

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Exploración y comparación de métodos o herramientas para servicios de comunicación industrial inteligente, haciendo uso del cloud, fog y edge computing, aplicado a un módulo de variables industriales en el laboratorio de docencia del ITM.

Juan Camilo Sepúlveda-Balbín

Trabajo de grado presentado como requisito parcial para optar al título de:
Ingeniero Mecatrónico

Asesor(es)

Jhon Alexander Isaza-Hurtado

Vanessa Botero-Gómez

Instituto Tecnológico Metropolitano - ITM
Facultad de Ingenierías
Departamento Mecatrónica y Electromecánica
Medellín, Colombia
2023

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Resumen

En este trabajo de grado, se aborda el tema de las comunicaciones industriales y su relación con el Internet Industrial de las Cosas (IIOT) y la computación en el borde. Se destaca la importancia de garantizar la sincronía, evitar la pérdida de información y minimizar la latencia en los procesos industriales, lo cual se logra mediante el uso de controladores industriales tipo PLC y sistemas embebidos. Estos dispositivos requieren una Interfaz de Programación de Aplicaciones (API) para comunicarse con servidores externos, ya sea en una red local o en la nube.

El enfoque principal del trabajo se centra en el análisis y evaluación de diversos protocolos de comunicación para establecer una conexión exitosa entre un controlador (PLC o sistema embebido) y una base de datos en la nube. Se presenta un procedimiento metodológico que divide el sistema en tres capas distintas: la capa de borde, la capa de niebla y la capa de nube. Cada capa se caracteriza con una taxonomía de software y hardware específica, proporcionando una guía clara y orientadora para la implementación de sistemas similares. Además, se implementa un sistema bidireccional de transmisión de información mediante una interfaz de usuario, con el objetivo de posteriormente alojarla permanentemente en un servicio en la nube en internet, con una dirección IP pública. Todas estas implementaciones se llevan a cabo en un módulo ubicado en el laboratorio de Automatización del Instituto Tecnológico Metropolitano de Medellín.

Además, se propone una modificación a la clásica pirámide de la automatización, que permite esclarecer la función de cada una de las etapas en un proyecto de automatización industrial y comprender el flujo de la transmisión de información.

Finalmente, como caso de aplicación, se presenta la construcción e implementación de todas las herramientas expuestas en este trabajo, estableciendo una comunicación entre un módulo de variables industriales y un servicio de almacenamiento de información o base de datos alojado en la nube. Asimismo, se muestra cómo se logra la comunicación con una interfaz gráfica que permite la gestión de la información de manera efectiva y remota. El conjunto de todas estas contribuciones proporciona una sólida base para implementar sistemas de comunicación industrial inteligente, acercándose a una visión completa y eficiente del Internet Industrial de las Cosas (IIOT).

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Reconocimientos

Este trabajo fue soportado por el Instituto Tecnológico Metropolitano bajo el proyecto PCI 22101 titulado "Modelo conceptual y sistémico para la gestión de información que permita la prolongación de la vida útil de las herramientas tecnológicas y estén a la vanguardia de la industrial digital" con el constante acompañamiento del Semillero en Automática SMINGE78.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Acrónimos

- IOT* Internet Of Things
- IIoT* Industrial Internet Of Things
- API* Application programming Interface
- SSH* Secure Shell
- PLC* Programmable Logic Controller
- PaaS* Platform as a Service
- IaaS* Infrastructure as a Service
- UART* Universal Asynchronous Receiver-Transmitter
- USART* Universal Synchro-nous/Asynchronous Receiver-Transmitter
- Profibus DP* Decentralized Periphery
- Profibus PA* Process Automation
- I2C* Inter-Integrated Circuit
- SPI* Serial Peripheral Interface
- TCP/IP* Transmission Control Proto-col/Internet Protocol
- HTTP* Hypertext Transfer Protocol
- MQTT* Message Queuing Telemetry Transport
- FBD* Function Block Diagram
- ST* Structure Text
- IL* Instruction List
- STL* Statement List
- AWS* Amazon Web Services
- SQL* Structured Query Language
- SSL/TLS* Secure Socket Layer/Transport Layer Security
- SSH* Secure Shell
- PID* Proporcional, integral, Derivativo

Tabla de contenido

1. Introducción	8
2. Protocolos de comunicación implementados en las arquitecturas de computación en la nube, la niebla y el borde	10
2.1 Protocolos de comunicación.....	10
2.2 Docker y el uso de contenedores	14
2.3 Computación en la nube.....	15
2.4 Computación en el borde	16
2.5 Computación en la niebla	18
2.6 Propuesta de modificación de la pirámide de la automatización	19
3. Arquitectura y protocolos de comunicación evaluadas y seleccionadas	22
3.1 Arquitectura seleccionada	22
3.2 Capa de borde.....	23
3.3 Capa de niebla.....	24
3.4 Capa de nube	24
3.5 Criterios de evaluación y selección.....	25
3.6 Caracterización y configuración de los protocolos de comunicación elegidos	27
4. Aplicación de las herramientas en el módulo de variables	32
4.1 Capa de borde para el caso de aplicación.	32
4.2 Capa de niebla para el caso de aplicación.	34
4.3 Capa de nube para el caso de aplicación.....	38
5. Conclusiones, recomendaciones y trabajo futuro	49
6. Referencias.....	51

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Listado de figuras

Figura 1 Modificación a la pirámide de la automatización.	20
Figura 2 Arquitectura de sistemas interconectados	23
Figura 3 Módulo de variables industrial del ITM.....	33
Figura 4 Conexiones y Protocolos	34
Figura 5 Interfaz de Docker, donde se presentan los contenedores.	35
Figura 6 Interfaz de general NodeRED	35
Figura 7 Plugin necesarios para el funcionamiento correcto de la aplicación.....	36
Figura 8 Nodo de comunicación Step 7.....	36
Figura 9 Configuración general del nodo Step 7	37
Figura 10 Configuración de variables protocolo Step 7	37
Figura 11 Nodo para el uso del protocolo HTTP.....	38
Figura 12 Nodo de configuración del protocolo HTTP	38
Figura 13 Información almacenada mostrada en la interfaz de MongoDB Atlas	47
Figura 14 Presentación de la información luego de un método GET común usando el buscador de URL del navegador.....	47
Figura 15 Presentación de la información luego de un método GET común usando el buscador de URL del navegador.....	48

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Listado de tablas

Tabla 1	Protocolos de comunicación.....	11
Tabla 2	Taxonomía de hardware y software de la capa de nube	16
Tabla 3	Taxonomía de hardware y software de la capa de borde	17
Tabla 4	Taxonomía de hardware y software de la capa de niebla.....	18
Tabla 5	Caracterización de protocolos de comunicación.....	30
Tabla 6	Caracterización de dispositivos del módulo de variables.....	32
Tabla 7	Métodos HTTP utilizado por la API para la comunicación módulo de variables industriales – base de datos	39
Tabla 8	Métodos HTTP utilizado por la API para la comunicación con el usuario	41
Tabla 9	Esquema de información sensores de presión y nivel.....	44
Tabla 10	Esquema de información sensores de Flujo y Temperatura	44
Tabla 11	Esquema de información para el envío de datos al módulo	45

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

1. Introducción

La evolución de la tecnología, a través de la aplicación de herramientas asociadas a la industria 4.0, ha generado grandes cambios en los paradigmas utilizados en la automatización de sistemas productivos. La industria 4.0 se refiere en gran medida a la integración de un conjunto de tecnologías digitales las cuales buscan alcanzar altos niveles de eficiencia operativa y productividad. Uno de los principales objetivos consta en acercar los sistemas al ciberespacio, teniendo un gran foco en el acceso, monitoreo y gestión de la información de manera remota y en tiempo real. Un ejemplo cercano se encuentra en los celulares su basta librería de aplicaciones, a través de las cuales se puede monitorear y controlar algún tipo de proceso que cuente con conexión a internet. Dado lo anterior, conceptos como el IOT e IIOT han tenido una gran acogida, generando la necesidad de integrar a las máquinas la capacidad de conectividad y transferencia de información entre dispositivos asociados al mismo sistema, además de la capacidad de almacenar y procesar información en servicios basados en la nube, revolucionando la manera en que se accede a la información, independientemente del lugar y el momento (Eryarsoy, Kilic, Zaim, S, & Doszhanova, 2022).

Una de las tecnologías que aparece como uno de los principales exponentes de la industria 4.0 es llamada la computación en la nube (Cloud Computing), ya que su arquitectura está basada en el uso de servidores remotos, los cuales se encuentran interconectados, permitiendo ajustar los recursos y la seguridad necesaria para almacenar y procesar un alto volumen de datos, haciendo uso de la red de Internet (Miralles, 2010) (Kim, y otros, 2022).

Cuando se habla de automatización en sistemas productivos o automatización industrial, el tráfico de información en la web genera altos niveles de latencia y posible saturación de los sistemas, lo que exige contar con mayores niveles de ancho de banda y estabilidad en la conexión a internet, ya que cualquier cambio brusco o pérdida de información puede llegar a ser un fallo grave del sistema o una pérdida económica considerable, por lo cual se hace necesario gestionar qué información será enviada a la nube y cuál procesará dentro de la infraestructura del sistema. Para este fin aparecen dos conceptos importantes, la computación en el borde (Edge computing) la cual se refiere al control y procesamiento de información que se realiza a nivel de planta, en otras palabras, se refiere a la adquisición de información de los sensores, el control que realiza el PLC o cualquier controlador industrial, y los actuadores que generan la acción con la que se desea realizar control del proceso. No es necesario que toda esta información viaje hacia la nube, es conveniente separar cual debe reflejarse en la nube y cual no, en este orden de ideas, esta capa cobra gran importancia para reducir de alguna forma, los paquetes de datos que viajan por la red (Kim, y otros, 2022). Por otro lado, se tiene el concepto de computación en la niebla (Fog Computing), este expande el concepto anterior el cual se presenta como una capa adicional de servidores informáticos los cuales pueden procesar información, además de realizar la

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

capa de comunicación con la nube, en otras palabras, es un modelo de informática distribuida o descentralizada en el que el procesamiento y almacenamiento de los datos se realiza en una capa externa a la nube (Garg & Dave, 2019).

En el Instituto Tecnológico Metropolitano (ITM) de Medellín, los módulos de control de variables industriales que están a la disposición de los estudiantes, para realizar sus prácticas de laboratorio, carecen de las propiedades como conexión a la nube o características IIOT, ya que los PLC (Programmable Logic Controller) que están a disposición en estos módulos de referencia CPU 314C -2DP y CPU 1214 DC/DC/DC de Siemens están algo desactualizados comparados con los controladores modernos. Dado lo anterior, se propone buscar métodos y tecnologías que permitan adquirir las características antes mencionadas y finalmente conectar de manera remota estos dispositivos industriales a una interfaz gráfica alojada en la nube, donde se espera monitorear información relevante del proceso, y así los estudiantes del ITM podrán realizar sus prácticas de manera remota.

Existen numerosas metodologías, tecnologías, conceptos y protocolos para realizar esta comunicación entre dispositivos, por lo que para este trabajo su objetivo principal será analizar métodos o herramientas para servicios de comunicación industrial inteligente, haciendo uso de la computación en la nube, computación en la niebla y la computación en el borde. Constantemente las personas están expuestas a dispositivos tecnológicos los cuales sin saberlo hacen uso de diferentes protocolos de comunicación, que permiten la vida como se conoce actualmente. En el contexto de este trabajo, los protocolos de comunicación desempeñan un papel fundamental. Los protocolos de comunicación establecen las reglas y estándares para el intercambio de información entre dispositivos de una red. La elección de los protocolos adecuados es crucial para garantizar una comunicación eficiente y confiable entre los sistemas, por lo cual este trabajo busca explorar y evaluar diferentes métodos y protocolos de comunicación implementados en las arquitecturas de computación en la nube, computación en el borde y computación en la niebla, con el objetivo de establecer una conexión efectiva entre el módulo de variables industriales del ITM y la nube. Esta investigación contribuirá a mejorar la capacidad de monitoreo y control remoto de los sistemas industriales, brindando a los estudiantes del ITM la oportunidad de realizar prácticas de laboratorio de manera remota.

A continuación, este documento está organizado de la siguiente manera: Sección 2, protocolos de comunicación implementados en las arquitecturas de computación en la nube, el borde y la niebla. Sección 3, métodos para implementar las herramientas. Sección 4, aplicación de las herramientas en el módulo de variables. Sección 5, conclusiones, recomendaciones y trabajo futuro. Sección 6, referencias.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

2. Protocolos de comunicación implementados en las arquitecturas de computación en la nube, la niebla y el borde

Al profundizar en las posibles soluciones para abordar la problemática de comunicación entre Internet y el módulo de variables industriales del ITM, se encontraron diferentes arquitecturas y paradigmas en la literatura, las cuales hacen uso de las nuevas tecnologías disponibles en la actualidad en diversas configuraciones para establecer la comunicación entre dispositivos y la nube.

En este capítulo, se presentará una revisión exhaustiva de los diferentes métodos y protocolos de comunicación implementados en las arquitecturas de computación en la nube, computación en el borde y computación en la niebla. Se analizarán las características, ventajas y desventajas de cada uno de ellos, con el objetivo de seleccionar las soluciones más adecuadas para establecer la comunicación entre el módulo de variables industriales del ITM y la nube.

Además, se propondrá una taxonomía o clasificación de hardware y software que componen cada uno de estos conceptos, así como los protocolos de comunicación que intervienen en su funcionamiento. Esta taxonomía permitirá comprender mejor las diferentes opciones disponibles y facilitará la elección de las tecnologías más apropiadas para el caso de estudio planteado. Por último, se propone una modificación a la pirámide de la automatización, siendo segmentada en tres capas, la capa del borde, la capa de niebla y la capa de la nube.

2.1 Protocolos de comunicación

En el contexto de la tecnología, un protocolo de comunicación se refiere a un conjunto de reglas y estándares establecidos que permiten que diferentes dispositivos, sistemas o componentes se comuniquen entre sí de manera eficiente y confiable. Estas reglas definen el formato, la secuencia y los procedimientos que deben seguirse durante la transmisión de datos, asegurando una interacción coherente y efectiva (Bayılmış, Ebleme, Küçük, & Sevin, 2022).

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Un protocolo de comunicación desempeña un papel fundamental en el intercambio de información dentro de un entorno tecnológico, ya sea en redes de computadoras, sistemas embebidos, Internet de las cosas u otras aplicaciones. Proporciona una estructura organizada para el envío, recepción y procesamiento de datos, estableciendo normas claras para la codificación, el direccionamiento, la autenticación, la corrección de errores y otros aspectos esenciales de la comunicación (Bayılmış, Ebleme, Küçük, & Sevin, 2022).

En la Tabla 1, se realiza una breve descripción y casos de uso de los protocolos de comunicación principales estudiados en este trabajo, en el capítulo 3 apartado de Criterios de evaluación y selección, se realizará una descripción más detallada de los protocolos seleccionados e implementados en el módulo de variables industriales.

Tabla 1 Protocolos de comunicación

Protocolos de comunicación	Descripción	Casos de uso	Referencia
Profibus DP	Profibus DP (Decentralized Periphery) es un protocolo de comunicación de bus de campo ampliamente utilizado en la automatización industrial. Proporciona una conexión de alta velocidad y confiable entre controladores y dispositivos de campo, como sensores y actuadores.	Se utiliza en aplicaciones de automatización industrial para permitir la comunicación entre sistemas de control centralizados y dispositivos de campo distribuidos. Por ejemplo, en sistemas de control de procesos, sistemas de control de motores, sistemas de transporte automatizado, etc.	(Vadi, Bayindir, Toplar, & Colak, 2022)
Profibus PA	Profibus PA (Process Automation) es una variante del protocolo Profibus diseñada para permitir la transmisión de datos y señales entre dispositivos de campo, como sensores y actuadores, y sistemas de control. A diferencia de Profibus DP mientras este se enfoca en aplicaciones que requieren alta velocidad y sincronización, Profibus PA está diseñado para procesos con requisitos de seguridad intrínseca y baja potencia.	Profibus PA se utiliza en plantas industriales para conectar sensores y actuadores en aplicaciones como control de nivel, presión, temperatura, caudal, entre otras.	(Vadi, Bayindir, Toplar, & Colak, 2022)

Protocolos de comunicación	Descripción	Casos de uso	Referencia
Modbus TCP/IP	Modbus TCP/IP es una variante del protocolo Modbus, que es un protocolo de comunicación ampliamente utilizado en la industria. Modbus TCP/IP utiliza la infraestructura de red TCP/IP para permitir la comunicación entre dispositivos a través de una red Ethernet.	Se utiliza en una amplia gama de aplicaciones industriales, como sistemas de monitoreo y control de energía, controladores de automatización, sistemas de gestión de edificios, entre otros. Permite la comunicación entre dispositivos de diferentes fabricantes y facilita la integración de diferentes componentes en un sistema industrial.	(Yuanyuan & Meng, 2021)
Industrial Ethernet	Conjunto de protocolos de comunicación basados en Ethernet utilizados en entornos industriales. Proporciona una conexión de red confiable y de alta velocidad para la comunicación entre dispositivos y sistemas en entornos de automatización industrial.	Se utiliza en sistemas de control industrial, como controladores lógicos programables (PLC), robots industriales y sistemas de supervisión y adquisición de datos (SCADA), para permitir la comunicación rápida y confiable entre diferentes componentes y sistemas en una planta industrial.	(Wu & Xie, 2019)
RS-232	RS-232 es un estándar de comunicación serie que se utiliza para la transmisión de datos entre dispositivos digitales. Define las conexiones eléctricas y los protocolos de comunicación para la transmisión de datos en serie.	RS-232 se utiliza comúnmente para la comunicación entre computadoras y periféricos, como impresoras, módems, dispositivos de adquisición de datos, entre otros. También se puede utilizar para la comunicación entre dispositivos de control, como PLC y HMI (Human-Machine Interface).	(Strangio, 2012)
UART/USART	UART (Universal Asynchronous Receiver-Transmitter) y USART (Universal Synchronous/Asynchronous Receiver-Transmitter) son estándares de comunicación serie que permiten la transmisión y recepción de datos en serie. UART se utiliza principalmente para la comunicación asíncrona, mientras que USART admite tanto la comunicación asíncrona como la síncrona.	Se utilizan en una amplia variedad de aplicaciones, como comunicación entre microcontroladores y periféricos, transferencia de datos en sistemas embebidos, interconexión entre componentes electrónicos, entre otros.	(Nanda & Pattnaik, 2016)

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Protocolos de comunicación	Descripción	Casos de uso	Referencia
I2C	I2C (Inter-Integrated Circuit) es un protocolo de comunicación serie de dos hilos utilizado para la comunicación entre chips en sistemas electrónicos. Permite la transmisión bidireccional de datos y soporta la conexión de múltiples dispositivos en el mismo bus.	Se utiliza comúnmente para la comunicación entre microcontroladores y dispositivos periféricos, como sensores, pantallas LCD, memorias EEPROM, entre otros. Es adecuado para aplicaciones donde se requiere la comunicación entre múltiples dispositivos en un sistema electrónico.	(Trivedi, Khade, Jain, & Jadhav, 2018)
SPI	SPI (Serial Peripheral Interface) es un protocolo de comunicación síncrono utilizado para la transferencia de datos entre dispositivos electrónicos. Se basa en una arquitectura maestro/esclavo y utiliza líneas de reloj, datos y control para la transmisión de información.	Es ampliamente utilizado en sistemas embebidos y aplicaciones que requieren una comunicación rápida y eficiente entre microcontroladores y periféricos, como pantallas LCD, sensores, memoria flash y convertidores analógico-digitales/digitales-analógicos.	(Trivedi, Khade, Jain, & Jadhav, 2018)
TCP/IP	TCP/IP (Transmission Control Protocol/Internet Protocol) es el conjunto de protocolos de comunicación utilizado en Internet y en muchas redes locales. Proporciona una base para la transferencia de datos y el enrutamiento de paquetes en redes de computadoras.	Se utiliza en prácticamente todas las comunicaciones en Internet, desde navegación web y correo electrónico hasta transferencia de archivos y servicios en la nube. También se utiliza en redes empresariales para permitir la comunicación entre diferentes dispositivos y sistemas.	(Bayılmış, Ebleme, Küçük, & Sevin, 2022).
HTTP	HTTP (Hypertext Transfer Protocol) es el protocolo de comunicación utilizado para la transferencia de información en la World Wide Web. Define la estructura de las solicitudes y respuestas entre un cliente y un servidor web.	Se utiliza en aplicaciones web para cargar y mostrar páginas web, enviar y recibir formularios, realizar solicitudes y recibir respuestas de servicios web, entre otros. Es el protocolo fundamental para la navegación y la interacción en la web.	(A. da Cruz, y otros, 2019), (Bayılmış, Ebleme, Küçük, & Sevin, 2022).

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Protocolos de comunicación	Descripción	Casos de uso	Referencia
MQTT	MQTT (Message Queuing Telemetry Transport) es un protocolo de mensajería ligero diseñado para la comunicación entre dispositivos conectados a redes con ancho de banda limitado. Está basado en el modelo de publicación/suscripción, donde los dispositivos pueden publicar mensajes en temas específicos y suscribirse a temas para recibir mensajes.	Se utiliza ampliamente en aplicaciones de Internet de las cosas (IoT), donde se requiere una comunicación eficiente y de bajo consumo de energía entre dispositivos IoT y servidores. Es adecuado para escenarios donde los dispositivos deben enviar y recibir datos de manera eficiente en redes con recursos limitados.	(Bayılmış, Ebleme, Küçük, & Sevin, 2022).
Websocket	Websocket es un protocolo de comunicación bidireccional que permite la comunicación en tiempo real entre un cliente y un servidor a través de una conexión persistente. A diferencia de HTTP, que es una comunicación de solicitud/respuesta, Websocket permite una comunicación continua y en tiempo real.	Se utiliza en aplicaciones web que requieren una comunicación bidireccional en tiempo real, como aplicaciones de chat en vivo, juegos en línea, monitoreo remoto, entre otros. Permite la transmisión de datos instantánea y la actualización en tiempo real de la interfaz de usuario.	(Bayılmış, Ebleme, Küçük, & Sevin, 2022).

En conclusión, los protocolos de comunicación desempeñan un papel crítico en el ámbito tecnológico al establecer un conjunto de reglas y estándares que permiten la comunicación eficiente y confiable entre diferentes dispositivos, sistemas o componentes. Estos protocolos garantizan una interacción coherente y efectiva en diversas aplicaciones tecnológicas, como redes de computadoras, sistemas embebidos, Internet de las cosas, entre otros.

Para el desarrollo de este trabajo se utilizó Docker como herramienta que permitió empaquetar la aplicación de NodeRED en un contenedor y luego desplegarla en un ordenador que prestaba los servicios de comunicación. A continuación, se definirá el concepto de contenedor, además de presentar la herramienta Docker como una de las principales herramientas para llevar a cabo este proceso.

2.2 Docker y el uso de contenedores

Los contenedores y Docker son conceptos relacionados que se utilizan para facilitar el despliegue y la ejecución de aplicaciones de manera eficiente y reproducible.

La creación de un contenedor es el proceso de empaquetar una aplicación y todas sus dependencias en un contenedor virtual. Un contenedor es una unidad de software

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

autónoma que contiene todo lo necesario para que la aplicación se ejecute, incluyendo el código, las bibliotecas, las configuraciones y otros archivos necesarios. Es como un paquete completo y autónomo que se puede mover e instalar fácilmente en diferentes entornos sin afectar a otras aplicaciones (Potdar, G, Kengond, & Mulla, 2020).

Docker, por otro lado, es uno de los proyectos más populares y ampliamente utilizado que automatiza el despliegue de aplicaciones dentro de contenedores de software, además es de código abierto. Proporciona herramientas y servicios para crear, gestionar y desplegar contenedores de manera eficiente. Docker permite empaquetar una aplicación y sus dependencias en un contenedor, lo que facilita su distribución y ejecución en diferentes sistemas operativos y entornos de manera consistente (Potdar, G, Kengond, & Mulla, 2020).

Una de las ventajas clave del empaquetamiento en contenedores y Docker es la portabilidad. Los contenedores son independientes del entorno subyacente y pueden ejecutarse en cualquier máquina que tenga Docker instalado. Esto simplifica el proceso de desarrollo, implementación y escalado de aplicaciones, ya que se eliminan las preocupaciones sobre las diferencias de configuración y las dependencias del sistema (Potdar, G, Kengond, & Mulla, 2020).

Además, Docker ofrece características como el aislamiento de recursos, lo que significa que los contenedores no afectan ni se ven afectados por otros contenedores en el mismo sistema, y la capacidad de escalar vertical y horizontalmente los contenedores según las necesidades de la aplicación.

2.3 Computación en la nube

La computación en la nube es uno de los paradigmas de computación más prometedores en el mercado tecnológico. Su potencial reside en que este ofrece diferentes servicios y herramientas con lo que antes solo se podía soñar. Como se menciona en (D & Soni, 2022), la computación en la nube ofrece infraestructura como servicio (IaaS), plataforma como servicio (PaaS) y software como servicio (SaaS) para proveer justo los requerimientos cómputo, redes y almacenamiento que el usuario necesita por medio del modelo de pago-por-uso, esto quiere decir que solo se le cobra al usuario explícitamente los recursos que está utilizando. Este es tan relevante que de este se emergen varios paradigmas integrados de computación, como lo son la computación en el borde, el internet de las cosas (IOT), gemelos digitales, entre otros.

Tabla 2 Taxonomía de hardware y software de la capa de nube

Hardware	Software	Protocolos de comunicación
<ul style="list-style-type: none"> – Equipos de red (conmutadores, enrutadores, firewalls, entre otros). – Servidores u ordenadores. – Matrices de almacenamiento. – Dispositivos de backup. 	<ul style="list-style-type: none"> – Bases de datos (PostgreSQL, MongoDB, MySQL, entre otros). – Microsoft Azure. – Amazon Web Services (AWS). – Google Cloud. – Entre otros. 	<ul style="list-style-type: none"> – TCP/IP – HTTP. – MQTT. – WebSocket.

La Tabla 2 presenta una taxonomía del software y hardware utilizado e investigado en este proyecto, centrándose en la capa de la nube. El principal objetivo de esta taxonomía es clasificar y organizar de manera sistemática los diversos elementos empleados para la implementación y configuración de la capa de nube en el contexto de este proyecto. Esta estructuración facilita no solo el entendimiento de los componentes utilizados, sino también la selección de software y hardware adecuado al momento de replicar este proceso en otros contextos.

En la siguiente sección, se realiza el mismo análisis detallado para la capa de borde, permitiendo comprender y categorizar el software y hardware involucrado en esta parte del proyecto. Esta metodología de análisis y clasificación en las tres capas de infraestructura garantiza un enfoque sistemático en la implementación tecnológica, lo que resulta esencial para la replicabilidad y eficiencia del proyecto en futuras aplicaciones y desarrollos.

2.4 Computación en el borde

La computación en el borde se presenta como una solución para abordar los desafíos de latencia y velocidad de transmisión al enviar grandes volúmenes de información a la nube. En este enfoque, los dispositivos locales, como controladores PLC o sistemas embebidos, asumen un papel más activo en la ejecución de los algoritmos principales de control.

La idea principal es que estos dispositivos realicen el procesamiento y análisis de los datos en el lugar donde se generan, es decir, en el borde de la red. Esto permite una mayor eficiencia en términos de tiempo de respuesta y reducción de la carga en la red. Los

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

resultados obtenidos localmente se pueden enviar posteriormente a una base de datos en la nube para su almacenamiento y análisis adicional (Kim, y otros, 2022).

Con la computación en el borde, se logra acercar el mundo físico al mundo digital, permitiendo una toma de decisiones más rápida y autónoma en tiempo real. Además, se reduce la dependencia de una conexión continua y estable a la nube, lo que aumenta la disponibilidad y la confiabilidad de los sistemas.

Tabla 3 Taxonomía de hardware y software de la capa de borde

Hardware	Software	Protocolos de comunicación
<ul style="list-style-type: none"> - PLC - Sistemas embebidos (microcontroladores como Arduino) - Sensores - Actuadores 	<ul style="list-style-type: none"> - Ladder - Lista de Instrucciones (IL o STL) - Texto Estructurado (ST) - Diagrama de Bloques (SFD) - C++ - Python 	<ul style="list-style-type: none"> - Industrial Ethernet - Profibus DP - Modbus TCP/IP - RS - 232 - UART/USART - I2C - TCP/IP - MQTT - HTTP

En la Tabla 3, al igual que en la sección anterior, se ofrece una taxonomía detallada del software y hardware que caracteriza a la capa de borde. En esta capa, se emplean dispositivos de nivel industrial o de planta, lo cual es importante tener en cuenta. Asimismo, se observa una amplia gama de protocolos de comunicación utilizados, ya que se deben considerar los estándares industriales.

Esta taxonomía permite precisar los elementos clave involucrados en la implementación de la capa de borde, lo que facilita la comprensión de su funcionamiento y la selección adecuada de componentes para futuras aplicaciones. Al tener en cuenta dispositivos industriales y estándares específicos de comunicación, se garantiza una integración efectiva y confiable en entornos industriales, lo que resulta esencial para el éxito del proyecto en el contexto de esta investigación.

En la siguiente sección, llevamos a cabo el análisis de la última capa presentada en este trabajo: la capa de niebla. En esta etapa es donde se lleva a cabo todo el proceso de separación y transferencia de información de manera detallada y precisa.

2.5 Computación en la niebla

La computación en la niebla, también conocida como Fog Computing, es una capa adicional que se superpone tanto a la capa de nube como a la capa de borde en la arquitectura de computación. Esta capa intermedia se encuentra entre los dispositivos en el borde de la red y los servidores en la nube.

La computación en la niebla combina las capacidades de procesamiento y almacenamiento local de los dispositivos en el borde con la capacidad de comunicación y procesamiento de la nube. Esto permite que los datos se procesen y analicen no solo en el borde, sino también en esta capa antes de ser enviados a la nube.

La principal ventaja de la computación en la niebla es la capacidad de reducir la carga en la red y mejorar la eficiencia en la transmisión de datos. Al procesar y analizar los datos en la capa de niebla, se puede realizar una filtración y selección de la información relevante antes de enviarla a la nube. Esto ayuda a optimizar el uso de ancho de banda y a reducir la latencia en las comunicaciones (Garg & Dave, 2019).

Además, la computación en la niebla ofrece una mayor capacidad de almacenamiento y procesamiento en comparación con los dispositivos en el borde, lo que permite ejecutar aplicaciones más complejas y realizar análisis más avanzados. La computación en la niebla desempeña un papel crucial en la optimización de la comunicación y el procesamiento de datos en entornos industriales.

Tabla 4 Taxonomía de hardware y software de la capa de niebla

Hardware	Software	Protocolos de comunicación
<ul style="list-style-type: none"> - Servidor u ordenador. - Microcomputadora o microordenador (Raspberry PI). 	<ul style="list-style-type: none"> - JavaScript. - NodeRED. - Node.js. - Python. - C++. - Entre otros. 	<ul style="list-style-type: none"> - Ethernet. - Industrial Ethernet. - Modbus TCP/IP - RS - 232 - UART/USART - TCP/IP - HTTP. - MQTT. - WebSocket.

La Tabla 4 presenta la taxonomía detallada de la capa de niebla, destacando aspectos fundamentales. En lo referente al hardware, es esencial contar con un servidor capaz de ejecutar procesos informáticos y conectado a la red local con acceso a internet, lo cual habilita la transferencia de información hacia la nube. Respecto al software, se permite la utilización de lenguajes de alto nivel, lo que simplifica en gran medida la programación

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

de estos sistemas. Por último, en cuanto a los protocolos de comunicación, esta capa destaca por su versatilidad, ya que admite una amplia gama de protocolos, incluyendo el uso de Ethernet industrial y Modbus TCP como protocolos con estándar industrial.

Esta taxonomía brinda una visión completa de los elementos clave que definen la capa de niebla. Al resaltar los elementos esenciales, se facilita una comprensión clara de cómo opera esta capa y cómo se relaciona con las demás en el contexto del proyecto. La capacidad de utilizar múltiples protocolos de comunicación, incluidos aquellos con estándares industriales, amplía las posibilidades de aplicación práctica y mejora la interoperabilidad con otros sistemas en entornos industriales y de comunicación avanzados. Con esta información detallada, se sientan las bases para una implementación exitosa y eficiente de la capa de niebla en el proyecto, permitiendo una gestión eficaz de la información.

En la siguiente sección, presentaremos una propuesta de modificación a la clásica pirámide de la automatización con el objetivo de mejorar la caracterización e identificación de los conceptos de las capas de borde, niebla y nube aplicados en el contexto de la automatización. Con esta propuesta, buscamos brindar una visión actualizada y enriquecida de la automatización, que refleje de manera más precisa la realidad tecnológica actual y fomente la adopción de estrategias de automatización más avanzadas y eficientes en el contexto de la industria moderna.

2.6 Propuesta de modificación de la pirámide de la automatización

Con el objetivo de integrar los nuevos desarrollos tecnológicos en los sistemas convencionales, se ha realizado una modificación en la clásica pirámide que describe los niveles de automatización de un sistema, tal como se muestra en la Figura 1. En esta nueva representación, se ha incorporado la capa de la niebla en el Nivel 3 de la pirámide, junto con la etapa de supervisión. También se ha incluido el límite que delimita la capa del borde, y se ha añadido la capa de la nube, mostrando claramente el flujo de la información.

Es relevante destacar que el límite del borde abarca también a la capa de niebla en ciertos casos, ya que pueden compartir ubicación física. Esto ocurre especialmente cuando la capa de niebla obtiene información directamente del controlador y comparte el nivel con las pantallas de supervisión. No obstante, es esencial distinguir estas capas según su función y los servicios que ofrecen. La capa de niebla, aunque puede estar presente en la planta como un ordenador instalado en la red local, desempeña el importante rol de enrutador de información. Al analizar minuciosamente la imagen, se observa que la información se dirige directamente desde el nivel 3, y es en esta capa donde se toma la decisión sobre la dirección que tomará la información, ya sea hacia la nube o hacia otro nivel superior de la pirámide. Por lo tanto, la distinción clara entre estas capas resulta

fundamental para comprender el flujo de datos y la función que cumplen en el sistema en su conjunto.

Por otro lado, la capa de borde comprende todas las necesidades vinculadas a la planta o al proceso que se desea automatizar, englobando elementos como sensores, actuadores y el controlador industrial. Es en esta capa donde se lleva a cabo la automatización tradicional de un proceso.

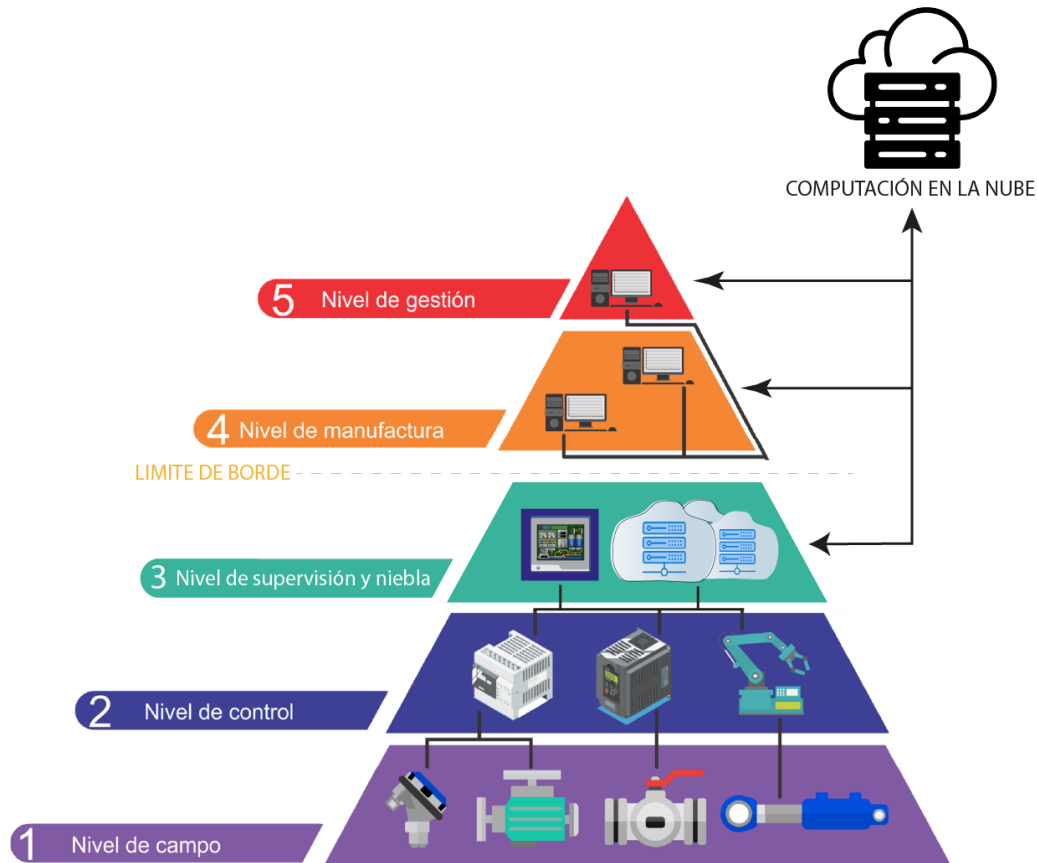


Figura 1 Modificación a la pirámide de la automatización.

La implementación de estas modificaciones facilita una gestión más efectiva de los datos generados en un proceso específico. A lo largo de las diferentes etapas de automatización, se generan volúmenes de información diversos, los cuales deben separarse para evitar ralentizar la operación del sistema. En otras palabras, cuando la información llega al nivel de supervisión y niebla (nivel 3), esta será gestionada por la capa de niebla, la cual decidirá qué información debe continuar su camino hacia los niveles superiores y cuál debe permanecer como parte del sistema. Esto asegura una respuesta inmediata y un control eficiente del proceso, evitando saturar la red del servidor local o depender exclusivamente de la conexión a internet en caso de necesitar información.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Existe información relacionada con la trazabilidad, el seguimiento y el control de calidad, esta información administrativa puede procesarse y almacenarse directamente en la nube. Esto optimiza el almacenamiento, reduce el tráfico de información en la red y mejora la seguridad. Sin embargo, es crucial analizar los niveles más altos de la pirámide, ya que manejan grandes volúmenes de información. En este contexto, es importante validar si toda la información debe transferirse a la nube o si ciertos datos requieren un almacenamiento local. Esta estrategia permite reducir las deficiencias asociadas al uso de nubes, como la necesidad de contar con altos anchos de banda, conectividad continua y la latencia inherente a la operación de la red (Kim, y otros, 2022).

En este capítulo, hemos proporcionado el contexto necesario para comprender la selección e implementación de la arquitectura y el modelo de comunicación que se presenta a continuación. Para ello, se han abordado conceptos fundamentales, como los contenedores, los protocolos de comunicación y las diferentes capas con su taxonomía de hardware y software, que serán manejados a lo largo de este trabajo.

Además, se ha presentado una propuesta de la clásica pirámide de la automatización, que resulta crucial para obtener una comprensión clara de estos conceptos y del flujo de automatización que buscamos establecer en este proyecto. Esta perspectiva proporcionará un marco teórico sólido para el desarrollo de la arquitectura y el modelo de conexión que se detallan en la próxima sección. Además, se presentarán los criterios de selección y los procedimientos de configuración de los métodos de comunicación estudiados y elegidos, lo cual fue un pilar fundamental en el planteamiento y desarrollo de este proyecto.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3. Arquitectura y protocolos de comunicación evaluadas y seleccionadas

En este capítulo, se presenta la arquitectura o configuración utilizada en el módulo de variables industriales del ITM, basada en los conceptos y tecnologías discutidos en el capítulo anterior. Se describirán los criterios de selección y los métodos de comunicación implementados en este caso de aplicación.

En resumen, en este capítulo se presenta la arquitectura y la configuración utilizada en el módulo de variables industriales del ITM. Se detallan las diferentes capas y los procesos involucrados en cada una de ellas, así como los métodos de comunicación implementados. Esta configuración permite el monitoreo y control remoto de la planta industrial a través de una interfaz gráfica accesible desde cualquier ubicación.

3.1 Arquitectura seleccionada

En el diagrama presentado en la Figura 2, se muestra una representación visual de la arquitectura utilizada en el módulo de variables industriales. Esta arquitectura se divide en tres capas: borde, niebla y nube, y muestra cómo interactúan entre sí. También se distinguen los distintos subprocesos que componen la configuración y cómo fluye la información entre ellos.

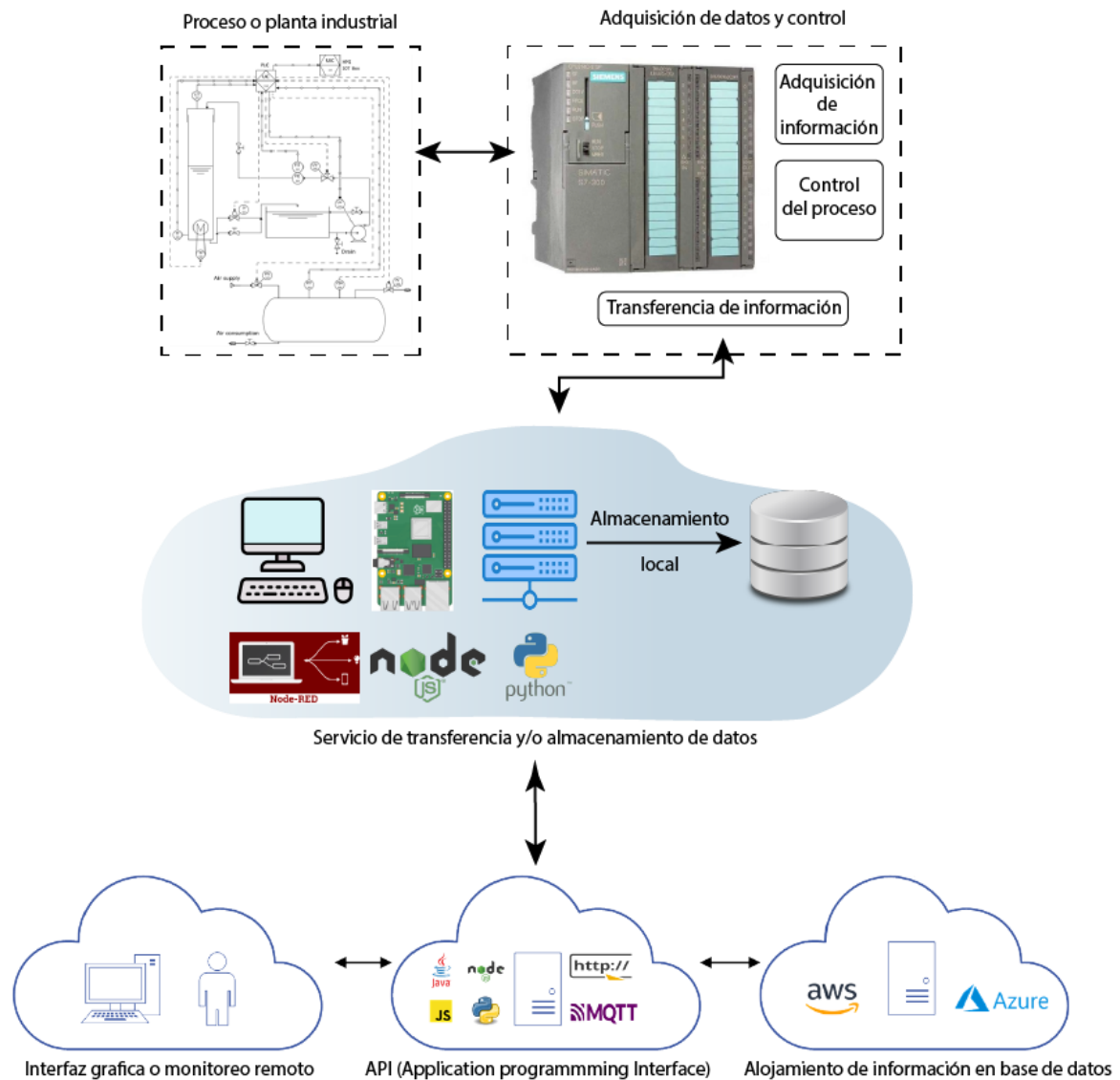


Figura 2 Arquitectura de sistemas interconectados

3.2 Capa de borde

Como se mencionó en la sección 2.4 La capa de borde representa todo lo relacionado con el nivel de planta o el proceso que se quiere automatizar. A continuación, se presenta siguiendo la arquitectura expuesta en la Figura 2 la caracterización de la capa de borde:

- **Planta industrial:** Representa el proceso que se desea automatizar, controlando variables con sensores y actuadores.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

- **Proceso de adquisición de datos y control:** En esta etapa, el controlador, en este caso un PLC, adquiere datos de los sensores y realiza el control necesario para el correcto funcionamiento del proceso.

3.3 Capa de niebla

Haciendo uso de la dinámica anterior, en esta subsección se presenta la caracterización de la capa de niebla teniendo en cuenta la arquitectura presentada en la Figura 2:

- **Almacenamiento y procesamiento de información:** En esta etapa, se utiliza un servidor u ordenador que ejecuta una aplicación llamada NodeRED. Esta se encarga de almacenar información que no requiere ser enviada a la nube y también permite realizar procesamiento adicional utilizando lenguajes de alto nivel para programar diversas tareas.
- **Interfaz de comunicación y transferencia de datos:** NodeRED se utiliza como una interfaz de comunicación entre aplicaciones. Permite configurar y utilizar diferentes protocolos de comunicación, como HTTP y MQTT. En este caso de estudio, se utiliza la comunicación por HTTP y se puede agregar código personalizado en Python u otro lenguaje para manipular manualmente las reglas de comunicación.

3.4 Capa de nube

Finalmente se presenta la caracterización de la capa de nube según la arquitectura antes expuesta:

- **Alojamiento de información en base de datos:** En la nube, se accede a servicios de base de datos. En este caso, se utiliza el servicio de MongoDB para alojar información en Internet y procesarla posteriormente con otra aplicación.
- **API (Application Programming Interface):** Es la interfaz de comunicación entre aplicaciones. En este caso particular, se establece la comunicación entre la planta industrial y una base de datos, y a su vez, entre la base de datos y una interfaz gráfica.
- **Interfaz gráfica o monitoreo remoto:** Es la interfaz final donde los usuarios pueden visualizar opciones en una página web y manipular el proceso de forma remota.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3.5 Criterios de evaluación y selección

En este apartado se presentarán los criterios utilizados para seleccionar los métodos de comunicación a emplear en la arquitectura del módulo de variables industriales. Estos criterios se basarán en las necesidades y requisitos específicos del sistema, así como en las características y ventajas de cada método.

- Velocidad de transferencia:** Se debe evaluar la velocidad de transferencia de datos requerida por un sistema. Si se necesita una comunicación rápida y eficiente, protocolos como Industrial Ethernet o Modbus TCP/IP podrían ser adecuados. Por otro lado, si la velocidad no es una preocupación principal, protocolos como UART/USART o I2C podrían ser suficientes.
- Distancia de transmisión:** Una variable a veces determinante es la distancia a la que se deben transmitir los datos. Si se necesita una comunicación a larga distancia, protocolos como Industrial Ethernet o incluso TCP/IP podrían ser más adecuados debido a su soporte para redes extensas. Para distancias más cortas, protocolos como UART/USART o I2C podrían ser suficientes.
- Capacidad de integración:** Se debe asegurar de que los protocolos seleccionados sean compatibles y se integren bien con los dispositivos y sistemas que se utilizan en un proyecto. Algunos protocolos, como Modbus TCP/IP o MQTT, tienen amplio soporte en diversos dispositivos y son fáciles de integrar.
- Flexibilidad y escalabilidad:** Se debe considerar si el protocolo seleccionado es lo suficientemente flexible y escalable para acomodar posibles expansiones o cambios futuros en el sistema. Algunos protocolos, como Industrial Ethernet y TCP/IP, son muy flexibles y se pueden adaptar a diferentes necesidades.
- Seguridad:** La seguridad es una preocupación importante en los proyectos industriales, al seleccionar los protocolos se debería asegurar de que ofrezcan mecanismos de cifrado y autenticación robustos. Protocolos como MQTT y HTTPS (utilizando TCP/IP) pueden proporcionar capas de seguridad adicionales.
- Compatibilidad con estándares existentes:** Si un proyecto se conecta a sistemas existentes o equipos de diferentes fabricantes, es fundamental seleccionar protocolos que sean compatibles con los estándares utilizados en esos sistemas. Este es un criterio importante en este trabajo, que el sistema implementado fuera transversal a múltiples equipos, ya que posteriormente se buscaba replicar en los diferentes controladores disponibles en los laboratorios. Por ejemplo, si estás trabajando con equipos industriales, es posible que desees optar por protocolos como Modbus TCP/IP o Profibus DP, que son ampliamente utilizados en la industria.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

- **Facilidad de implementación:** Facilidad de implementación es un criterio clave al seleccionar un protocolo de comunicación, ya que impacta directamente la implementación y el mantenimiento del sistema. Para que un protocolo sea óptimo en este aspecto, se deben considerar los siguientes puntos:
 - **Configuración sencilla:** El protocolo debe contar con una configuración clara y comprensible.
 - **Documentación detallada:** Es importante que el protocolo cuente con guías de usuario y manuales técnicos bien elaborados.
 - **Herramientas de diagnóstico:** Debería proporcionar herramientas que faciliten a los usuarios la identificación y resolución rápida de problemas de comunicación.
 - **Interoperabilidad:** Un protocolo fácil de usar debe ser compatible con una amplia variedad de dispositivos y sistemas de diferentes fabricantes, sin requerir configuraciones adicionales complicadas.
 - **Manejo de errores:** El protocolo debe tener mecanismos que detecten y manejen errores de manera elegante, proporcionando mensajes de error claros y útiles para facilitar la depuración y corrección de problemas.
 - **Soporte de la comunidad:** Es beneficioso que el protocolo cuente con una comunidad activa de usuarios y desarrolladores, para obtener respuestas a preguntas y soluciones a problemas comunes.
 - **Interfaces de usuario amigables:** Si el protocolo incluye herramientas o software de configuración, estas interfaces deben ser intuitivas y fáciles de usar, facilitando así el proceso de configuración y monitoreo del sistema.
 - **Costo:** Finalmente, evalúa el costo asociado con la implementación de cada protocolo. Algunos protocolos pueden requerir hardware adicional o licencias especiales, lo que podría influir en la decisión final. En este trabajo se buscó que todos los protocolos de comunicación utilizados fueran de uso gratuito y sin licencias, para permitir esa transversalidad y replicabilidad que se buscaba desde el planteamiento del problema.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

3.6 Caracterización y configuración de los protocolos de comunicación elegidos

En esta sección, se ofrece una comparativa entre los protocolos de comunicación que han sido estudiados y seleccionados para este proyecto, siguiendo los criterios previamente mencionados. A continuación, se presentan dichos protocolos, acompañados de una breve descripción o procedimiento de su configuración. Además, en la Tabla 5, se muestra la caracterización de estos protocolos según los criterios mencionados anteriormente.

- **Profibus DP:** Cada dispositivo conectado al bus Profibus DP debe tener una dirección única y esto es importante tenerlo en cuenta en su configuración para que no haya conflictos. Para configurar y utilizar este protocolo de manera general, se puede seguir los siguientes pasos:
 - a) Configuración del maestro: En el dispositivo maestro se habilita la opción que permite iniciar la comunicación con los dispositivos esclavos en el bus Profibus DP.
 - b) Topología del bus: Decidir la topología física del bus Profibus DP. Según los requisitos del sistema, se puede utilizar una configuración lineal, en estrella o mixta, entre otros.
 - c) Conexión y verificación: Se conecta el bus de datos correspondiente a la topología elegida, se verifica que las conexiones sigan los estándares industriales y que permita la comunicación efectiva entre los dispositivos del sistema.

- **Profibus PA:** Este protocolo de comunicación está muy dirigido a los dispositivos y sensores conectados en campo, para configurar este protocolo se debe tener en cuenta:
 - a) Configuración del segmento Profibus PA: Se define la topología de la red, luego se asigna direcciones únicas a cada dispositivo y se configuran los parámetros de comunicación.
 - b) Configuración del maestro Profibus PA: Se establece la dirección y los parámetros de comunicación del dispositivo maestro.
 - c) Configuración de los dispositivos esclavos Profibus PA: Se definen cuáles van a ser los dispositivos esclavos configurando la dirección y los parámetros de comunicación, finalmente se asignan las variables que se desean intercambiar.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

d) Prueba y verificación: Se realizan pruebas de comunicación y se verifica que haya intercambio de información, con el fin de asegurar una comunicación estable y confiable.

- **Industrial Ethernet:** Este es uno de los protocolos más relevantes en la actualidad. Para configurar y utilizar este protocolo de manera general, se puede seguir estos pasos:

- a) Selección de tipo de red: Definir el tipo de red industrial Ethernet que se requiere utilizar, como Ethernet/IP, PROFINET o Modbus TCP, entre otros, según los requerimientos del sistema.
- b) Configuración de direcciones IP: Se debe asignar las direcciones IP únicas a cada dispositivo que participe en la red industrial Ethernet.
- c) Configuración de switches: Si se utilizan switches Ethernet para la conexión de múltiples dispositivos, se debe verificar que estén adecuadamente configurados para permitir que haya comunicación efectiva entre los dispositivos de la red.

- **Modbus TCP/IP:** El protocolo Modbus TCP/IP se basa en Ethernet y TCP/IP. Para configurar y utilizar este protocolo de manera general, se puede seguir estos pasos:

- a) Asignación de direcciones IP: Se asignan direcciones IP únicas a cada dispositivo que participe en la comunicación Modbus TCP/IP.
- b) Puertos TCP: Modbus TCP/IP utiliza el puerto 502 por defecto. Por lo que se debe verificar que este puerto esté abierto y disponible para la comunicación.
- c) Modo maestro/esclavo: Definir cuales dispositivos actuarán como maestros y cuáles como esclavos. El maestro iniciará las solicitudes y el esclavo responderá a ellas.
- d) Direcciones de dispositivos: Cada dispositivo esclavo debe tener una dirección única para que el maestro pueda dirigir sus solicitudes a dispositivos específicos.

- **HTTP:** Este protocolo, ampliamente utilizado en Internet, permite una comunicación efectiva para enviar y recibir archivos desde cualquier ordenador en cualquier parte del mundo que esté conectado a la red. Para utilizar este protocolo de manera adecuada, es recomendable seguir las siguientes pautas:

- a) Configuración del servidor HTTP: Instalar y configurar el servidor HTTP en el sistema que actuará como servidor. Establecer los puertos y direcciones IP para escuchar las solicitudes de los clientes.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

- b) Configuración del cliente HTTP: Implementar el cliente HTTP en el dispositivo o sistema que enviará solicitudes al servidor. Definir los parámetros y datos para enviar en las solicitudes HTTP.
- c) Definición de las rutas y recursos: En el servidor, definir las rutas y recursos a los que el cliente podrá acceder a través de las solicitudes HTTP. Establecer los métodos permitidos para cada recurso (GET, POST, PUT, DELETE, etc.).
- d) Autenticación y seguridad: Configurar la autenticación y autorización para controlar el acceso a ciertos recursos y funciones del servidor. Implementar medidas de seguridad, como encriptación y certificados SSL/TLS, para proteger la comunicación entre el cliente y el servidor.
- e) Prueba y verificación: Realizar pruebas para asegurarse de que el servidor y el cliente puedan comunicarse correctamente a través de HTTP. Verificar que las solicitudes se procesen correctamente y que las respuestas contengan la información esperada.

Es importante destacar que Profinet y Modbus TCP/IP son protocolos basados en Industrial Ethernet, lo que significa que comparten algunas características y aspectos similares. Su principal diferencia radica en el tipo de cable utilizado y en la forma en que se empaqueta la información para su transmisión.

Tabla 5 Caracterización de protocolos de comunicación

Protocolo de comunicación	Velocidad de transferencia	Distancia de transmisión	Capacidad de integración	Flexibilidad y escalabilidad	Seguridad	Facilidad de implementación	Costo	Referencia
Profibus DP	500 kbps; 1.5 Mbps; 12 Mbps	Cables eléctricos: 9.6 km; Cables ópticos: 90 km	Solo Industrial	Poco flexible Max 32 dispositivos	Sin cifrado	Difícil de configurar	Altos	(Vadi, Bayindir, Toplar, & Colak, 2022)
Profibus PA	31.25 kbps	Cables eléctricos: 9.6 km; Cables ópticos: 90 km	Solo Industrial	Poco flexible Max 32 dispositivos	Sin cifrado	Difícil de configurar	Altos	(Vadi, Bayindir, Toplar, & Colak, 2022)
Modbus TCP/IP	100 Mbps	100 mts	Solo Industrial	Alta Indefinido	Sin cifrado	Medio (Depende del hardware)	Medio (Hardware)	(Yuanyuan & Meng, 2021)
Industrial Ethernet	10 Mbps - 1 Gbps	100 mts	Versátil	Alta Indefinido	Cifrado adicional TSL/SSL	Medio (Depende del hardware)	Medio (Hardware)	(Wu & Xie, 2019)
HTTP	Mayor a 100 Mbps	Indefinido	Versátil	Alta Indefinido	Cifrado adicional TSL/SSL	Fácil	Bajo (Cobros por uso en la nube)	(A. da Cruz, y otros, 2019)

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

En este capítulo se presentaron los criterios utilizados para la selección de los métodos de comunicación en la arquitectura del módulo de variables industriales. Se consideraron las necesidades y requisitos del sistema, así como las características y ventajas de cada método. Además, se describieron los procedimientos necesarios para configurar y establecer la comunicación entre los diferentes elementos de la arquitectura. La aplicación de estos métodos permitirá una comunicación eficiente y confiable, facilitando el monitoreo y control remoto de la planta industrial.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

4. Aplicación de las herramientas en el módulo de variables

En esta sección, se va a aplicar la arquitectura presentada en la figura 2, de manera práctica en uno de los módulos de los laboratorios del ITM. Además, se dividirá por las etapas anteriormente estudiadas en este documento Borde, Niebla y Nube, donde se presentará de manera específica los puntos y protocolos de comunicación utilizados en cada una de las etapas de esta arquitectura.

4.1 Capa de borde para el caso de aplicación.

En el borde el módulo de variables industriales es el que realiza todas las operaciones de adquisición y procesamiento de información de los sensores. El PLC actualmente programado en el módulo implementa algoritmos de control P, PI y PID, los cuales realizan acciones de control sobre los actuadores para mantener un set point o punto de control preestablecido. En la tabla 6 se puede observar los diferentes dispositivos de los que este dispone, y su función en el proceso de control. En la Figura 3 se puede ver una fotografía del módulo (a), además se puede apreciar un diagrama de cómo están dispuestos sus diferentes dispositivos que realizan la etapa de control (b).

Tabla 6 Caracterización de dispositivos del módulo de variables

Descripción	Etiqueta	Referencia	Función
PLC	UC	SIEMENS 314C2DP	Controlador del proceso
HMI	UI	KTP700-Basic	Interfaz para el control local del proceso
IOT Box	UIC	Raspberry pi 4 B	Interfaz de comunicación entre aplicaciones
Transmisor indicador de flujo	FIT	MAG6000 PA Extended	Mide el flujo de bombeo hacia el tanque vertical
Transmisor indicador de nivel	LIT	SITRANS PROBE LU	Mide el nivel del fluido en el tanque vertical

Descripción	Etiqueta	Referencia	Función
Transmisor de temperatura	TT	SITRANS T3K PA	Mide la temperatura del fluido
Relé de estado solido	TY	OPTO 22 240D45	Activa la resistencia de calentamiento
Válvula Proporcional	UV	Danfoss EV260B	Regula el flujo de la bomba hacia el tanque vertical
Electro válvula	LV	KSD - FB2E	Válvula de descarga del tanque vertical
Bomba y variador de frecuencia	UY	Micromaster 440	Regula el flujo de la bomba hacia el tanque vertical

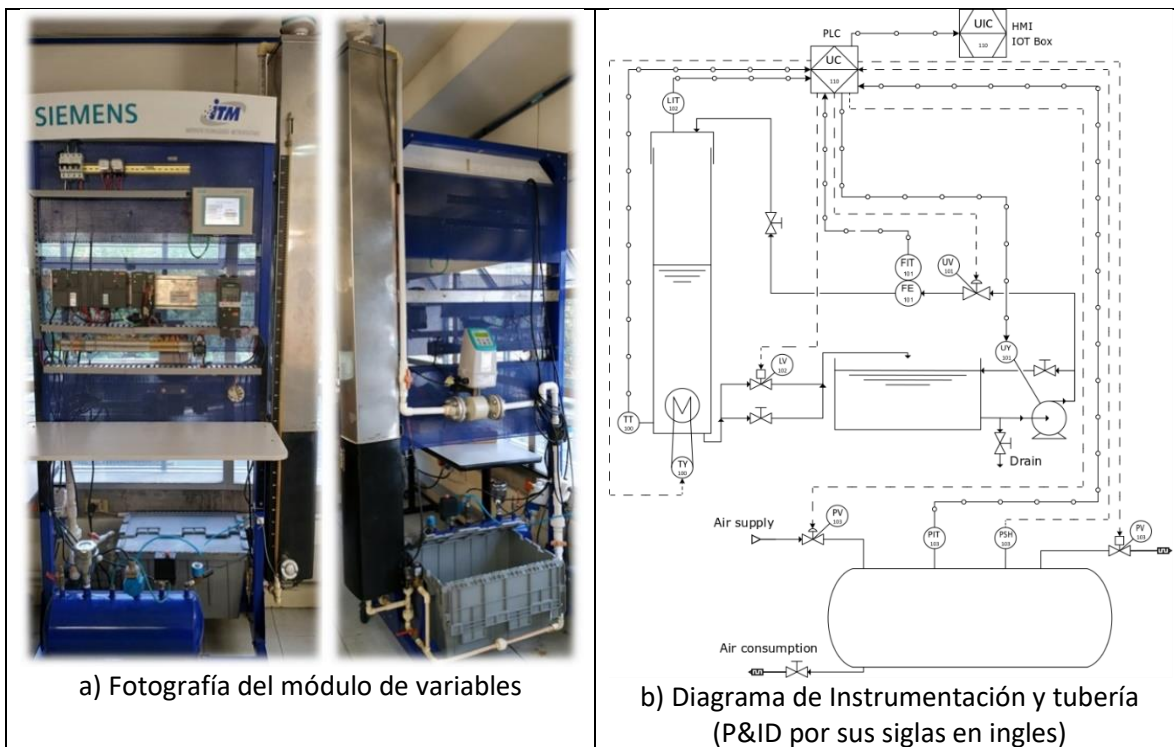


Figura 3 Módulo de variables industrial del ITM

En los procesos industriales, estas operaciones de control son especialmente críticas. Para que un control funcione correctamente, la información adquirida y analizada debe ser prácticamente en tiempo real. Esto significa que hay procesos que no toleran retrasos

o latencias, además el procesamiento y la acción de control deben ser precisos y tener una respuesta en el menor tiempo posible. Por esta razón, existen protocolos y estándares establecidos en la industria que permiten garantizar estas condiciones. En este punto, es donde el procesamiento en el borde cobra gran importancia.

En la Figura 4 se pueden observar las conexiones de los sensores y actuadores etiquetados con sus respectivos protocolos de comunicación. Estos protocolos son estándares de la industria y permiten la transmisión confiable y precisa de la información entre los dispositivos en el borde.

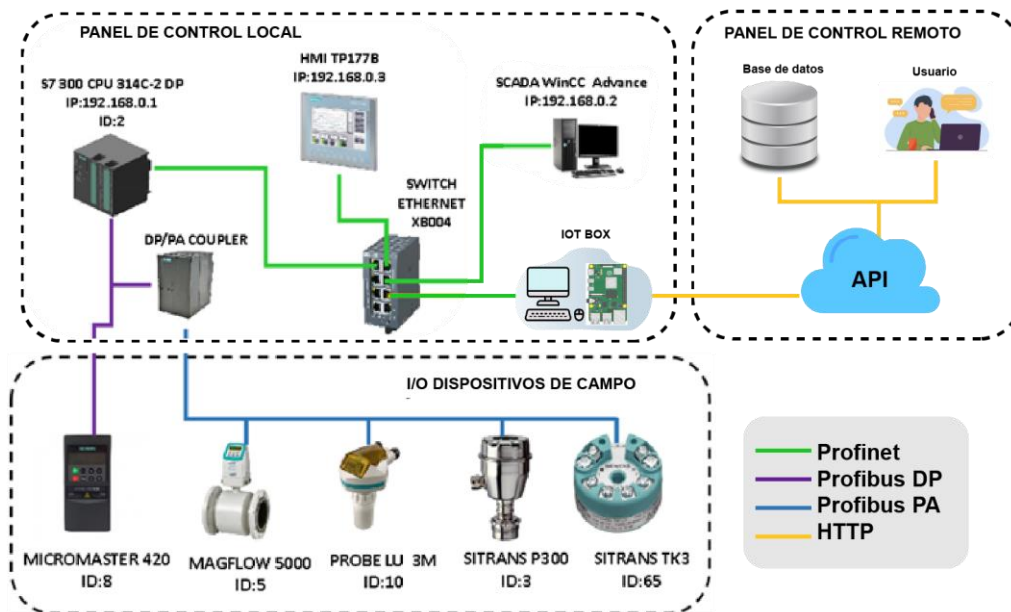


Figura 4 Conexiones y Protocolos

4.2 Capa de niebla para el caso de aplicación.

Para la etapa de niebla, se utilizó una Raspberry Pi 4 modelo B. Se siguió la documentación oficial del fabricante para instalar el sistema operativo recomendado (Raspberry Pi, 2023). Además, se habilitó la comunicación SSH (Secure Shell) para poder acceder a la Raspberry a través de su dirección IP en una red local. Esto permite establecer una conexión de consola o terminal de comunicación mediante la cual se instaló el programa Docker con la imagen de NodeRED instalada, preparada para su uso.

NodeRED es una herramienta que al instalarse permite crear flujos de programación mediante diagramas de flujo. En la Figura 5 y 6 se muestra respectivamente la interfaz de inicio de Docker y NodeRED, mientras en la Figura 7 se presentan los plugin necesarios para el desarrollo de este proyecto.

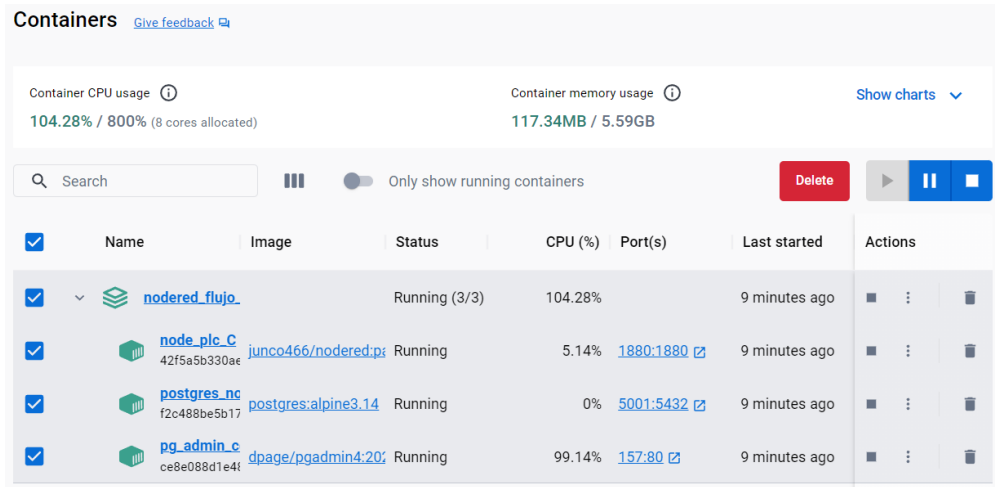


Figura 5 Interfaz de Docker, donde se presentan los contenedores.

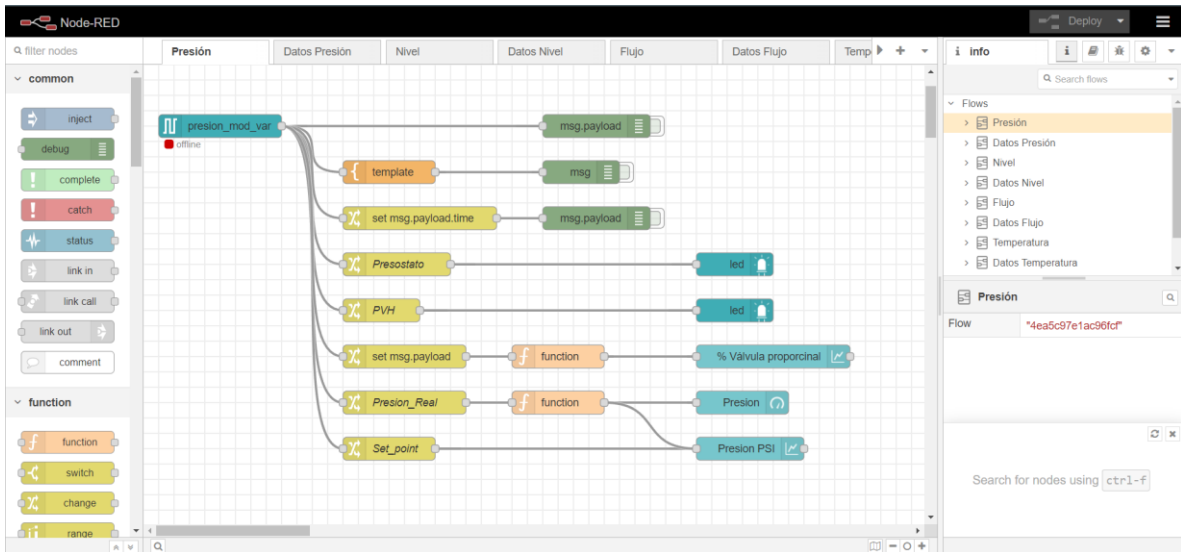


Figura 6 Interfaz de general NodeRED

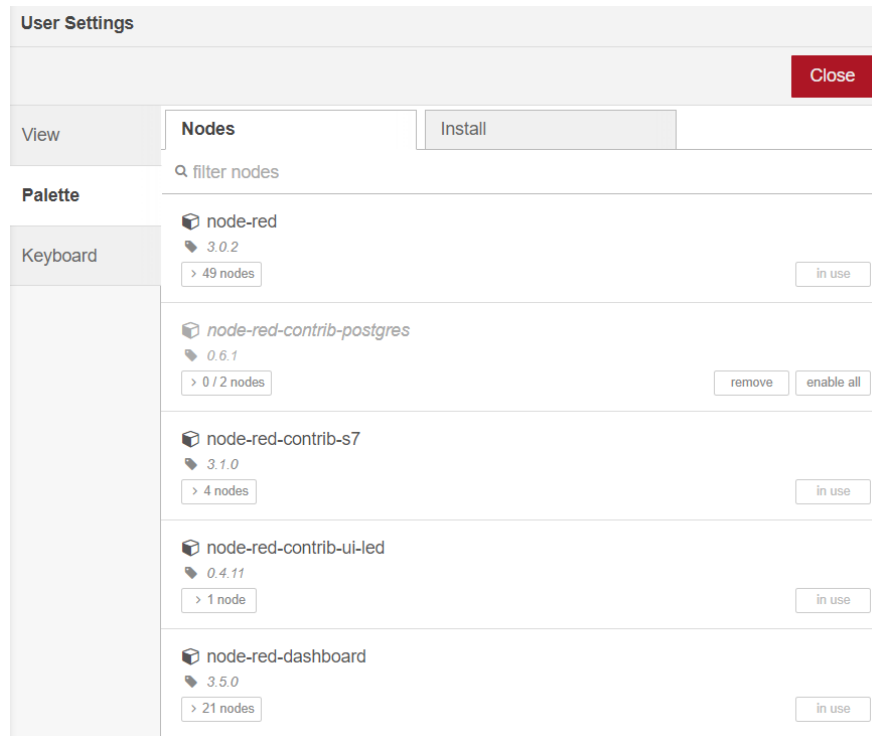


Figura 7 Plugin necesarios para el funcionamiento correcto de la aplicación

Luego de instalados los plugin, NodeRED proporciona diferentes nodos, uno de ellos es el nodo S7, el cual permitirá la comunicación con el módulo mediante el protocolo Step 7 de Siemens, en la Figura 8 se presenta como se ve el nodo en la aplicación. Allí configurando la dirección IP y el rack que se designó en el PLC al momento de su programación, se puede establecer comunicación como se muestra en la Figura 9, además en la Figura 10, se presenta la manera en que se puede configurar las variables que vienen del PLC.

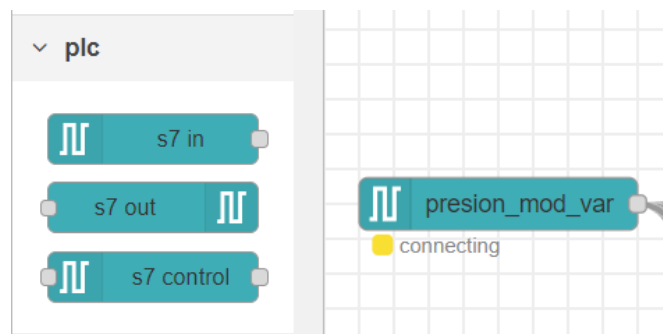
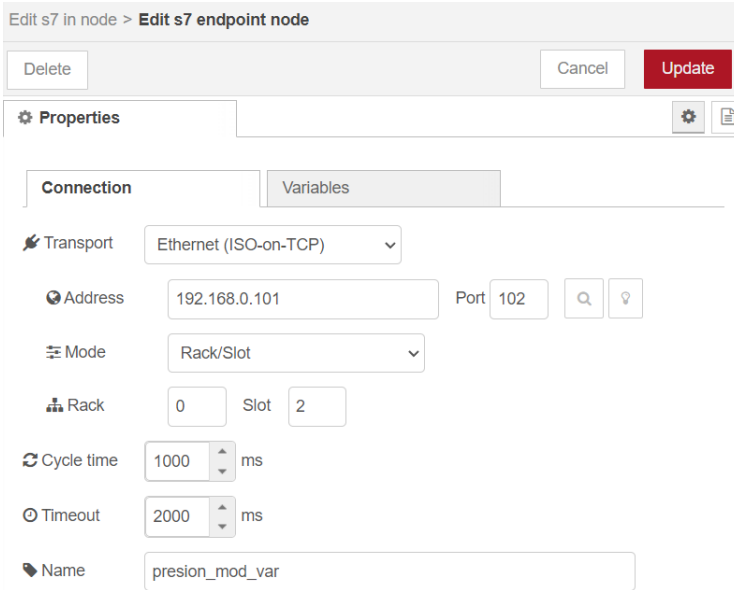


Figura 8 Nodo de comunicación Step 7



Edit s7 in node > Edit s7 endpoint node

Delete Cancel Update

Properties

Connection Variables

Transport: Ethernet (ISO-on-TCP)

Address: 192.168.0.101 Port: 102

Mode: Rack/Slot

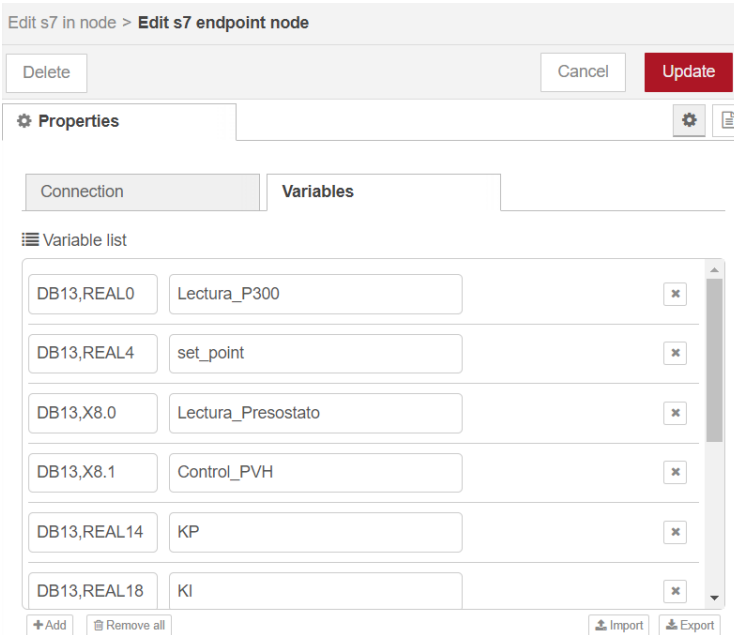
Rack: 0 Slot: 2

Cycle time: 1000 ms

Timeout: 2000 ms

Name: presion_mod_var

Figura 9 Configuración general del nodo Step 7



Edit s7 in node > Edit s7 endpoint node

Delete Cancel Update

Properties

Connection Variables

Variable list

DB13,REAL0	Lectura_P300	x
DB13,REAL4	set_point	x
DB13,X8.0	Lectura_Presostato	x
DB13,X8.1	Control_PVH	x
DB13,REAL14	KP	x
DB13,REAL18	KI	x

+ Add Remove all Import Export

Figura 10 Configuración de variables protocolo Step 7

Se preparan los nodos de comunicación HTTP donde se determina que método de comunicación se va a utilizar y el "Endpoints" y así determinar la dirección o ruta que va a tomar la información. En la Figura 11 se presenta el nodo que permite la configuración HTTP, además en la Figura 12 se muestra su configuración el nodo de HTTP.

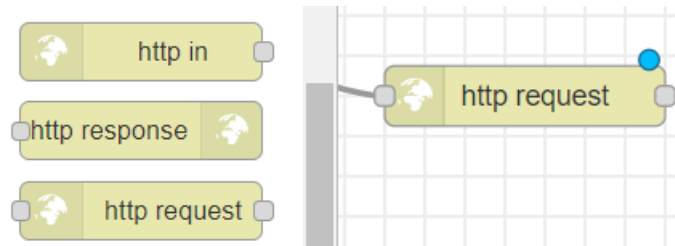


Figura 11 Nodo para el uso del protocolo HTTP

The screenshot shows the configuration interface for an 'http request' node. At the top, there are 'Delete', 'Cancel', and 'Done' buttons. Below is a 'Properties' section with the following settings:

- Method:** POST
- URL:** 127.0.0.1/presion
- Enable secure (SSL/TLS) connection
- Use authentication
- Enable connection keep-alive
- Use proxy
- Only send non-2xx responses to Catch node
- Disable strict HTTP parsing
- Return:** a UTF-8 string
- Headers:** (empty list)

Figura 12 Nodo de configuración del protocolo HTTP

4.3 Capa de nube para el caso de aplicación.

Se desarrolla una API o aplicación de interfaz utilizando el lenguaje de programación JavaScript con la herramienta Node.js. Esta API cumple la función de ser un puente de comunicación entre las distintas aplicaciones que forman parte del sistema completo. Finalmente, se aloja en un servidor que forma parte de la nube, con el propósito de facilitar el acceso de cualquier dispositivo a las reglas de comunicación implementadas para el enrutamiento de información.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Además, esta API permite configurar la estructura de información que posteriormente será migrada a la base de datos. Los detalles específicos de esta estructura se detallarán más adelante en el desarrollo del proyecto.

Como se mencionó anteriormente, la API cumple un papel fundamental en el establecimiento de reglas y protocolos necesarios para garantizar una transmisión segura y confiable de los datos. Los "Endpoints" representan URL o enlaces que se encargan de dirigir el flujo de información de acuerdo con las reglas definidas en la API. Gracias a estos "Endpoints", se puede tanto leer como escribir información y así lograr una comunicación efectiva entre los distintos componentes del sistema, permitiendo una transferencia de información eficiente.

En la Tabla 7 se exponen los métodos HTTP utilizados para transferir información desde el módulo de variables industriales hacia la base de datos alojada en la nube. Es importante destacar que esta comunicación se realiza de forma automática, sin intervención del usuario, estableciendo una conexión directa entre el módulo y la base de datos. Por otro lado, en la Tabla 8 se detallan los métodos empleados para la comunicación entre el módulo y el usuario, donde se encuentran los parámetros que se modificarán constantemente.

Nota: Con el propósito de facilitar la comprensión, se emplean las variables "[servidor]" para representar la dirección IP del servidor que contiene la programación de la API, y "[sensor]" para denotar el sensor específico que se desea seleccionar para cambiar sus parámetros en el módulo. De esta manera, proporcionamos una referencia clara y concisa a los elementos involucrados en la explicación del proceso.

Tabla 7 Métodos HTTP utilizado por la API para la comunicación módulo de variables industriales – base de datos

Gestión de información del modulo			
Variable	Método HTTP	URL o Enlace	Función o Tarea
PRESIÓN	GET	[servidor]/presion	Permite obtener toda la información relevante del sensor de presión (medida del sensor, estado del actuador, fecha de toma de la información, entre otros)
		[servidor]/presion/sensor	Permite obtener la medida del sensor de presión de manera individual, sin información extra
	POST	[servidor]/presion	Permite enviar la información de presión desde la IOT box (capa de niebla) a la nube. Esto funciona de manera interna para el envío de información a la nube, el usuario no interviene en este proceso.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Gestión de información del módulo			
Variable	Método HTTP	URL o Enlace	Función o Tarea
NIVEL	GET	[servidor]/nivel	Permite obtener toda la información relevante del sensor de nivel (medida del sensor, estado del actuador, fecha de toma de la información, entre otros)
		[servidor]/nivel/sensor	Permite obtener la medida del sensor de nivel de manera individual, sin información extra
	POST	[servidor]/nivel	Permite enviar la información de nivel desde la IOT box (capa de niebla) a la nube. Esto funciona de manera interna para el envío de información a la nube, el usuario no interviene en este proceso.
FLUJO	GET	[servidor]/flujo	Permite obtener toda la información relevante del sensor de flujo (medida del sensor, estado del actuador, fecha de toma de la información, entre otros)
		[servidor]/flujo/sensor	Permite obtener la medida del sensor de flujo de manera individual, sin información extra
	POST	[servidor]/flujo	Permite enviar la información de flujo desde la IOT box (capa de niebla) a la nube. Esto funciona de manera interna para el envío de información a la nube, el usuario no interviene en este proceso.
TEMPERATURA	GET	[servidor]/temperatura	Permite obtener toda la información relevante del sensor de temperatura (medida del sensor, estado del actuador, fecha de toma de la información, entre otros)
		[servidor]/temperatura/sensor	Permite obtener la medida del sensor de temperatura de manera individual, sin información extra
	POST	[servidor]/temperatura	Permite enviar la información de temperatura desde la IOT box (capa de niebla) a la nube. Esto funciona de manera interna para el envío de información a la nube, el usuario no interviene en este proceso.

Tabla 8 Métodos HTTP utilizado por la API para la comunicación con el usuario

Gestión de información por parte del usuario			
Variable	Método HTTP	URL o Enlace	Función o Tarea
PRESIÓN	GET	[servidor]/enviar/presion	Permite obtener los parámetros actuales con los que está trabajando el sensor de presión (modo de operación, P, PI, PID, conexión remota disponible, entre otros).
FLUJO		[servidor]/enviar/flujo	Permite obtener los parámetros actuales con los que está trabajando el sensor de flujo (modo de operación, P, PI, PID, conexión remota disponible, entre otros).
NIVEL		[servidor]/enviar/nivel	Permite obtener los parámetros actuales con los que está trabajando el sensor de nivel (modo de operación, P, PI, PID conexión remota disponible, entre otros).
TEMPERATURA		[servidor]/enviar/temperatura	Permite obtener los parámetros actuales con los que está trabajando el sensor de temperatura (modo de operación, P, PI, PID, conexión remota disponible, entre otros).
TODOS		[servidor]/enviar	Permite obtener los parámetros actuales de trabajo de todos los sensores en un paquete de datos (modo de operación, P, PI, PID, conexión remota disponible, entre otros).

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Gestión de información por parte del usuario			
Variable	Método HTTP	URL o Enlace	Función o Tarea
TODOS	PATCH	[servidor]/enviar/[sensor]	Este es el método más importante para la comunicación remota, ya que en este método interviene el usuario, y se podrá modificar los parámetros de funcionamiento de cualquier sensor del módulo, dependiendo del sensor que escriba al final de la url

Para utilizar las tablas anteriores, es importante entender que el enrutamiento de la información es gestionado por la URL o enlace que se utilice. Por ejemplo, si se asume que el servidor que alberga la API se encuentra en la red local, se podrá utilizar "127.0.0.1" o "localhost" como dirección IP. Si se desea obtener información sobre el sensor de presión almacenada en la base de datos, se utiliza el primer ítem de la Tabla 7.

Para realizar la petición, se utiliza el método GET apuntando a la URL mencionada. Para mayor practicidad, se utiliza el buscador de URL de cualquier navegador web, ya que por defecto realiza un método GET a la URL proporcionada. En este caso, la URL quedaría de la siguiente forma:

http://127.0.0.1:3000/presion

De esta manera, se puede acceder y obtener la información requerida sobre el sensor de presión desde el servidor local donde se encuentra alojada la API.

Por otro lado, si el usuario desea cambiar los parámetros del módulo, se utiliza el ítem 6 de la Tabla 8, que emplea el método PATCH para reescribir la información almacenada en la base de datos. La estructura de datos que contiene los parámetros de la variable seleccionada se muestra en la Tabla 10, la cual se explica a detalle más adelante.

Para efectuar la operación con el método PATCH, se requiere el uso de herramientas especiales. En este caso, se utilizó NodeRED, que permite configurar nodos para el manejo de estos métodos. Por ejemplo, para realizar la petición PATCH del sensor de flujo, la URL quedaría de la siguiente forma:

http://127.0.0.1:3000/enviar/flujo

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Al realizar la petición con el protocolo HTTP, podemos establecer un cuerpo para el mensaje, conocido como "body". Este mensaje se envía en formato JSON. A continuación, presentamos un ejemplo:

```

"flujo": {
  "enable": False,
  "modo": "PID",
  "control": {
    "sp": 10,
    "P": 10,
    "I": 5,
    "D": 5,
    "manual": 0
  }
}

```

De esta manera, se puede actualizar y enviar la información deseada para el sensor de flujo a través del método PATCH, utilizando la estructura de datos y formato JSON correspondientes.

Con el objetivo de asegurar un almacenamiento adecuado de la información en la nube, se implementa un esquema de la base de datos. Este esquema actúa como un modelo de empaquetamiento para cada una de las variables que se espera recibir como respuesta. Los detalles de este esquema de empaquetamiento se presentan en las Tablas 9 y 10, que muestran cómo se organiza la información para este proyecto.

Cuando se hace uso de la API, dependiendo del "Endpoint" utilizado, se obtiene toda o una porción específica de la información, tal como se muestra en el ejemplo anterior. Estos datos se devuelven en consonancia con el esquema de empaquetamiento establecido, tal como se detalla en las tablas mencionadas.

Nota: Las bases de datos no relacionales empaquetan la información en instancias llamadas colecciones, cabe tener en cuenta que estos nombres no pueden llevar caracteres especiales ni espacios. En las Tablas 9, 10 y 11 que se presentan a continuación, en el nombre de las columnas, se utiliza la escritura precisa con la cual se le dio nombre a la colección correspondiente a la estructura de datos que se presenta debajo de cada una de estas.

Tabla 9 Esquema de información sensores de presión y nivel

presion	nivel
<pre> { "sensor": Number, "presostato": Boolean, "controlPVH": Boolean, "valv_proporcional": Number, "time": { "fecha": String, "hora": String, }, "control": { "enable": Boolean, "modo": String, "sp": Number, "P": Number, "I": Number, "D": Number, "manual": Number, } } </pre>	<pre> { "sensor": Number, "time": { "fecha": String, "hora": String, }, "variador": { "frecuencia": Number, "valorPLC": Number, }, "control": { "enable": Boolean, "modo": String, "sp": Number, "P": Number, "I": Number, "D": Number, "manual": Number, } } </pre>

Tabla 10 Esquema de información sensores de Flujo y Temperatura

flujo	temperatura
<pre> { "sensor": Number, "time": { "fecha": String, "hora": String, }, "variador": { "frecuencia": Number, "valorPLC": Number, }, "control": { "enable": Boolean, "modo": String, "sp": Number, "P": Number, "I": Number, "D": Number, } } </pre>	<pre> { "sensorTemp": Number, "sensorNiv": Number, "nivMinimo": Boolean, "PWM": Boolean, "time": { "fecha": String, "hora": String, }, "control": { "enable": Boolean, "modo": String, "sp": Number, "P": Number, "I": Number, "D": Number, "manual": Number, } } </pre>

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

<pre> "manual": Number, } } </pre>	<pre> } } </pre>
------------------------------------	------------------

Por otro lado, en la Tabla 11 se muestra como esquema la información que representa los parámetros con los que va a interactuar directamente el usuario de la aplicación o interfaz gráfica que se genere para el control del módulo de variables industriales.

Tabla 11 Esquema de información para el envío de datos al módulo

enviar
<pre> { "presion": { "enable": Boolean, "modo": String, "control": { "sp": Number, "P": Number, "I": Number, "D": Number, "manual": Number } }, "flujo": { "enable": Boolean, "modo": String, "control": { "sp": Number, "P": Number, "I": Number, "D": Number, "manual": Number } }, "nivel": { "enable": Boolean, "modo": String, "control": { "sp": Number, "P": Number, "I": Number, "D": Number, </pre>

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

```

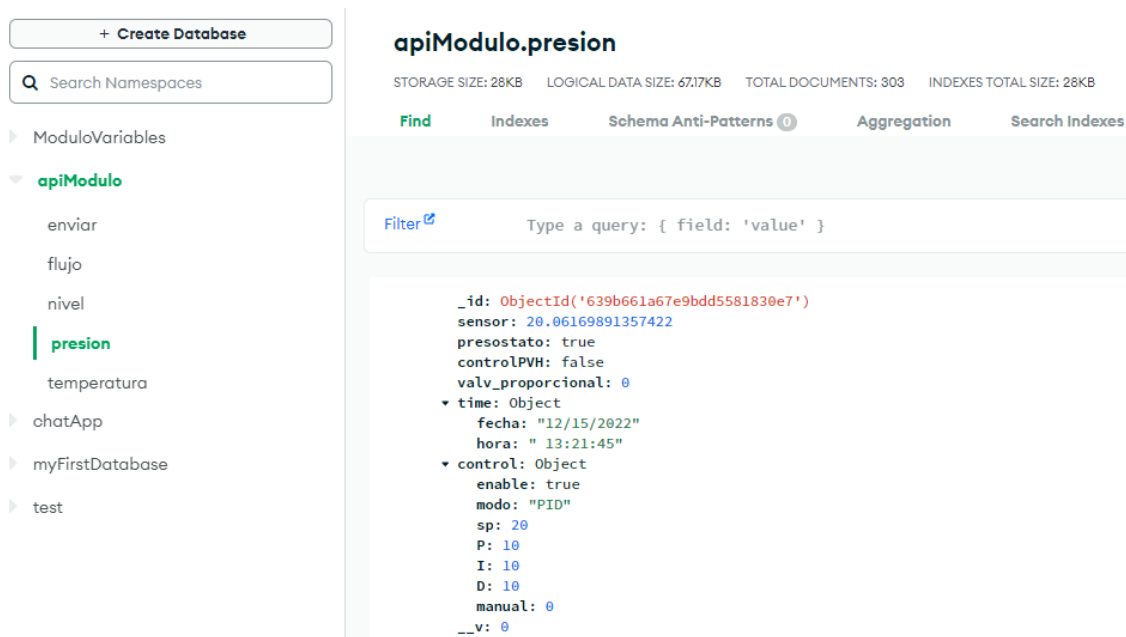
    "manual": Number
  }
},
"temperatura": {
  "enable": Boolean,
  "modo": String,
  "control": {
    "sp": Number,
    "P": Number,
    "I": Number,
    "D": Number,
    "manual": Number
  },
}
}
}

```

Para este caso de aplicación se decide utilizar como servicio de base de datos a MongoDB Atlas, la cual tiene una capa de uso gratuito, la cual permite realizar pruebas de peticiones, acceso y transferencia de la información almacenada. En la Figura 13 se presenta un ejemplo de cómo llega un mensaje de presión en la interfaz de MongoDB Atlas.

La Figura 14 muestra un ejemplo de cómo la base de datos retorna información cuando se realiza un método GET. En este caso, se utiliza el buscador de URL del navegador, ya que por defecto está configurado para realizar un método GET al enlace propuesto.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020



apiModulo.presion
 STORAGE SIZE: 28KB LOGICAL DATA SIZE: 67.17KB TOTAL DOCUMENTS: 303 INDEXES TOTAL SIZE: 28KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

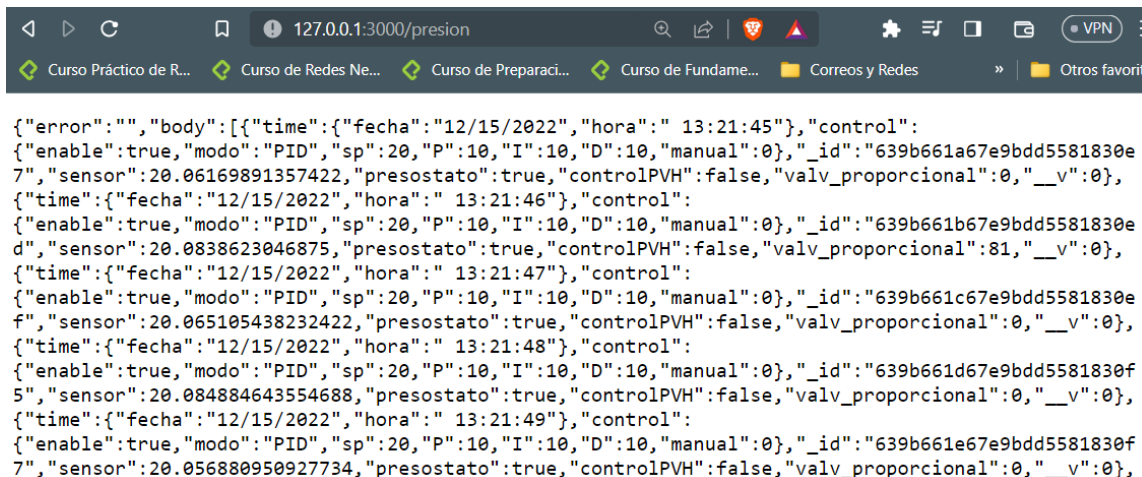
Filter Type a query: { field: 'value' }

```

_id: ObjectId('639b661a67e9bdd5581830e7')
sensor: 20.06169891357422
presostato: true
controlPVH: false
valv_proporcional: 0
time: Object
  fecha: "12/15/2022"
  hora: " 13:21:45"
control: Object
  enable: true
  modo: "PID"
  sp: 20
  P: 10
  I: 10
  D: 10
  manual: 0
  __v: 0

```

Figura 13 Información almacenada mostrada en la interfaz de MongoDB Atlas



```

{"error":"","body":[{"time":{"fecha":"12/15/2022","hora":" 13:21:45"},"control":{"enable":true,"modo":"PID","sp":20,"P":10,"I":10,"D":10,"manual":0},"_id":"639b661a67e9bdd5581830e7","sensor":20.06169891357422,"presostato":true,"controlPVH":false,"valv_proporcional":0,"__v":0}, {"time":{"fecha":"12/15/2022","hora":" 13:21:46"},"control":{"enable":true,"modo":"PID","sp":20,"P":10,"I":10,"D":10,"manual":0},"_id":"639b661b67e9bdd5581830ed","sensor":20.0838623046875,"presostato":true,"controlPVH":false,"valv_proporcional":81,"__v":0}, {"time":{"fecha":"12/15/2022","hora":" 13:21:47"},"control":{"enable":true,"modo":"PID","sp":20,"P":10,"I":10,"D":10,"manual":0},"_id":"639b661c67e9bdd5581830ef","sensor":20.065105438232422,"presostato":true,"controlPVH":false,"valv_proporcional":0,"__v":0}, {"time":{"fecha":"12/15/2022","hora":" 13:21:48"},"control":{"enable":true,"modo":"PID","sp":20,"P":10,"I":10,"D":10,"manual":0},"_id":"639b661d67e9bdd5581830f5","sensor":20.084884643554688,"presostato":true,"controlPVH":false,"valv_proporcional":0,"__v":0}, {"time":{"fecha":"12/15/2022","hora":" 13:21:49"},"control":{"enable":true,"modo":"PID","sp":20,"P":10,"I":10,"D":10,"manual":0},"_id":"639b661e67e9bdd5581830fd","sensor":20.056880950927734,"presostato":true,"controlPVH":false,"valv_proporcional":0,"__v":0},

```

Figura 14 Presentación de la información luego de un método GET común usando el buscador de URL del navegador

Por último, la Figura 15 muestra una interfaz gráfica que se desarrolló para mayor practicidad utilizando la herramienta NodeRED. Inicialmente, estaba diseñada para funcionar en la red local en la etapa de la niebla. Sin embargo, al disponer de la API, se realizó una conexión directa con el nodo HTTP que se configuró anteriormente, lo que permite que esta interfaz pueda desplegarse en cualquier servidor y enviar y recibir información con la única condición de tener una conexión activa a internet. En esta

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

interfaz, se observa la información correspondiente a la variable de flujo controlada por el módulo de variables industriales. Se presentan gráficas que permiten una fácil visualización y el usuario puede enviar valores con comodidad.



Figura 15 Presentación de la información luego de un método GET común usando el buscador de URL del navegador

El objetivo principal de este proyecto fue lograr una transferencia exitosa de información entre una aplicación industrial física y su acceso remoto desde un servidor u ordenador con conexión a internet. Además, se buscaba identificar y probar varios protocolos de comunicación para determinar cuál de ellos permitiría una conexión estable, segura, transversal y fácil de usar para este caso de estudio.

En este capítulo, se expuso el caso de aplicación que muestra cómo se establece una conexión exitosa del módulo de variables industriales de los laboratorios del ITM con una base de datos en la nube, y cómo se representa esta información en una interfaz gráfica accesible desde cualquier ordenador con conexión a internet. Se detalló el proceso de configuración y conexión, así como los dispositivos utilizados en cada capa de la arquitectura seleccionada.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

5. Conclusiones, recomendaciones y trabajo futuro

- Se llevó a cabo una investigación exhaustiva sobre los métodos y herramientas más utilizados para brindar servicios de comunicación industrial inteligente, haciendo uso de cloud y edge computing. Se introdujo la capa de niebla, que permitió una caracterización más precisa del flujo de la información en el contexto de la arquitectura propuesta.
- Se propuso una arquitectura que tiene como objetivo facilitar la construcción de un sistema que conecte un proceso industrial a la nube, marcando un primer paso hacia la implementación de IIOT (Internet de las cosas industriales).
- Se presentaron los principales protocolos de comunicación y se caracterizaron según criterios de selección que resultan fundamentales al considerar su implementación.
- Se diseñó un procedimiento metodológico que divide el sistema en tres capas principales: capa de borde, capa de niebla y capa de nube. Se expuso la taxonomía de software y hardware necesaria para su implementación y cómo se deben disponer en el sistema.
- Se propusieron modificaciones a la clásica pirámide de la automatización, con el propósito de brindar una comprensión más sencilla de la función de cada capa y el flujo de la información en el contexto de la arquitectura propuesta.
- Se llevó a cabo la aplicación de todas las herramientas propuestas en el módulo de variables industriales de los laboratorios del ITM. Se describió paso a paso el procedimiento seguido para lograr esta comunicación efectiva.
- Se definió un esquema o modelo de base de datos para el almacenamiento de la información en la nube, asegurando que este sea adecuado y organizado.
- Se desarrolló una API que permite la comunicación fluida entre la aplicación, la base de datos y una interfaz gráfica. Esta interfaz gráfica permite a los usuarios acceder al módulo de forma remota, brindando una experiencia de usuario amigable y fácil de usar.
- En conjunto, todas estas contribuciones proporcionan una base sólida para implementar sistemas de comunicación industrial inteligente que permitan conectar procesos industriales con la nube, avanzando hacia un panorama de IIOT cada vez más completo y efectivo.
- Durante el desarrollo de este trabajo, se descubrió que existen diversas herramientas y tecnologías que pueden realizar la misma tarea. Sin embargo, se evidenció la falta de una metodología o guía clara para replicar estos proyectos. Por esta razón, como recomendación en este trabajo, se propone un procedimiento metodológico y una clasificación de tecnologías con el objetivo de facilitar el montaje de estos sistemas.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

Estas propuestas buscan proporcionar una estructura coherente y orientadora que permita a otros implementar proyectos similares de manera más eficiente y efectiva.

- Aunque este trabajo presenta una interfaz gráfica desarrollada con el software NodeRED, el objetivo es crear una interfaz gráfica completa que pueda alojarse de forma permanente en un servicio en la nube en internet. De esta manera, mediante el uso de un dominio común, los usuarios podrían acceder al sistema de forma remota sin necesidad de realizar procesos adicionales. Esta mejora permitiría una mayor accesibilidad y conveniencia para los usuarios, al proporcionar un acceso sencillo y directo al sistema desde cualquier lugar y en cualquier momento.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

6. Referencias

- A. da Cruz, M. A., Rodrigues, J. J., Lorenz, P., Solic, P., Al-Muhtadi, J., & Albuquerque, V. C. (2019). A proposal for bridging application layer protocols to HTTP on IoT solutions. *Future Generation Computer Systems*, 97, 145-152.
- Babkin, N., Mylnikova, A., Chernovalova, E., Belmas, G., & Nagibina, S. (2022). Information-Infrastructure Mechanism for Managing Industrial Enterprise Self-Development in the Setting of Digitization. *XIV International Scientific Conference "INTERAGROMASH 2021"* (págs. 762-770). N/A: N/A.
- Bayılmış, C., Ebleme, M. A., Küçük, K., & Sevin, A. (2022). A survey on communication protocols and performance evaluations for. *Digital Communications and Networks*, 1094-1104.
- D, & Soni, K. N. (2022). Machine learning techniques in emerging cloud computing integrated paradigms: A survey and taxonomy. *Journal of Network and Computer Applications*, 205, N/A.
- E, J, A., & Perez, S. C. (2009). Automatas programables y sistemas de automatización. *Marcombo*, N/A.
- Eryarsoy, M., Kilic, E., Zaim, H. S., S, & Doszhanova. (2022). Assessing IoT challenges in supply chain: A comparative study before and during-COVID-19 using interval valued neutrosophic analytical hierarchy process. *Journal of Business Research*, 147, 108-123.
- Garg, H., & Dave, M. (2019). Securing iot devices and securely connecting the dots using rest api and middleware. *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoTSIU)* (págs. 1-6). N/A: N/A.
- Kim, P., Park, Y., Shakarami, S., Sankaran, S., Ferrier, R., & Beckman, N. (2022). Goal-Driven Scheduling Model in Edge Computing for Smart City Applications. *Journal of Parallel and Distributed Computing*, 167, 97-108.
- Miralles, R. (N/A de N/A de 2010). *Cloud computing y protección de datos*. Obtenido de IDP. Revista de Internet: Available: <https://bit.ly/3mmNew3>
- Nanda, U., & Pattnaik, S. K. (2016). Universal Asynchronous Receiver and Transmitter (UART). *2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS)*, (págs. 1-5). Coimbatore.
- Potdar, A. M., G, N. D., Kengond, S., & Mulla, M. M. (2020). Performance Evaluation of Docker Container and Virtual Machine. *Third International Conference on Computing and Network Communications (CoCoNet'19)*. 171, págs. 1419-1428. India: Procedia Computer Science.
- Raspberry Pi. (N/A de N/A de 2023). *Raspberry Pi Documentation - Getting Started*. Obtenido de Raspberry Pi Documentation: <https://www.raspberrypi.com/documentation/computers/getting-started.html>
- Strangio, C. E. (2012). *The RS232 Standard A Tutorial With Signal*. Obtenido de CAMI Research Inc. Acton Massachusetts:

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	24-02-2020

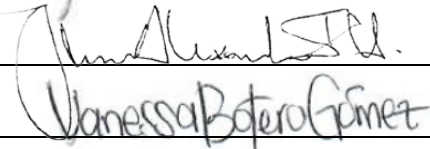
http://www.camiresearch.com/Data_Com_Basics/RS232_standard.html#anchor1154232

- Trivedi, D., Khade, A., Jain, K., & Jadhav, R. (2018). SPI to I2C Protocol Conversion Using Verilog. *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, (págs. 1-4). Pune.
- Vadi, S., Bayindir, R., Toplar, Y., & Colak, I. (2022). Induction motor control system with a Programmable Logic Controller. *ISA Transactions*, *122*, 459-471.
- Wilmshurst, T. (2006). Designing embedded systems with PIC microcontrollers: principles and applications. *Elsevier*, N/A.
- Wu, X., & Xie, L. (2019). Performance evaluation of industrial Ethernet protocols for networked control application. *Control Engineering Practice*, *84*, 208-217.
- Yuanyuan, Y., & Meng, C. (2021). The design of adaptive communication frame supporting high-speed transmission based on ModBus protocol. *10th International Conference of Information and Communication Technology (ICICT-2020)*. *183*, págs. 551-556. China: Procedia Computer Science.

FIRMA ESTUDIANTES



FIRMA ASESORES



FECHA ENTREGA: 31-07-2023