre, lo que ha causado que la

parte central no se divida en dos partes y el texto acabe en la sección inferior. A su vez, aunque ha realizado el iframe, no se ve en la web y el enlace que ha insertado es más extenso que un simple “www.google.es”.

Por último, se puede observar que aunque sí que se han coloreado las secciones, la sección superior-izquierda y superior-derecha tienen el mismo color.

```
60 <body>
61   <div class="container">
62     <div class="top">
63       <div class="top-left">
64         Foto
65       </div>
66       <div class="top-right">
67         <button class="button">SP</button>
68         <button class="button">ENG</button>
69         <button class="button">EUS</button>
70       </div>
71     </div>
72     <div class="center">
73       <div class="iframe-container">
74         <iframe src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d...</iframe>
75       </div>
76       <div class="bottom">
77         Texto
78       </div>
79     </div>
80     <div class="bottom" style="background-color: #0000FF;"> <!-- Azul -->
81     Bottom
82   </div>
83 </div>
84 </body>
85 </html>
86
```

Ilustración 83 - Código de la web fallida por ChatGPT.

La siguiente sección a desarrollar sería la creación de la base de datos con el fin de almacenar y consultar la generación eléctrica, en primera instancia las respuestas eran genéricas y poco claras, pero una vez especificado que se quería una herramienta sencilla y visual ofrecía la opción de usar InfluxDB para la realización del proyecto.

Al igual que sucedió con Node-RED, se muestra el pantalla el proceso de instalación y configuración para que Node-RED e InfluxDB se comuniquen de manera exitosa. Cabe mencionar que no se hicieron preguntas del tipo “¿Qué es una API?” ya que se ha comprobado anteriormente que los conceptos teóricos los define de manera correcta en la mayoría de los casos.

8. COMPARATIVA PRACTICAS UPNA

En este capítulo, se abordará la posibilidad de poder usar la monitorización realizada en el proyecto como un añadido en las prácticas de las asignaturas asociadas a la energía fotovoltaica impartidas en la Universidad Pública de Navarra, como serían “Instalaciones Eólicas y Fovoltáicas” o “Sistemas de energías renovables”.

En ambas asignaturas, se pudo observar que las prácticas introductorias a los sistemas de energías renovables compartían similitudes. Es decir, los estudiantes debían llevar a cabo un proceso de dimensionamiento de una instalación fotovoltaica, considerando el consumo energético de dicha instalación. Para llevar a cabo este proceso, se empleaba una herramienta en línea llamada PVGIS.

PVGIS es un sistema de información geográfica desarrollado por el Centro Común de Investigación (JRC) de la Comisión Europea. Este sistema tiene como objetivo proporcionar datos relacionados con la radiación solar y se utiliza para estimar la producción de energía generada por paneles solares en función de la ubicación geográfica especificada.

Esta plataforma web ofrece la capacidad de generar datos, como la producción mensual de energía en un sistema fotovoltaico ubicado en el Aulario, optimizando tanto la inclinación como el ángulo azimutal de los paneles solares. Esto se puede ver en la Ilustración 84.

RENDIMIENTO DE UN SISTEMA FV CONECTADO A RED

Base de datos de radiación solar* PVGIS-SARAH2

Tecnología FV* Silicio cristalino

Potencia FV pico instalada [kWp]* 134,64

Pérdidas sistema [%]* 4

Opciones de montaje fijo

Posición de montaje* Sobre el tejado / integrado en el edificio

Inclinación [°]* 35 Optimizar inclinación

Azimut [°]* 0 Optimizar inclinación y azimut

Ilustración 84 - Rendimiento fotovoltaico PVGIS.

De esta manera, se obtendrían los datos presentados en la Ilustración 85, que corresponden a la producción de energía en cada mes. Estos valores podrían ser contrastados directamente con la información proporcionada por el inversor, específicamente en el apartado denominado "Energía generada desde el inicio del inversor". Bastaría con buscar este valor al comienzo y al final de cada mes, restar ambos valores y con ello se obtendría la producción de energía mensual.

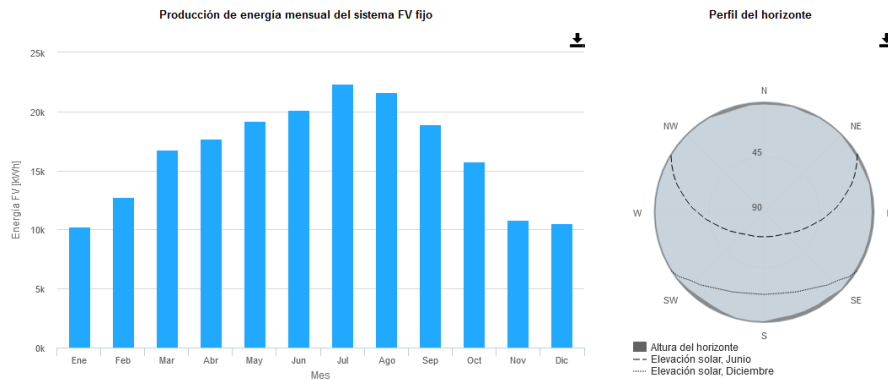


Ilustración 85 - Producción de energía mensual PVGIS.

Además, en el aulario se dispone de un sensor de irradiancia que posibilita el cálculo de parámetros como la temperatura de la célula (T_{cell}), el voltaje de circuito abierto (V_{oc}), la corriente de cortocircuito (I_{sc}) y la potencia máxima (P_{mpp}), utilizando la irradiancia promedio anual. Esto significa que, al conocer el modelo de los paneles fotovoltaicos instalados y disponer de dicho valor de irradiancia, sería viable realizar una estimación aproximada de la energía total generada.

Estos datos calculados podrían ser cotejados con los resultados obtenidos anteriormente mencionados. De esta forma, se podría determinar la discrepancia entre estos tres enfoques y evaluar su coherencia y precisión. Para ello se tendría que acceder a la sección de “Datos horarios de radiación” en PVGIS y hacer una descarga del fichero CSV entre las fechas seleccionadas indicando la inclinación y el ángulo azimutal, tal y como se muestra en la Ilustración 86.

DATOS HORARIOS DE RADIACIÓN

Base de datos de radiación solar* PVGIS-SARAH2

Año inicial:* 2020 Año final:* 2020

Tipo de montaje:*
 Fijo Eje vertical Eje inclinado Dos ejes

Inclinación [°] 34 Optimizar inclinación

Azimut [°] (-180-180) Optimize slope and azimuth

Potencia FV

Tecnología FV [...]

Potencia FV pico instalada [kWp] [...]

Pérdidas sistema [%] [...]

Componentes de la radiación

Ilustración 86 - Datos anuales de PVGIS.

Una vez que se haya calculado la energía generada utilizando PVGIS, la monitorización y las estimaciones del sensor de irradiancia, los datos podrían ser manipulados según se prefiera (siempre que exista un período de tiempo lo bastante extenso). Por ejemplo, se podrían utilizar para llevar a cabo análisis como la degradación del rendimiento de los paneles solares a lo largo del tiempo o el cálculo del ahorro económico a través de la disminución en la factura eléctrica y con ello ver la rentabilidad de la instalación.

9. CONCLUSIONES

En el contexto de este proyecto, se ha realizado una exploración informática que ha sido enriquecida por los conocimientos adquiridos en el Grado Superior de Administración de Sistemas Informáticos en Red. Dichos conocimientos previos sentaron las bases para abordar la creación de un sistema de monitorización para la Universidad Pública de Navarra. Sin embargo, la realización del proyecto no se limitó a la simple aplicación de conocimientos previos, sino que se llevó a cabo un estudio de las diversas herramientas de monitorización disponibles en la actualidad. Cada herramienta fue evaluada, sopesando su utilidad y potencial en relación con los objetivos del proyecto, garantizando así una solución final que cumpliera con los requisitos y expectativas establecidos.

Con respecto a los objetivos planteados al comienzo del proyecto, se puede concluir lo siguiente:

El componente esencial de este trabajo ha sido la implementación y desarrollo de la infraestructura de monitorización. Para lograrlo se ha seguido el sistema de Node-RED, InfluxDB y Grafana debido a que mantenía una interfaz amigable y de fácil comprensión, que tenía como propósito permitir que futuros interesados puedan utilizar, replicar o mejorar este proyecto.

La configuración de Node-RED, InfluxDB y Grafana, aunque en apariencia sencilla, se reveló como un proceso con múltiples posibilidades, algunas de las cuales se mostraron más efectivas y seguras que otras. Aunque se logró concluir el proyecto de manera satisfactoria, es válido considerar que existen enfoques potencialmente óptimos para llevar a cabo esta implementación, quizás reduciendo pasos intermedios u optimizando la utilización de recursos y seguridad.

En el desarrollo de la página web, aunque se partió de una base proporcionada por Alberto Berrueta y Leyre Catalán, se optó por reconstruir la interfaz para una mejor comprensión del código subyacente. Sin embargo, se mantuvo la misma apariencia visual con el fin de aprovechar la inversión previa tanto en publicidad como en desarrollo. Dejando así una interfaz tanto atractiva, como funcional y con una modificación posterior bastante sencilla, siempre y cuando se tenga un conocimiento básico del lenguaje HTML.

La inclusión de la herramienta ChatGPT en el proceso ha sido reveladora, debido a que se ha podido observar y comprender mejor el funcionamiento y potencial de dicha herramienta.

Sus aspectos positivos incluyen su capacidad para facilitar procesos de instalación y corrección de errores, la búsqueda simplificada y resumida de la información partiendo de una base nula o prácticamente inexistente, así como su comprensión y la ayuda que brinda en tareas relacionadas sobre todo con los entornos basados en comandos. Por ejemplo, el desarrollo de la base del entorno web, aun habiendo sido un proceso de reiteración y reformulación, se ha completado con éxito.

Sin embargo, se ha notado que la información puede estar desactualizada o desfasada debido a que solo proporciona información hasta Septiembre de 2021 y que el desarrollo en un entorno visual como Node-RED puede ser desafiante o en algún caso nulo. También se observaron respuestas repetitivas, respuestas complejas para un usuario con base inexistente o respuestas en algunos casos no funcionales, lo que llevaba a desechar la conversación mantenida y empezar con una nueva, siempre y cuando se resumiera lo realizado anteriormente. También se ha podido observar que al intentar obtener las fuentes de información de donde provienen las respuestas obtenidas de ChatGPT en la mayoría de casos suelen ser erróneas, debido a que se basan en patrones y ejemplos vistos en el entrenamiento de la herramienta, lo que incurre en un proceso estadístico que puede llegar a dar tanto respuestas coherentes como falacias inexistentes.

En última instancia, esta experiencia ha sido enriquecedora y fascinante. Ha permitido observar la instalación de un campus fotovoltaico, experimentar con secciones de pruebas y comprobar cómo los conceptos teóricos adquiridos en la formación académica se manifiestan en la realidad. La conclusión es que, si bien este proyecto se ha desarrollado con éxito tanto habiendo sido desarrollada por un estudiante de la Universidad Pública de Navarra o utilizando la herramienta ChatGPT, es importante señalar que en este último caso las respuestas en ciertos casos han sido influenciadas con el fin de seguir el camino específico de este proyecto. Dicho esto, el aprendizaje ha sido profundo y gratificante, y ha demostrado como las herramientas tecnológicas pueden ser efectivas para llevar a cabo proyectos prácticos y significativos en el campo de las energías renovables y la monitorización.

En resumen, este proyecto ha sido un viaje de descubrimiento y aprendizaje que ha permitido comprender mejor la integración de tecnologías en un contexto real. Ha confirmado la utilidad y las limitaciones de herramientas como ChatGPT, reforzando la importancia de la adaptación y la exploración continua en la implementación de soluciones innovadoras. Y a su vez dicho proyecto deja la puerta abierta a futuros recursos académicos tanto en la sección práctica como en la teórica.

10. LÍNEAS FUTURAS

A pesar de haber finalizado el proyecto con éxito, ciertos aspectos quedaron como ideas debido a la restricción temporal en el cual se llevó a cabo. En este contexto, se identifican varias áreas de mejora y posibles extensiones:

- Creación de un servicio Nagios: Un servicio Nagios se caracteriza por ser una herramienta de monitoreo. Su implementación permitiría una supervisión más exhaustiva y detallada, no solo brindando los valores del inversor, sino también notificando y localizando con precisión las áreas donde el servicio ha fallado. Su aplicación podría abarcar todos los edificios, brindando una vista integral de la estructura en una sola interfaz.
- Exportación de la base de datos a CSV: A su vez una de las propuestas más prometedoras que no se han podido concluir, sería la exportación de franjas temporales de los datos de InfluxDB, para tener un fichero Excel de producción entre dos fechas marcadas. Con el fin de hacer un estudio de generación y poder observar la degradación de las placas fotovoltaicas con el paso del tiempo.
- Integración de Node-Red con Bot de Telegram: La asociación de Node-Red con un Bot en Telegram habría habilitado la comunicación periódica de mensajes de estado o alertas de emergencia. Esta extensión habría involucrado el diseño y la implementación de un flujo de comunicación efectivo, así como la configuración adecuada del Bot.
- Comparación entre base de datos y valores teóricos de PVGIS: A pesar de ser una propuesta tentadora para la práctica de asignaturas, no se pudo llevar a cabo debido a las limitaciones de datos en PVGIS, ya que su información se encuentra limitada hasta el año 2021. Además, la falta de suficientes datos comparativos del inversor del aulario obstaculizó la realización de una comparación.
- Implementación de elementos observados en la universidad: La integración de dispositivos como el sensor de radiación en el proyecto habría requerido un análisis y configuración del elemento, con el fin de tener la radiación en el momento de la producción.
- Comparativa de generación y consumo entre departamentos: Esta comparativa no pudo llevarse a cabo debido a la falta de finalización de obras en la UPNA. La implementación de esta extensión habría involucrado el monitoreo y registro detallado de datos de generación y consumo en diferentes departamentos. Con el fin de ver que departamentos tienen un mayor gasto energético y si la localización entre edificios afecta a la hora de la generación.
- Exploración de la API desarrollada por el departamento informático: La API desarrollada por el departamento informático de la Universidad Pública de Navarra, a pesar de ofrecer un nivel adicional de seguridad, no se pudo implementar debido a las restricciones temporales. Su implementación requeriría un estudio exhaustivo, incluyendo la comprensión del lenguaje necesario, así como el desarrollo de visualizaciones funcionales en HTML utilizando los datos JSON proporcionados por la API.

En resumen, a pesar de los logros obtenidos en el proyecto, existen numerosas oportunidades de mejora y expansión que podrían haber sido abordadas en una etapa futura con más tiempo y recursos disponibles.

11. BIBLIOGRAFÍA Y REFERENCIAS

- [1] J. P. A. B. M. A. L. C. P. A. y. D. A. Javier Samanes, *Energía sostenible. Sin malos humos*, Pamplona, 2019.
- [2] E. Mugarza, «Diario de noticias: Alternativas del carbon.,» 18 Abril 2022. [En línea]. Available: <https://www.noticiasdenavarra.com/actualidad/2022/04/18/upna-plantea-alternativas-depender-carbon-2084742.html>. [Último acceso: 25 Agosto 2023].
- [3] Anonimo, «Diario de noticias: Linea 9 electrica.,» 21 Marzo 2019. [En línea]. Available: <https://www.noticiasdenavarra.com/pamplona/2019/03/21/linea-9-primera-100-electrica-2419419.html>. [Último acceso: 25 Agosto 2023].
- [4] J. Roche, «Delloite: Que es el desarrollo en espiral,» [En línea]. Available: <https://www2.deloitte.com/es/es/pages/technology/articles/que-es-el-desarrollo-en-espiral.html>. [Último acceso: 11 Agosto 2023].
- [5] S. Cope, “Youtube: MQTT Publish and Subscribe Using Node Red,” 17 Marzo 2018. [Online]. Available: <https://www.youtube.com/watch?v=XDrwgMSQrEY>. [Accessed 05 Julio 2023].
- [6] C. Mahler, “InfluxData: Node-RED e InfluxDB,” [Online]. Available: <https://www.influxdata.com/blog/iot-easy-node-red-influxdb/>. [Accessed 23 Junio 2023].
- [7] APM-DT, «Youtube: Install InfluxDB on Ubuntu Server - Influxdb oss-2.7,» 9 Abril 2023. [En línea]. Available: <https://www.youtube.com/watch?v=yiiwhxKNfxw>. [Último acceso: 5 Julio 2023].
- [8] J. Schaller, 5 Julio 2023. [En línea]. Available: <https://unix.stackexchange.com/questions/537546/e-unmet-dependencies-try-apt-fix-broken-install-with-no-packages-or-speci>.
- [9] E. Glass, 22 Junio 2023. [En línea]. Available: <https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-ubuntu-20-04-es>.
- [10] InfluxDB, «InfluxDB: Instalación 2.7 Linux,» [En línea]. Available: <https://docs.influxdata.com/influxdb/v2.7/install/?t=Linux>. [Último acceso: 5 Julio 2023].
- [11] R. Morales, «Ticarte: Tipos de conexiones de red en VirtualBox y VMware,» 16 Junio 2015. [En línea]. Available: <https://www.ticarte.com/contenido/tipos-de-conexiones-de-red-en-virtualbox-y-vmware>. [Último acceso: 22 Junio 2023].

- [12] Grafana, “Grafana: Dashboard anything. Observe everything.,” [Online]. Available: <https://grafana.com/grafana/?pg=oss-graf&plcmt=hero-btn-1>. [Accessed 26 Junio 2023].
- [13] I. & I. E. Solutions, “Youtube: How to install Node-RED on Ubuntu 20.04 with username and password authentication??,” 16 Octubre 2021. [Online]. Available: <https://www.youtube.com/watch?v=Y2ttfN9mOyQ>. [Accessed 23 Junio 2023].
- [14] Grafana, «Grafana: Setup,» [En línea]. Available: <https://grafana.com/docs/grafana/latest/setup-grafana/installation/debian/>. [Último acceso: 26 Junio 2023].
- [15] F. Code, “Youtube: NVM | Node version Manager (Multiples versiones de Node),” 20 Mayo 2020. [Online]. Available: https://www.youtube.com/watch?v=M_asn_Vr3Xs. [Accessed 23 Junio 2023].
- [16] J. Ellingwood, «DigitalOcean: How to use systemctl to manage systemd services and units es,» 2 Diciembre 2020. [En línea]. Available: <https://www.digitalocean.com/community/tutorials/how-to-use-systemctl-to-manage-systemd-services-and-units-es>. [Último acceso: 5 Julio 2023].
- [17] InfluxDB, «InfluxDB: Install via Linux.,» [En línea]. Available: <https://docs.influxdata.com/influxdb/v2.0/install/?t=Linux>. [Último acceso: 28 Junio 2023].
- [18] InfluxDB, «InfluxDB: InfluxDB como servicio de Systemd,» [En línea]. Available: <https://docs.influxdata.com/influxdb/v2.7/install/?t=Linux#install-influxdb-as-a-service-with-systemd>. [Último acceso: 26 Junio 2023].
- [19] SuperHouseTv, “Youtube: Datalogging with MQTT, Node-RED, InfluxDB, and Grafana,” 20 Diciembre 2020. [Online]. Available: https://www.youtube.com/watch?v=ffg3_1AgtyA. [Accessed 28 Junio 2023].
- [20] V. Lab, “Youtube: Manually Parsing HTML in Node.js,” 24 Octubre 2023. [Online]. Available: <https://www.youtube.com/watch?v=PozG7Sva270>. [Accessed 28 Julio 2023].
- [21] V. Jha, “Youtube: Grafana 8 InfluxDB 2 and Telegraf | Create First Grafana Dashboard,” 23 Septiembre 2021. [Online]. Available: <https://www.youtube.com/watch?v=NGv7LeCaPtc>. [Accessed 28 Junio 2023].
- [22] Ionos, «Ionos: Configuración SSH ubuntu.,» 5 Enero 2023. [En línea]. Available: <https://www.ionos.es/digitalguide/servidores/configuracion/ubuntu-ssh/>. [Último acceso: 5 Julio 2023].

- [23] S. Cope, "Youtube: Node-Red- Convert JSON to CSV," 10 Julio 2023. [Online]. Available: <https://www.youtube.com/watch?v=zCV1bLuDxo0>. [Accessed 27 Julio 2023].
- [24] Solar, Dómotica, «Youtube: NodeRed: Escritura en InfluxDb y Grafana. Introducción,» 23 Enero 2021. [En línea]. Available: <https://www.youtube.com/watch?v=EfcKAHMWCjM>. [Último acceso: 26 Junio 2023].
- [25] InfluxDB, "InfluxData: Unviarsity Catalog," [Online]. Available: <https://university.influxdata.com/catalog/>. [Accessed 22 Junio 2023].
- [26] S. Cope, "Steve blog," 27 Julio 2023. [Online]. Available: <http://stevesnoderedguide.com/storing-iot-data-sql-database>. [Accessed 11 Agosto 2023].
- [27] Vishal, "Stackoverflow: Errores de InfluxDB," 15 Diciembre 2017. [Online]. Available: <https://stackoverflow.com/questions/47833787/how-to-uninstall-influxdb-with-all-its-data-and-configuration-files>. [Accessed 28 Junio 2023].
- [28] Anonimo, «Ionos: Dirección IP en Linux,» 23 Enero 2023. [En línea]. Available: <https://www.ionos.es/digitalguide/hosting/cuestiones-tecnicas/direccion-ip-en-linux/>. [Último acceso: 22 Junio 2023].
- [29] EEA, «EEA: CO2 intensidad de emision.,» EEA, 12 Junio 2023. [En línea]. Available: https://www.eea.europa.eu/data-and-maps/daviz/co2-emission-intensity-13/#tab-chart_4. [Último acceso: 28 Agosto 2023].

12. ANEXOS

12.1. Configuración VMware Workstation Pro 17	77
12.2. Instalación Ubuntu	83
12.3. Ubuntu server.....	92
12.4. Instalar ssh.....	98
12.5. Utilización de Putty a través de SSH	100
12.7. Instalación Node-RED	104
12.8. Configuración Node-RED	107
12.9. Instalación InfluxDB	111
12.10. Error de InfluxDB	115
12.11. Instalar Grafana	116
12.12. Error en la instalación de Grafana	119
12.13. Instalación de Apache2	120
12.15. Datos del inversor Ingeteam.....	123
12.16. Conversación con ChatGPT.....	130

12.1. Configuración VMware Workstation Pro 17

VMware Workstation Pro es un software de virtualización de escritorios que permite crear y ejecutar máquinas virtuales en cualquier ordenador o servidor. Dicho software está desarrollado por VMware. Es comúnmente utilizada por los departamentos de sistemas en las empresas, con el fin de poder administrar y configurar de manera sencilla los servicios de las empresas. Debido a que en un solo ordenador físico se pueden tener múltiples máquinas virtuales y en cada una de ellas tener distintos sistemas operativos con distintos servicios.

Se ha seleccionado este software en particular debido a su facilidad de uso (ya sea a la hora de configurar la interfaz de red, la instalación de las guest-addition, duplicar o exportar máquinas virtuales) e integración con el entorno VMware vSphere. El cual permitiría integrar de forma sencilla este proyecto a la red de la Universidad Pública de Navarra.

Nada más abrir el programa se encontrará en el panel frontal y seleccionará "Create a New Virtual Machine". Tal y como se muestra en la Ilustración 87.

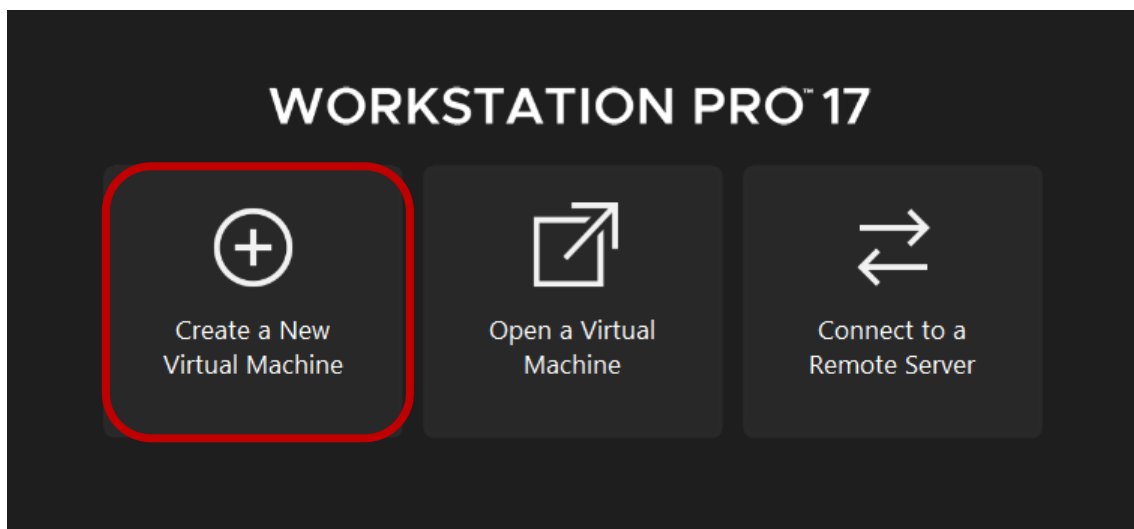


Ilustración 87 – Panel frontal de VMware Workstation Pro 17.

Debido a que no se tiene ninguna necesidad de configuración de la máquina virtual se seleccionará la opción "Typical", ya que con unos sencillos clics se tendrá configurada de forma bastante óptima el entorno de trabajo, tal y como se muestra en la Ilustración 88.

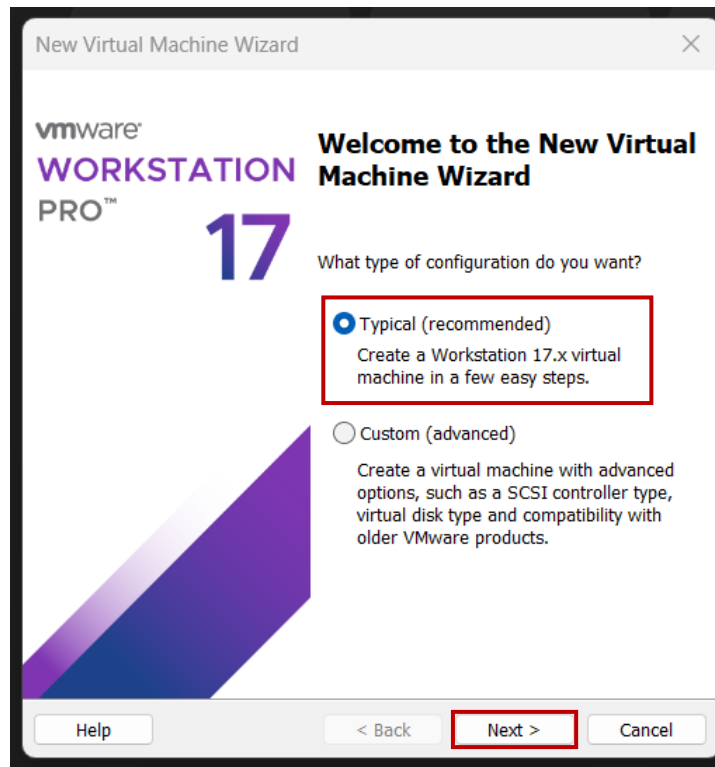


Ilustración 88 – Panel de nueva máquina virtual VMware Workstation Pro 17.

Con el fin de tener un mayor control de la máquina virtual, y debido a que aún no se ha descargado la imagen del sistema operativo a utilizar, se seleccionará la opción de posterior selección del sistema operativo “I will install the operating system later”, como se aprecia en la Ilustración 89.

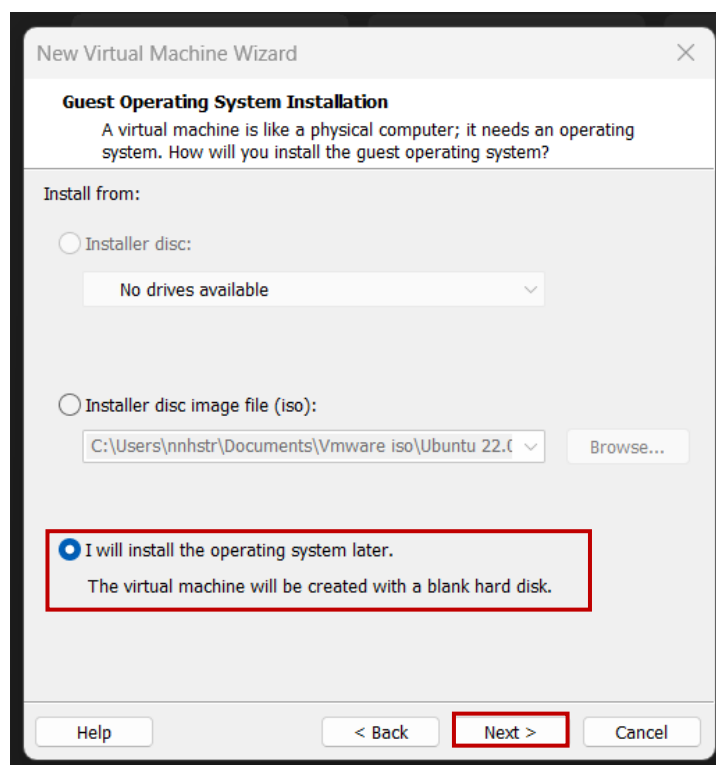


Ilustración 89 – Selección del sistema operativo VMware Workstation Pro 17.

Debido a que no se ha seleccionado ningún sistema operativo, el programa ofrece la opción de seleccionar el sistema que se va a utilizar. Dicha selección nos dará una configuración base, la cual permitirá un uso fluido de dicha máquina, como se muestra en la Ilustración 90. Siempre se ha de tener en cuenta los requisitos mínimos del sistema operativo para evitar problemas de rendimiento.

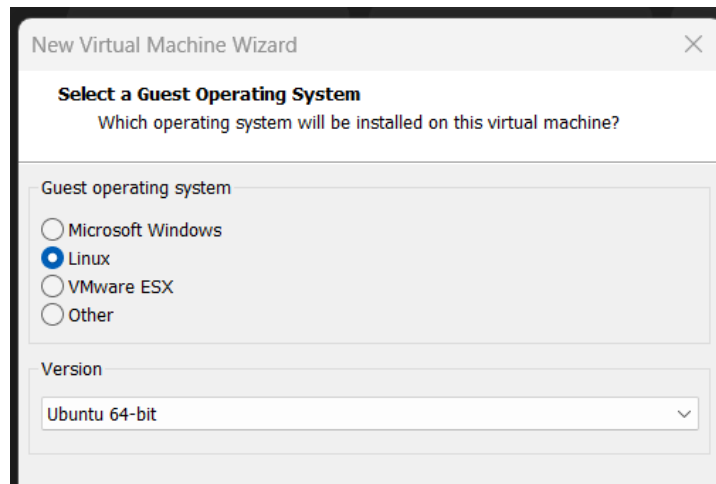


Ilustración 90 – Selección de tipo de sistema operativo VMware Workstation Pro 17.

Posteriormente, se nombrará la máquina virtual con el objetivo de facilitar su posterior identificación y se seleccionará la ubicación de los ficheros de configuración y almacenamiento, como se muestra en la Ilustración 91.

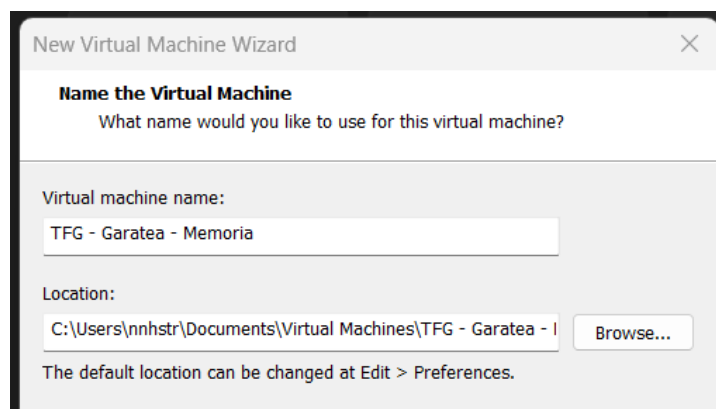


Ilustración 91 – Selección de nombre y localización de archivos VMware Workstation Pro 17.

Por último, se tendrá que seleccionar la forma de guardado. Para ello, se tendrán dos opciones "Store virtual disk as a single file" y "Split virtual disk into multiple files" la diferencia radica en como se organiza y almacena el archivo de disco virtual de una máquina virtual. Al almacenar el disco virtual como un único archivo, se simplifica la administración y respaldo, pero puede generar un archivo grande.

Por otro lado, al dividir el disco virtual en múltiples archivos, se facilita la gestión del tamaño y ubicación de almacenamiento, aunque puede generar mayor complejidad en la administración y fragmentación. En este caso y debido a que no se va a usar una gran

capacidad de almacenamiento, se utilizará la opción de dividir el disco en múltiples ficheros “Split virtual disk into multiple files”, como se muestra en la Ilustración 92.

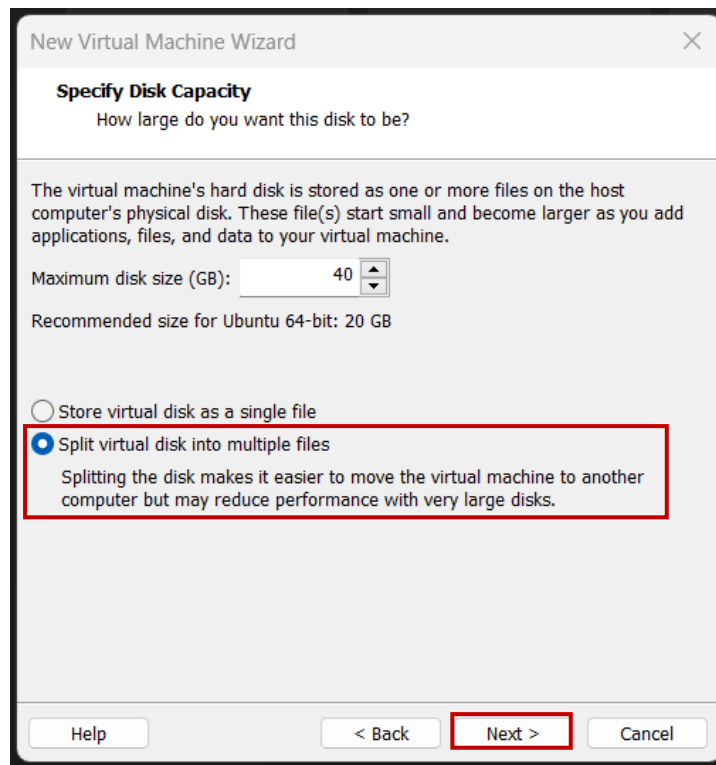


Ilustración 92 – Selección de tipo de ficheros VMware Workstation Pro 17.

Se mostrará un resumen de toda la configuración realizada anteriormente. En caso de querer personalizarla o modificar algún parámetro, se utilizará la opción de “Customize Hardware...”, como se observa en la Ilustración 93.

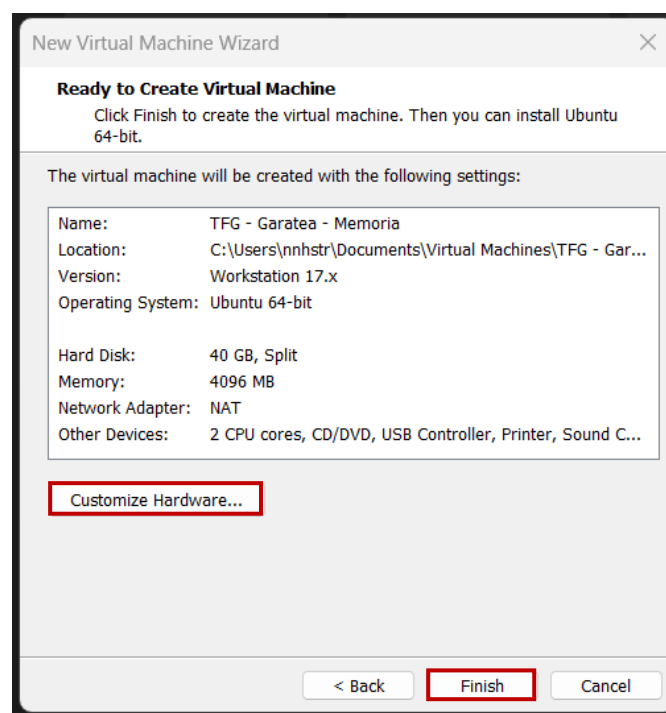


Ilustración 93 – Resumen de la configuración VMware Workstation Pro 17.

Para tener una mayor agilidad a la hora de instalación y configuración se ha decidido aumentar el número de núcleos de procesador de 2 a 4, como se observa en la Ilustración 94.

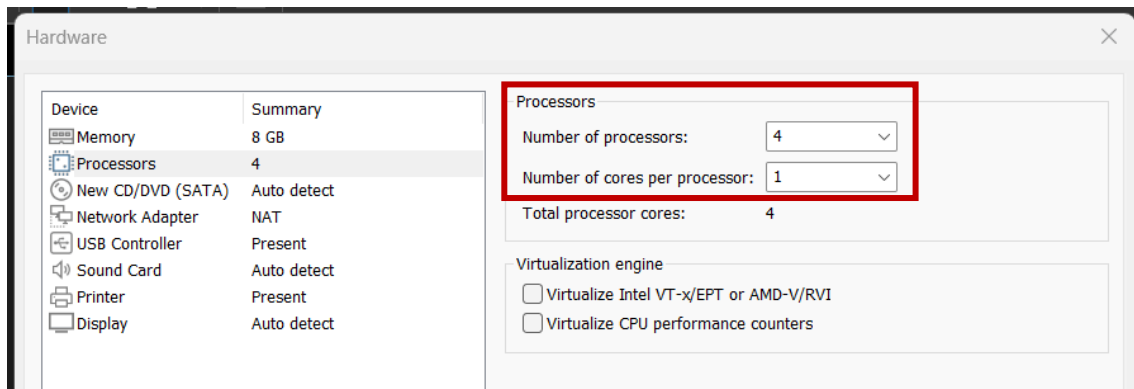


Ilustración 94 – Modificación de núcleos VMware Workstation Pro 17.

Con ello concluiría la configuración del programa VMware Workstation Pro y la creación de la máquina virtual, al igual que en la Ilustración 95.

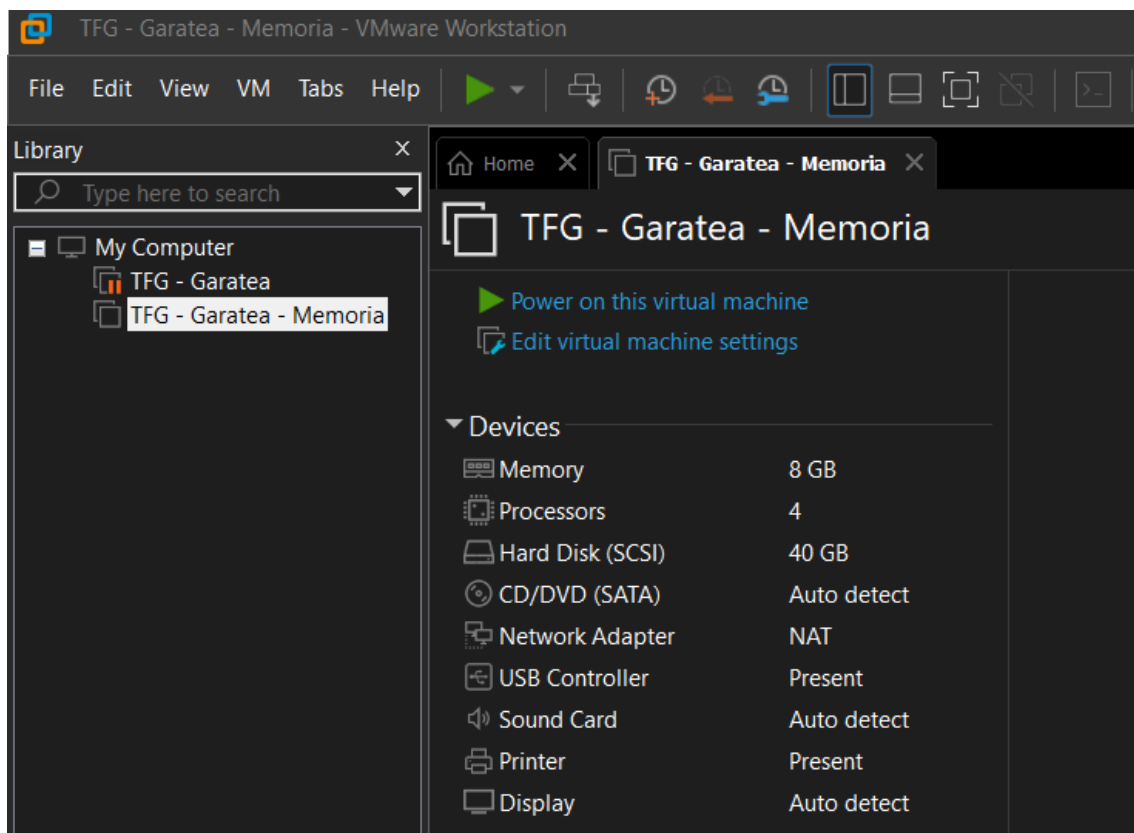


Ilustración 95 – Visualización final de la máquina virtual VMware Workstation Pro 17.

En caso de querer apagar el ordenador o pausar la máquina virtual, simplemente se tendrá que cerrar la pestaña del programa VMware. Al clicar en la “x” aparecerá una ventana emergente que proporcionará distintas opciones [10], como se observa en la Ilustración 96.

- **Suspend:** Esta opción guarda el estado actual de la máquina virtual y la pone en un modo de suspensión. Todos los procesos y aplicaciones en ejecución se detienen y se guarda el estado de la memoria en el disco. Al reanudar la máquina virtual, se restaura el estado en el que se encontraba antes de la suspensión.
- **Power off:** Esta opción fuerza el cierre de la máquina virtual sin realizar un proceso de apagado ordenado. Es similar a desconectar la alimentación o seleccionar el botón de apagar.
- **Run in background:** La máquina virtual se ejecuta en segundo plano. Generalmente, se puede acceder a ella nuevamente a través del software de virtualización o mediante la apertura de su ventana o panel de control correspondiente. También se puede realizar tareas y acciones en la máquina virtual, como interactuar con las aplicaciones y archivos que se ejecutan en ella.
- **Cancel:** Esta opción permite cancelar la operación y mantener la máquina virtual funcionando.

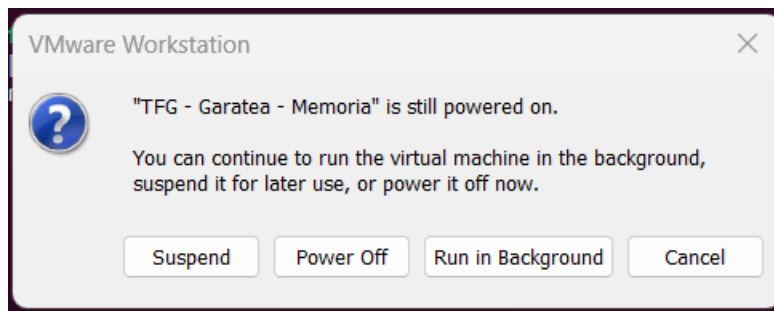


Ilustración 96 – Menú de cierre de la máquina virtual VMware Workstation Pro 17.

12.2. [Instalación Ubuntu](#)

Para hacer funcionar la máquina virtual, se necesita la imagen de disco del sistema operativo pertinente, para ello y como norma general, se puede obtener desde la página oficial del desarrollador. En este caso, debido a que se ha seleccionado Ubuntu como base del proyecto, se accederá a www.ubuntu.com.

En este caso, se pueden observar dos secciones “Ubuntu Desktop” que está diseñado para ofrecer una interfaz gráfica de usuario amigable y funcionalidades completas para el usuario final. Incluye aplicaciones y herramientas preinstaladas para tareas comunes. Por otro lado, “Ubuntu Server” se enfoca en proporcionar una plataforma estable y segura para servidores y entornos de red. Carece de una interfaz gráfica por defecto y se gestiona principalmente a través de la línea de comandos. Ubuntu Server se optimiza para el rendimiento y la escalabilidad. En este caso, se seleccionará “Ubuntu Desktop” por facilitar la instalación y la configuración del servicio, similar a la Ilustración 97.

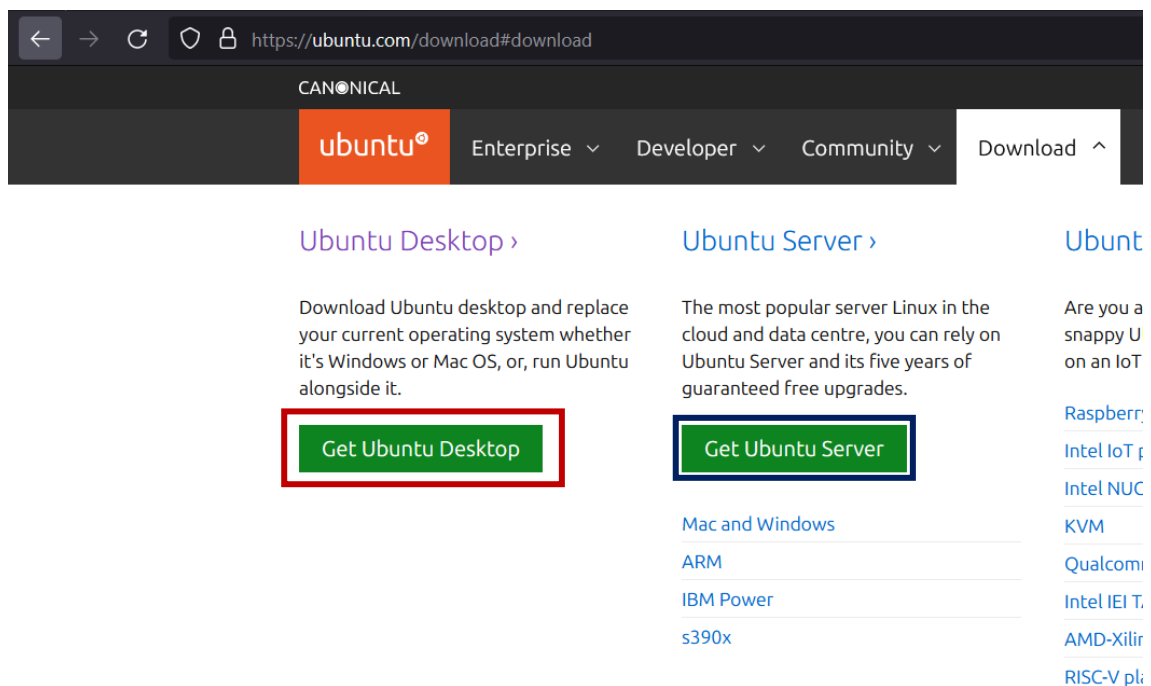
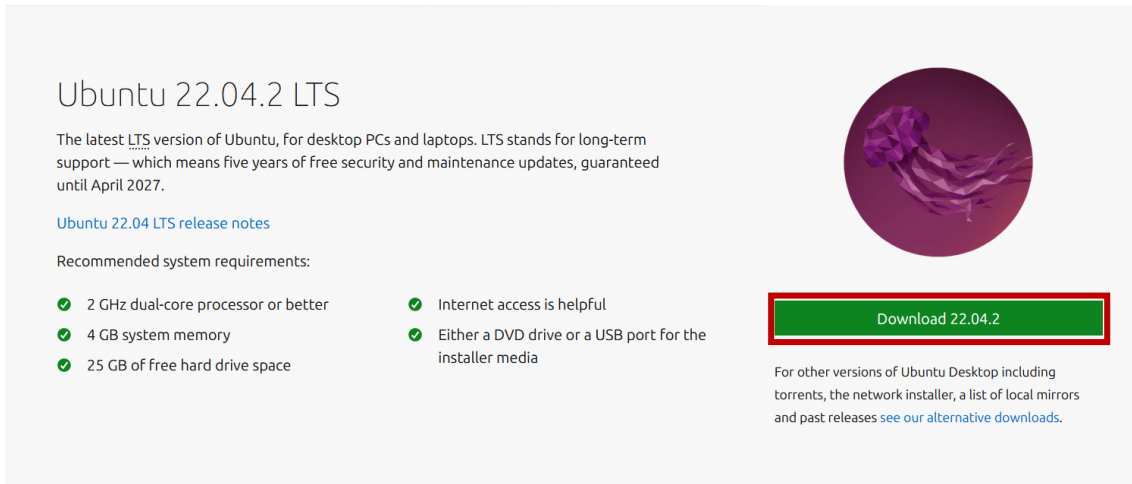


Ilustración 97 – Descarga de Ubuntu.

Para este proyecto se ha seleccionado la versión LTS (Long-term support), al igual que la Ilustración 98, la cual en principio se centra en la estabilidad y duración del soporte a largo plazo, por lo que garantizará actualizaciones y mantenimiento hasta 5 años después de su salida.



Ubuntu 22.04.2 LTS

The latest LTS version of Ubuntu, for desktop PCs and laptops. LTS stands for long-term support — which means five years of free security and maintenance updates, guaranteed until April 2027.

[Ubuntu 22.04 LTS release notes](#)

Recommended system requirements:

- ✓ 2 GHz dual-core processor or better
- ✓ 4 GB system memory
- ✓ 25 GB of free hard drive space
- ✓ Internet access is helpful
- ✓ Either a DVD drive or a USB port for the installer media

[Download 22.04.2](#)

For other versions of Ubuntu Desktop including torrents, the network installer, a list of local mirrors and past releases [see our alternative downloads](#).

Ilustración 98 - Ubuntu LTS.

Para poder utilizar el sistema operativo en el entorno de VMware se deberá dirigir a la máquina virtual en cuestión, y seleccionar el botón de edición “Edit virtual machine settings”, al igual que en la Ilustración 99.

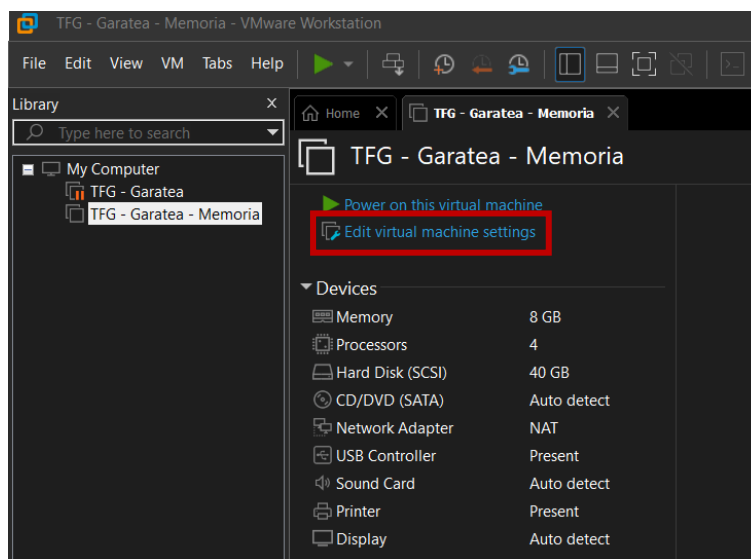


Ilustración 99 – Visualización final de la máquina virtual VMware Workstation Pro 17.

En la ventana desplegada, deberá seleccionarse la opción “CD/DVD (SATA)”, en columna derecha se clicará “Use ISO Image file:” y posteriormente “Browse...”, al igual que en la Ilustración 100. Esta acción abrirá una ventana aparte, en la cual se deberá seleccionar la imagen descargada anteriormente.

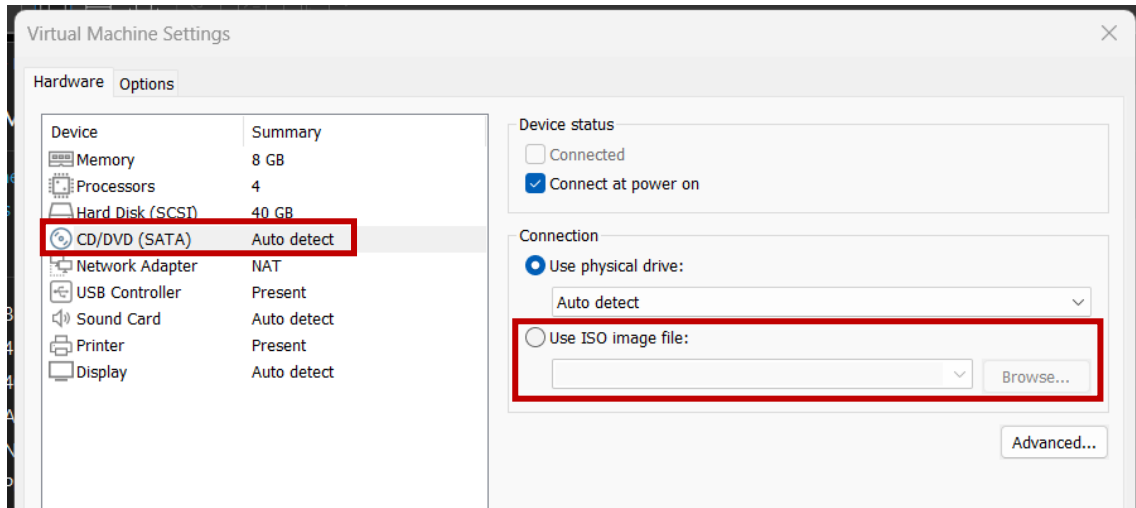


Ilustración 100 – Selección de la imagen del sistema VMware Workstation Pro 17.

Obteniendo así la Ilustración 101.

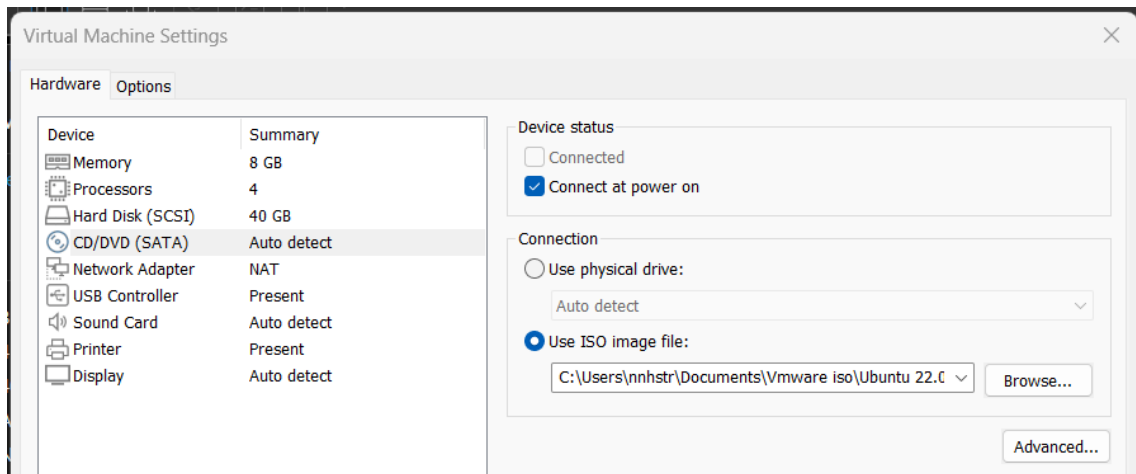


Ilustración 101 – Imagen del sistema seleccionada VMware Workstation Pro 17.

Se aplicarán los cambios en el botón “Ok” situado en la sección inferior y se encenderá la máquina virtual en “Power on this virtual machine”, como en la Ilustración 102.

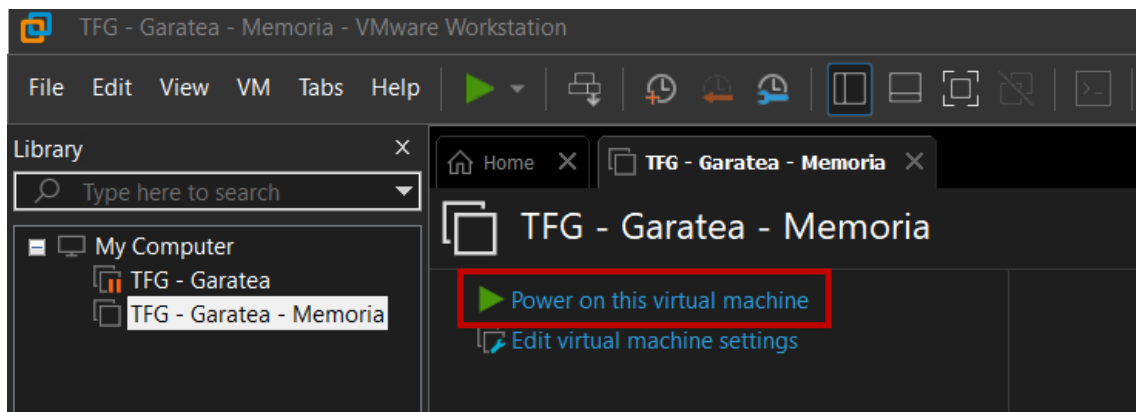


Ilustración 102 – Encendido de la máquina virtual VMware Workstation Pro 17.

Una vez encendida la máquina virtual, se mostrará el menú Grub de la iso seleccionada. Debido a que la intención es instalar el sistema se seleccionará la primera opción “Try or Install Ubuntu”, al igual que en la Ilustración 103.

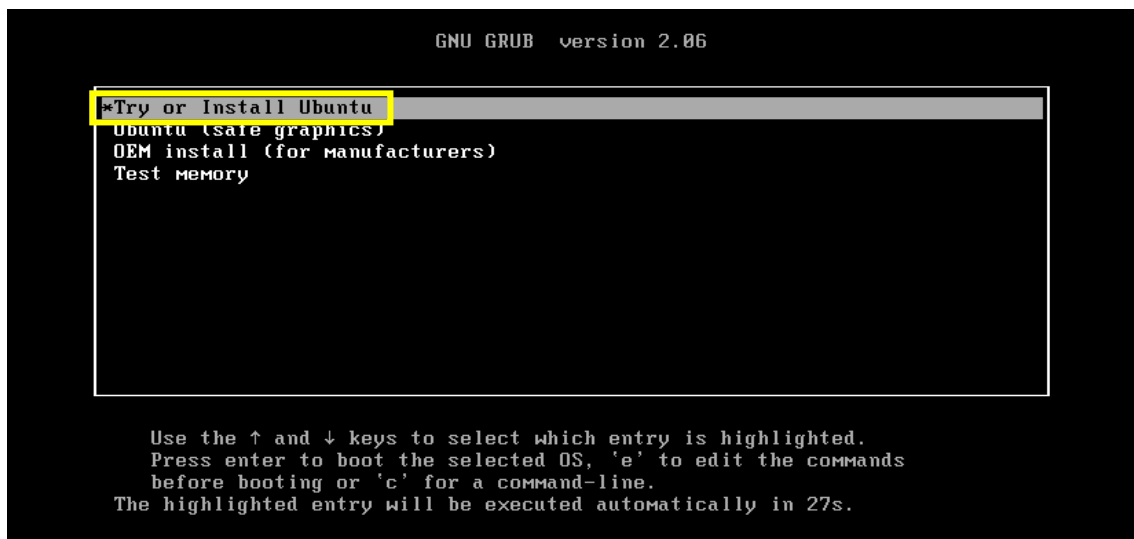


Ilustración 103 – Grub de la máquina virtual Ubuntu.

Una vez cargada la opción seleccionada, se seleccionará el idioma en el cual se quiere tener el sistema operativo, en este caso “Español” y se seleccionará la opción de “Instalar Ubuntu”, al igual que en la Ilustración 104.



Ilustración 104 – Selección de idioma y opción de Ubuntu.

En este apartado se seleccionará la distribución del teclado que se va a usar, en el caso del portátil utilizado se tiene una distribución de Español – España, al igual que en la Ilustración 105.

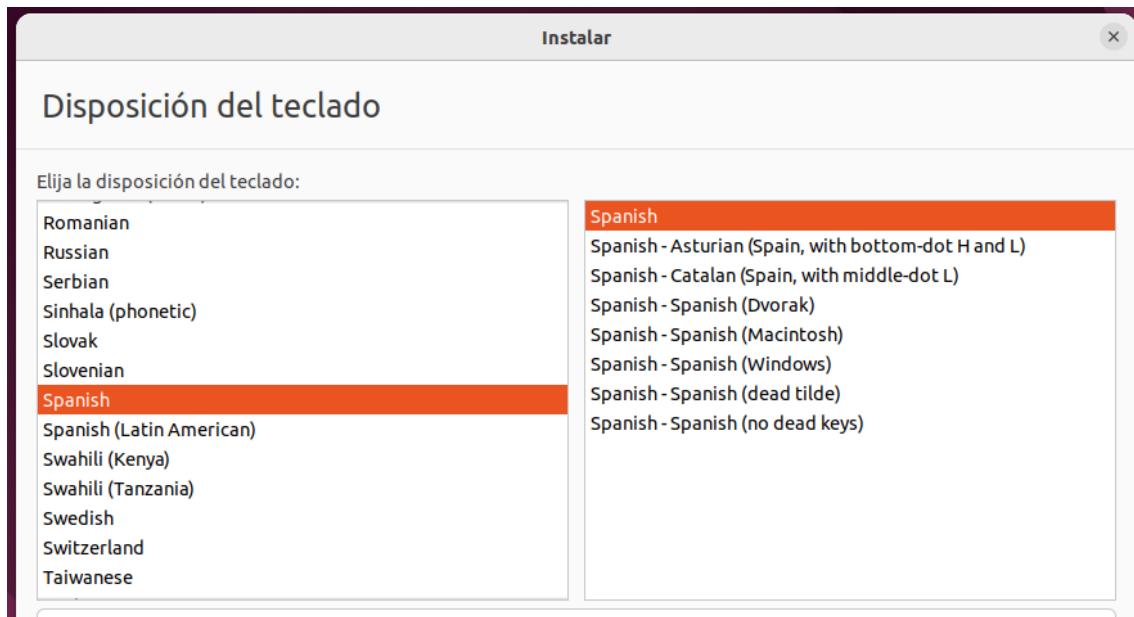


Ilustración 105 – Selección de distribución de teclado Ubuntu.

Debido a que se va a realizar una instalación limpia y controlada, se seleccionará la opción de “Instalación mínima” y la opción de “Descargar actualizaciones al instalar Ubuntu”. En caso de que se quiera utilizar el servidor para cualquier otra tarea multimedia o para usuarios nuevos en este entorno es preferible seleccionar la opción de “Instalación normal”, al igual que en la Ilustración 106.

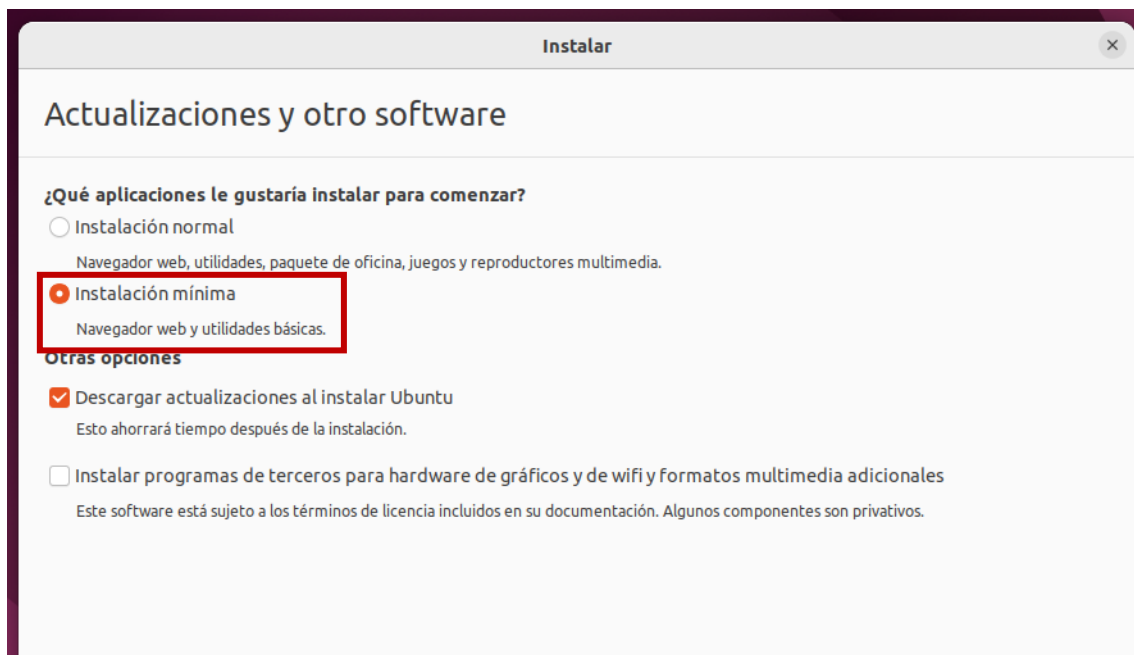


Ilustración 106 – Opciones de la instalación Ubuntu.

Debido a que es una máquina virtual, el formateo del disco duro no es necesario, pero en caso de estar utilizando un disco duro físico es aconsejable seleccionar la opción de “Borrar disco e instalar Ubuntu”. Si lo que se quiere es tener la opción de multiboot o una partición para el almacenamiento de datos se tendría que seleccionar “Más opciones”, al igual que en la Ilustración 107.

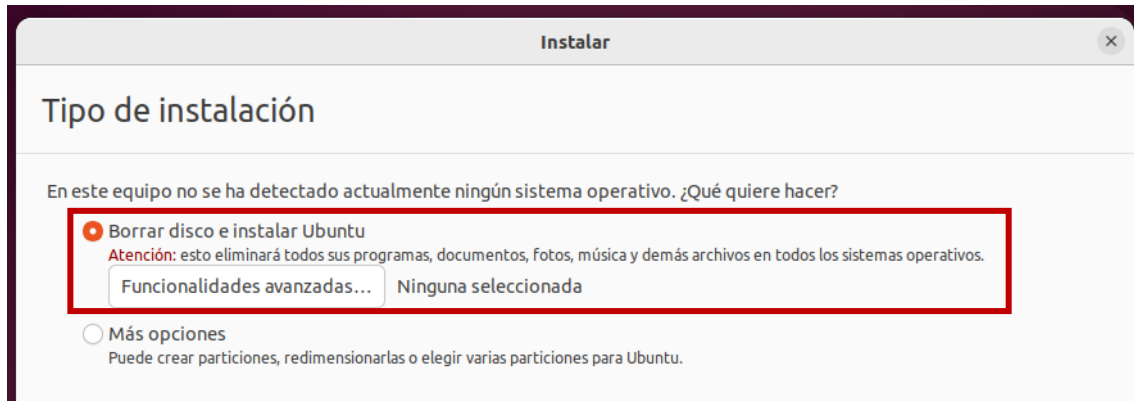


Ilustración 107 – Particiones de disco duro Ubuntu.

Al clicar “Instalar ahora” se mostrará una ventana emergente con un resumen con las particiones a realizar con su respectiva configuración, se comprueba que todo este correctamente y se selecciona “Continuar”, como en la Ilustración 108.

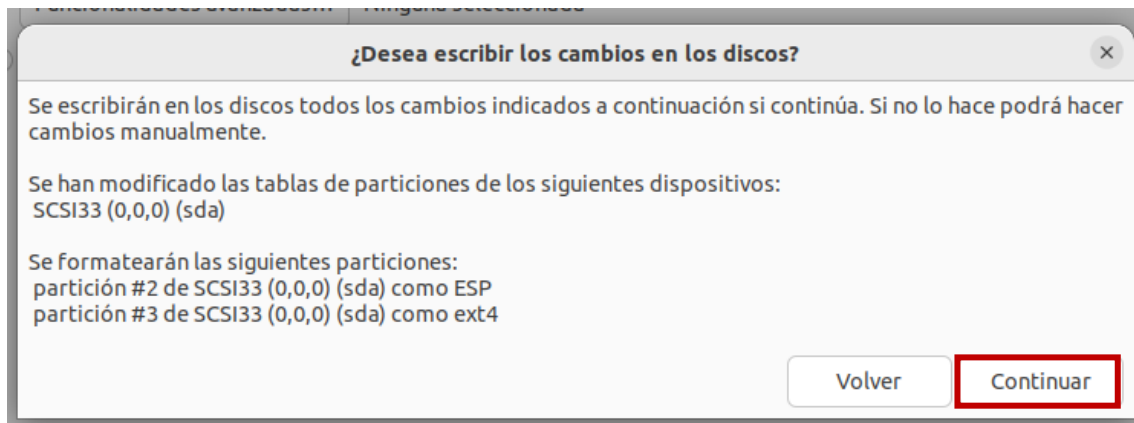


Ilustración 108 – Resumen de particiones Ubuntu.

Una vez realizado el dimensionamiento de discos, se tendrá que seleccionar la franja horaria en el cual se va a instalar el servidor, como en la Ilustración 109.

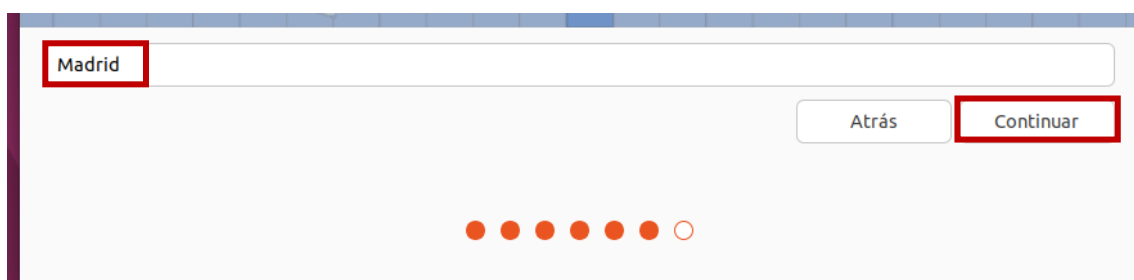


Ilustración 109 – Selección de franja horaria Ubuntu.

Por último, se creará el usuario administrador y se seleccionará el nombre del equipo, en este caso se ha decidido el nombre de usuario **fvarrosadia** con contraseña **Qwerty1**. Actualmente es una máquina de pruebas, pero en caso de querer publicarla posteriormente, sería recomendable cambiar la contraseña. Para tener agilidad a la hora de trabajar se seleccionará la opción de “Iniciar sesión automáticamente”, como se puede observar en la Ilustración 110.

Ilustración 110 – Creación de usuario Ubuntu.

Una vez completada la creación del usuario, comenzará la instalación, como en la Ilustración 111.

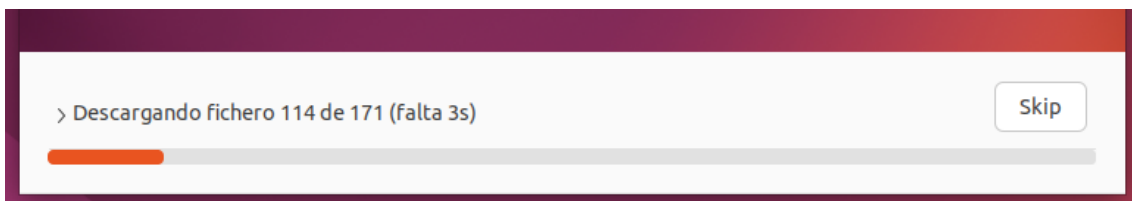


Ilustración 111 – Proceso de instalación Ubuntu.

A continuación, con la instalación realizada se reiniciará la máquina virtual y se quitará la iso seleccionada, debido a que no se va a utilizar más, como se ve en la Ilustración 112.

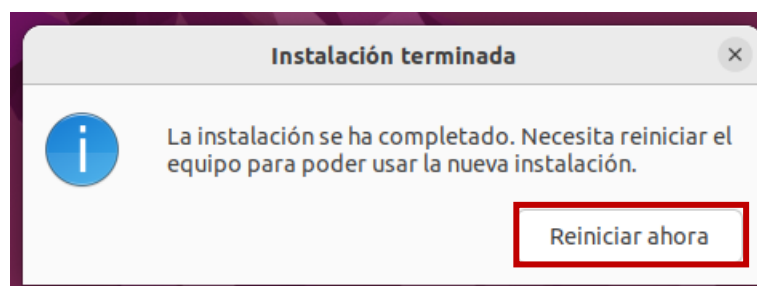


Ilustración 112 – Instalación completada Ubuntu.

Una vez completada la instalación se asegurará que la máquina está completamente actualizada. Para ello, se tendrá que seleccionar el botón de cuadrícula situado en la sección inferior izquierda de la pantalla. Como se puede observar en la Ilustración 113.

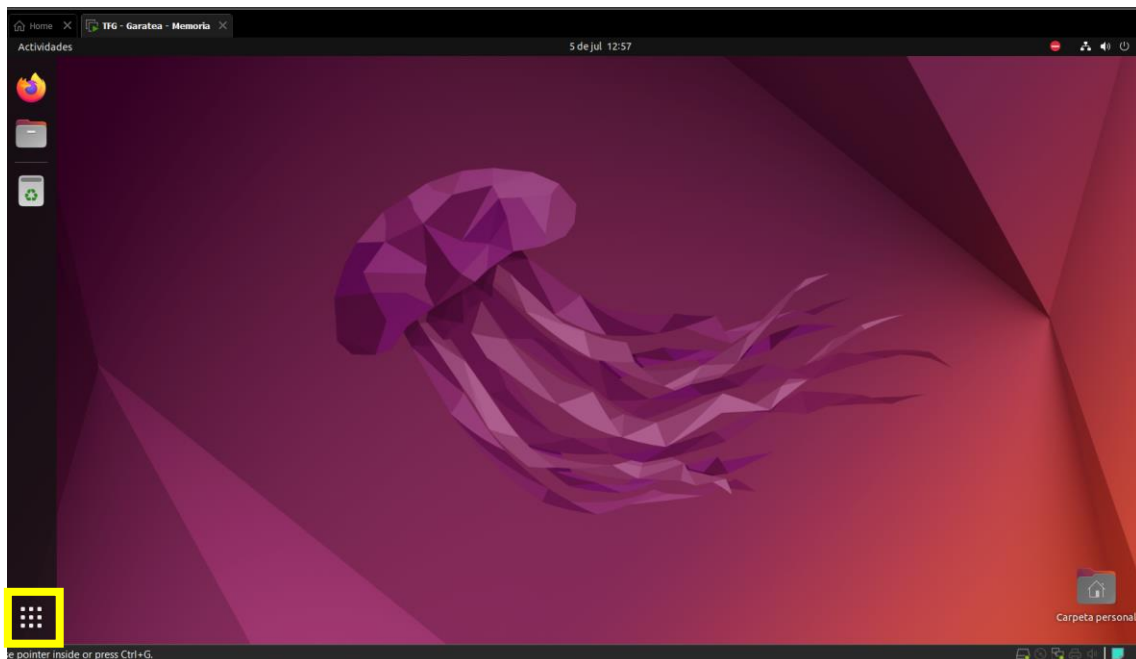


Ilustración 113 – Escritorio de Ubuntu.

Aparecerá en pantalla el menú de inicio y se seleccionará la aplicación “Terminal”, la cual es la ventana de comandos de Ubuntu, situado en la Ilustración 114.

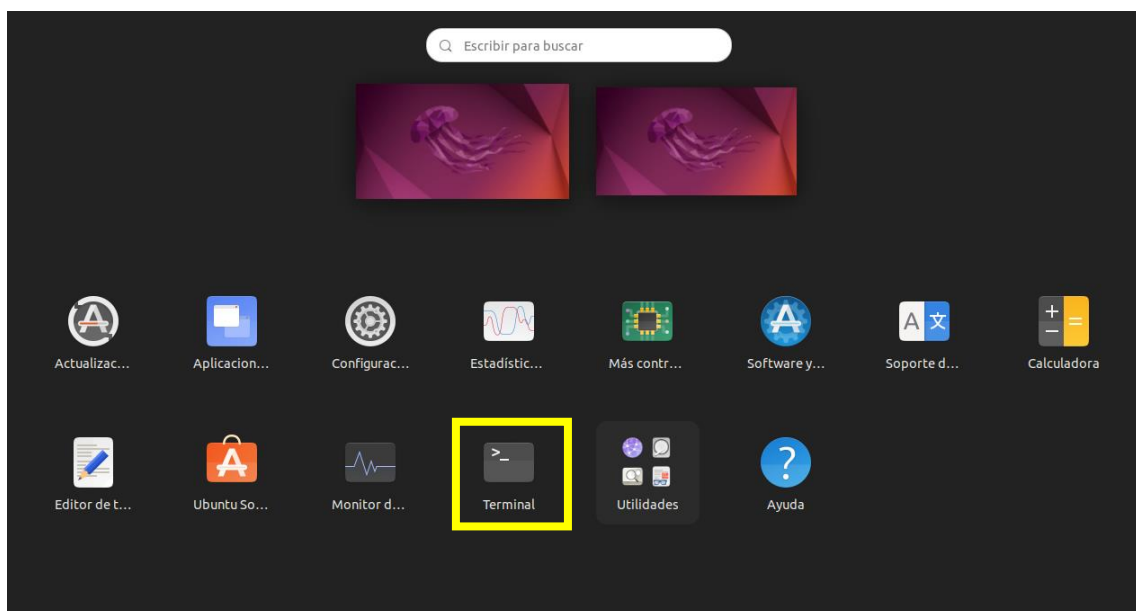
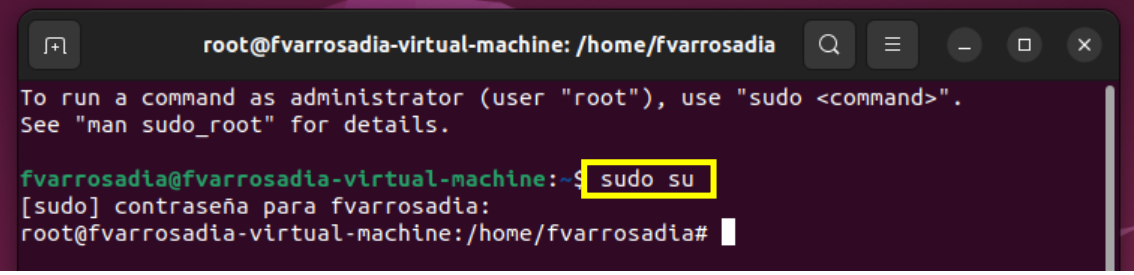


Ilustración 114 – Menu de inicio Ubuntu.

Una vez abierto el programa “terminal” se escribirán los siguientes comandos:

```
sudo su
```

Dicho comando pone el “terminal” en modo administrador, de esta manera se tendrán permisos para poder instalar cualquier paquete, al igual que en la Ilustración 115.



```
root@fvarrosadia-virtual-machine: /home/fvarrosadia
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

fvarrosadia@fvarrosadia-virtual-machine:~$ sudo su
[sudo] contraseña para fvarrosadia:
root@fvarrosadia-virtual-machine:~#
```

Ilustración 115 – comando sudo su Ubuntu.

Una vez en el modo super usuario se escribirán los siguientes comandos, en el cual el primero busca las distintas actualizaciones de los distintos servicios que se tendrán instalados en el ordenador y el segundo da la orden de instalación de las actualizaciones:

```
apt-get update
```

```
apt-get upgrade
```

12.3. Ubuntu server

Para realizar la instalación de Ubuntu server se tendrán que seguir los pasos de Instalación Ubuntu hasta la Ilustración 103. Pero teniendo en cuenta que en la Ilustración 97 se seleccionará Ubuntu server.

Una vez seguidos dichos pasos se tendrá que seleccionar el idioma de la máquina virtual, recordar que al ser Ubuntu server el ratón no está disponible. Por lo que únicamente se utilizará el teclado, en este caso se seleccionará la opción de “Español”, al igual que en la Ilustración 116.

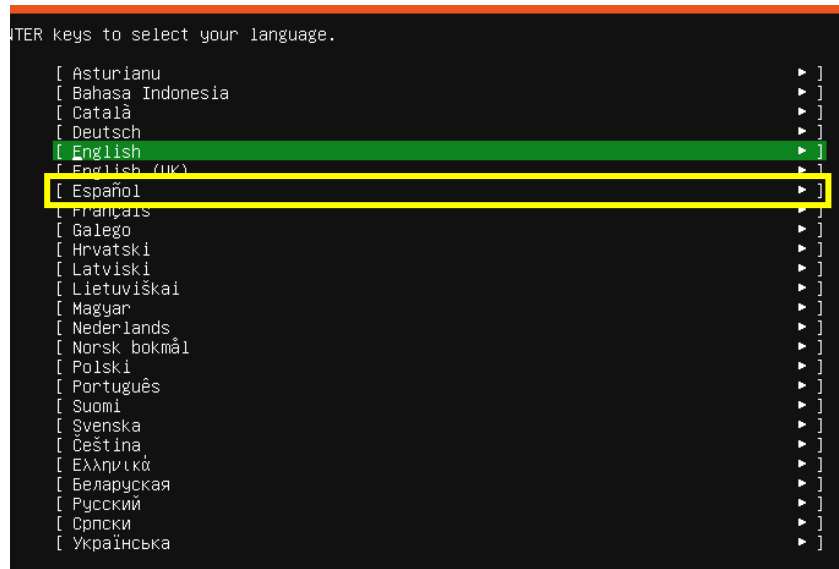


Ilustración 116 – Selección de idioma de Ubuntu server.

Posteriormente, se mostrará un mensaje de actualización del sistema operativo. En este caso, ya que la iso descargada es la última versión de la página oficial se obviará dicho mensaje, al igual que en la Ilustración 117.

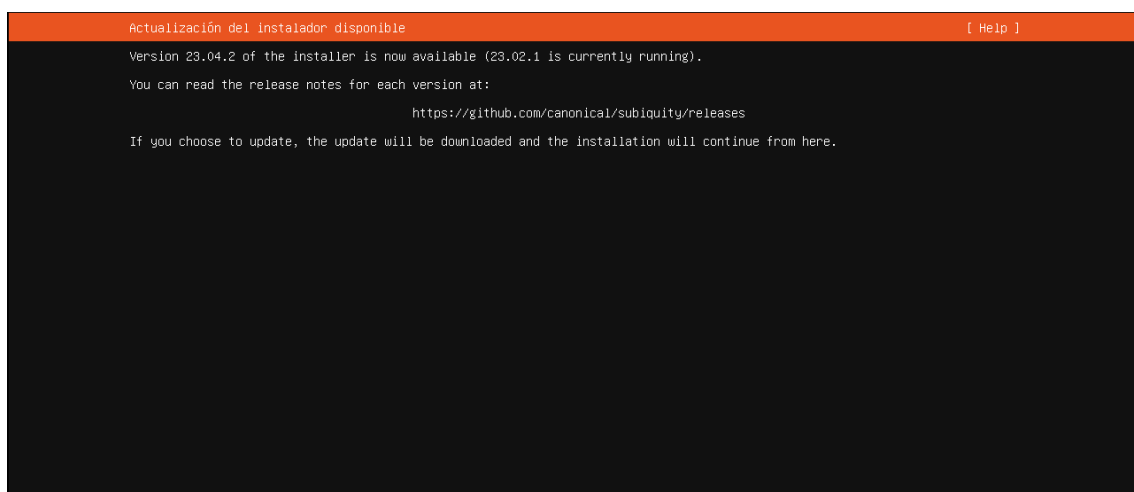


Ilustración 117 – Ventana update Ubuntu server.

En este apartado se seleccionará la distribución del teclado que se va a usar, en el caso del portátil utilizado se tiene una distribución de Español – España, como se aprecia en la Ilustración 118.

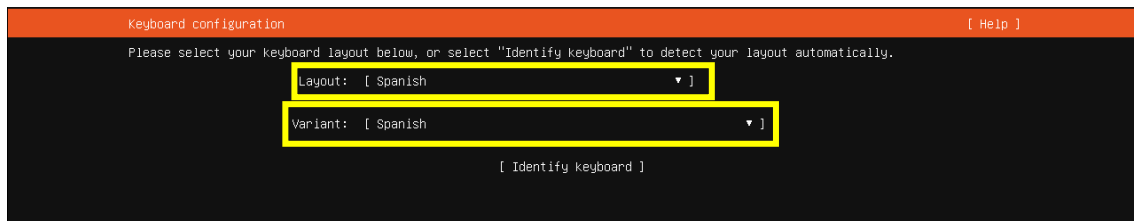


Ilustración 118 – Selección de idioma Ubuntu server.

Debido a que se va a realizar una instalación limpia y controlada, se seleccionará la opción de “Ubuntu server (minimized)”. En caso de que se quiera utilizar el servidor para cualquier otra tarea multimedia o para usuarios nuevos, en este entorno es preferible seleccionar la opción de “Search for third-party drivers”, al igual que en la Ilustración 119.

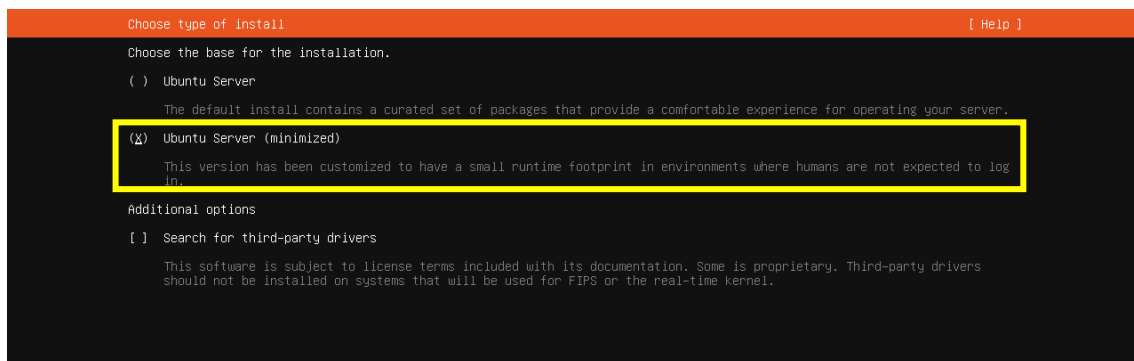


Ilustración 119 – Opciones de instalación Ubuntu server.

En caso de ser requerida la configuración del puerto ethernet se realizará desde esta ventana, pero al tratarse de una máquina de pruebas en un entorno controlado y red cerrada se dejará que el DHCP se encargue de asignar una @ip adecuada, como en la Ilustración 120.

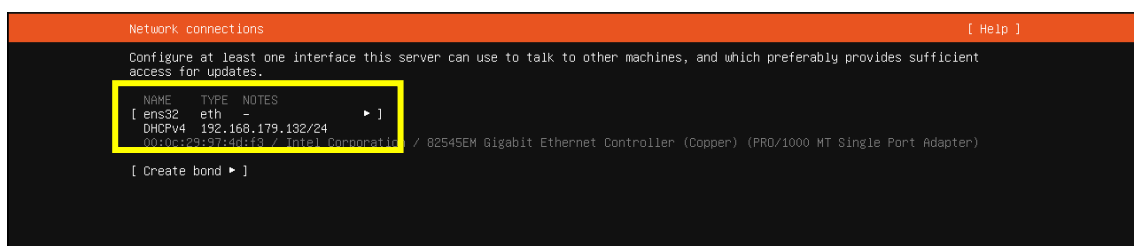


Ilustración 120 – Configuración de red Ubuntu server.

Al estar diseñado como servidor, es común que en las redes en las que va a estar integrado tengan un proxy, por lo que se brinda la posibilidad de configurarlo, en este caso se obviará dicha opción. Pero de ser requerido en un futuro, el departamento de informática será el encargado de configurarlo, como se aprecia en la Ilustración 121.

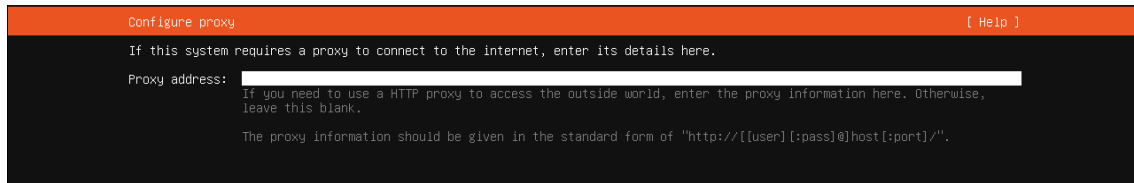


Ilustración 121 – Selección de proxy Ubuntu server.

A su vez, dicha instalación brinda la posibilidad de seleccionar un mirror en concreto, pero se dejará el default debido a que en la red actual funciona correctamente, como esta en la Ilustración 122.

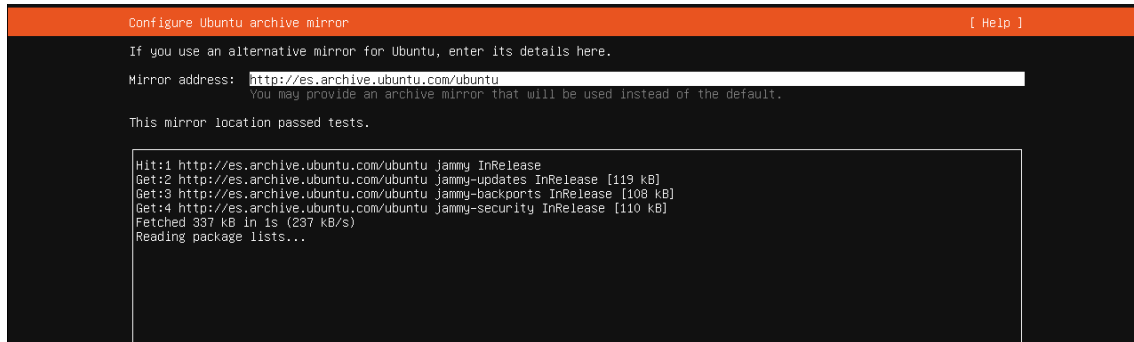


Ilustración 122 – Selección de mirror en Ubuntu server.

En esta sección se configurará la distribución de discos duros. Ya que es una máquina virtual se trata como un disco duro nuevo y en blanco, por lo que se seleccionará la opción de “Use an entire disk” junto con “Set up this disk as an LVM group”. Esto significa configurar este disco como un grupo LVM (Logical Volume Manager). LVM es una tecnología de gestión de volúmenes lógicos que permite combinar varios discos físicos en un único grupo de almacenamiento lógico. Al configurar el disco como un grupo LVM, se podrá utilizar para crear volúmenes lógicos, que a su vez se pueden particionar y utilizar como unidades de almacenamiento flexibles y escalables en el sistema operativo, al igual que en la Ilustración 123.

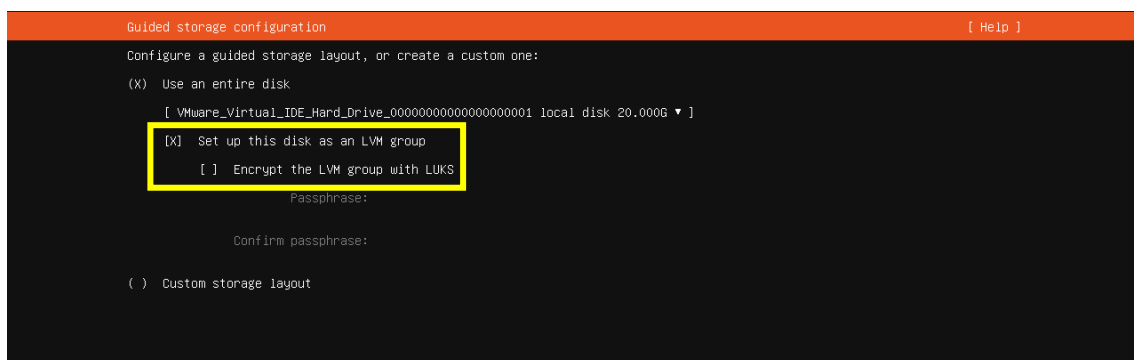


Ilustración 123 – Particiones de disco duro en Ubuntu server.

Al clicar “Instalar ahora” se mostrará una ventana con un resumen con las particiones a realizar con su respectiva configuración, como se ve en la Ilustración 124. Se comprueba que todo este correctamente y se selecciona “Continuar”.

```

Storage configuration [ Help ]

FILE SYSTEM SUMMARY

MOUNT POINT  SIZE  TYPE  DEVICE TYPE
[ /          10.000G new ext4 new LVM logical volume ▶ ]
[ /boot     1.771G new ext4 new partition of local disk ▶ ]

AVAILABLE DEVICES

DEVICE  TYPE  SIZE
[ ubuntu-vg (new)  LVM volume group  18.222G ▶ ]
free space  8.222G ▶

[ Create software RAID (md) ▶ ]
[ Create volume group (LVM) ▶ ]

USED DEVICES

DEVICE  TYPE  SIZE
[ ubuntu-vg (new)  LVM volume group  18.222G ▶ ]
ubuntu-lv  new, to be formatted as ext4, mounted at /  10.000G ▶

[ VMware_Virtual_IDE_Hard_Drive_00000000000000000001  local disk  20.000G ▶ ]
partition 1  new, BIOS grub spacer  1.000M ▶
partition 2  new, to be formatted as ext4, mounted at /boot  1.771G ▶
partition 3  new, PV of LVM volume group ubuntu-vg  18.225G ▶

```

Ilustración 124 – Resumen de las particiones en Ubuntu server.

Aparecerá una ventana con un botón de confirmación debido a que se va a borrar completamente el disco duro. Por lo que se seleccionará “Continue”, al igual que en la Ilustración 125, ya que el disco duro virtual está en blanco.



Ilustración 125 – Confirmación de formateo Ubuntu server.

Por último, se creará el usuario administrador y se seleccionará el nombre del equipo, en este caso se ha decidido el nombre de usuario **fvarrosadia** con contraseña **Qwerty1**. Debido a que actualmente es una máquina de pruebas, en caso de querer publicarla posteriormente sería recomendable cambiar la contraseña. Con el fin de tener agilidad a la hora de trabajar se seleccionará la opción de “Iniciar sesión automáticamente”, al igual que en la Ilustración 126.

```

Profile setup [ Help ]

Enter the username and password you will use to log in to the system. You can configure SSH access on the next screen but a
password is still needed for sudo.

Your name: fvarrosadia

Your server's name: fvarrosadiauona
The name it uses when it talks to other computers.

Pick a username: fvarrosadia

Choose a password: *****

Confirm your password: *****

```

Ilustración 126 – Creación de usuario Ubuntu server.

Dicha instalación a su vez nos brinda la posibilidad de pasar a Ubuntu pro, una distribución diseñada específicamente para entornos empresariales, las cuales requieren una mayor seguridad y configuración adicional. En este caso, se evitará por el momento, por lo que se seleccionará “Skip for now”, como en la Ilustración 127.

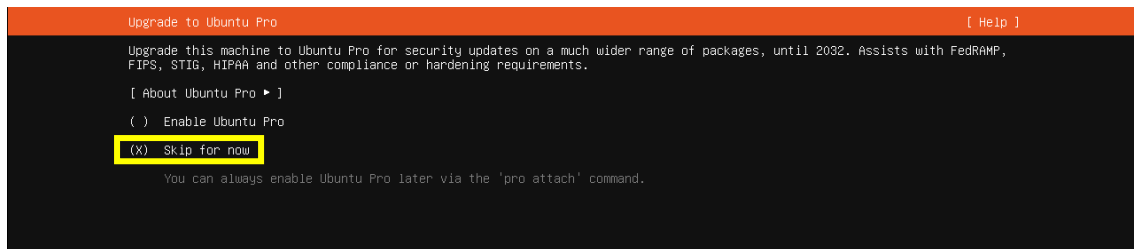


Ilustración 127 – Ubuntu pro Ubuntu server.

Por último, se brinda la opción de poder instalar OpenSSH al servidor, lo que ofrecerá la opción de poder conectarse por Putty de forma remota a controlar el servidor. En este caso, se realizará la instalación del paquete con el fin de facilitar el uso de la máquina virtual, como en la Ilustración 128, aunque no sería necesario.

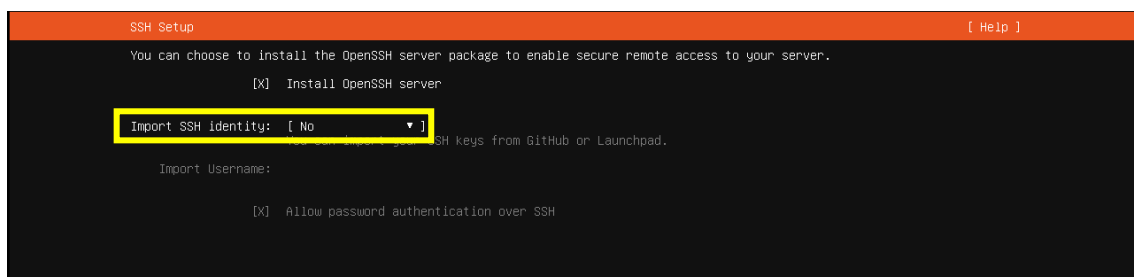


Ilustración 128 – Instalación de SSH en Ubuntu server.

Una vez finalizada la configuración del SSH, comenzará la instalación, al igual que en la Ilustración 129.

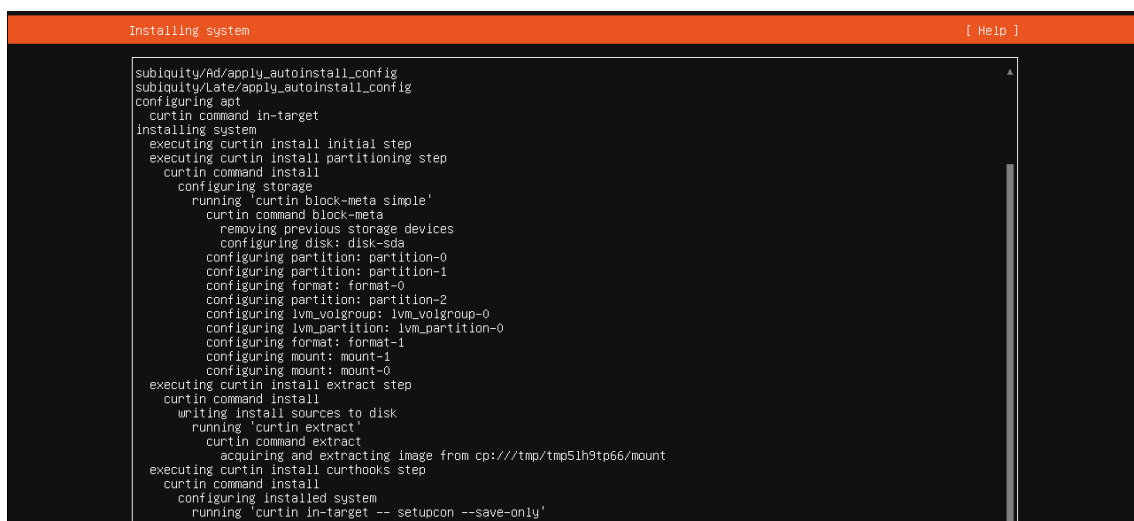


Ilustración 129 – Instalación de ubuntu server.

Una vez realizada la instalación se reiniciará la máquina virtual, como en la Ilustración 130, y se quitará la iso seleccionada, ya que no se va a utilizar más.


```

Install complete! [ Help ]

curtin command install
writing install sources to disk
running 'curtin extract'
curtin command extract
acquiring and extracting image from cp:///tmp/tmp5lh9tp66/mount
executing curtin install curthooks step
curtin command install
configuring installed system
running 'curtin in-target -- setupcon --save-only'
curtin command in-target
running 'curtin curthooks'
curtin command curthooks
configuring apt
configuring apt
installing missing packages
configuring lsscsi service
configuring raid (mdadm) service
installing kernel
setting up swap
apply networking config
writing etc/fstab
configuring multipath
updating packages on target system
configuring pollinate user-agent on target
updating initramfs configuration
configuring target system bootloader
installing grub to target devices
final system configuration
configuring cloud-init
calculating extra packages to install
installing openssh-server
retrieving openssh-server
curtin command system-install
unpacking openssh-server
curtin command system-install
downloading and installing security updates
curtin command in-target
restoring apt configuration
curtin command in-target
subiquity/Late/run

[ Reboot Now ]

```

Ilustración 130 – Instalación completada de Ubuntu server.

Una vez reiniciada la máquina virtual ya se podrá utilizar correctamente, al igual que en la Ilustración 131.

```

Ubuntu 22.04.2 LTS fvarrosadiaupna tty1
fvarrosadiaupna login: fvarrosadia
Password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-76-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

fvarrosadia@fvarrosadiaupna:~$ sudo su
[sudo] password for fvarrosadia:
root@fvarrosadiaupna:/home/fvarrosadia# _

```

Ilustración 131 – Entorno visual de ubuntu server.

Sería recomendable actualizar el sistema tal y como se muestra en la Ilustración 115 y posteriores.

12.4. Instalar ssh

OpenSSH es una herramienta de acceso de remoto seguro a través de conexiones SSH [11]. Dicho acceso remoto seguro permite iniciar sesiones remotas de servidores de manera cifrada, lo que garantiza la privacidad e integridad de los datos transmitidos.

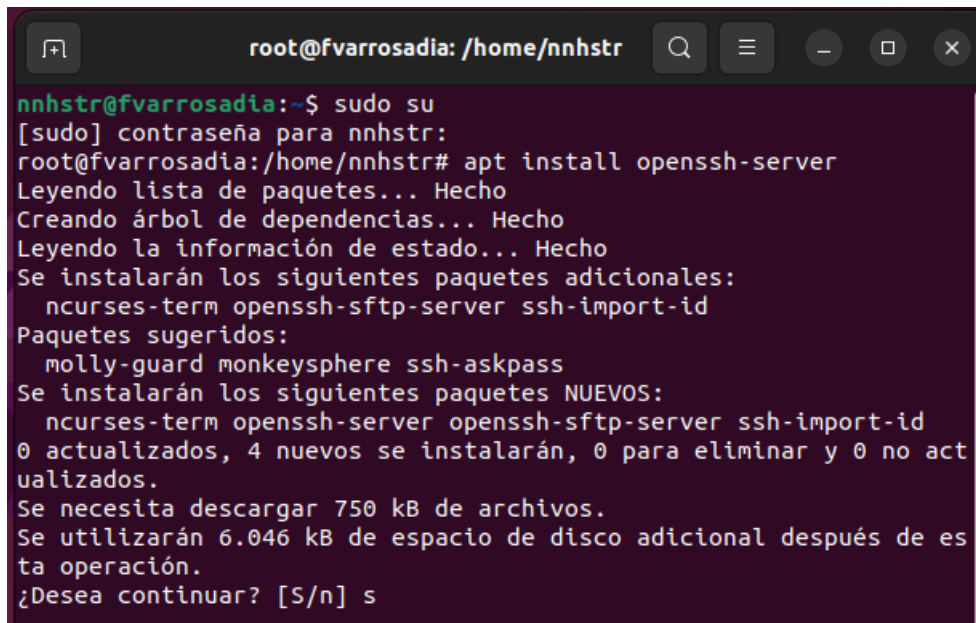
Por lo que, para su instalación se abrirá el “terminal” y se entrará en modo super usuario con el siguiente comando:

```
sudo su
```

Una vez situado como super usuario se deberán introducir los siguientes comandos:

```
apt install openssh-server
```

Al igual que en la Ilustración 132.



```
root@fvarrosadia: /home/nnhstr
nnhstr@fvarrosadia:~$ sudo su
[sudo] contraseña para nnhstr:
root@fvarrosadia:~/home/nnhstr# apt install openssh-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
 ncurses-term openssh-sftp-server ssh-import-id
Paquetes sugeridos:
 molly-guard monkeysphere ssh-askpass
Se instalarán los siguientes paquetes NUEVOS:
 ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 0 no act
ualizados.
Se necesita descargar 750 kB de archivos.
Se utilizarán 6.046 kB de espacio de disco adicional después de es
ta operación.
¿Desea continuar? [S/n] s
```

Ilustración 132 – Instalación de SSH server en Ubuntu.

Con el fin de saber si se ha instalado correctamente y está funcionando, tal y como se ve en la Ilustración 133, se hará uso del comando:

```
systemctl status ssh
```

```

root@fvarrosadia:/home/nnhstr# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-07-05 14:58:42 CEST; 54s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 8792 (sshd)
     Tasks: 1 (limit: 9430)
    Memory: 1.7M
       CPU: 40ms
   CGroup: /system.slice/ssh.service
           └─8792 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

jul 05 14:58:42 fvarrosadia systemd[1]: Starting OpenBSD Secure Shell server...
jul 05 14:58:42 fvarrosadia sshd[8792]: Server listening on 0.0.0.0 port 22.

```

Ilustración 133 – Comprobación de estado SSH server en Ubuntu.

En caso de no estar activo se tendrá que hacer uso de los siguientes comandos:

```
sudo systemctl enable ssh
```

```
sudo systemctl start ssh
```

Si por alguna razón llegara a dar error a la hora de activarlo, se deberá agregar la regla en Ubuntu para activar el puerto 22, con el objetivo de que lleguen las comunicaciones correctamente, para ello y al igual que en la Ilustración 134, se hará uso del comando:

```
ufw allow ssh
```

```

root@fvarrosadia:/home/nnhstr# ufw allow ssh
Reglas actualizadas
Reglas actualizadas (v6)
root@fvarrosadia:/home/nnhstr#

```

Ilustración 134 – Configuración de puerto ssh en Ubuntu.

En caso de querer hacer una mayor configuración o ver exactamente como está realizada la configuración por defecto, se tendrá que hacer uso del comando mostrado en la Ilustración 135.

```
nano /etc/ssh/sshd_config
```

```

root@fvarrosadia:/home/nnhstr# nano /etc/ssh/sshd_config
root@fvarrosadia:/home/nnhstr#

```

Ilustración 135 – Configuración SSH en Ubuntu.

Si se hubiera realizado cualquier configuración adicional se deberá reiniciar el servicio, como en la Ilustración 136, con el comando:

```
service ssh restart
```

```

root@fvarrosadia:/home/nnhstr# nano /etc/ssh/sshd_config
root@fvarrosadia:/home/nnhstr# service ssh restart
root@fvarrosadia:/home/nnhstr# █

```

Ilustración 136 – Reinicio del servicio SSH en Ubuntu.

12.5. Utilización de Putty a través de SSH

El SSH (Secure Shell) es un protocolo de red que se utiliza para hacer una conexión remota y cifrada, por lo que sería perfecta para que algún departamento gestione la máquina virtual sin necesidad de recurrir al servicio informático de la universidad. Para ello, se hará uso de los siguientes comandos por “terminal”.

El primero de ellos sería “sudo su” con el fin de que se coloque como administrador y posteriormente el comando “apt install openssh-server”.

```
sudo su
```

```
apt install openssh-server
```

Una vez instalado correctamente se hará uso del comando ufw, que permite abrir el puerto 22, necesario para las comunicaciones SSH y permitir que todo funcione correctamente.

```
ufw allow ssh
```

En caso de ser necesaria alguna configuración adicional se puede hacer desde /etc/ssh/sshd_config, para ello se hará uso del comando:

```
nano /etc/ssh/sshd_config
```

Una vez hecho cualquier cambio en la configuración es aconsejable reiniciar el servicio para un correcto funcionamiento:

```
service ssh restart
```

Debido a que no se ha configurado ningún alias y tampoco está dentro de un dominio, se conectará a la máquina virtual con direccionamiento IP. Para ello, primero de todo se deberá conocer la dirección IP. En este caso, se podría usar el siguiente comando que hace un display de las distintas interfaces de red y los parámetros de cada una de ellas, al igual que en la Ilustración 137:

```
ifconfig
```

```
root@fvarrosadia:/home/nhstr# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.179.128 netmask 255.255.255.0 broadcast 192.168.179.255
    inet6 fe80::f163:1612:2fda:1eae prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:0e:65:95 txqueuelen 1000 (Ethernet)
    RX packets 213534 bytes 91682871 (91.6 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 204352 bytes 91084854 (91.0 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 156072 bytes 55814953 (55.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 156072 bytes 55814953 (55.8 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
Mostrar aplicaciones
root@fvarrosadia:/home/nhstr#
```

Ilustración 137 - Uso de ifconfig.

Se puede observar que la dirección IP de la máquina virtual sería entonces la 192.168.179.128, simplemente se rellenaran los campos pertinentes, como en la Ilustración 138:

- Hostname: 192.168.179.128
- Port: 22 (Debido a que no se ha realizado ningún cambio en la configuración)
- Connection type: SSH

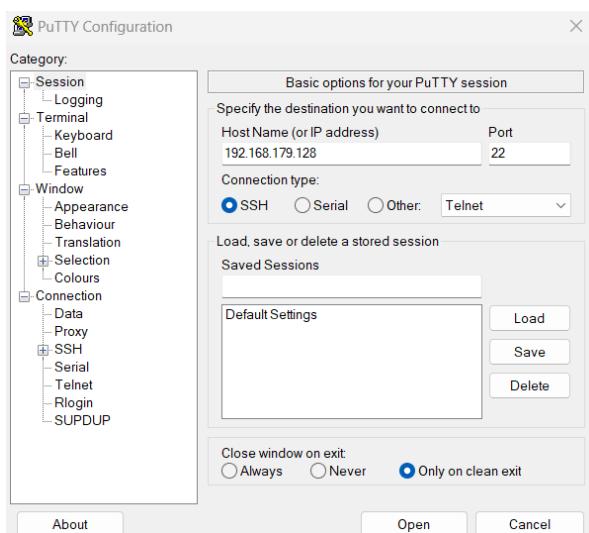


Ilustración 138 – Conexión a través de Putty.

Una vez aceptada la conexión, mostrará una ventana emergente el cual hace un resumen de la conexión y la fingerprint de la misma, debido a que es un dispositivo de confianza se le dará a “Accept”. Tal y como se ve en la Ilustración 139.

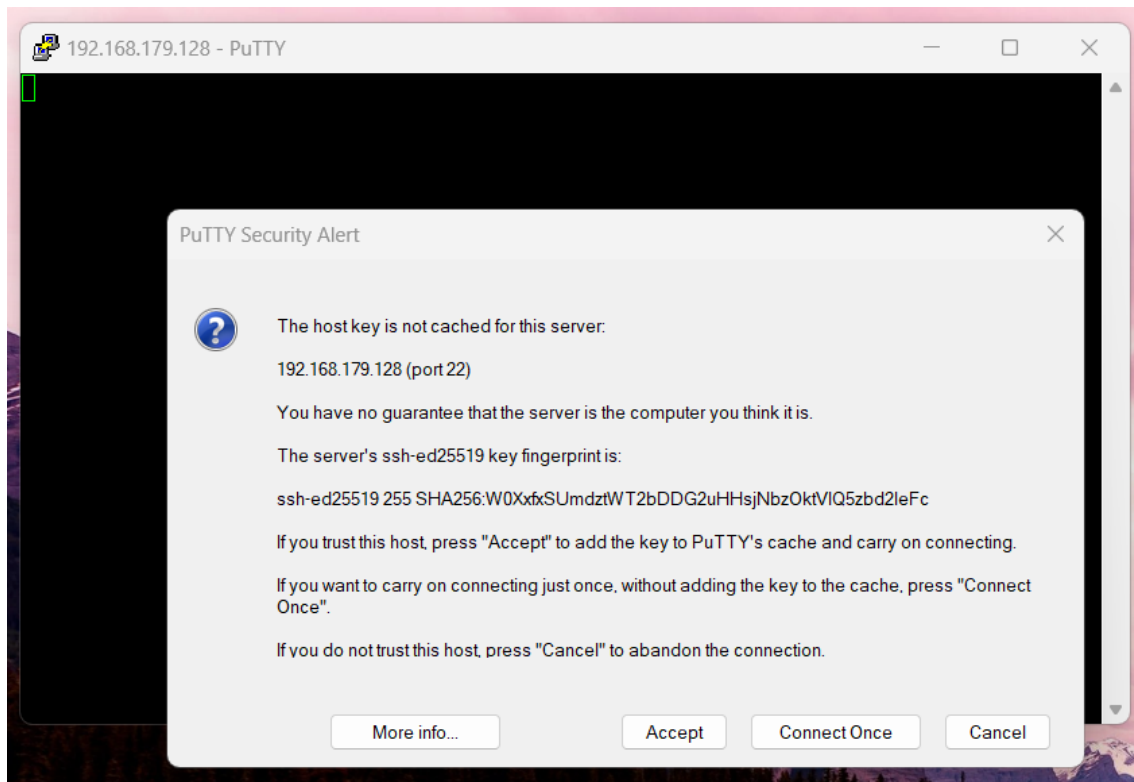


Ilustración 139 – Dipslay de conexión SSH.

Y ya estaría el acceso al “terminal” de la máquina virtual. Por lo que se podría reiniciar cualquier servicio bloqueado o hacer modificaciones o actualizaciones de programas, como se aprecia en la Ilustración 140.

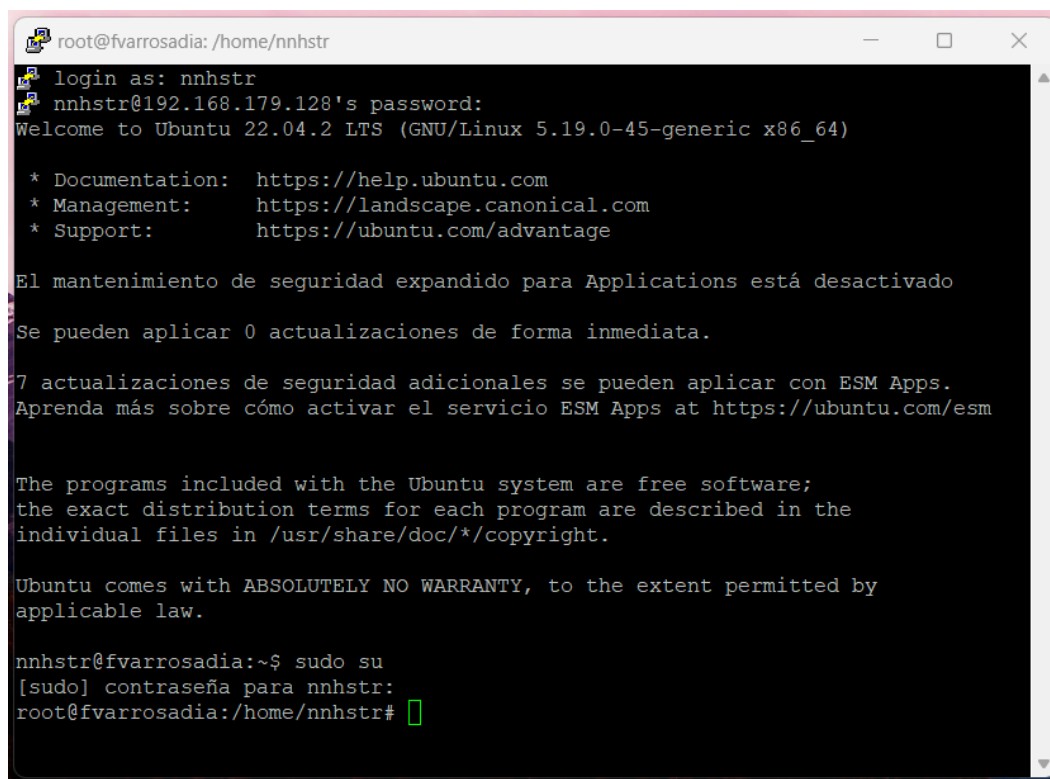


Ilustración 140 – “terminal” a traves de putty.

12.6. Network tools

Para poder hacer uso del comando `ifconfig` [12], como se ve en la Ilustración 141, se necesitará el paquete de Network tools, el cual se instalará con el siguiente comando:

```
sudo su
```

```
apt install net-tools
```

```
root@fvarrosadia:/home/fvarrosadia# apt install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
 net-tools
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.
Need to get 204 kB of archives.
```

Ilustración 141 – Instalación de las Network tools en Ubuntu.

Para poder hacer uso del paquete al igual que en la Ilustración 142, se tendrá que usar el comando, como en la Ilustración 142:

```
ifconfig
```

```
root@fvarrosadia:/home/fvarrosadia# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.179.133 netmask 255.255.255.0 broadcast 192.168.179.255
    inet6 fe80::7703:ac39:57a4:9131 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:b2:bb:db txqueuelen 1000 (Ethernet)
    RX packets 308164 bytes 456513149 (456.5 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 33816 bytes 2660764 (2.6 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Ilustración 142 – Comando `ifconfig` en Ubuntu.

Como se puede observar el comando equivale a `ipconfig /all` en Windows, proporciona detalles sobre las interfaces de red, como direcciones IP, máscaras de subred, dirección MAC, estadísticas de paquetes, entre otros.

12.7. Instalación Node-RED

Para poder hacer uso de Node-RED [13], el cual es un entorno Node.js basado en programación visual, se deberá hacer uso de la documentación oficial que se encuentra en el siguiente [enlace](#).

Antes de poder realizar la instalación de Node-RED, se tendrá que comprobar que están los paquetes de Nodejs y NPM preinstalados en el sistema. Para ello, se hará uso de los siguientes comandos.

El comando `sudo su` cambiará la “terminal” a modo administrador, por lo que dará permisos especiales y se podrá instalar correctamente cualquier paquete. El segundo comando permitirá descargar e instalar el paquete deseado, en este caso Nodejs, al igual que en la Ilustración 143.

```
sudo su
```

```
apt install nodejs
```

```
root@fvarrosadia:/home/fvarrosadia# apt install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  javascript-common libc-ares2 libjs-highlight.js libnode72 nodejs-doc
Suggested packages:
  apache2 | lighttpd | httpd npm
The following NEW packages will be installed:
  javascript-common libc-ares2 libjs-highlight.js libnode72 nodejs nodejs-doc
0 upgraded, 6 newly installed, 0 to remove and 7 not upgraded.
Need to get 13,7 MB of archives.
After this operation, 53,9 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Ilustración 143 – Instalación de Nodejs en Ubuntu.

Con el fin de comprobar que versión se ha instalado en el sistema se podrá hacer uso del comando visto en la Ilustración 144:

```
apt install nodejs
```

```
root@fvarrosadia:/home/fvarrosadia# node -v
v12.22.9
root@fvarrosadia:/home/fvarrosadia#
```

Ilustración 144 – Comprobación de la versión de aplicación Nodejs.

A su vez, se tendrá que instalar el paquete npm, como en la Ilustración 145, por lo que se hará uso del siguiente comando:

apt install npm

```
root@fvarrosadia:/home/fvarrosadia# apt install npm
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev
  fakeroot g++ g++-11 gcc gcc-11 git git-man gyp libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan6 libbinutils
```

Ilustración 145 – Instalación del paquete NPM en Ubuntu.

Con el fin de comprobar que versión se ha instalado en el sistema se podrá hacer uso del comando, al igual que en la Ilustración 146:

npm -v

```
root@fvarrosadia:/home/fvarrosadia# npm -v
8.5.1
root@fvarrosadia:/home/fvarrosadia#
```

Ilustración 146 – Comprobación de versión de NPM en Ubuntu.

Una vez que se ha comprobado que se tienen los paquetes de Nodejs y Npm se procede a hacer la instalación de node-red. En este caso es una instalación un poco especial debido a que se hace uso del comando:

npm install -g --unsafe-perm node-red

La sección **-g** permite hacer una instalación de forma global en el sistema, en lugar de un directorio actual. Es decir, que podrá ser utilizado por cualquier proyecto o aplicación y a su vez permite que esté disponible como comando desde cualquier ubicación, al igual que en la Ilustración 147.

De vez en cuando al hacer uso del comando npm en conjunto con el comando sudo su (que cambia de usuario y grupo a “root” temporalmente), podría ocasionar problemas de permisos al acceder a diversos archivos o directorios. Por ello, se hace uso de la opción “--unsafe-perm”, que resuelve este problema asegurándose de que “npm” mantenga los privilegios y permisos del usuario actual.

```
root@fvarrosadia:/home/fvarrosadia# npm install -g --unsafe-perm node-red
( ) :. idealTree:lib: sill idealTree buildDeps
```

Ilustración 147 – Instalación de Node-RED a través de Npm en Ubuntu.

Una vez finalizada la instalación de Node-RED, se tendrá que iniciar la aplicación manualmente. Para ello simplemente se tendrá que hacer uso del comando “node-red” en el “terminal”, al igual que en la Ilustración 148.

node-red

```
root@fvarrosadia:/home/fvarrosadia# node-red
5 Jul 18:23:51 - [info]

Welcome to Node-RED
=====

5 Jul 18:23:51 - [info] Node-RED version: v3.0.2
5 Jul 18:23:51 - [info] Node.js version: v12.22.9
5 Jul 18:23:51 - [info] Linux 5.19.0-46-generic x64 LE
5 Jul 18:23:52 - [info] Loading palette nodes
5 Jul 18:23:52 - [info] Settings file : /root/.node-red/settings.js
5 Jul 18:23:52 - [info] Context store : 'default' [module=memory]
5 Jul 18:23:52 - [info] User directory : /root/.node-red
5 Jul 18:23:52 - [warn] Projects disabled : editorTheme.projects.enabled=false
5 Jul 18:23:52 - [info] Flows file : /root/.node-red/flows.json
5 Jul 18:23:52 - [info] Creating new flow file
5 Jul 18:23:52 - [warn]
```

Ilustración 148 – Inicio de la aplicación Node-RED en Ubuntu.

Para poder hacer uso de node-red o cualquier configuración o instalación de paquetes se tendrá que hacer mediante el navegador web, mediante el puerto 1880. Por lo que para acceder se tendrá que ir al navegador e ir a cualquiera de las siguientes direcciones, y se obtendrá el resultado de la Ilustración 149.

- localhost:1880
- 127.0.0.1:1880
- 192.168.179.128:1880 (dicha dirección se puede obtener en el “terminal” mediante el comando ifconfig)

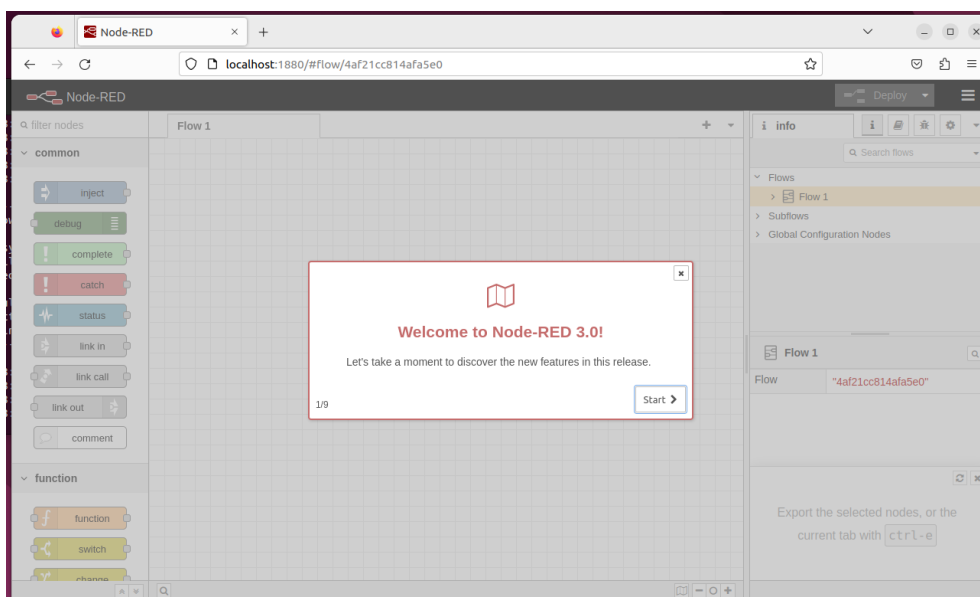


Ilustración 149 – Acceso a Node-RED mediante firefox.

12.8. Configuración Node-RED

Para la realización de este proyecto, se utilizará la aplicación de Node-RED, que facilitará la comunicación entre un inversor fotovoltaico con un protocolo de comunicación tipo Modbus y la base de datos InfluxDB. Para habilitar esta comunicación, será imprescindible incorporar ciertos paquetes adicionales en la aplicación. Los cuales son:

- node-red: Este es el paquete principal de Node-RED que proporciona todos los nodos básicos necesarios para crear flujos de trabajo. Incluye nodos para la manipulación de datos, flujos de control, operaciones lógicas, integración con HTTP, MQTT y otros protocolos, entre muchas otras funciones.
- node-red-contrib-buffer-parser: Este paquete extiende Node-RED para permitir el análisis y manipulación de datos binarios. Es especialmente útil cuando se necesita trabajar con este tipo de datos, como aquellos provenientes de sensores. Permite interpretar, extraer y transformar datos para que puedan ser procesados por los flujos de Node-RED.
- node-red-contrib-influxdb: Este paquete facilita la integración de la base de datos InfluxDB con Node-RED. InfluxDB es una base de datos diseñada para almacenar y consultar datos de series temporales, como mediciones de sensores o registros de eventos que tienen una marca de tiempo. Con este paquete, se pueden enviar datos a una base de datos InfluxDB o realizar consultas para recuperar información almacenada previamente.
- node-red-contrib-modbus: Este paquete permite la comunicación entre dispositivos y sistemas a través del protocolo Modbus en Node-RED. Modbus es un protocolo de comunicación utilizado ampliamente en la automatización industrial para intercambiar datos entre dispositivos electrónicos. Con este paquete, se pueden leer y escribir registros Modbus, lo que permite la integración con una amplia variedad de dispositivos compatibles con este protocolo.

Para proceder a la instalación de estos paquetes en la aplicación, en primera instancia se deberá de tener la aplicación de Node-RED abierta en un navegador web. Posteriormente, se tendrá que ir a la parte superior derecha de la web, y desplegar el menú de configuración con el botón que muestra tres barras horizontales. Finalmente, se clicará el botón de “Settings”, como se ve en la Ilustración 150.

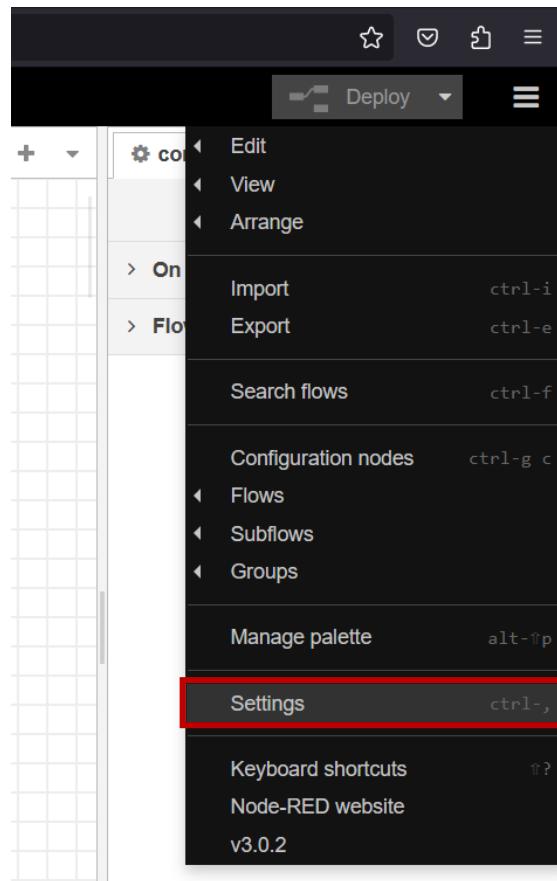


Ilustración 150 – Settings de Node-RED.

Una vez clicado el botón de “Settings” aparecerá en pantalla el siguiente cuadro, y se deberá ir a la segunda pestaña denominada “Palette”. En esta sección se podrá comprobar los paquetes que se tienen actualmente instalados en la aplicación, como se observa en la Ilustración 151.

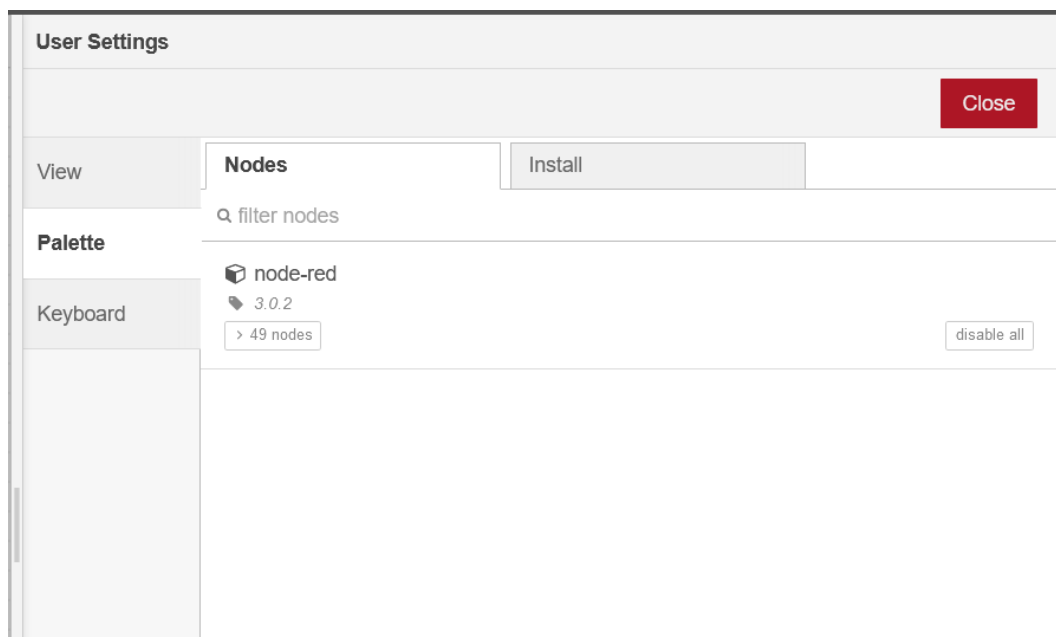


Ilustración 151 – Paquetes instalados acualmente en Node-RED.

En caso de que falte cualquiera de los paquetes anteriormente mencionados, se deberá ir a la segunda sección denominada “Install”. Y en el buscador se escribirá el nombre del paquete pertinente y se seleccionará la opción de “Install”, al igual que en la Ilustración 152.

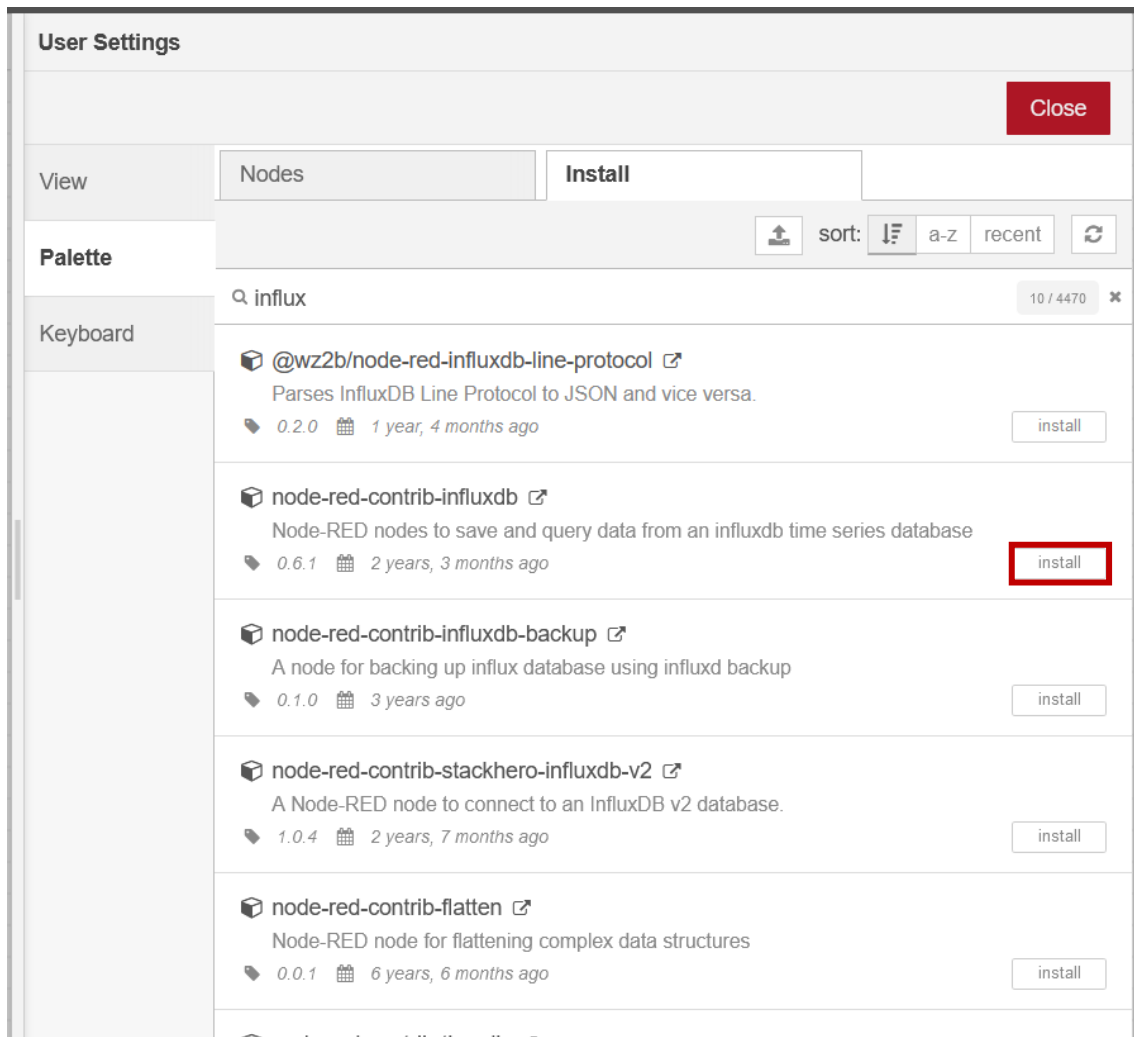


Ilustración 152 – Búsqueda de paquetes en Node-RED.

Una vez realizada la instalación de todos los paquetes se deberá volver a la sección de “Nodes”, con el fin de comprobar que todos los paquetes se han instalado correctamente, al igual que en la Ilustración 153.

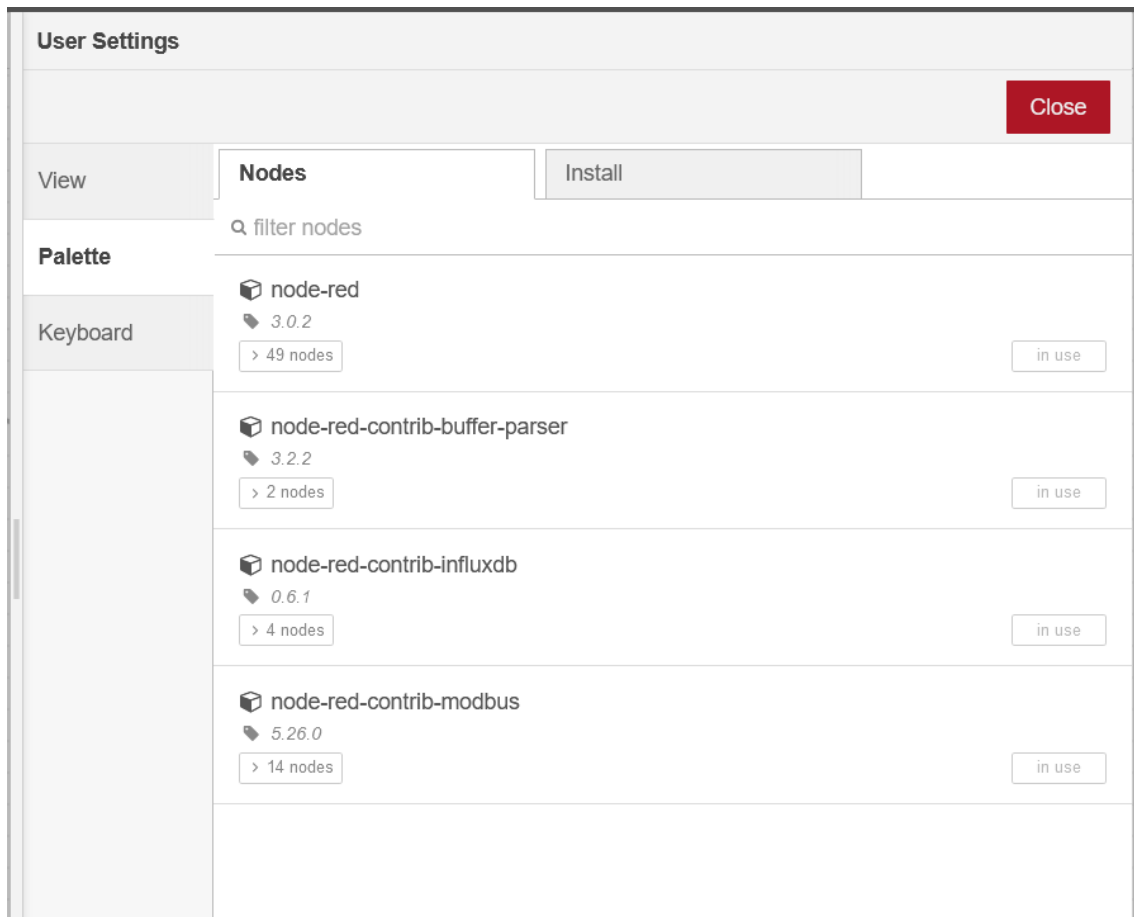


Ilustración 153 – Comprobación de los paquetes instalados en Node-RED.

12.9. Instalación InfluxDB

Para la instalación de InfluxDB [14] se va a seguir el procedimiento de instalación desde navegador, debido a que es una instalación más rápida y sencilla.

Para ello, se deberá ir a la [página web oficial de Influxdb](#) y se seleccionará la descarga manual. Una vez en la web, se seleccionará la opción (amd64), como en la Ilustración 154.

Manually download and install the influxd binary

1. Download the InfluxDB binary.

Download the InfluxDB binary [from your browser](#) or [from the command line](#).

Download from your browser



Ilustración 154 – Descarga manual de Influxdb Ubuntu.

Una vez descargado, se llevará el archivo “.tar.gz” al directorio en el que se quiere guardar la base de datos. Este método permite tener los archivos más localizados, para poder hacer back-ups o exportación de la base de datos.

Se abrirá el archivo “.tar.gz” y se clicará a “Extract” para descomprimir los ficheros pertinentes, al igual que en la Ilustración 155.

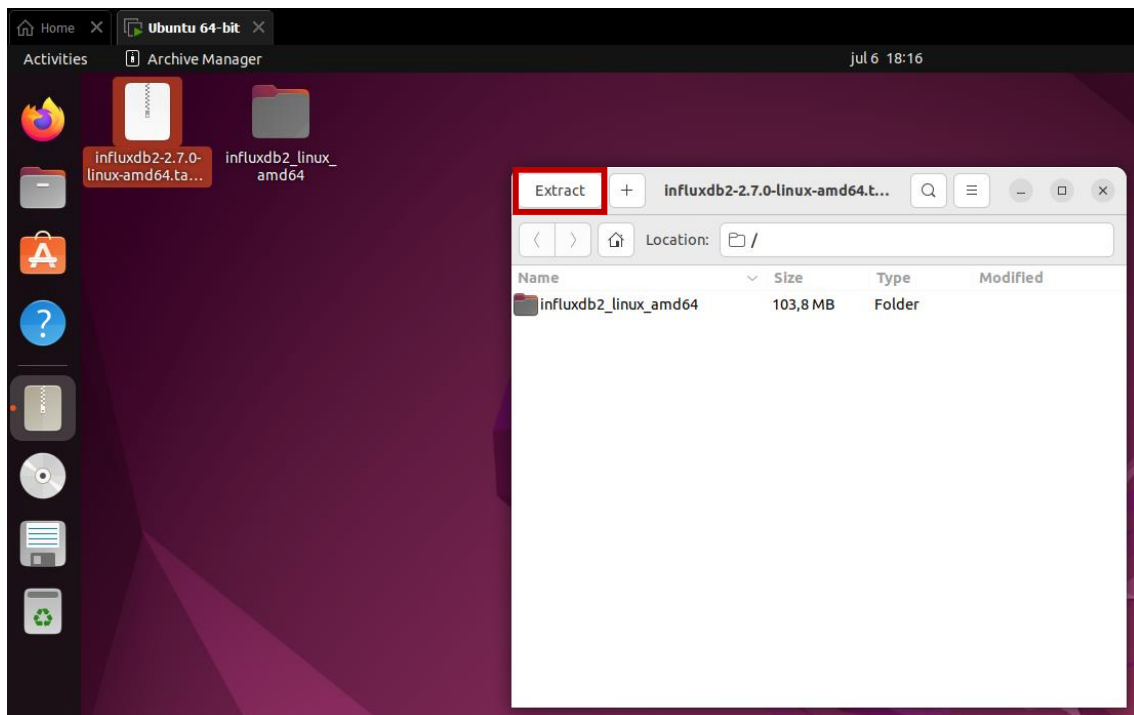


Ilustración 155 – Descompresión de InfluxDB en Ubuntu.

Una vez extraído el contenido de la carpeta y tal y como se observa en la Ilustración 156, se tendrá que dar botón derecho y “Open in terminal”.

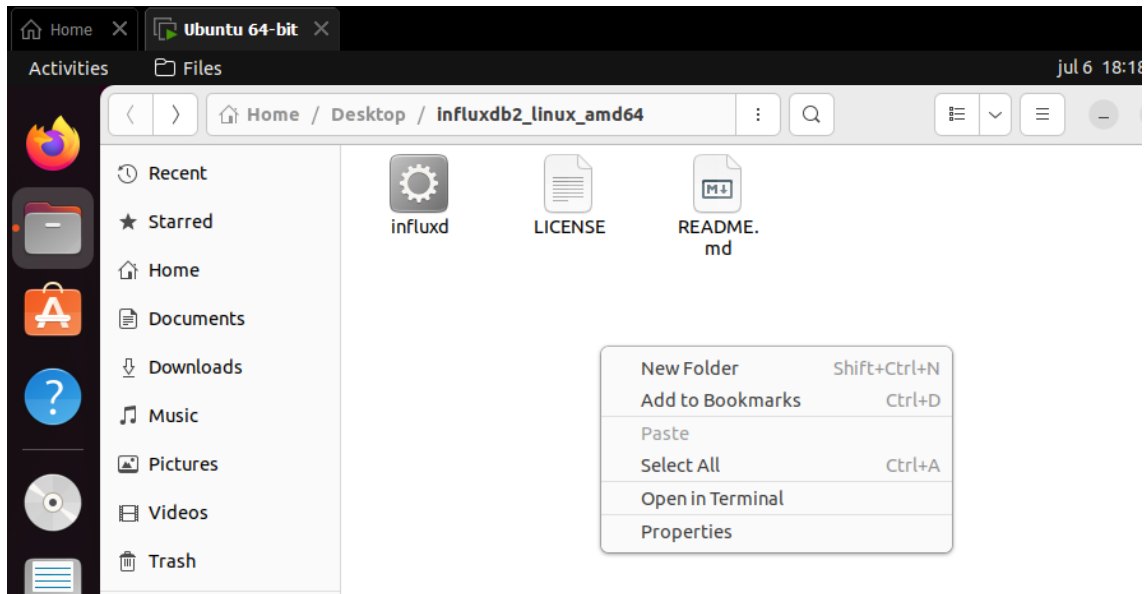


Ilustración 156 – Archivos descomprimidos de InfluxDB.

En caso de querer usar los comandos de InfluxDB desde “terminal” sin tener que cambiar la ruta por defecto, se puede hacer una copia del fichero Influxd en el directorio “/home/%username%/influxd” con el comando mostrado en la Ilustración 157:

```
cp influxd /home/%username%/influxd
```

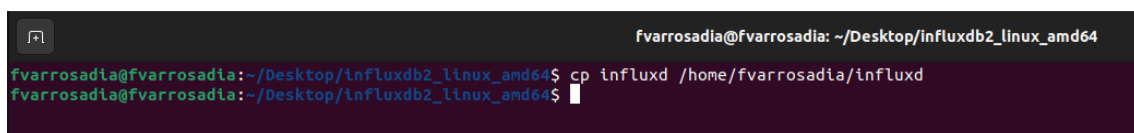


Ilustración 157 – Copia de la aplicación en la ruta de “terminal”.

Para poder ejecutar el programa de Influxdb simplemente se tiene que abrir el “terminal” en el directorio donde se encuentre la base de datos (en este caso, debido a que ya está copiado el fichero en la ruta que se abre por defecto en el terminal), se tiene que escribir el siguiente comando, que arrancara la base de datos al igual que en la Ilustración 158:

```
influxd
```



```
fvarrosadia@fvarrosadia:~$ influxd
2023-07-06T16:21:30.659999Z info Welcome to InfluxDB {"log_id": "0ir~2"}
{"log_level": "info"}
2023-07-06T16:21:30.661785Z info Resources opened {"log_id": "0ir~2"}
2023-07-06T16:21:30.662430Z info Resources opened {"log_id": "0ir~2"}
2023-07-06T16:21:30.667222Z info Using data dir {"log_id": "0ir~2"}
{"ine/data"}
2023-07-06T16:21:30.667727Z info Compaction settings {"log_id": "0ir~2"}
"throughput_bytes_per_second": 50331648, "throughput_bytes_per_second_burst": 50331648, "op_elapsed": "0.499ms"}
2023-07-06T16:21:30.668179Z info Open store (start) {"log_id": "0ir~2"}
: "start"}
2023-07-06T16:21:30.668676Z info Open store (end) {"log_id": "0ir~2"}
: "end", "op_elapsed": "0.499ms"}
2023-07-06T16:21:30.669177Z info Starting retention policy enforcement {"log_id": "0ir~2"}
2023-07-06T16:21:30.669577Z info Starting precreation service {"log_id": "0ir~2"}
{"30m"}
2023-07-06T16:21:30.672864Z info Starting query controller {"log_id": "0ir~2"}
s_quota_per_query": 9223372036854775807, "memory_bytes_quota_per_query": 9223372036854775807}
2023-07-06T16:21:30.686153Z info Configuring InfluxQL statement executor {"log_id": "0ir~2"}
series": 0, "max_select_buckets": 0}
2023-07-06T16:21:30.695769Z info Starting {"log_id": "0ir~2"}
2023-07-06T16:21:30.695911Z info Listening {"log_id": "0ir~2"}
```

Ilustración 158 – Ejecución de Influxd en Ubuntu.

Para poder hacer uso de InfluxDB o cualquier configuración o instalación de paquetes se tendrá que hacer mediante el navegador web, mediante el puerto 8086. Por lo que para acceder se tendrá que ir al navegador e ir a cualquiera de las siguientes direcciones.

- localhost:8086
- 127.0.0.1:8086
- 192.168.179.128:8086 (dicha dirección se puede obtener en el “terminal” mediante el comando `ifconfig`)

Realizando este proceso, se obtendrá lo mostrado en la Ilustración 159.

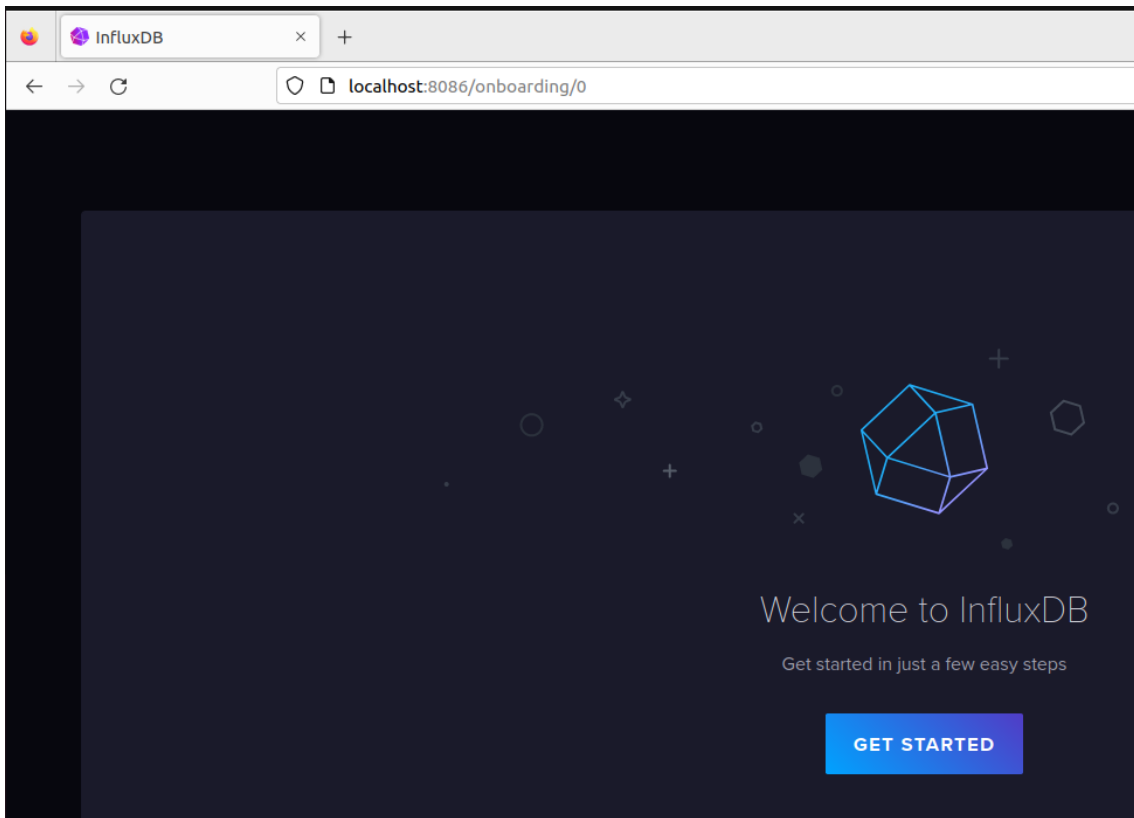


Ilustración 159 – Aplicación de InfluxDB en Ubuntu.

12.10. Error de InfluxDB

En algunas ocasiones, si la aplicación de InfluxDB no se ha cerrado correctamente, al intentar volver a ejecutarla puede mostrar el error que el puerto 8086 está en uso, por lo que no se puede volver a ejecutar la aplicación, como se puede observar en la Ilustración 160.

```
cp :8086: bind: address already in use"}
Error: listen tcp :8086: bind: address already in use
See 'influxd -h' for help
```

Ilustración 160 – Error de puerto en uso.

Para ello, se deberá utilizar el comando `lsof`, el cual mostrará una lista de las conexiones de red activas y los programas asociados a ellas. Los resultados mostrarán detalles como el nombre del programa o proceso, su PID (identificador de proceso), el usuario que inició el proceso, el tipo de conexión (TCP o UDP), la dirección IP y el número de puerto local y remoto, entre otros datos, como se aprecia en la Ilustración 161.

```
root@fvarrosadia:/home/nnhstr# sudo lsof -i :8086
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
influxd 2535 influxdb 50u IPv6 25821 0t0 TCP *:8086 (LISTEN)
influxd 2535 influxdb 54u IPv4 27667 0t0 TCP localhost:51990->localhost:8086 (ESTABLISHED)
influxd 2535 influxdb 55u IPv6 32080 0t0 TCP localhost:8086->localhost:51990 (ESTABLISHED)
```

Ilustración 161 – Comando `lsof -i` en Ubuntu.

Una vez muestre la lista en la pantalla se buscará que PID está en modo “Listen”, debido a que será el que cause el error anteriormente mencionado. En este caso sería el PID 2535.

Por lo que para solucionarlo se deberá usar el comando seguido del PID localizado anteriormente, como en la Ilustración 162:

```
sudo kill 2535
```

```
root@fvarrosadia:/home/nnhstr# sudo kill 2535
```

Ilustración 162 - Finalización del proceso InfluxDB.

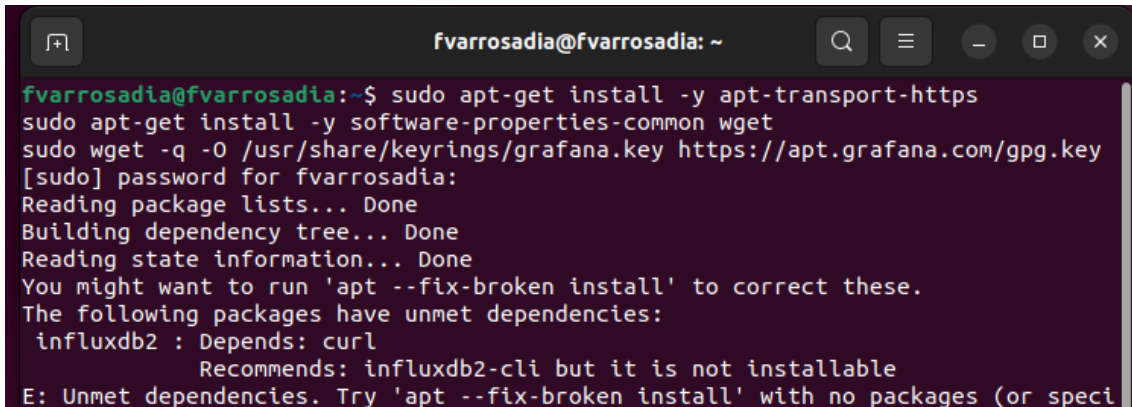
Por lo que ya se podría volver a usar el comando “Influxd”, sin problema alguno.

12.11. Instalar Grafana

Para poder hacer uso de Grafana [15] se seguirá el procedimiento mencionado en su web oficial, dicho proceso es vía “terminal”.

Los primeros comandos servirán para descargar e instalar los repositorios de Grafana, como se realiza en la Ilustración 163:

```
sudo apt-get install -y apt-transport-https
sudo apt-get install -y software-properties-common wget
sudo wget -q -O /usr/share/keyrings/grafana.key https://apt.grafana.com/gpg.key
```

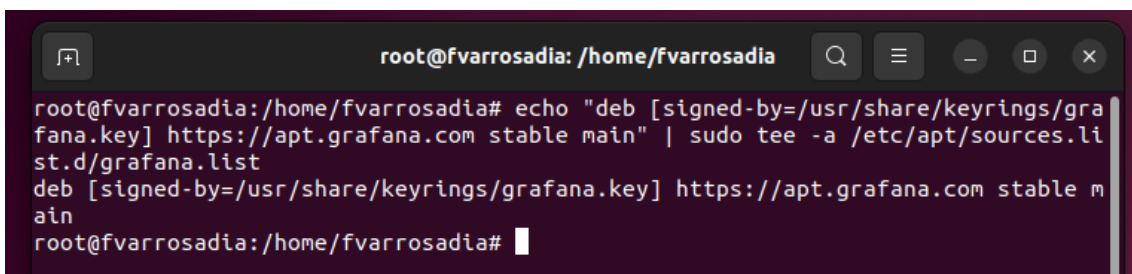


```
fvarrosadia@fvarrosadia: ~
fvarrosadia@fvarrosadia:~$ sudo apt-get install -y apt-transport-https
sudo apt-get install -y software-properties-common wget
sudo wget -q -O /usr/share/keyrings/grafana.key https://apt.grafana.com/gpg.key
[sudo] password for fvarrosadia:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
You might want to run 'apt --fix-broken install' to correct these.
The following packages have unmet dependencies:
 influxdb2 : Depends: curl
               Recommends: influxdb2-cli but it is not installable
E: Unmet dependencies. Try 'apt --fix-broken install' with no packages (or speci
```

Ilustración 163 – Actualizar los repositorios de Grafana en Ubuntu.

Se agregan los repositorios estables de Grafana mediante el comando, como se ve en la Ilustración 164:

```
echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com
stable main" | sudo tee -a /etc/apt/sources.list.d/grafana.list
```



```
root@fvarrosadia: /home/fvarrosadia
root@fvarrosadia:/home/fvarrosadia# echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com stable main" | sudo tee -a /etc/apt/sources.list.d/grafana.list
deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com stable main
root@fvarrosadia:/home/fvarrosadia#
```

Ilustración 164 – Agregar los repositorios estables de Grafana en Ubuntu.

```
echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com beta
main" | sudo tee -a /etc/apt/sources.list.d/grafana.list
```

En caso de querer hacer uso de los repositorios en beta, se deberá usar el comando de la Ilustración 165:

```
root@fvarrosadia:/home/fvarrosadia# echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com beta main" | sudo tee -a /etc/apt/sources.list.d/grafana.list
deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com beta main
root@fvarrosadia:/home/fvarrosadia#
```

Ilustración 165 – Agregar los repositorios beta de Grafana en Ubuntu.

Una vez agregados los repositorios se actualizará la lista de paquetes disponibles en los servidores, como se observa en la Ilustración 166.

sudo apt-get update

```
root@fvarrosadia:/home/fvarrosadia# apt-get update
Hit:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 https://apt.grafana.com stable InRelease [5.983 B]
Get:3 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 https://apt.grafana.com beta InRelease [5.976 B]
Get:5 https://apt.grafana.com stable/main amd64 Packages [132 kB]
0% [3 InRelease 94,1 kB/119 kB 79%] [Connecting to security.ubuntu.com] [5 Pack
```

Ilustración 166 – Actualización de los repositorios en Ubuntu.

Una vez actualizados los repositorios, se procederá con la instalación de Grafana con el comando mostrado en la Ilustración 167:

sudo apt-get install grafana

```
root@fvarrosadia:/home/fvarrosadia# apt-get install grafana
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  grafana
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.
Need to get 83,4 MB of archives.
After this operation, 306 MB of additional disk space will be used.
Get:1 https://apt.grafana.com stable/main amd64 grafana amd64 10.0.1 [83,4 MB]
58% [1 grafana 60,3 MB/83,4 MB 72%] 1.344 kB/s 17s
```

Ilustración 167 – Instalación de Grafana en Ubuntu.

Para poder hacer uso de Grafana o cualquier configuración o instalación de paquetes se tendrá que hacer mediante el navegador web, mediante el puerto 3000. Por lo que, para acceder se tendrá que ir al navegador e ir a cualquier a de las siguientes direcciones.

- localhost:3000
- 127.0.0.1:3000
- 192.168.179.128:3000 (dicha dirección se podrá obtener en el “terminal” mediante el comando ifconfig)

A fin de obtener acceso a la plataforma, tal y como se aprecia en la Ilustración 168.

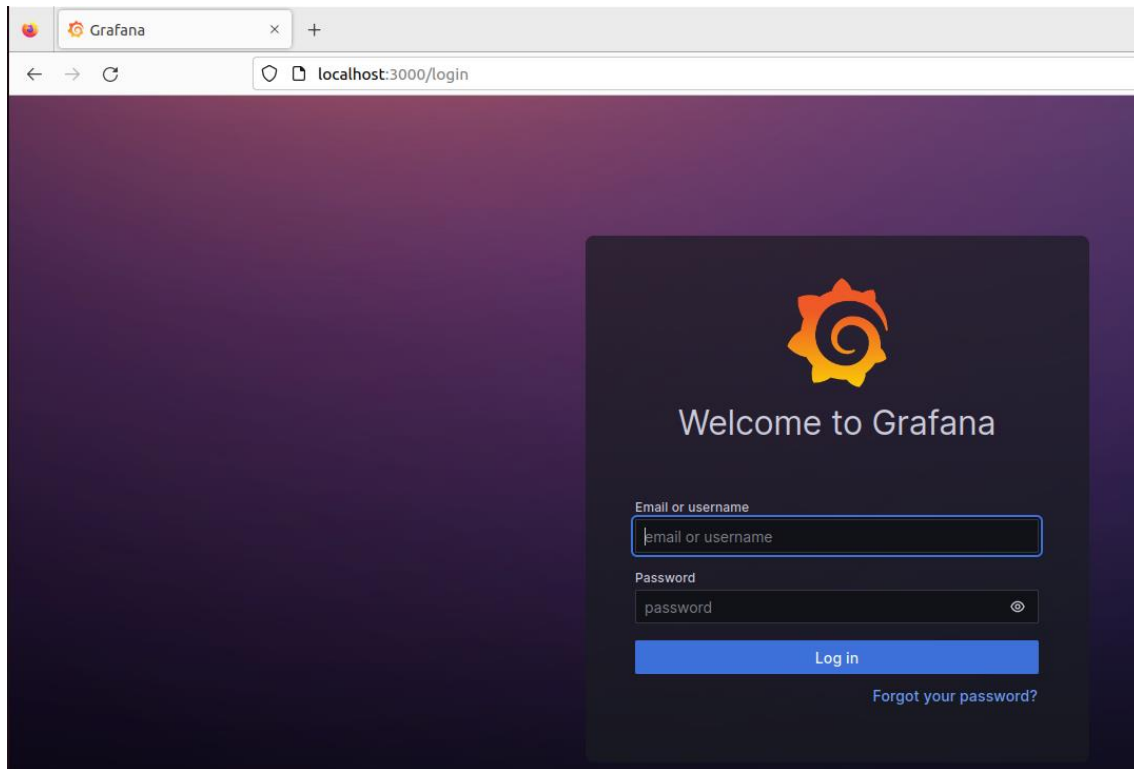


Ilustración 168 – Uso de grafana por navegador web en Ubuntu.

En caso de querer que Grafana se abra automáticamente junto con el sistema se deberán de usar los comandos:

```
sudo /bin/systemctl daemon-reload  
sudo /bin/systemctl enable grafana-server  
sudo /bin/systemctl start grafana-server
```

12.12. [Error en la instalación de Grafana](#)

En caso de dar error al usar el comando “apt-get update” se deberá recuperar la lista de repositorios con el comando mostrado en la Ilustración 169:

```
sudo apt install --fix-broken
```

```
ty & solution).
root@fvarrosadia:/home/fvarrosadia# sudo apt install --fix-broken
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Correcting dependencies... Done
The following additional packages will be installed:
  curl
The following NEW packages will be installed:
```

Ilustración 169 – Recuperar los repositorios dañados en Ubuntu.

12.13. Instalación de Apache2

Para poder realizar la instalación de Apache2 en Ubuntu, se hará por vía ‘terminal’, por lo que después de abrir dicha aplicación se tendrá que hacer uso de los siguientes comandos con el fin de situarse como administrador y hacer una actualización completa del sistema para evitar errores entre aplicaciones.

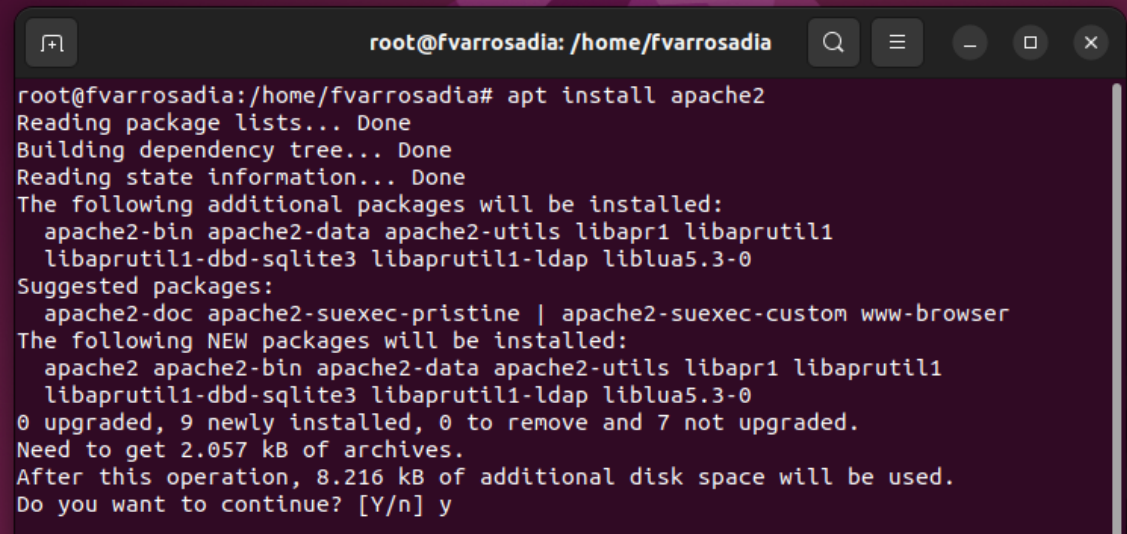
```
sudo su
```

```
apt-get update
```

```
apt-get upgrade
```

Una vez que ha finalizado la actualización del sistema, se hará uso del comando que descarga e instala Apache2 del repositorio oficial de Ubuntu, como se observa en la Ilustración 170.

```
apt install apache2
```



```
root@fvarrosadia: /home/fvarrosadia
root@fvarrosadia:/home/fvarrosadia# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.3-0
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.3-0
0 upgraded, 9 newly installed, 0 to remove and 7 not upgraded.
Need to get 2.057 kB of archives.
After this operation, 8.216 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Ilustración 170 – Instalación de Apache2.

Después de esta breve instalación se comprobará el estado de funcionamiento del servicio con el comando, al igual que en la Ilustración 171:

systemctl status apache2

```
root@fvarrosadia:/home/fvarrosadia# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese
   Active: active (running) since Wed 2023-08-09 16:45:14 CEST; 1min 7s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 51383 (apache2)
    Tasks: 55 (limit: 4599)
   Memory: 5.2M
      CPU: 107ms
   CGroup: /system.slice/apache2.service
           └─51383 /usr/sbin/apache2 -k start
             └─51384 /usr/sbin/apache2 -k start
               └─51385 /usr/sbin/apache2 -k start

ago 09 16:45:14 fvarrosadia systemd[1]: Starting The Apache HTTP Server...
ago 09 16:45:14 fvarrosadia apachectl[51382]: AH00558: apache2: Could not relia
ago 09 16:45:14 fvarrosadia systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

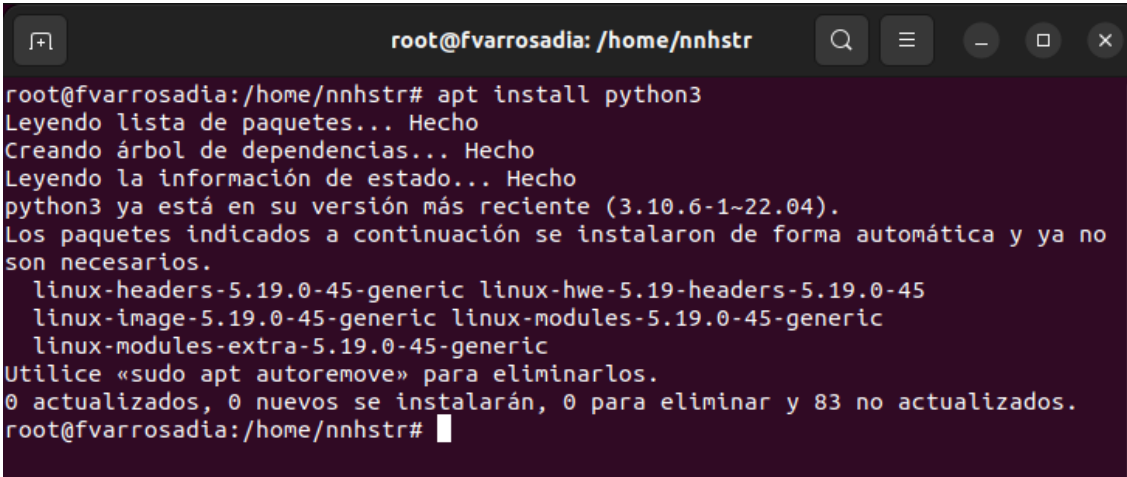
Ilustración 171 – Comprobación del estado de Apache2.

12.14. Instalación de Python

A fin de poder instalar Python en el servidor de Ubuntu, se tendrá que abrir una nueva ventana de “Terminal”. Posteriormente se accederá a modo super usuario con el comando “sudo su” y después se escribirá el comando de “apt install python” para descargar e instalar la última versión. Tal y como se aprecia en la Ilustración 172.

```
sudo su
```

```
apt install python
```



```
root@fvarrosadia: /home/nnhstr
root@fvarrosadia:/home/nnhstr# apt install python3
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
python3 ya está en su versión más reciente (3.10.6-1~22.04).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  linux-headers-5.19.0-45-generic linux-hwe-5.19-headers-5.19.0-45
  linux-image-5.19.0-45-generic linux-modules-5.19.0-45-generic
  linux-modules-extra-5.19.0-45-generic
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 83 no actualizados.
root@fvarrosadia:/home/nnhstr#
```

Ilustración 172 - Instalación de Python en Ubuntu.

12.15. Datos del inversor Ingeteam

7. REGISTER MAP AND DESCRIPTION FOR 3PLAY 100TL AND 160TL (ABS1004)

7.1 ONLINE DATA REV.0

Start Address	MB Modicon Address	Display order	Show when	Description	Magnitude	Usual Min Value (*)	Usual Max Value (*)
0	30001	1	always	Actual Date. Year (Uint16).	Year	1900	2100
1	30002			Actual Date. Month of the Year.(Uint16).	Month	1	12
2	30003			Actual Date. Day of the month.(Uint16).	Day	1	31
3	30004			Actual Date. Hour of the day. (Uint16).	Hour	0	23
4	30005			Actual Date: Minute. (Uint16).	Minute	0	59
5	30006			Actual Date: Second. (Uint16).	Second	0	59
6	30007	6	always	Partial Energy delivered by the unit since user reset. (high part of Uint32. Bits 31-16).	kW*h	0	2^32-2
7	30008			Partial Energy delivered by the unit since user reset. (low part of Uint32. Bits 15-0).			
8	30009	7	always	Partial number of hours of operation. (high part of Uint32. Bits 31-16).	hours	0	2^32-2
9	30010			Partial number of hours of operation. (low part of Uint32. Bits 15-0).			
10	30011	8	always	Partial number of connection since user reset. (high part of Uint32. Bits 31-16).	n/a	0	2^32-2
11	30012			Partial number of connection since user reset. (low part of Uint32. Bits 15-0).			
12	30013	5	always	Daily energy value. (bits 32-16, Uint32)	KWh*100	0	2^32-2
13	30014			Daily energy value. (bits 15-0, Uint32)			

14	30015	12	always	Instantaneous alarm code 1 bits.	Flag bits	0	0xFFFF	
15	30016	13	always	Instantaneous alarm code 2 bits.	Flag bits	0	0xFFFF	
16	30017	14	always	Instantaneous alarm code 3 bits.	Flag bits	0	0xFFFF	
17	30018	15	always	Instantaneous alarm code 4 bits.	Flag bits	0	0xFFFF	
18	30019	10	always	Instantaneous stop event (Uint16).	Code	0	0xFFFF	
19	30020	11	always	Warning code (Uint16).	Flag bits	0	0xFFFF	
20	30021	9	always	Status Unit 1		States	0	2
				0	Not Ready to Connect			
				1	Waiting to Connect			
				2	Connected to the Grid			
21	30022	30	always	Output grid RMS current of phase 1 (Uint16)	Amps x 100	0	20000	
22	30023	31	always	Output grid RMS current of phase 2 (Uint16)	Amps x 100	0	20000	
23	30024	32	always	Output grid RMS current of phase 3 (Uint16)	Amps x 100	0	20000	
24	30025	27	always	Grid RMS voltage of phase 1. (Uint16).	Volts x 10	0	5000	
25	30026	28	always	Grid RMS voltage of phase 2. (Uint16).	Volts x 10	0	5000	
26	30027	29	always	Grid RMS voltage of phase 3. (Uint16).	Volts x 10	0	5000	
27	30028	33	always	Grid frequency (Uint16)	Hz x 100	0	7000	
28	30029	34	always	Output apparent power (Int16)	VA/10	0	15000	
29	30030	35	always	Output active power (Int16).	Watt/10	-1000	15000	
30	30031	36	always	Output reactive power (Int16).	VAr/10	-15000	15000	
31	30032	37	always	Cosine of Phi. (Int16)	x1000	-1000	1000	
32	30033	40	always	Input 1 current (Int16).	Amps x 100	-100	20000	
33	30034	41	always	Input 1 voltage (Uint16)	Vdc	0	1200	
34	30035	42	always	Input 1 power (Int16).	Wats/10	-1000	15000	
35	30036	43	always	Isolation resistance of PV field (Uint16)	kOhm	0	65000	
36	30037	38	Direct	Wattmeter measurement (Self-Consumption mode) (Int16)	Watt/10	-20000	20000	



Functionality Specifications

		Self-Consum				
37	30038	39	Direct Self-Consum	Self Consumption ratio	%	0 100
38	30039	44	String kit on	Current of PV string 2_1 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
39	30040	45		Current of PV string 4_3 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
40	30041	46		Current of PV string 6_5 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
41	30042	47		Current of PV string 8_7 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
42	30043	48		Current of PV string 10_9 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
43	30044	49		Current of PV string 12_11 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
44	30045	50		Current of PV string 14_13(Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
45	30046	51		Current of PV string 16_15(Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
46	30047	52		Current of PV string 18_17 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
47	30048	53		Current of PV string 20_19 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
48	30049	54		Current of PV string 22_21 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
49	30050	55		Current of PV string 24_23 (Unt16) (8bits,8bits)	A x10 – A x 10	0 - 0 255-255
50	30051	20		Warning String kit code1.		Code 0 0xFFFF
51	30052	21	Warning String kit code2.		Code 0 0xFFFF	
52	30053	24	always	Active Power Reduction rate(Uint16)	% 0 100	
53	30054	25	always	Active Power Reduction Reason (Uint16)		Reason 0 12
				0	Not reduction ()	
				1	Not reduction ()	
				2	Not defined.	
				3	Not defined.	
				4	Not defined.	
				5	Communications (C)	
6	Grid FrequencyActive Power control (F)					



Functionality Specifications

				7	Grid Voltage/Active Power control(V)				
				8	Reactive Priority and Output Current Limit(Q)				
				9	Configuration (A)				
				10	Connection Initial Ramp (R)				
				11	Self Consumption mode (S)				
				12	Manufactured reserved (M)				
				13	Pac vs Vac algorithm				
				14	Phi cosine or tangent target				
				15	Low VAC to reach nominal power				
54	30055	26	always	Reactive Power Reference type (Uint16)		Reason	0	6	
				0	CosPhi Configuration				
				1	Qref Manual				
				2	CosPhi Manual				
				3	Qref Communications				
				4	CosPhi Communications				
				5	Qref (QvsV)				
				6	CosPhi (CosPhiVsP)				
55	30056	22	always	Unit flags 1. To indicate any feature.		Flag bits	0	0xFFFF	
				f0					
				f1					
				f2					
				f3					
				f4					
				f5					
				f6					

				f7			
				f8			
				f9			
				f10			
				f11			
				f12			
				f13			
				f14			
				f15			
56	30057	23	always	Unit flags 2. To indicate any feature.			
				f0	FW update result bit 0		
				f1	FW update result bit 1		
				f2	FW update result bit 2		
				f3	FW update result bit 3		
				f4			
				f5			
				f6			
				f7		Flag bits	0
				f8			0xFFFF
				f9			
				f10			
				f11			
				f12			
				f13			
				f14			
				f15			

57	30058	2	always	Total Energy delivered by the unit. (bits 31-16, Uint32).	kW*h	0	2^32-2
58	30059			Total Energy delivered by the unit. (bits 15-0, Uint32).			
59	30060	3	always	Total hours of operation. (high part of Uint32. Bits 31-16).	Hours	0	2^32-2
60	30061			Total hours of operation. (low part of Uint32. Bits 15-0).			
61	30062	4	always	Total number of connections since launching. (high part of Uint32. Bits 31-16).	n/a	0	2^32-2
62	30063			Total number of connections since launching. (low part of Uint32. Bits 15-0).			
63	30064			Reserved value (Uint16). Not implemented (0xFFFF)	Not defined	0xFFFF	0xFFFF
---	---			---	---	-	-
124	30125			Reserved value (Uint16). Not implemented (0xFFFF)	Not defined	0xFFFF	0xFFFF

Table 4: Register description of online address range 30001-30125 for 3Play 100TL.

(*)This Max/Min values are only approximated values, useful for programmers who need to know the typical range of each register.

7.2 STRING KIT DATA REV.0

String kit data are also available in registers 860-984 with different format. These values are available since FW version ABS1004_F.

Start Address	MB Modicon Address	Show when	Description	Magnitude	Usual Min Value (*)	Usual Max Value (*)
---------------	--------------------	-----------	-------------	-----------	---------------------	---------------------



12.16. Conversación con ChatGPT

Pregunta: Buenos días, acaban de instalarme 10 paneles fotovoltaicos en mi casa, y me gustaría realizar una página web para poder ver la generación de energía eléctrica que hago en cada momento. No tengo conocimiento previo sobre el tema... por lo que necesito un poco de ayuda. Sé que el modelo del inversor es el INGECON SUN 3Play y que tiene la dirección IP 192.168.1.10.

Respuesta: ¡Felicitaciones por la instalación de tus paneles fotovoltaicos! Para monitorear la generación de energía en tiempo real, puedes seguir estos pasos:

1. **Asegúrate de la conexión:** Asegúrate de que tu dispositivo (computadora, tablet o smartphone) esté conectado a la misma red Wi-Fi que tu inversor, que en este caso tiene la dirección IP 192.168.1.10.
2. **Accede a la interfaz del inversor:** Abre un navegador web en tu dispositivo y escribe la dirección IP del inversor (192.168.1.10) en la barra de direcciones. Esto debería llevarte a la interfaz de configuración del inversor.
3. **Inicia sesión:** Si se te solicita un nombre de usuario y contraseña, utiliza las credenciales proporcionadas por el instalador o consulta el manual del inversor para obtener esta información.
4. **Explora la interfaz:** Una vez que hayas iniciado sesión, deberías poder acceder a los datos de generación de energía. La interfaz puede variar según el modelo, pero generalmente podrás ver información como la producción actual, la producción acumulada y posiblemente gráficos que muestren el rendimiento en diferentes intervalos de tiempo.
5. **Opción avanzada - Registro de datos:** Si deseas crear una página web para mostrar estos datos de manera más accesible, puedes explorar si el inversor tiene una función de exportación de datos o registro en tiempo real. Si es así, podrías programar un proceso para tomar esos datos y mostrarlos en tu página web.
6. **Desarrollo de la página web:** Para crear la página web, tendrás que aprender sobre desarrollo web. Puedes utilizar HTML, CSS y posiblemente JavaScript para crear la interfaz de usuario y mostrar los datos. Si no tienes experiencia en programación web, es posible que quieras considerar la contratación de un desarrollador web o explorar herramientas y plataformas que te ayuden a crear sitios web sin programar.

Recuerda que si tienes problemas técnicos o necesitas asistencia específica con la configuración de tu inversor, es recomendable contactar al instalador o al fabricante para obtener soporte técnico.

Pregunta: De acuerdo, creo que voy a optar por la opción de realizar mi propia página web. Dispongo de un ordenador que actualmente no tiene ningún sistema operativo. ¿Cómo empezamos?

Respuesta: Excelente elección. Aquí tienes los pasos básicos para comenzar a crear tu propia página web desde cero:

1. Instala un sistema operativo: Para empezar, necesitas un sistema operativo en tu ordenador. Puedes optar por sistemas como Windows, macOS o una distribución de Linux. Si no tienes preferencias y estás buscando una opción gratuita y versátil, puedes considerar instalar Ubuntu, que es una distribución de Linux popular y amigable para principiantes.

2. Aprende HTML y CSS: HTML (HyperText Markup Language) es el lenguaje utilizado para crear la estructura de una página web, mientras que CSS (Cascading Style Sheets) se utiliza para darle estilo y formato. Hay numerosos recursos en línea para aprender HTML y CSS de manera gratuita, como tutoriales en línea, cursos en plataformas como Codecademy o freeCodeCamp, y documentación oficial.

3. Elige un editor de código: Necesitarás un editor de código para escribir y editar tu código HTML y CSS. Hay opciones gratuitas y de pago disponibles. Algunas opciones populares son Visual Studio Code, Sublime Text y Atom. Estos editores ofrecen resaltado de sintaxis y otras características útiles para el desarrollo web.

4. Crea tu página web: Empieza con una página simple. Abre tu editor de código y crea un archivo HTML con la estructura básica. Luego, agrega contenido y utiliza CSS para darle estilo. Puedes comenzar por crear una página que muestre información sobre tu generación de energía solar.

5. Prueba tu página: Abre tu archivo HTML en un navegador web para ver cómo se ve y funciona. Realiza ajustes en tu código según sea necesario para lograr el diseño deseado.

6. Amplía tus conocimientos: A medida que ganes más confianza, puedes aprender más sobre JavaScript para agregar interactividad y funcionalidad avanzada a tu página web. También puedes explorar frameworks y bibliotecas populares como Bootstrap, que te ayudarán a crear diseños responsivos y atractivos de manera más sencilla.

7. Consigue un dominio y alojamiento: Una vez que hayas desarrollado tu página web, si deseas que esté en línea, necesitarás un nombre de dominio (la dirección web) y un servicio de alojamiento web donde puedas publicar tus archivos. Hay muchas empresas que ofrecen estos servicios.

Recuerda que la creación de una página web lleva tiempo y práctica, así que no te desanimes si al principio encuentras desafíos. ¡Con paciencia y dedicación, podrás lograrlo!

Pregunta: Guay, pues vamos a instalar Ubuntu. ¿Cómo lo hago?

Respuesta: ¡Genial! Instalar Ubuntu en tu ordenador es un proceso bastante sencillo. Aquí tienes los pasos generales que debes seguir:

Descarga la imagen de Ubuntu:

Ve al sitio web oficial de Ubuntu (<https://ubuntu.com/download>) desde otro dispositivo con conexión a Internet.

Descarga la versión de Ubuntu que desees. Recomendaría la versión LTS (Soporte a Largo Plazo) para mayor estabilidad.

Prepara una unidad USB de arranque:

Necesitarás una unidad USB vacía con al menos 4 GB de capacidad.

Descarga el programa "Etcher" (<https://www.balena.io/etcher/>) en tu dispositivo actual.

Ejecuta Etcher y utiliza la imagen de Ubuntu que descargaste para crear una unidad USB de arranque.

Arranca desde la unidad USB:

Inserta la unidad USB en tu ordenador.

Reinicia el ordenador y accede a la configuración de arranque. Esto generalmente se hace presionando una tecla como F2, F12 o Supr durante el proceso de inicio. La tecla exacta varía según la marca y modelo de tu computadora.

Selecciona la unidad USB como dispositivo de arranque.

Inicia el proceso de instalación:

El sistema arrancará desde la unidad USB y te presentará la opción de probar Ubuntu o instalarlo. Elige "Instalar Ubuntu".

Configura el idioma y la ubicación:

Sigue las instrucciones en pantalla para seleccionar tu idioma y ubicación.

Configura la conexión a Internet:

Si estás conectado a Internet durante la instalación, podrás descargar actualizaciones y software adicionales. Esto es recomendable.

Configura la instalación:

En la pantalla de instalación, elige "Instalar Ubuntu junto a [otro sistema operativo]" si tienes otro sistema instalado, o "Borrar disco e instalar Ubuntu" si deseas reemplazar todo.

Configura la zona horaria y el teclado:

Selecciona tu zona horaria y el tipo de teclado que usas.

Crea un usuario y contraseña:

Ingresa tus datos para crear un nombre de usuario y contraseña.

Espera a que finalice la instalación: Ubuntu comenzará a instalarse en tu disco duro. Esto puede llevar algunos minutos.

Reinicia tu ordenador:

Recuerda respaldar tus datos importantes antes de comenzar la instalación, ya que el proceso puede involucrar formateo de disco. Si tienes dudas o problemas durante la instalación, no dudes en buscar guías en línea específicas para tu situación o consultar el soporte de Ubuntu.

Pregunta: ¡Listo! Ya tengo la última versión de Ubuntu LTS en mi ordenador. ¿Cuál sería el siguiente paso para poder crear la página web con los datos de generación eléctrica?

Respuesta: ¡Excelente! Ahora que tienes Ubuntu instalado en tu ordenador, el siguiente paso es configurar un entorno de desarrollo web y comenzar a crear tu página web. Aquí tienes los pasos básicos para hacerlo:

Instala un servidor web:

En Ubuntu, puedes instalar Apache, que es uno de los servidores web más populares. Abre una terminal y ejecuta el siguiente comando para instalarlo:

```
sudo apt-get update
```

```
sudo apt-get install apache2
```

Crea una carpeta para tu sitio web:

En tu directorio personal (normalmente /home/tu_usuario), crea una carpeta para tu sitio web. Por ejemplo:

```
mkdir sitio_web
```

Coloca tus archivos web:

Coloca los archivos HTML, CSS y otros recursos de tu página web en la carpeta que creaste en el paso anterior. Puedes utilizar el editor de código que prefieras para crear y editar estos archivos.

Accede a tu sitio localmente:

Abre un navegador web en tu Ubuntu e ingresa la siguiente dirección en la barra de direcciones:

```
http://localhost/
```

Deberías ver tu página web local alojada en tu servidor Apache.

Conecta tu página a los datos del inversor:

Para mostrar los datos de generación eléctrica, necesitarás obtener esos datos desde el inversor. Puedes utilizar lenguajes de programación como Python para acceder a la dirección IP del inversor y extraer los datos.

Aprende sobre bases de datos (opcional):

Si deseas almacenar y mostrar los datos históricos de generación eléctrica, puedes explorar el uso de bases de datos como MySQL o SQLite para almacenar estos datos y luego mostrarlos en tu página web.

Estiliza y mejora tu página:

Utiliza tus conocimientos de HTML y CSS para darle estilo a tu página web y mejorar su apariencia.

Aprende JavaScript (opcional):

Si deseas agregar interactividad a tu página web, puedes aprender JavaScript para crear elementos interactivos, gráficos dinámicos y otras características.

Prueba en diferentes dispositivos:

Asegúrate de que tu página web se vea y funcione bien en diferentes dispositivos y tamaños de pantalla.

Considera alojamiento en línea:

Una vez que estés satisfecho con tu página web, podrías considerar alojarla en línea para que otros puedan acceder a ella. Hay varios servicios de alojamiento web disponibles.

Recuerda que la creación de una página web puede ser un proceso gradual, y siempre puedes mejorar y expandir tu página a medida que adquieres más conocimientos y habilidades en desarrollo web. ¡Buena suerte!

Pregunta: ¡Vale! Ya tengo instalado Apache2. Por lo que me gustaría saber cómo hago para obtener los datos del inversor y mostrarlos, tal y como has indicado en el paso 5.

Respuesta: Claro, obtener los datos del inversor y mostrarlos en tu página web requiere programación. En este caso, te guiaré a través de un enfoque general utilizando Python para obtener datos a través de la dirección IP del inversor. Ten en cuenta que los detalles específicos pueden variar dependiendo del modelo de inversor y cómo esté configurada su interfaz.

Aquí hay una idea de cómo podrías proceder:

Asegúrate de que Python esté instalado:

Si aún no tienes Python instalado en tu sistema Ubuntu, puedes instalarlo con el siguiente comando en la terminal:

```
sudo apt-get install python3
```

Utiliza la biblioteca requests:

La biblioteca requests en Python te permitirá hacer solicitudes HTTP a la dirección IP del inversor para obtener los datos. Si no la tienes instalada, puedes instalarla con el siguiente comando:

pip3 install requests

Escribe el código para obtener los datos:

Aquí hay un ejemplo simplificado de cómo podrías usar requests para obtener datos del inversor:

```
import requests
```

```
# URL para obtener los datos del inversor (esto puede variar según tu inversor)
```

```
url = "http://192.168.1.10/datos"
```

```
try:
```

```
response = requests.get(url)
```

```
data = response.json() # Suponiendo que los datos se devuelven en formato JSON
```

```
# Aquí procesas los datos y los muestras en tu página web
```

```
# Puedes usar Flask u otro enfoque para conectar estos datos con tu página web
```

```
except requests.exceptions.RequestException as e:
```

```
print("Error al obtener los datos:", e)
```

Conecta los datos a tu página web:

Para mostrar los datos en tu página web, tendrás que incorporarlos en tu código HTML. Puedes usar un enfoque como Flask para crear una aplicación web en Python que sirva tus datos y los muestre en tu página.

Actualiza los datos periódicamente:

Para mostrar los datos en tiempo real, tendrás que establecer un proceso que obtenga los datos del inversor a intervalos regulares y actualice tu página web con esos datos. Puedes utilizar JavaScript para realizar actualizaciones sin necesidad de recargar la página completa.

Ten en cuenta que este es un enfoque general y podrías necesitar adaptarlo a tu situación específica y las características de tu inversor. También, ten en cuenta las consideraciones de seguridad al trabajar con direcciones IP y datos de dispositivos conectados a tu red local. Si no te sientes cómodo programando esto por tu cuenta, considera buscar asistencia de un desarrollador web o un profesional en la materia.

Pregunta: *Huy... los comandos que me has enviado son un poco confusos. ¿No hay ninguna aplicación de programación visual para poder capturar los datos del inversor?*

Respuesta: *¡Por supuesto! Si prefieres una solución más visual y amigable para capturar los datos del inversor y mostrarlos en tu página web, puedes considerar utilizar un servicio de desarrollo web sin programación o una plataforma de creación de*

aplicaciones en línea. Algunas de estas opciones pueden ayudarte a lograr tu objetivo sin tener que escribir código manualmente. Aquí hay algunas opciones:

Google Sheets y Apps Script:

Puedes utilizar Google Sheets para capturar los datos y Apps Script para automatizar la extracción de datos desde tu inversor y su presentación en una hoja de cálculo. Luego, puedes usar Google Charts para crear gráficos y luego embeberlos en tu página web.

Node-RED:

Node-RED es una plataforma de programación visual que te permite crear flujos de datos de manera intuitiva. Puedes usarlo para capturar datos de tu inversor y mostrarlos en una interfaz gráfica.

IoT Plataformas:

Plataformas de Internet de las Cosas (IoT) como ThingSpeak o Blynk ofrecen herramientas para capturar y visualizar datos fácilmente. Puedes conectar tu inversor a estas plataformas y luego mostrar los datos en tu página web.

Plataformas de Monitoreo Energético:

Algunas plataformas específicas de monitoreo energético pueden ofrecer interfaces gráficas para capturar y mostrar los datos de generación eléctrica. Investigar si tu inversor es compatible con alguna de estas plataformas podría ser útil.

Estas alternativas pueden simplificar el proceso para alguien que no está familiarizado con la programación, aunque es posible que debas investigar un poco sobre cómo configurar y usar cada una de estas opciones. Recuerda que cada enfoque tendrá sus propias ventajas y limitaciones, por lo que es recomendable investigar y probar antes de decidir cuál utilizar.

Pregunta: En su momento usé Google Sheets, pero no he terminado de estar conforme con el resultado que obtuve. Podemos probar el segundo punto que has mencionado, Node-RED, que además has dicho que es intuitivo y tiene una interfaz gráfica, por lo que puede ser más sencillo. ¿Cómo lo instalo?

Respuesta: ¡Por supuesto! Node-RED es una excelente opción para crear flujos de datos sin programar de manera convencional. Aquí te explico cómo instalar Node-RED en tu sistema Ubuntu:

Instala Node.js:

Node-RED se ejecuta sobre Node.js, por lo que necesitarás tener Node.js instalado en tu sistema. Si aún no lo tienes, puedes instalarlo usando el siguiente comando en la terminal:

```
sudo apt-get install nodejs
```

Instala npm (Node Package Manager):

npm es el administrador de paquetes para Node.js. Puedes instalarlo con el siguiente comando:

```
sudo apt-get install npm
```

Instala Node-RED:

Una vez que tengas Node.js y npm instalados, puedes usar npm para instalar Node-RED. Ejecuta el siguiente comando:

```
csharp
```

```
sudo npm install -g --unsafe-perm node-red
```

Inicia Node-RED:

Una vez instalado, puedes iniciar Node-RED con el siguiente comando:

```
node-red
```

Accede a la interfaz gráfica:

Abre un navegador web en tu Ubuntu e ingresa la siguiente dirección en la barra de direcciones:

```
http://localhost:1880
```

Esto te llevará a la interfaz gráfica de Node-RED.

Crea tus flujos:

En Node-RED, puedes crear flujos de trabajo arrastrando y soltando nodos en la interfaz gráfica. Los nodos representan diferentes acciones y funciones. Puedes usar nodos para obtener datos del inversor y luego procesarlos y mostrarlos en tu página web.

Conecta con tu inversor:

Busca nodos en Node-RED que te permitan conectarte con tu inversor. Puedes explorar nodos relacionados con HTTP, MQTT u otros protocolos que puedan ser compatibles con tu inversor. Configura estos nodos para que obtengan los datos necesarios.

Crea una interfaz web:

Utiliza nodos para crear una interfaz web en Node-RED que muestre los datos de tu inversor. Puedes usar nodos de gráficos, tablas y otros elementos para mostrar la información de manera visual.

Despliega tus flujos:

Una vez que hayas creado tus flujos en Node-RED, desplégalos para que estén activos. Los cambios que hagas en la interfaz se reflejarán en tiempo real.

Recuerda que Node-RED ofrece mucha flexibilidad y versatilidad para crear flujos de datos, pero puede requerir cierto tiempo para familiarizarse con su interfaz y sus capacidades. Puedes encontrar documentación y tutoriales en línea para ayudarte a aprovechar al máximo esta plataforma.

Pregunta: *Entendido. La instalación de Node-Red ha sido exitosa y puedo acceder a su sitio web para realizar acciones. Sin embargo, me encuentro con una pantalla en blanco que muestra la etiqueta "Flow". ¿Podrías proporcionar una explicación más detallada sobre qué representa exactamente esta etiqueta "Flow"?*

Respuesta: *La ventana "Flow" en Node-RED es donde puedes diseñar y construir tus flujos de trabajo. Un flujo en Node-RED es una serie de nodos interconectados que realizan tareas específicas. Cada nodo representa una función o acción, y los flujos te permiten conectarlos para lograr tareas más complejas.*

Cuando abres Node-RED en tu navegador y ves la ventana blanca con el título "Flow", estás viendo el lienzo de diseño principal. Aquí es donde arrastrarás y conectarás los nodos para crear tus flujos. Para crear flujos y realizar acciones, sigue estos pasos básicos:

Agregar nodos:

En el panel izquierdo, encontrarás una lista de nodos disponibles. Puedes buscar nodos específicos o explorar las categorías para encontrar el nodo que necesitas.

Arrastra un nodo desde el panel izquierdo y suéltalo en el lienzo.

Conectar nodos:

Los nodos se conectan arrastrando un cable desde la salida de un nodo y soltándolo en la entrada de otro nodo. Los cables representan el flujo de datos entre los nodos.

Configurar nodos:

Doble clic en un nodo para abrir su configuración. Aquí puedes ajustar las propiedades del nodo, como las URL, los datos que se procesarán, etc.

Desplegar flujos:

Una vez que hayas configurado tus nodos como desees, asegúrate de hacer clic en el botón "Deploy" en la esquina superior derecha. Esto guardará y activará tus cambios, poniendo en funcionamiento los flujos.

Observar el flujo:

Siempre que hayas desplegado tu flujo, los nodos comenzarán a ejecutarse según la lógica que hayas diseñado. Puedes observar el flujo de datos en tiempo real a medida que pasan a través de los nodos.

En cuanto a la ventana blanca en la interfaz, esta es donde construyes visualmente tus flujos. Puedes agregar nodos, conectarlos y diseñar la lógica sin necesidad de escribir

código manualmente. Esta es la principal característica de Node-RED: su enfoque de programación visual que permite a personas sin experiencia en programación crear flujos complejos.

Explora las categorías de nodos y experimenta con diferentes configuraciones para familiarizarte con la forma en que funciona Node-RED. Puedes encontrar muchos tutoriales y recursos en línea para ayudarte a aprender y aprovechar esta poderosa herramienta.

Pregunta: *Vale, ¿qué nodos serían necesarios para conectarme con el inversor de Ingeteam?*

Respuesta: *Para conectarte con el inversor Ingeteam en Node-RED, necesitarás usar nodos que sean compatibles con el protocolo de comunicación que utiliza tu inversor. Según la información que proporcionaste, el modelo de tu inversor es el "INGECON SUN 3Play", pero es importante saber si este inversor utiliza algún protocolo específico para la comunicación, como HTTP, MQTT, Modbus, etc.*

Sin embargo, aquí te proporcionaré una orientación general sobre algunos nodos comunes que podrías considerar según diferentes protocolos:

HTTP Request Node:

Si el inversor tiene una interfaz web a la que puedas acceder a través de HTTP, puedes usar el nodo "HTTP Request" para hacer solicitudes GET o POST a la dirección IP del inversor. Esto te permitirá obtener los datos desde la interfaz web.

MQTT Nodes:

Si tu inversor es compatible con el protocolo MQTT, puedes usar nodos MQTT para suscribirte a los datos del inversor y recibirlos en tiempo real.

Modbus Nodes:

Si el inversor utiliza el protocolo Modbus, necesitarás nodos Modbus para comunicarte con él. Puedes utilizar nodos Modbus TCP o Modbus RTU según el tipo de comunicación que utilice tu inversor.

Exec Node:

Si el inversor proporciona datos mediante un programa o script ejecutable, puedes usar el nodo "Exec" para ejecutar ese programa y capturar la salida.

Recuerda que la elección de los nodos dependerá del protocolo de comunicación que use tu inversor. Debes investigar más sobre el protocolo específico y cómo acceder a los datos del inversor utilizando ese protocolo. Una vez que comprendas cómo comunicarte con el inversor, podrás seleccionar los nodos apropiados en Node-RED y configurarlos según tus necesidades.

Es importante mencionar que Node-RED tiene una amplia comunidad y muchos usuarios han compartido flujos y ejemplos para diferentes casos de uso, incluyendo la

comunicación con dispositivos como inversores solares. Puedes buscar ejemplos específicos en la comunidad de Node-RED para obtener orientación adicional.

Pregunta: De acuerdo, ¿Cómo se el protocolo que usa mi inversor?

Respuesta: Para determinar el tipo de protocolo que utiliza tu inversor, en este caso el INGECON SUN 3Play de Ingeteam, tendrás que consultar la documentación proporcionada por el fabricante o el instalador. Aquí hay algunas formas de obtener información sobre el protocolo de comunicación:

Manual del Usuario: Revisa el manual del usuario proporcionado por Ingeteam para el inversor. En esta guía, deberías encontrar detalles sobre cómo comunicarte con el inversor, incluyendo el tipo de protocolo que utiliza y las instrucciones para acceder a los datos.

Contacto con el Instalador o Fabricante: Si el manual no proporciona suficiente información, ponte en contacto con el instalador que colocó los paneles o directamente con el fabricante (Ingeteam). Ellos pueden proporcionarte información específica sobre cómo acceder a los datos del inversor y el protocolo de comunicación utilizado.

Interfaz Web del Inversor: Intenta acceder a la interfaz web del inversor utilizando su dirección IP en un navegador web. En algunos casos, podrás encontrar información sobre el protocolo utilizado y las opciones disponibles para la comunicación.

Etiqueta en el Inversor: A veces, los inversores pueden tener etiquetas o información en la carcasa que indique el protocolo de comunicación utilizado. Busca cualquier información técnica visible en el inversor.

Recursos en Línea: Busca en línea información relacionada con tu modelo de inversor. Puede haber discusiones en foros o sitios web especializados que mencionen el protocolo de comunicación que utiliza el inversor.

Una vez que tengas información sobre el protocolo de comunicación (por ejemplo, HTTP, MQTT, Modbus, etc.), podrás investigar cómo configurar los nodos adecuados en Node-RED para comunicarte con el inversor y obtener los datos de generación eléctrica.

Pregunta: De acuerdo, he mirado el manual y creo que es comunicación Modbus, ¿qué nodos necesito?

Respuesta: Si tu inversor utiliza comunicación Modbus, puedes utilizar nodos Modbus en Node-RED para establecer la comunicación y obtener los datos. A continuación, te describo los pasos y los nodos que podrías necesitar para configurar la comunicación Modbus en Node-RED:

Instalar Nodos Modbus:

Para comenzar, necesitarás instalar los nodos Modbus en Node-RED. Puedes hacerlo a través de la pestaña "Manage palette" en la interfaz de Node-RED. Busca "modbus" y selecciona e instala los nodos que sean apropiados para tu caso (Modbus TCP o Modbus RTU).

Configurar Nodo Modbus:

Después de instalar los nodos Modbus, arrastra un nodo Modbus desde el panel izquierdo al lienzo de Node-RED. Doble clic en el nodo para abrir su configuración. Aquí es donde establecerás los detalles de comunicación para conectarte con tu inversor Modbus.

Configurar Parámetros de Comunicación:

Dentro de la configuración del nodo Modbus, deberás especificar los parámetros de comunicación, como la dirección IP del inversor (si es Modbus TCP) o el puerto serie (si es Modbus RTU). Además, establece el número de dispositivo Modbus y el tipo de datos que deseas obtener.

Obtener Datos del Inversor:

Una vez que hayas configurado el nodo Modbus, tendrás la opción de elegir qué tipo de datos deseas obtener del inversor. Esto podría incluir datos como la producción actual, la producción acumulada, el voltaje, la corriente, etc.

Procesar y Mostrar Datos:

Después de obtener los datos del inversor, puedes utilizar otros nodos de Node-RED para procesar y formatear los datos según lo que desees mostrar en tu página web. Puedes usar nodos de función, nodos de cambio, y más para manipular los datos.

Crear una Interfaz Web:

Utiliza nodos en Node-RED para crear una interfaz web que muestre los datos que has obtenido del inversor. Puedes utilizar nodos de gráfico, tabla u otros elementos visuales para mostrar la información de manera atractiva.

Desplegar el Flujo:

Una vez que hayas configurado todos los nodos y diseñado tu flujo en Node-RED, asegúrate de hacer clic en el botón "Deploy" en la esquina superior derecha para activar tus cambios y ver cómo funcionan los nodos en tiempo real.

Recuerda que la configuración específica puede variar según los detalles del inversor y los nodos Modbus disponibles en Node-RED. Asegúrate de consultar la documentación del fabricante y de los nodos Modbus que estás utilizando para obtener instrucciones detalladas sobre la configuración.

Pregunta: De acuerdo, ya tengo los paquetes de Modbus instalados, ¿Qué nodo selecciono para comenzar?

Respuesta: Si ya tienes los paquetes de Modbus instalados en Node-RED y estás listo para comenzar a configurar la comunicación con tu inversor, el nodo que debes seleccionar depende del tipo de comunicación Modbus que utilice tu inversor (TCP o RTU/Serial).

A continuación, te proporciono información sobre los dos tipos principales de comunicación Modbus y los nodos que podrías considerar para cada caso:

Comunicación Modbus TCP:

Si tu inversor utiliza Modbus TCP para la comunicación, necesitas nodos que admitan esta variante del protocolo. Puedes considerar los siguientes nodos:

1. *node-red-contrib-modbus*: Este conjunto de nodos es muy versátil y ampliamente utilizado. Puedes utilizar nodos como "Modbus Read" para leer registros Modbus y "Modbus Write" para escribir en ellos. Configura estos nodos con los detalles de dirección IP y otros parámetros de comunicación.
2. *node-red-contrib-modbus-flex-getter*: Este nodo es altamente configurable y te permite trabajar con varios tipos de registros Modbus. Es especialmente útil cuando necesitas acceder a diferentes tipos de datos en tu inversor.

Comunicación Modbus RTU/Serial:

Si tu inversor utiliza Modbus RTU a través de una conexión serie (RS-232 o RS-485), necesitarás nodos que admitan este tipo de comunicación. Puedes considerar estos nodos:

1. *node-red-contrib-modbus-serial*: Este conjunto de nodos es específico para la comunicación Modbus RTU a través de conexiones serie. Configura los detalles de la conexión serie y los parámetros de comunicación en el nodo.
2. *node-red-contrib-serial-modbus*: Otro conjunto de nodos que también permite la comunicación Modbus RTU a través de conexiones serie. Puedes configurar los nodos para leer y escribir registros Modbus.

El nodo que elijas dependerá de la configuración específica de tu inversor y de tus necesidades de comunicación. Consulta la documentación de cada nodo para obtener instrucciones detalladas sobre cómo configurarlos correctamente, ya que deberás proporcionar detalles como direcciones de dispositivos, registros Modbus, velocidades de conexión serie, etc.

Una vez que hayas configurado los nodos según las especificaciones de tu inversor, podrás comenzar a obtener datos y trabajar con ellos en Node-RED.

Pregunta: Creo que usa Modbus TCP.

Respuesta: Si tu inversor utiliza Modbus TCP para la comunicación, puedes comenzar utilizando los nodos del paquete *node-red-contrib-modbus*, ya que este conjunto de nodos es adecuado para la comunicación Modbus TCP. A continuación, te guiaré a través de los pasos para configurar y utilizar los nodos Modbus TCP en Node-RED:

Agregar un Nodo Modbus Read:

Arrastra un nodo "Modbus Read" desde el panel izquierdo y colócalo en el lienzo de Node-RED.

Configurar el Nodo Modbus Read:

Doble clic en el nodo "Modbus Read" para abrir su configuración.

En la pestaña "Connection", configura la dirección IP y el puerto de tu inversor.

En la pestaña "Modbus", elige la función de lectura adecuada (por ejemplo, "Read Input Registers" o "Read Holding Registers").

Especifica la dirección del registro Modbus que deseas leer y la cantidad de registros.

Procesar los Datos:

Después de obtener los datos del inversor utilizando el nodo "Modbus Read", puedes procesarlos según tus necesidades. Puedes usar nodos de función, cambio, gráficos, etc., para darles formato y mostrarlos en tu interfaz web.

Desplegar el Flujo:

Haga clic en el botón "Deploy" en la esquina superior derecha para activar tus cambios y poner en funcionamiento el flujo.

Observar los Resultados:

Una vez que hayas desplegado el flujo, podrás observar los resultados en la interfaz web de Node-RED. Asegúrate de revisar la pestaña "Debug" para ver los datos obtenidos del inversor.

Ten en cuenta que los pasos pueden variar ligeramente dependiendo de la configuración específica de tu inversor y la versión de los nodos Modbus que estás utilizando. Asegúrate de consultar la documentación de los nodos y la información proporcionada por el fabricante de tu inversor para asegurarte de que estás configurando los parámetros correctos.

Con esta configuración, estarás listo para obtener datos de tu inversor utilizando la comunicación Modbus TCP en Node-RED y mostrarlos en tu interfaz web.

Pregunta: Vale, ya he configurado el nodo Modbus read, ahora ¿cómo compruebo que datos estoy recibiendo?

Respuesta: Después de haber configurado el nodo Modbus Read en Node-RED, puedes verificar los datos que estás recibiendo y utilizando el nodo de depuración (Debug) en Node-RED. Aquí te explico cómo hacerlo:

Agregar un Nodo Debug:

Desde el panel izquierdo de Node-RED, busca el nodo "Debug" y arrástralo al lienzo.

Conectar el Nodo Debug:

Conecta la salida del nodo Modbus Read a la entrada del nodo Debug. Esto se hace arrastrando desde la pequeña bola de salida del nodo Modbus Read hacia la entrada del nodo Debug.

Desplegar el Flujo:

Asegúrate de hacer clic en el botón "Deploy" en la esquina superior derecha para activar tus cambios y poner en funcionamiento el flujo.

Observar la Salida en la Consola:

Cuando el flujo esté en funcionamiento, puedes ver la salida del nodo Modbus Read en la consola de depuración. La consola de depuración se encuentra en la parte inferior de la interfaz de Node-RED. Puedes abrirlo haciendo clic en el botón "Debug" en la esquina inferior derecha.

Visualizar Datos en la Pestaña Debug:

Cuando los datos sean leídos por el nodo Modbus Read, verás información en la pestaña Debug. Allí podrás ver los datos obtenidos y verificar si son los datos que esperabas.

Usar los Datos en Otros Nodos:

Puedes conectar el nodo Debug a otros nodos en Node-RED para procesar y visualizar los datos de manera más conveniente. Por ejemplo, puedes usar nodos de función para dar formato a los datos antes de mostrarlos en una interfaz web.

Recuerda que los datos que verás en la consola de depuración dependerán de la configuración que hayas realizado en el nodo Modbus Read y de los registros Modbus que hayas seleccionado. Asegúrate de verificar la documentación de tu inversor para comprender cómo se estructuran los datos en los registros Modbus y cómo debes interpretarlos.

Usar el nodo Debug es una excelente manera de monitorear los datos en tiempo real y verificar que estás obteniendo la información correcta de tu inversor.

Pregunta: *Ya lo he puesto, pero algo no funciona bien. Creo que no estoy cogiendo los datos del inversor, volvamos al paso anterior. Acabo de arrastrar el nodo Modbus-read y lo estoy configurando ¿qué pongo en esos campos?*

Respuesta: *¡Por supuesto! Configurar correctamente el nodo Modbus Read es esencial para obtener los datos del inversor de manera exitosa. Aquí te guiaré a través de los campos principales que debes configurar en el nodo Modbus Read en Node-RED para una comunicación Modbus TCP:*

Name: *Proporciona un nombre descriptivo para el nodo, por ejemplo, "Lectura Inversor".*

Server: *Aquí debes configurar la dirección IP del inversor que estás tratando de conectar. Ingresa la dirección IP del inversor Modbus TCP.*

Port: *Especifica el puerto Modbus TCP que utiliza tu inversor. El puerto por defecto suele ser 502.*

Unit ID: *Ingresa el número de identificación de la unidad Modbus. En la mayoría de los casos, se utiliza 1, pero debes verificar la configuración de tu inversor.*

Function Code: *Selecciona el código de función Modbus adecuado para leer datos. Por ejemplo, si estás leyendo registros de entrada, puedes seleccionar "Read Input Registers". Si estás leyendo registros mantenidos, puedes seleccionar "Read Holding Registers".*

Data Address: *Ingresa la dirección de inicio del registro Modbus que deseas leer. Esta dirección puede variar según la configuración de tu inversor y los datos que necesitas.*

Data Length: *Especifica la cantidad de registros Modbus que deseas leer a partir de la dirección especificada. Esto depende de la estructura de los datos y de la cantidad de información que estás solicitando.*

Output Format: *Puedes seleccionar "Number" si esperas que los datos sean interpretados como números, o "Buffer" si deseas recibirlos como un búfer de bytes.*

Output: *Define en qué formato deseas recibir los datos. Puedes seleccionar "msg.payload" para recibir los datos en el campo "payload" del mensaje de salida.*

Interval: *Si deseas leer los datos en intervalos regulares, puedes habilitar esta opción y especificar el intervalo en milisegundos.*

Recuerda que los detalles específicos pueden variar según la configuración de tu inversor y la versión de los nodos Modbus que estás utilizando. Consulta la documentación de los nodos y de tu inversor para asegurarte de que estás configurando los parámetros correctos.

Una vez que hayas configurado los campos, asegúrate de hacer clic en "Done" para guardar la configuración. Luego, despliega el flujo haciendo clic en el botón "Deploy" para activar la comunicación con el inversor y ver los resultados en la consola de depuración.

Pregunta: *¿Qué campo pongo en Data Length?*

Respuesta: *El campo "Data Length" en el nodo Modbus Read se refiere a la cantidad de registros Modbus que deseas leer a partir de la dirección especificada. La elección del valor correcto en este campo depende de la estructura de los datos que estás solicitando y cómo están organizados en los registros Modbus.*

Un registro Modbus generalmente consiste en 16 bits (2 bytes), y los datos pueden estar organizados de diferentes maneras según el inversor y el tipo de información que estás tratando de obtener. Aquí hay algunos ejemplos para ayudarte a determinar el valor correcto para "Data Length":

Lectura de un Registro:

Si estás leyendo un solo registro Modbus, debes configurar "Data Length" en 1, ya que solo estás leyendo un conjunto de datos de 16 bits (2 bytes).

Lectura de Varios Registros:

Si estás leyendo varios registros consecutivos, debes considerar cuántos registros de 16 bits (2 bytes) estás leyendo. Por ejemplo, si deseas leer 3 registros consecutivos, configurarás "Data Length" en 3.

Lectura de Datos más Complejos:

Si estás leyendo datos más complejos, como valores flotantes o datos estructurados, deberás considerar cómo se almacenan y cuántos registros Modbus abarcan esos datos. Es posible que necesites leer varios registros para obtener un solo valor.

Es importante consultar la documentación de tu inversor para comprender cómo se organizan los datos en los registros Modbus y cuántos registros necesitas leer para obtener la información deseada. Una vez que comprendas cómo se estructuran los datos, podrás configurar correctamente el campo "Data Length" en el nodo Modbus Read en Node-RED.

Pregunta: ¿Cuál sería el data address?

Respuesta: El campo "Data Address" en el nodo Modbus Read se refiere a la dirección de inicio del registro Modbus que deseas leer. La dirección que debes ingresar en este campo depende de cómo se organizan los datos en tu inversor y cómo se mapean a los registros Modbus.

En el protocolo Modbus, los registros son direcciones numéricas que se utilizan para acceder a diferentes tipos de datos almacenados en el dispositivo. Estas direcciones pueden variar según el fabricante y el modelo del dispositivo. Aquí hay algunas pautas generales para determinar la dirección que debes usar:

Consulta la Documentación del Inversor: La forma más segura de determinar la dirección correcta es consultando la documentación del fabricante o el manual del usuario de tu inversor. Por lo general, encontrarás una tabla que muestra cómo se mapean los datos a las direcciones Modbus.

Identifica el Tipo de Dato: Si estás buscando un dato específico, como la producción actual de energía, consulta cómo se mapea ese dato en los registros Modbus en la documentación del inversor.

Ten en Cuenta la Dirección Base: Algunos inversores comienzan a mapear registros Modbus desde una dirección base, como 40001 o 30001. A partir de esa dirección base, cada registro subsiguiente se incrementa en uno. Por ejemplo, el registro 40001 se refiere a la dirección base, el registro 40002 se encuentra en la siguiente dirección, y así sucesivamente.

Observa la Estructura de Datos: Si tienes acceso a los registros Modbus en tu inversor, puedes examinar la estructura de los datos para determinar qué registros y direcciones están asociados con la información que necesitas.

En resumen, la dirección que debes ingresar en el campo "Data Address" dependerá de cómo se organizan los datos en el inversor y de cómo se mapean a los registros Modbus. Consulta la documentación del inversor o los datos técnicos para obtener la información precisa sobre cómo acceder a los datos que deseas.

Pregunta: *De acuerdo, ya está mostrando en pantalla algunos valores que espero que estén bien. ¿Cómo sé que los valores son esos?*

Respuesta: *Es una excelente noticia que estés viendo valores en pantalla. Ahora, para asegurarte de que los valores sean correctos, es importante verificarlos utilizando la documentación de tu inversor y tener un conocimiento básico de cómo se almacenan los datos en los registros Modbus. Aquí hay algunos pasos que puedes seguir para verificar los valores:*

Consulta la Documentación del Inversor:

Revisa la documentación proporcionada por el fabricante de tu inversor. Busca la tabla de registros Modbus o la descripción de cómo se mapean los datos. Esto te permitirá comprender qué datos corresponden a las direcciones Modbus que estás leyendo.

Comprende la Estructura de los Datos:

Asegúrate de comprender cómo se almacenan los datos en los registros Modbus. Por ejemplo, algunos valores pueden estar codificados como números de punto flotante o enteros sin signo. Asegúrate de aplicar la interpretación adecuada a los datos que estás viendo.

Comparación con Valores Reales:

Si tienes la capacidad de comparar los valores mostrados en pantalla con valores reales, como medidores físicos, esto puede ser una forma efectiva de verificar su precisión. Compara la producción de energía mostrada en la interfaz de Node-RED con la producción de energía que estás obteniendo en la vida real.

Realiza Pruebas de Cambio:

Si es posible, realiza cambios en el inversor que deberían afectar los valores mostrados. Por ejemplo, si cambias la configuración de la producción de energía, verifica si los valores en la interfaz de Node-RED también cambian de acuerdo a lo esperado.

Compara con Otras Fuentes:

Si tienes acceso a otras fuentes de datos, como sistemas de monitoreo independientes, compara los valores mostrados en Node-RED con los valores de esas fuentes para corroborar su precisión.

Recuerda que la precisión de los valores mostrados en Node-RED dependerá de cómo se hayan configurado los nodos Modbus Read y cómo se interpreten los datos. Si tienes dudas sobre los resultados, consulta la documentación de tu inversor y, si es posible, busca la asistencia de un profesional o un experto en sistemas de energía solar.

Pregunta: *¿Cómo comparo con los valores reales?*

Respuesta: *Comparar los valores mostrados en Node-RED con los valores reales requerirá acceso a datos medidos o estimados de manera independiente. Aquí tienes algunas formas de llevar a cabo esta comparación:*

Medidores Físicos: *Si tienes medidores físicos instalados en tu sistema solar que miden la producción de energía, como medidores de corriente o voltaje, puedes comparar los valores que muestran estos medidores con los valores que ves en la interfaz de Node-RED. Esto te permitirá verificar si los valores son consistentes.*

Medidores Independientes: *Si tienes acceso a sistemas de monitoreo independientes de terceros que miden la producción de energía, como sistemas de monitoreo en línea o aplicaciones móviles proporcionadas por el fabricante de tus paneles solares o inversor, compara los datos de estos sistemas con los valores en Node-RED.*

Datos Históricos: *Si has estado registrando datos históricos de producción de energía, compara los registros históricos con los valores que estás viendo en Node-RED. Esto te ayudará a determinar si los valores actuales son consistentes con el rendimiento previo.*

Cambios en la Configuración: *Realiza cambios en la configuración de tu sistema solar y observa cómo responden los valores en Node-RED. Por ejemplo, si ajustas la inclinación de los paneles solares o cambias la configuración del inversor, deberías ver cambios correspondientes en la producción de energía.*

Profesional de Energía Solar: *Si tienes dudas sobre la precisión de los valores y la comparación con los valores reales, considera consultar con un profesional de energía solar. Un experto podrá ayudarte a verificar la precisión de tus mediciones y realizar pruebas para asegurarse de que tu sistema esté funcionando correctamente.*

Recuerda que la comparación con valores reales puede ser importante para verificar la precisión de tus mediciones y para identificar cualquier problema en tu sistema solar. Siempre es recomendable contar con una fuente de datos confiable y realizar pruebas y ajustes cuando sea necesario para garantizar un funcionamiento óptimo.

Pregunta: *¿Cómo veo los datos reales de inversor con el fin de compararlos con el de Node-RED?*

Respuesta: *Para comparar los datos reales de tu inversor con los datos mostrados en Node-RED, necesitas acceder a la interfaz o plataforma de monitoreo proporcionada por el fabricante de tu inversor. Aquí hay pasos generales que puedes seguir para acceder a los datos reales del inversor:*

Consultar la Documentación: Revisa el manual del usuario o la documentación proporcionada por el fabricante de tu inversor. La documentación generalmente incluirá información sobre cómo acceder a la interfaz de monitoreo y obtener los datos en tiempo real.

Acceder a la Interfaz Web del Inversor:

Abre un navegador web en tu computadora o dispositivo.

Ingresa la dirección IP del inversor en la barra de direcciones del navegador. La dirección IP es la misma que configuraste en el nodo Modbus Read en Node-RED.

Es posible que necesites proporcionar credenciales de inicio de sesión si el inversor está protegido por contraseña.

Explorar la Interfaz de Monitoreo:

Una vez que hayas ingresado a la interfaz web del inversor, explora las opciones y pestañas disponibles.

Busca secciones que muestren la producción de energía actual, la producción acumulada, la tensión, la corriente y otros datos relevantes.

Compara los Valores:

Mientras estás en la interfaz de monitoreo del inversor, compara los valores mostrados en esta interfaz con los valores que estás viendo en Node-RED.

Presta atención a las unidades de medida (vatios, kilovatios, amperios, etc.) para asegurarte de que estás comparando valores equivalentes.

Realiza Cambios y Observa las Respuestas:

Si es posible, realiza cambios en la configuración del inversor (por ejemplo, ajusta la inclinación de los paneles) y observa cómo responden los valores en tiempo real en la interfaz de monitoreo.

Recuerda que cada fabricante de inversores puede tener una interfaz de monitoreo diferente, por lo que los pasos exactos pueden variar. Si tienes dificultades para acceder a la interfaz de monitoreo o interpretar los valores, te recomiendo consultar la documentación del inversor o contactar al servicio de atención al cliente del fabricante para obtener asistencia específica.

Pregunta: *Hola, estoy realizando un proyecto para ver los datos de generación eléctrica de mis paneles fotovoltaicos de casa. He estado hablando contigo anteriormente e hicimos una instalación de Node-RED.*

Respuesta: *¡Hola de nuevo! Claro, recuerdo haber hablado contigo sobre Node-RED y tu proyecto de paneles fotovoltaicos. Me alegra saber que estás avanzando con tu proyecto. ¿En qué puedo ayudarte esta vez? ¿Necesitas asistencia con alguna parte específica de la visualización de los datos de generación eléctrica de tus paneles solares?*

Pregunta: Me gustaría pasar esos datos a una web, para ello tengo un Ubuntu con Apache2.

Respuesta: Entiendo, deseas mostrar los datos de generación eléctrica de tus paneles solares en una página web utilizando un servidor Apache en Ubuntu. Aquí hay un enfoque general para lograr esto:

Obtener los Datos: Asegúrate de que los datos de generación eléctrica de tus paneles fotovoltaicos estén siendo capturados y almacenados en algún formato. Puedes usar Node-RED para recopilar estos datos y guardarlos en una base de datos, un archivo CSV o en algún otro formato que sea conveniente para ti.

Crear una Página Web: Diseña una página web simple que muestre los datos de generación eléctrica. Puedes usar HTML, CSS y JavaScript para crear una interfaz agradable y fácil de entender. Si tienes experiencia con frameworks de frontend como React, Vue.js o Angular, también podrías considerar utilizarlos.

Configurar Apache: Asegúrate de que Apache esté instalado y funcionando en tu servidor Ubuntu. Puedes instalarlo yendo a la terminal y ejecutando los siguientes comandos:

```
sudo apt update
```

```
sudo apt install apache2
```

Ubicación de la Página Web: Coloca los archivos HTML, CSS y JavaScript de tu página web en el directorio raíz de tu servidor web Apache. Por defecto, este directorio suele ser `/var/www/html/`. Puedes copiar los archivos a ese directorio.

Acceso a los Datos: En tu página web, necesitarás acceder a los datos de generación eléctrica. Puedes hacerlo utilizando tecnologías como AJAX para hacer solicitudes al servidor donde Node-RED está almacenando los datos.

Actualización Automática: Si deseas que los datos se actualicen en tiempo real en la página web, puedes usar técnicas como AJAX polling o WebSockets para obtener actualizaciones periódicas de los datos.

Proteger el Acceso: Dependiendo de tus necesidades, podrías considerar la autenticación y autorización para proteger el acceso a la página web y a los datos.

Configurar Dominio (Opcional): Si deseas acceder a la página web utilizando un nombre de dominio en lugar de la dirección IP del servidor, puedes configurar un dominio y configurar la configuración de DNS para apuntar al servidor.

Ten en cuenta que este es un proceso general y puede haber detalles específicos en función de tus necesidades y configuración. Si tienes preguntas específicas o encuentras obstáculos en algún paso, no dudes en preguntar para obtener orientación más detallada.

Pregunta: Bien. Vamos a crear la página web entonces. Me gustaría que la página web tenga el siguiente formato:

Quiero que dicha web este organizada de la siguiente forma: Tres divs principales (superior, medio e inferior). Los cuales tendrán 15vh 70vh y 15vh y que ocupen todo el ancho de la pantalla. Y un margen de 5% a izquierda y derecha. Y que el superior tenga un color de fondo rosa, medio un color de fondo azul e inferior un color de fondo rojo.

A su vez quiero que el div superior este dividido en superior-izquierda y superior-derecha, los cuales ocuparán el 50% del ancho del div superior. A su vez quiero que superior-izquierda sea de color verde y superior-derecha sea de color naranja.

A su vez quiero que el div inferior este dividido en inferior-izquierda e inferior-derecha, los cuales ocuparán el 50% del ancho del div inferior. A su vez quiero que inferior-izquierda sea de color oro e inferior-derecha sea de color blanco.

Por último en el superior-derecha quiero añadir tres botones "SP", "ENG" y "EUS", los cuales estarán centrados verticalmente y situados a la derecha del div.

Quiero también añadir en el div superior-izquierdo dos fotos que ocupen el 80% de la altura del div, pero que el ancho se ajuste a la escala real de la foto. Ambas fotos tienen que ir una al lado de la otra separadas por un div negro de 2 pixeles de ancho.

Quiero también añadir en el div inferior-derecha tres fotos que ocupen el 80% de la altura del div, pero que el ancho se ajuste a la escala real de la foto. Ambas fotos tienen que ir una al lado de la otra separadas por un div negro de 2 pixeles de ancho.

Respuesta: Para crear la página web con las especificaciones que has dado, necesitarás los archivos `index.html` y `styles.css`. A continuación, te proporcionaré el código que cumple con todas las condiciones que has mencionado:

`index.html:`

```
html
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <link rel="stylesheet" href="styles.css">
```

```
  <title>Mi página web</title>
```

```
</head>
```

```
<body>
```

```
  <div class="main-container">
```

```
    <div class="superior">
```

```
<div class="superior-izquierda">
  
  <div class="separator"></div>
  
</div>

<div class="superior-derecha">
  <button>SP</button>
  <button>ENG</button>
  <button>EUS</button>
</div>

</div>

<div class="medio">
  <!-- Contenido del div medio (opcional) -->
</div>

<div class="inferior">
  <div class="inferior-izquierda">
    
    <div class="separator"></div>
    
  </div>

  <div class="inferior-derecha">
    
    <div class="separator"></div>
    
  </div>
</div>

</div>

</body>
</html>
```



```
styles.css:
css
body, html {
  margin: 0;
  padding: 0;
}
.main-container {
  display: flex;
  flex-direction: column;
  height: 100vh;
  margin: 0 5%;
}
.superior, .medio, .inferior {
  width: 100%;
  display: flex;
  align-items: center;
  justify-content: center;
}
.superior {
  height: 15vh;
  background-color: pink;
}
.medio {
  height: 70vh;
  background-color: blue;
}
.inferior {
  height: 15vh;
  background-color: red;
```

```
}  
.superior-izquierda, .superior-derecha, .inferior-izquierda, .inferior-derecha {  
  height: 100%;  
}  
.superior-izquierda {  
  width: 50%;  
  background-color: green;  
}  
.superior-derecha {  
  width: 50%;  
  background-color: orange;  
  display: flex;  
  justify-content: flex-end;  
  align-items: center;  
}  
.inferior-izquierda {  
  width: 50%;  
  background-color: gold;  
}  
.inferior-derecha {  
  width: 50%;  
  background-color: white;  
}  
.separator {  
  width: 2px;  
  height: 80%;  
  background-color: black;  
  margin: 0 10px;  
}
```

```
img {
```

```
  height: 80%;
```

```
  width: auto;
```

```
}
```

Asegúrate de reemplazar los nombres de las imágenes (imagen1.jpg, imagen2.jpg, etc.) en el archivo index.html con las ubicaciones de tus imágenes reales. Además, puedes añadir el contenido adicional al div .medio si lo necesitas.

Este código cumple con todas tus especificaciones, creando la página web dividida en los tres divs principales con sus respectivas subdivisiones y estilos de color y margen. Las imágenes en los divs .superior-izquierda e .inferior-derecha ocuparán el 80% de la altura y ajustarán automáticamente su ancho para mantener su escala original.

Pregunta: *¿Qué tendría que cambiar en el css de img para que se centren verticalmente en altura?*

Respuesta: *Para centrar verticalmente las imágenes dentro de sus contenedores, puedes utilizar el siguiente código CSS en la sección de estilos:*

```
css
```

```
img {
```

```
  height: 80%;
```

```
  width: auto;
```

```
  display: block;
```

```
  margin: auto;
```

```
}
```

Agregamos la propiedad display: block; para asegurarnos de que la imagen se comporte como un bloque y ocupe todo el ancho disponible dentro de su contenedor. Luego, utilizamos margin: auto; para centrar la imagen vertical y horizontalmente dentro de su contenedor. Con estos cambios, las imágenes se centrarán tanto vertical como horizontalmente en la altura del div contenedor.

Pregunta: *Cambiar el img la parte de margin ya que solo quiero que se centre en altura.*

Respuesta: *Mis disculpas por la confusión. Para centrar verticalmente las imágenes sin afectar su posición horizontal, debes utilizar la propiedad display: flex; en el contenedor de las imágenes y luego establecer la propiedad align-items: center; en ese contenedor. De esta forma, las imágenes se centrarán verticalmente en su div contenedor sin afectar su posición horizontal. Aquí está el código actualizado para lograr esto:*

```
styles.css:
css
body, html {
  margin: 0;
  padding: 0;
}
.main-container {
  display: flex;
  flex-direction: column;
  height: 100vh;
  margin: 0 5%;
}
.superior, .medio, .inferior {
  width: 100%;
  display: flex;
  align-items: center;
  justify-content: center;
}
.superior {
  height: 15vh;
  background-color: pink;
}
.medio {
  height: 70vh;
  background-color: blue;
}
.inferior {
  height: 15vh;
  background-color: red;
```

```
}  
.superior-izquierda, .superior-derecha, .inferior-izquierda, .inferior-derecha {  
  height: 100%;  
}  
.superior-izquierda {  
  width: 50%;  
  background-color: green;  
}  
.superior-derecha {  
  width: 50%;  
  background-color: orange;  
  display: flex;  
  justify-content: flex-end;  
  align-items: center;  
}  
.inferior-izquierda {  
  width: 50%;  
  background-color: gold;  
}  
.inferior-derecha {  
  width: 50%;  
  background-color: white;  
  display: flex;  
  align-items: center;  
}  
.separator {  
  width: 2px;  
  height: 80%;  
  background-color: black;
```

```
margin: 0 10px;
}
img {
height: 80%;
width: auto;
display: block;
margin: auto 0; /* Solo centramos verticalmente */
}
```

Con estos cambios, las imágenes dentro de los divs `.superior-izquierda` e `.inferior-derecha` se centrarán verticalmente en la altura de sus respectivos contenedores. La propiedad `margin: auto 0;` establece un margen automático en la parte superior e inferior de las imágenes, lo que centra verticalmente las imágenes.

Pregunta: Pero las imágenes tienen que ir una al lado de la otra, no una encima de la otra.

Respuesta: Mis disculpas nuevamente por la confusión. Para colocar las imágenes una al lado de la otra en lugar de una encima de la otra, debes usar `display: flex;` en los contenedores `.superior-izquierda` y `.inferior-derecha`, y además, establecer la propiedad `flex-direction: row;` para que los elementos se coloquen en una fila. Aquí tienes el código actualizado para lograr esto:

styles.css:

css

```
body, html {
margin: 0;
padding: 0;
}
.main-container {
display: flex;
flex-direction: column;
height: 100vh;
margin: 0 5%;
}
.superior, .medio, .inferior {
```

```
width: 100%;  
display: flex;  
align-items: center;  
justify-content: center;  
}  
.superior {  
height: 15vh;  
background-color: pink;  
}  
.medio {  
height: 70vh;  
background-color: blue;  
}  
.inferior {  
height: 15vh;  
background-color: red;  
}  
.superior-izquierda, .superior-derecha, .inferior-izquierda, .inferior-derecha {  
height: 100%;  
display: flex; /* Usamos flexbox para colocar las imágenes una al lado de la otra */  
align-items: center;  
}  
.superior-izquierda {  
width: 50%;  
background-color: green;  
flex-direction: row; /* Colocar los elementos en una fila */  
}  
.superior-derecha {  
width: 50%;
```

```
background-color: orange;
display: flex;
justify-content: flex-end;
align-items: center;
}
.inferior-izquierda {
width: 50%;
background-color: gold;
flex-direction: row; /* Colocar los elementos en una fila */
}
.inferior-derecha {
width: 50%;
background-color: white;
display: flex;
align-items: center;
flex-direction: row; /* Colocar los elementos en una fila */
}
.separator {
width: 2px;
height: 80%;
background-color: black;
margin: 0 10px;
}
img {
height: 80%;
width: auto;
display: block;
margin: auto 5px; /* Centrar verticalmente con un pequeño espacio entre las imágenes
*/
}
```


Con estos cambios, las imágenes dentro de los divs `.superior-izquierda` e `.inferior-derecha` se colocarán una al lado de la otra con un pequeño espacio entre ellas, manteniendo la alineación vertical dentro de sus respectivos contenedores.

Pregunta: Pásame el color R 250 G 247 B 233 a hexadecimal.

Respuesta: Para convertir los valores de color RGB (Rojo, Verde, Azul) en hexadecimal, primero tienes que dividir cada valor entre 16 y luego obtener el residuo (resto) en cada caso. Luego, esos residuos se convierten en sus equivalentes hexadecimales. Aquí está el cálculo:

$R = 250 \rightarrow 250 / 16 = 15$ con residuo 10 (A en hexadecimal)

$G = 247 \rightarrow 247 / 16 = 15$ con residuo 7 (7 en hexadecimal)

$B = 233 \rightarrow 233 / 16 = 14$ con residuo 9 (9 en hexadecimal)

Entonces, el valor hexadecimal sería #FAF7E9.

Pregunta: Quiero que una foto en concreto de la web tenga otro porcentaje de altura, de un 70%.

Respuesta: Para ajustar la altura de una foto específica de la web a un 70% del tamaño del contenedor en lugar del 80% utilizado anteriormente, simplemente necesitas agregar una clase especial para esa imagen y establecer el porcentaje de altura deseado en esa clase. Aquí te muestro cómo hacerlo:

index.html:

```
html
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <link rel="stylesheet" href="styles.css">
```

```
  <title>Mi página web</title>
```

```
</head>
```

```
<body>
```

```
  <div class="main-container">
```

```
    <div class="superior">
```

```
      <div class="superior-izquierda">
```

```

<div class="separator"></div>

</div>
<div class="superior-derecha">
  <button>SP</button>
  <button>ENG</button>
  <button>EUS</button>
</div>
</div>
<div class="medio">
  <!-- Contenido del div medio (opcional) -->
</div>
<div class="inferior">
  <div class="inferior-izquierda">
    
    <div class="separator"></div>
    
  </div>
  <div class="inferior-derecha">
    
    <div class="separator"></div>
    
  </div>
</div>
</div>
</body>
</html>
styles.css:
```

```
css
body, html {
  margin: 0;
  padding: 0;
}
.main-container {
  display: flex;
  flex-direction: column;
  height: 100vh;
  margin: 0 5%;
}
.superior, .medio, .inferior {
  width: 100%;
  display: flex;
  align-items: center;
  justify-content: center;
}
.superior {
  height: 15vh;
  background-color: pink;
}
.medio {
  height: 70vh;
  background-color: blue;
}
.inferior {
  height: 15vh;
  background-color: red;
}
```

```
.superior-izquierda, .superior-derecha, .inferior-izquierda, .inferior-derecha {  
    height: 100%;  
    display: flex; /* Usamos flexbox para colocar las imágenes una al lado de la otra */  
    align-items: center;  
}  
  
.superior-izquierda {  
    width: 50%;  
    background-color: green;  
    flex-direction: row; /* Colocar los elementos en una fila */  
}  
  
.superior-derecha {  
    width: 50%;  
    background-color: orange;  
    display: flex;  
    justify-content: flex-end;  
    align-items: center;  
}  
  
.inferior-izquierda {  
    width: 50%;  
    background-color: gold;  
    flex-direction: row; /* Colocar los elementos en una fila */  
}  
  
.inferior-derecha {  
    width: 50%;  
    background-color: white;  
    display: flex;  
    align-items: center;  
    flex-direction: row; /* Colocar los elementos en una fila */  
}
```

```
.separator {  
    width: 2px;  
    height: 80%;  
    background-color: black;  
    margin: 0 10px;  
}  
  
img {  
    height: 80%;  
    width: auto;  
    display: block;  
    margin: auto 5px; /* Centrar verticalmente con un pequeño espacio entre las imágenes */  
*/*  
}  
  
.special-image {  
    height: 70%; /* Establecer el porcentaje de altura deseado para la imagen especial */  
}
```

Ahora, la clase `.special-image` se aplicará solo a las imágenes dentro del `div .superior-izquierda`, y ajustará su altura al 70% del tamaño del contenedor. El resto de las imágenes dentro de `.superior-izquierda` y `.inferior-derecha` conservarán la altura del 80% como se había establecido inicialmente.

Pregunta: Quiero que los botones sean visualmente más atractivos, que solo se vea el texto.

Respuesta: Para hacer que los botones sean visualmente más atractivos y solo muestren el texto sin ningún fondo ni borde visible, podemos aplicar un estilo más sutil usando CSS. Puedes utilizar pseudo-elementos (`::before` y `::after`) para crear un efecto de resaltado en los botones al pasar el cursor sobre ellos. Aquí tienes el código para lograrlo:

`index.html:`

```
html
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="styles.css">
<title>Mi página web</title>
</head>
<body>
  <div class="main-container">
    <div class="superior">
      <div class="superior-izquierda">
        
        <div class="separator"></div>
        
      </div>
      <div class="superior-derecha">
        <button>SP</button>
        <button>ENG</button>
        <button>EUS</button>
      </div>
    </div>
    <div class="medio">
      <!-- Contenido del div medio (opcional) -->
    </div>
    <div class="inferior">
      <div class="inferior-izquierda">
        
        <div class="separator"></div>
        
      </div>
      <div class="inferior-derecha">
        
      </div>
    </div>
  </div>
</body>
```

```
<div class="separator"></div>

</div>
</div>
</div>
</body>
</html>
```

styles.css:

css

```
body, html {
```

```
  margin: 0;
```

```
  padding: 0;
```

```
}
```

```
.main-container {
```

```
  display: flex;
```

```
  flex-direction: column;
```

```
  height: 100vh;
```

```
  margin: 0 5%;
```

```
}
```

```
.superior, .medio, .inferior {
```

```
  width: 100%;
```

```
  display: flex;
```

```
  align-items: center;
```

```
  justify-content: center;
```

```
}
```

```
.superior {
```

```
  height: 15vh;
```

```
  background-color: pink;
```

```
}
```

```
.medio {  
    height: 70vh;  
    background-color: blue;  
}  
.inferior {  
    height: 15vh;  
    background-color: red;  
}  
.superior-izquierda, .superior-derecha, .inferior-izquierda, .inferior-derecha {  
    height: 100%;  
    display: flex; /* Usamos flexbox para colocar las imágenes una al lado de la otra */  
    align-items: center;  
}  
.superior-izquierda {  
    width: 50%;  
    background-color: green;  
    flex-direction: row; /* Colocar los elementos en una fila */  
}  
.superior-derecha {  
    width: 50%;  
    background-color: orange;  
    display: flex;  
    justify-content: flex-end;  
    align-items: center;  
}  
.inferior-izquierda {  
    width: 50%;  
    background-color: gold;  
    flex-direction: row; /* Colocar los elementos en una fila */
```



```
}  
.inferior-derecha {  
    width: 50%;  
    background-color: white;  
    display: flex;  
    align-items: center;  
    flex-direction: row; /* Colocar los elementos en una fila */  
}  
.separator {  
    width: 2px;  
    height: 80%;  
    background-color: black;  
    margin: 0 10px;  
}  
img {  
    height: 80%;  
    width: auto;  
    display: block;  
    margin: auto 5px; /* Centrar verticalmente con un pequeño espacio entre las imágenes */  
}  
.special-image {  
    height: 70%; /* Establecer el porcentaje de altura deseado para la imagen especial */  
}  
/* Estilos para los botones */  
button {  
    background: transparent;  
    border: none;  
    color: #000;  
    font-size: 16px;
```

```
    cursor: pointer;
    position: relative;
    z-index: 1;
    padding: 5px 15px;
}
button::before {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    right: 0;
    bottom: 0;
    z-index: -1;
    background: #e1e1e1;
    border-radius: 5px;
    opacity: 0;
    transition: opacity 0.2s ease;
}
button:hover::before {
    opacity: 1;
}
```

Con estos cambios, los botones se mostrarán como simples textos hasta que el cursor se sitúe sobre ellos, momento en el cual aparecerá un efecto de resaltado suave alrededor del texto del botón. Puedes ajustar el color de fondo del efecto de resaltado modificando el valor en la propiedad `background` de `button::before`.

Pregunta: Color de texto del botón.

Respuesta: Para cambiar el color de texto del botón, puedes agregar la propiedad `color` en la regla de estilo CSS para los botones. Por ejemplo, si deseas que el color de texto sea blanco, puedes hacer lo siguiente:

CSS

```
/* Estilos para los botones */
```

```
button {  
    background: transparent;  
    border: none;  
    color: white; /* Cambia el color de texto del botón a blanco */  
    font-size: 16px;  
    cursor: pointer;  
    position: relative;  
    z-index: 1;  
    padding: 5px 15px;  
}  
/* Resto de estilos para el efecto de resaltado (puedes mantenerlo igual) */  
button::before {  
    content: "";  
    position: absolute;  
    top: 0;  
    left: 0;  
    right: 0;  
    bottom: 0;  
    z-index: -1;  
    background: #e1e1e1;  
    border-radius: 5px;  
    opacity: 0;  
    transition: opacity 0.2s ease;  
}  
button:hover::before {  
    opacity: 1;  
}
```

Con estos cambios, los botones tendrán un texto blanco y mantendrán el efecto de resaltado que aparece al pasar el cursor sobre ellos. Puedes ajustar el color de texto a cualquier otro valor válido en CSS para lograr el aspecto deseado.

Pregunta: Hacer que un div se vea www.google.es.

Respuesta: Para mostrar el contenido de una página web dentro de un div de tu sitio, puedes utilizar un elemento `iframe` en HTML. El `iframe` te permite cargar una página web externa dentro de tu página. Aquí tienes el código para lograrlo:

`index.html:`

```
html
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <link rel="stylesheet" href="styles.css">
```

```
  <title>Mi página web</title>
```

```
</head>
```

```
<body>
```

```
  <div class="main-container">
```

```
    <div class="superior">
```

```
      <div class="superior-izquierda">
```

```
        
```

```
        <div class="separator"></div>
```

```
        
```

```
      </div>
```

```
    <div class="superior-derecha">
```

```
      <button>SP</button>
```

```
      <button>ENG</button>
```

```
      <button>EUS</button>
```

```
    </div>
```

```
</div>
```

```
  <div class="medio">
```

```
    <iframe src="https://www.google.es" frameborder="0"></iframe>
```

```
</div>
<div class="inferior">
  <div class="inferior-izquierda">
    
    <div class="separator"></div>
    
  </div>
  <div class="inferior-derecha">
    
    <div class="separator"></div>
    
  </div>
</div>
</div>
</div>
</body>
</html>
styles.css:
css
/* Tu CSS existente aquí (no es necesario cambiarlo para mostrar la página
www.google.es) */
En este ejemplo, hemos agregado un iframe dentro del div con la clase .medio. El atributo
src del iframe apunta a la URL de www.google.es. Al cargar la página, verás que el
contenido de www.google.es se mostrará dentro de ese div.
Es importante mencionar que cargar una página externa en un iframe puede tener
limitaciones de seguridad y puede depender de la política de seguridad del sitio web que
estás utilizando. Algunos sitios web pueden bloquear el contenido dentro de un iframe
por razones de seguridad (CORS). Si estás trabajando con tu propio contenido o sitios
web que permiten su visualización en iframes, entonces no deberías tener problemas. Sin
embargo, si deseas cargar sitios web de terceros, es posible que encuentres restricciones.
```

Una vez pegado el código de ChatGPT en un fichero HTML y CSS, se obtiene la web mostrada en la Ilustración 173.

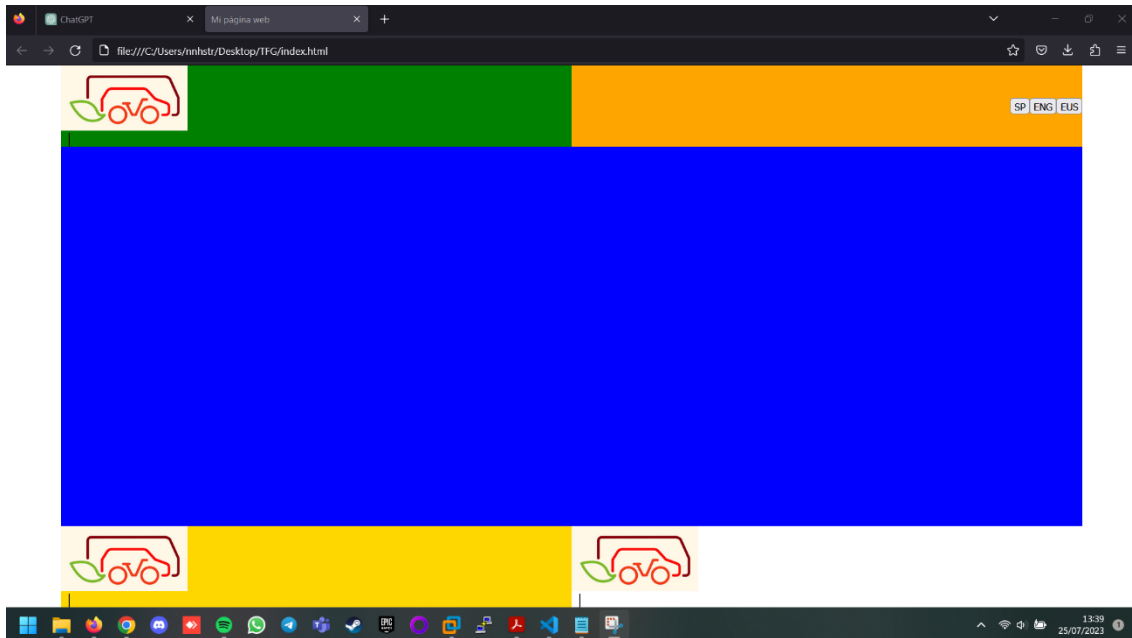


Ilustración 173 - HTML obtenido de ChatGPT.

Pregunta: *Hola, te pongo en contexto. Estoy haciendo la monitorización de una instalación fotovoltaica de casa. Para ello estoy haciendo uso de Node-RED y Apache2 en Ubuntu. Tengo ya el Node-RED configurado y veo los datos del inversor y también tengo la página web creada. En la web tengo un div actualmente en azul y me gustaría ver un gráfico de 24h con los datos de potencia generada. ¿Cómo lo hago?*

Respuesta: *¡Hola! Parece que estás trabajando en un proyecto interesante. Para mostrar un gráfico de 24 horas con los datos de potencia generada en tu página web usando Node-RED, puedes seguir los siguientes pasos generales:*

Obtener los Datos:

Asegúrate de que Node-RED esté obteniendo correctamente los datos de potencia generada del inversor. Puede ser a través de una API, MQTT, u otra fuente de datos. Si ya estás viendo los datos en Node-RED, estás en el camino correcto.

Preparar los Datos:

Los datos obtenidos necesitarán ser preparados en el formato adecuado para que puedan ser utilizados por la librería de gráficos que elijas. Normalmente, los datos necesitarán estar en forma de pares de tiempo y valor para cada punto de datos.

Elegir una Librería de Gráficos:

Hay varias librerías de gráficos disponibles para crear visualizaciones interactivas en tu página web. Algunas opciones populares son Chart.js, D3.js y Highcharts. Elige la que mejor se adapte a tus necesidades y habilidades.

Configurar el Gráfico:

Una vez que hayas elegido una librería, tendrás que configurar el gráfico para mostrar los datos de potencia generada en un período de 24 horas. Esto puede incluir la configuración de ejes, etiquetas de tiempo y valores, colores, leyendas, etc.

Integración con tu Página Web:

Una vez que el gráfico esté configurado, necesitarás integrarlo en tu página web. Puedes hacerlo insertando el código HTML y JavaScript necesario en el lugar donde deseas que aparezca el gráfico en tu página.

Actualizar Datos:

Si deseas que el gráfico se actualice automáticamente con nuevos datos, puedes configurar un mecanismo para que Node-RED actualice los datos del gráfico a intervalos regulares. Esto puede hacerse utilizando solicitudes HTTP, MQTT o cualquier otro método de comunicación.

Dado que mencionaste que ya tienes una página web con un div azul, supongo que ya tienes conocimientos de HTML y JavaScript. Aquí hay un ejemplo simple utilizando la librería Chart.js para crear un gráfico de línea que muestra los datos de potencia generada en las últimas 24 horas:

```
html
<!DOCTYPE html>
<html>
<head>
  <title>Gráfico de Potencia Generada</title>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
<body>
  <div style="width: 80%; margin: auto;">
    <canvas id="powerChart"></canvas>
  </div>
  <script>
    // Ejemplo de datos (sustituye esto por tus propios datos)
    const data = {
      labels: ["Hora 1", "Hora 2", /* ... */, "Hora 24"],
      datasets: [{
        label: "Potencia Generada",
```

```
data: [10, 20, /* ... */, 15], // Sustituye con tus datos
borderColor: "rgba(75, 192, 192, 1)",
backgroundColor: "rgba(75, 192, 192, 0.2)",
fill: true,
}}
};

const ctx = document.getElementById("powerChart").getContext("2d");

const powerChart = new Chart(ctx, {
  type
```

Pregunta: Hay alguna forma de hacer que en mi web aparezca un gráfico de potencia generada en las últimas horas. Me gustaría usar una herramienta o aplicación con el fin de evitar programar más en HTML.

Respuesta: ¡Por supuesto! Si prefieres evitar programar en HTML y JavaScript, puedes utilizar plataformas y herramientas que te permitan crear gráficos de forma visual y luego integrarlos en tu página web. Aquí hay algunas opciones que podrían ser útiles:

Google Sheets y Google Charts:

Puedes utilizar Google Sheets para ingresar tus datos de potencia generada y luego utilizar la herramienta "Google Charts" para crear gráficos visuales. Una vez que hayas creado el gráfico en Google Sheets, puedes embeberlo en tu página web utilizando el código de inserción proporcionado por Google.

Microsoft Excel y Office Online:

Similar a la opción de Google Sheets, puedes utilizar Microsoft Excel para crear gráficos y luego cargar el archivo a OneDrive o SharePoint para acceder a él en línea. Office Online te permite embeber los gráficos en tu página web.

Tableau Public:

Tableau Public es una plataforma de visualización de datos que te permite crear gráficos y tableros interactivos. Puedes cargar tus datos en Tableau Public y luego embeber los gráficos generados en tu página web.

Data Studio de Google:

Google Data Studio es una herramienta gratuita que te permite crear informes y paneles interactivos con tus datos. Puedes conectar tus datos de potencia generada y diseñar visualizaciones personalizadas sin necesidad de programación.

Power BI de Microsoft:

Power BI es una herramienta de análisis y visualización de datos de Microsoft. Puedes crear visualizaciones atractivas con tus datos y luego embeberlas en tu página web.

Estas opciones te permiten crear gráficos sin necesidad de programar en HTML y JavaScript. Sin embargo, ten en cuenta que es posible que debas aprender cómo usar estas herramientas y plataformas para crear los gráficos que necesitas. Cada herramienta tiene su propio conjunto de características y limitaciones, así que asegúrate de elegir la que mejor se adapte a tus requisitos y habilidades.

Pregunta: *m... ¿Habría otra forma de guardar los datos del inversor en algún sitio y después hacer un gráfico de ello?*

Respuesta: *Sí, hay varias formas de almacenar los datos del inversor y luego generar gráficos a partir de ellos. Aquí te presento algunas opciones:*

Base de Datos:

Puedes configurar una base de datos (como MySQL, PostgreSQL, MongoDB, etc.) para almacenar los datos del inversor a medida que se generan. Luego, puedes utilizar lenguajes de programación como Python, PHP o Node.js para extraer los datos de la base de datos y generar gráficos utilizando bibliotecas como Matplotlib (Python), Chart.js (JavaScript), etc.

Archivos CSV o Excel:

Puedes guardar los datos del inversor en archivos CSV o Excel a medida que se generan. Luego, puedes utilizar herramientas como Microsoft Excel, Google Sheets, Python o R para cargar los datos y crear gráficos a partir de ellos.

Plataformas IoT y Cloud:

Utiliza plataformas de Internet de las cosas (IoT) como ThingSpeak, Adafruit IO o incluso plataformas en la nube como AWS IoT. Estas plataformas pueden ayudarte a almacenar y visualizar los datos generados por el inversor, y algunas de ellas también ofrecen opciones para generar gráficos.

Servicios de Monitorización Energética:

Existen servicios específicos para monitorizar instalaciones fotovoltaicas y generar gráficos a partir de los datos recopilados. Algunos ejemplos incluyen SolarEdge, Enphase Enlighten y otros servicios de gestión energética.

Plataformas de Análisis y Visualización de Datos:

Utiliza herramientas de análisis y visualización de datos como Tableau, Power BI o Google Data Studio. Puedes conectar tus datos almacenados a estas herramientas y crear visualizaciones gráficas personalizadas.

La elección de la mejor opción dependerá de tus habilidades técnicas, la cantidad de datos que desees manejar, la frecuencia de actualización de los datos y otros requisitos específicos de tu proyecto. Es importante considerar factores como la escalabilidad, la seguridad de los datos y la facilidad de mantenimiento al elegir la solución adecuada para tu caso.

Pregunta: *Bien, me gustaría realizar una base de datos con el fin de almacenar todos los datos. ¿Qué hago para pasar los datos de Node-RED a una base de datos? ¿Qué aplicaciones puedo usar para hacer la base de datos?*

Respuesta: *Para pasar los datos de Node-RED a una base de datos, puedes seguir estos pasos generales:*

Selecciona una Base de Datos:

Primero, elige la base de datos que desees utilizar. Algunas opciones populares incluyen MySQL, PostgreSQL, SQLite, MongoDB, InfluxDB, entre otras. La elección dependerá de factores como el tipo de datos que estás almacenando, la escalabilidad requerida y tus preferencias personales.

Configura la Conexión a la Base de Datos:

Debes configurar una conexión entre Node-RED y la base de datos que hayas elegido. Esto generalmente implica proporcionar la dirección, el puerto, las credenciales de autenticación y otros detalles necesarios para establecer la comunicación.

Transforma y Almacena los Datos:

Utiliza los nodos de Node-RED para transformar los datos obtenidos del inversor en un formato adecuado y luego guárdalos en la base de datos. Puedes utilizar nodos específicos que te permitan interactuar con la base de datos que has elegido.

Programa el Flujo en Node-RED:

Crea un flujo en Node-RED que tome los datos de entrada del inversor, los transforme si es necesario y luego los almacene en la base de datos. Utiliza nodos como "Function", "Database", "HTTP Request", "MQTT" u otros según la base de datos y la fuente de datos que estés utilizando.

Monitorea y Gestiona los Datos:

Una vez que los datos estén almacenados en la base de datos, puedes utilizar consultas SQL u otras operaciones para analizar, gestionar y generar informes a partir de los datos almacenados.

En cuanto a las aplicaciones que puedes usar para crear y administrar la base de datos, aquí hay algunas opciones comunes:

MySQL:

MySQL es una base de datos relacional de código abierto muy popular. Puedes instalarlo en tu servidor y utilizar herramientas como phpMyAdmin o MySQL Workbench para administrar la base de datos.

PostgreSQL:

PostgreSQL es otra base de datos relacional de código abierto que ofrece características avanzadas. Puedes utilizar herramientas como pgAdmin para administrarla.

MongoDB:

MongoDB es una base de datos NoSQL orientada a documentos. Puedes instalar MongoDB y utilizar MongoDB Compass para administrar y visualizar los datos.

InfluxDB:

InfluxDB es una base de datos de series temporales diseñada para almacenar y consultar datos temporales. Es ideal para datos de sensores y monitoreo. También puedes usar Grafana para visualizar los datos almacenados en InfluxDB.

SQLite:

SQLite es una base de datos ligera y autónoma que puede ser útil para proyectos más pequeños. No requiere un servidor separado y puede ser administrada a través de herramientas como DB Browser for SQLite.

Recuerda investigar y seleccionar la base de datos que mejor se adapte a tus necesidades y nivel de experiencia. Cada base de datos tiene sus propias ventajas y desventajas, así que elige la que mejor se ajuste a tu proyecto.

Pregunta: *Bien, me has dado esta lista MySQL, PostgreSQL, SQLite, MongoDB, InfluxDB. Explícame las opciones.*

Respuesta: *¡Por supuesto! Aquí tienes una breve explicación de cada una de las opciones de bases de datos que mencionaste:*

MySQL:

MySQL es una base de datos relacional de código abierto muy popular. Es conocida por ser rápida, segura y confiable. Es ampliamente utilizada en una variedad de aplicaciones y es compatible con muchos lenguajes de programación. MySQL utiliza el lenguaje SQL (Structured Query Language) para realizar consultas y operaciones en la base de datos.

PostgreSQL:

PostgreSQL es otra base de datos relacional de código abierto. Es conocida por su robustez y capacidad para manejar cargas de trabajo complejas. Ofrece características avanzadas como soporte para tipos de datos personalizados, funciones almacenadas y consultas complejas. PostgreSQL también utiliza SQL como lenguaje de consulta.

SQLite:

SQLite es una base de datos incorporada en bibliotecas que no requiere un servidor de base de datos separado. Es ideal para proyectos más pequeños o aplicaciones móviles. Aunque es ligera y de un solo archivo, todavía puede manejar una cantidad considerable de datos. SQLite también utiliza SQL para consultas y operaciones.

MongoDB:

MongoDB es una base de datos NoSQL orientada a documentos. En lugar de usar tablas y filas, almacena los datos en documentos en formato JSON o BSON. Es flexible y escalable, ideal para datos no estructurados o semiestructurados. MongoDB es popular para aplicaciones web y móviles que requieren un almacenamiento de datos rápido y flexible.

InfluxDB:

InfluxDB es una base de datos de series temporales diseñada específicamente para el almacenamiento y análisis de datos temporales o en secuencia, como datos de sensores, métricas de monitoreo y registros de tiempo. Ofrece una estructura optimizada para datos que cambian con el tiempo y es compatible con una integración fácil con herramientas de visualización como Grafana.

La elección de la base de datos dependerá de factores como el tipo de datos que manejas, la escala de tu proyecto, la experiencia que tengas con bases de datos y las características específicas que necesites (por ejemplo, si los datos son estructurados o no estructurados, si necesitas alta escalabilidad, etc.).

Cada una de estas opciones tiene sus propias ventajas y desventajas, por lo que te recomiendo investigar más sobre cada una y evaluar cuál se adapta mejor a tus necesidades y capacidades técnicas.

Pregunta: *¿De esas opciones cuál es la más sencilla?*

Respuesta: *La simplicidad de cada opción de base de datos puede variar según tus habilidades técnicas y la naturaleza de tu proyecto. Sin embargo, en términos generales, si estás buscando una opción simple para comenzar, aquí están algunas recomendaciones:*

SQLite:

SQLite es muy sencillo de usar, especialmente para proyectos pequeños o aplicaciones individuales. No requiere configuración de servidor y se almacena en un único archivo. Es perfecto si estás buscando una solución rápida y no necesitas gestionar múltiples usuarios o altas cargas de trabajo.

MongoDB:

MongoDB es bastante amigable para los principiantes, especialmente si estás trabajando con datos no estructurados o semiestructurados. Su formato de documento JSON/BSON es intuitivo, y puedes comenzar a trabajar con él con relativamente poca

configuración. Sin embargo, si planeas un crecimiento significativo, debes considerar la planificación adecuada para su escalabilidad.

InfluxDB:

Si estás trabajando con datos de series temporales, InfluxDB es una opción simple y efectiva. Está diseñado específicamente para manejar este tipo de datos, lo que facilita su configuración y uso. Además, su integración con Grafana para la visualización lo convierte en una opción poderosa para monitoreo y análisis de tiempo.

Cada opción tiene su propio conjunto de características y ventajas, por lo que es importante evaluar cuál se alinea mejor con tus necesidades y nivel de experiencia. Si eres nuevo en las bases de datos, SQLite puede ser una excelente opción para comenzar debido a su simplicidad y portabilidad. Sin embargo, si estás dispuesto a aprender un poco más y necesitas flexibilidad, MongoDB también podría ser una buena elección.

Pregunta: Debido a que es simple y efectiva, ¿Cómo instalo InfluxDB?

Respuesta: Instalar InfluxDB es relativamente sencillo. Te proporcionaré instrucciones básicas para instalar InfluxDB en sistemas basados en Ubuntu, que parece ser el sistema operativo que estás utilizando.

Agregar el Repositorio:

Abre una terminal en tu sistema y ejecuta los siguientes comandos para agregar el repositorio oficial de InfluxDB:

```
sudo apt update
```

```
sudo apt install -y curl
```

```
curl -sL https://repos.influxdata.com/influxdb.key | sudo gpg --dearmor -o /usr/share/keyrings/influxdb-archive-keyring.gpg
```

```
echo "deb [signed-by=/usr/share/keyrings/influxdb-archive-keyring.gpg] https://repos.influxdata.com/debian $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/influxdb.list
```

Instalar InfluxDB:

Una vez que hayas agregado el repositorio, actualiza la lista de paquetes e instala InfluxDB:

```
sudo apt update
```

```
sudo apt install influxdb
```

Iniciar y Habilitar el Servicio:

Inicia el servicio de InfluxDB y habilítalo para que se inicie automáticamente al arrancar el sistema:

```
sudo systemctl start influxdb
```

```
sudo systemctl enable influxdb
```

Verificar el Estado del Servicio:

Puedes verificar si InfluxDB se está ejecutando correctamente utilizando el siguiente comando:

```
sudo systemctl status influxdb
```

Acceder a la Interfaz Web:

InfluxDB proporciona una interfaz web llamada InfluxDB Chronograf para la administración y visualización de los datos. Puedes acceder a ella a través de tu navegador web visitando <http://localhost:8888>.

Ten en cuenta que estos pasos son para sistemas basados en Ubuntu. Si estás utilizando un sistema operativo diferente, los comandos y los pasos podrían variar. Una vez que hayas instalado InfluxDB, puedes comenzar a utilizarlo para almacenar tus datos de series temporales y conectarlo con Node-RED u otras herramientas según tus necesidades.

Pregunta: Ya tengo InfluxDB instalado, ahora cómo lo comunico con Node-RED.

Respuesta: Genial, ahora puedes configurar la comunicación entre InfluxDB y Node-RED para almacenar los datos generados en la base de datos. Aquí tienes los pasos generales para hacerlo:

Instala los Nodos de InfluxDB en Node-RED:

Para interactuar con InfluxDB desde Node-RED, necesitarás instalar los nodos correspondientes. En Node-RED, ve al menú en la esquina superior derecha y selecciona "Manage Palette". Busca "node-red-contrib-influxdb" y procede a instalarlo.

Configura el Nodo de InfluxDB:

Agrega un nodo "InfluxDB Out" a tu flujo. Configura los detalles de conexión al servidor de InfluxDB, como la dirección, el puerto y las credenciales. También deberás especificar la base de datos y la medición (tabla) en la que deseas almacenar los datos.

Transforma los Datos en Node-RED:

Asegúrate de que tus datos en Node-RED estén en el formato correcto antes de enviarlos a InfluxDB. Puedes utilizar nodos de transformación como "Function" o "Change" para dar formato a los datos antes de enviarlos al nodo "InfluxDB Out".

Conecta los Nodos:

Conecta el nodo que produce los datos con el nodo "InfluxDB Out". Asegúrate de que la estructura de los datos coincida con la configuración del nodo "InfluxDB Out".

Despliega el Flujo:

Despliega tu flujo en Node-RED para que los datos fluyan desde la fuente hasta el nodo de InfluxDB. Verifica que los datos se estén almacenando correctamente en la base de datos.

Verifica los Datos en Chronograf:

Si has configurado InfluxDB Chronograf (la interfaz web para InfluxDB), puedes verificar que los datos se están almacenando correctamente al acceder a Chronograf y ejecutar consultas para ver los datos almacenados en tu base de datos.

Recuerda que estos pasos son generales y pueden variar según la versión específica de los nodos de InfluxDB que estés utilizando y la configuración de tu proyecto. Asegúrate de consultar la documentación de los nodos de InfluxDB para obtener información detallada sobre su configuración y uso dentro de Node-RED.

Pregunta: ¿Cómo uso InfluxDB con Grafana?

Respuesta: *InfluxDB y Grafana son dos herramientas muy populares en el mundo de la monitorización y visualización de datos en tiempo real. InfluxDB es una base de datos de series temporales diseñada para almacenar y consultar datos con una marca de tiempo, mientras que Grafana es una plataforma de visualización que permite crear paneles y gráficos interactivos a partir de diversos orígenes de datos, incluido InfluxDB.*

Aquí tienes una guía básica sobre cómo usar InfluxDB con Grafana:

Instalación y configuración:

InfluxDB: Primero, debes instalar y configurar InfluxDB. Puedes seguir la documentación oficial para realizar la instalación en tu sistema operativo: <https://docs.influxdata.com/influxdb/>

Grafana: Luego, instala y configura Grafana siguiendo su documentación oficial: <https://grafana.com/docs/grafana/latest/getting-started/getting-started-home/>

Crear una base de datos en InfluxDB:

Después de configurar InfluxDB, crea una base de datos donde almacenarás los datos. Puedes usar la interfaz de línea de comandos de InfluxDB o la interfaz web.

Conectar Grafana a InfluxDB:

Abre la interfaz web de Grafana y accede a tu instancia.

Ve a la sección "Configuration" (Configuración) y luego selecciona "Data Sources" (Orígenes de datos).

Haz clic en "Add data source" (Agregar origen de datos) y elige "InfluxDB".

Completa los detalles de la conexión, como el nombre, la URL de la base de datos InfluxDB, y las credenciales de acceso.

Configurar un panel en Grafana:

Crea un nuevo panel en Grafana seleccionando "Create" > "Dashboard" (Crear > Panel de control).

Agrega un panel vacío y haz clic en "Add Query" (Agregar consulta).

Elige el origen de datos InfluxDB que configuraste previamente.

*Escribe tu consulta InfluxQL para recuperar los datos de la base de datos InfluxDB. Por ejemplo: `SELECT * FROM measurement_name`.*

Ajusta la configuración de visualización, como el tipo de gráfico, el intervalo de tiempo, las etiquetas y más.

Personalizar el panel:

Agrega más consultas al mismo panel si deseas mostrar múltiples conjuntos de datos.

Añade leyendas, títulos y anotaciones para hacer que el panel sea más informativo.

Utiliza las opciones de formato para ajustar los colores, tamaños y estilos de los gráficos.

Guardar y compartir:

Una vez que estés satisfecho con tu panel, guárdalo haciendo clic en el icono de disco en la parte superior.

Puedes compartir el panel con otros proporcionándoles el enlace o incluso incrustándolo en otros sitios web.

Recuerda que esta es una guía básica para comenzar. Tanto InfluxDB como Grafana tienen muchas funciones avanzadas que te permitirán crear paneles de visualización más sofisticados y personalizados. Te recomiendo explorar las documentaciones oficiales de ambas herramientas para obtener información más detallada sobre sus características y posibilidades.

Pregunta: *Quiero que me hagas un script en python que coja la primera fila de un fichero denominado actuales.csv y haga una operación matemática de sumarle 2 al primer elemento del array y posteriormente pegue ese resultado en un fichero denominado energia.csv.*

Respuesta: *# Leer la primera fila del archivo actuales.csv*

with open('actuales.csv', 'r') as archivo_entrada:

primera_fila = archivo_entrada.readline().strip().split(',')

Convertir los valores de la primera fila a números

valores = [float(valor) for valor in primera_fila]

Realizar la operación de suma

resultado = valores[0] + 2

Escribir el resultado en el archivo energia.csv

with open('energia.csv', 'w') as archivo_salida:

archivo_salida.write(str(resultado))

Asegúrate de que los archivos actuales.csv y energia.csv estén en el mismo directorio que el script. Además, este ejemplo asume que el archivo actuales.csv solo contiene una línea de números separados por comas en la primera fila. Si la estructura de tu archivo es diferente, es posible que necesites ajustar el código para que funcione correctamente.