

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**DESARROLLO DE APLICACIONES DE SUPERVISIÓN
INDUSTRIAL DE CÓDIGO ABIERTO PARA EL LABORATORIO DE
REDES INDUSTRIALES**

**DISEÑO E IMPLEMENTACIÓN DE UN INTERFAZ DE OPERADOR
INDUSTRIAL WEB USANDO SOFTWARE DE CÓDIGO ABIERTO**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERÍA EN
ELECTRÓNICA Y AUTOMATIZACIÓN**

CYNTHIA MARISOL MEDINA GUTIÉRREZ

cynthia.medina@epn.edu.ec

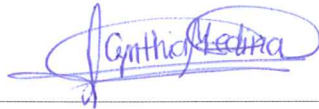
DIRECTOR: DRA. SILVANA DEL PILAR GAMBOA BENÍTEZ

silvana.gamboa@epn.edu.ec

DMQ, octubre 2022

CERTIFICACIONES

Yo, Cynthia Marisol Medina Gutiérrez declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



Srta. Cynthia Marisol Medina Gutiérrez

Certifico que el presente trabajo de integración curricular fue desarrollado por Cynthia Marisol Medina Gutiérrez, bajo mi supervisión.



Dra.- Ing. Silvana del Pilar Gamboa Benítez
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

Srta. Cynthia Marisol Medina Gutiérrez

Dra.- Ing. Silvana del Pilar Gamboa Benítez

DEDICATORIA

A toda mi familia que siempre confió en mí y que me dio todo el apoyo necesario para continuar en este largo proceso.

AGRADECIMIENTO

A mis padres, Martha Gutiérrez y Marco Medina, por su confianza, apoyo incondicional y por haber creído en mí.

A mi hermano, Kevin Medina, por ser mi motivación para seguir adelante.

A mi abuelita, Blanca Caisaguano, por su cariño y por enseñarme que para lograr algo en esta vida se necesita de un gran esfuerzo y constancia.

A mi enamorado y mejor amigo, Ronny Camacho, por haberme acompañado en este proceso de aprendizaje y por siempre haber estado conmigo.

A mis compañeros de estudio, por hacer ameno el tiempo que pasé en la universidad.

A mi tutora, Dra. Silvana Gamboa, por la paciencia y la guía que me dio para culminar con este proyecto.

ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN	VII
ABSTRACT	VIII
1. INTRODUCCIÓN.....	1
1.1 Objetivo general.....	2
1.2 Objetivos específicos	2
1.3 Alcance	2
1.4 Marco teórico	4
1.4.1 Interfaz de operador industrial.....	4
1.4.2 Componentes más usados en la industria	7
1.4.3 Editores de código.....	8
1.4.4 Lenguajes de programación más utilizados	10
1.4.5 Base de datos	11
1.4.6 Sistemas de gestión de base de datos	12
1.4.7 Arquitectura cliente – servidor de software	12
2. METODOLOGÍA.....	14
2.1 Definición de Requerimientos	14
2.1.1 Requerimientos de la interfaz de operador web	14
2.1.2 Requerimientos de funcionamiento de la interfaz de operador web	15
2.2 Arquitectura de la aplicación web.....	16
2.3 Selección de herramientas de software	16
2.4 Diseño de la aplicación web.....	18
2.4.1 Diseño de la pestaña de configuración	18
2.4.2 Diseño de las pestañas de variables analógicas y digitales.....	21
2.4.3 Diseño de las pestañas de componentes analógicas y digitales.....	22
2.4.4 Diseño de la pestaña de alarmas.....	22
2.4.5 Diseño de la pestaña de históricos	23
2.4.6 Diseño de la pestaña de inicio.....	24
2.4.7 Diseño de la base de datos Local	26

2.5	Implementación de la aplicación web.....	28
2.5.1	Implementación de la pestaña de configuración	28
2.5.2	Implementación de la pestaña de variables analógicas	30
2.5.3	Implementación de la pestaña de variables digitales	33
2.5.4	Implementación de la pestaña de componentes analógicos	34
2.5.5	Implementación de la pestaña de componentes digitales	36
2.5.6	Implementación de la pestaña de alarmas	37
2.5.7	Implementación de la pestaña de históricos	38
2.5.8	Implementación de la pestaña de mesa de trabajo estático.....	38
2.5.9	Implementación de la pestaña de mesa de trabajo dinámico.....	39
3.	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	40
3.1	Resultados	40
3.1.1	Pestaña de Configuración	40
3.1.2	Pestaña de Variables analógicas.....	42
3.1.3	Pestaña de Variables digitales	46
3.1.4	Pestaña de Componentes analógicos.....	47
3.1.5	Pestaña de Componentes digitales	48
3.1.6	Pestaña de alarmas	48
3.1.7	Pestaña de históricos.....	49
3.1.8	Pestaña de inicio.....	49
3.2	Conclusiones.....	52
3.3	Recomendaciones.....	54
4.	REFERENCIAS BIBLIOGRÁFICAS	55
5.	ANEXOS.....	57

RESUMEN

En la actualidad, a nivel industrial muchas empresas hacen uso de interfaces de operador para automatizar sus procesos, sin embargo, el alto costo de las licencias de dichas interfaces en ocasiones resulta un obstáculo para las empresas que recién están empezando.

En vista de esta problemática el presente proyecto trata de una interfaz de operador web, que se traduce como una página web, en donde se puede monitorear un proceso industrial. Esta interfaz cuenta con el control de variables analógicas como digitales, trabaja con los componentes que usualmente se usan en las industrias, cuenta con un registro de alarmas en donde éstas se van ordenando por fecha, tiene las gráficas de datos históricos, además cuenta con una vista principal en donde se encuentran todos los componentes creados junto con las variables vinculadas a los mismos, por último se ha creado una pestaña de ajustes para colocar todos los datos de la base de datos remota, por otro lado se encuentra una sección de la vinculación de variables locales con las externas.

Este proyecto trabaja con el modelo cliente servidor de software de 2 niveles, en donde el cliente es la parte visual que se le presenta al usuario, es decir, la página web, mientras que el servidor es el back-end, la lógica del programa que trabaja con toda la información de las bases de datos. Se opera con dos bases de datos, una local, en donde se va guardando la información del proceso industrial, por ejemplo: los datos de los valores de cada variable, las alarmas, los históricos, entre otros y una remota en donde se extrae toda la información de las variables que están vinculadas directamente con el proceso.

PALABRAS CLAVE: Interfaz, página web, back-end, base de datos.

ABSTRACT

At present, at an industrial level, many companies use operator interfaces to automate their processes, however, the high cost of licenses for those interfaces is sometimes an obstacle for companies that are just starting out.

In view of this problem, the present project is about a web operator interface, which is as a web page, where an industrial process can be monitored. This interface has the control of analog and digital variables, it works with the components that are usually used in industries, it has an alarm register where they are sorted by date, it has historical data graphs, it also has a main view where all the created components are found along with the variables linked to them, finally a settings tab has been created to place all the data from the remote database, on the other hand, there is a linking section of local variables with external ones.

This project works with the 2-level software client-server model, where the client is the visual part that is presented to the user, which is the web page, while the server is the Back end, the logic of the program that works with all the information in the databases. It operates with two databases, a local one, where the information of the industrial process is stored, such as the data of the values of each variable, the alarms, the historical data, among others, and a remote one where all the information is extracted from variables that are directly linked to the process.

KEYWORDS: Interface, web page, back-end, database.

1. INTRODUCCIÓN

La primera revolución industrial introdujo máquinas a producción, las cuales no utilizaban ningún HMI porque los operadores controlaban las máquinas directamente a través de pedales o palancas. La segunda revolución trabajó con la electricidad, gracias a la misma, la gente empezó a utilizar los primeros elementos HMI, tales como luces para señalización de que algo está bajo alimentación e interruptores para encender o apagar. Así es como se conoce el origen del HMI [1].

La primera interfaz hombre máquina fue incorporada por la tercera revolución industrial, en donde se conocían como salas de control, las mismas que se utilizaban para visualización y control remoto. Luego revolucionaron y pasaron de ser grandes salones a computadoras personales, la visualización remota se ejecutaba en aplicaciones de escritorio, mientras que la visualización local se ejecutaba en paneles táctiles. La visualización y el control remotos en las computadoras solo podían trabajar con redes. Las primeras generaciones de control remoto utilizaban redes RS-232, RS485 o DH++. Más tarde, se crearon nuevas posibilidades para un HMI, al cambiar las redes por Ethernet [1].

Hoy en día, se conoce a un HMI como una interfaz hombre máquina que sirve para vincular al operador con el proceso industrial. Puede ser usada para indicar datos de un proceso, supervisar entradas y salidas de las máquinas industriales, mostrar información de una máquina en específico, optimizar procesos industriales por medio de digitalización, administrar alarmas, visualizar tendencias, reducir la manipulación de la maquinaria por operadores, entre otros [2].

Un HMI juega un papel importante en la automatización industrial, sin embargo, la adquisición e implementación de un HMI en ocasiones resulta difícil, dado que al ser un software se necesita contar con una licencia, la misma que suele ser demasiado costosa como para ser adquirida por empresas de bajo poder adquisitivo como es el caso de la pequeña industria. Incluso, para las empresas que ya llevan años en la industria les resulta complicado pagar esta licencia anualmente. Dada esta problemática, se ha visto prudente desarrollar un prototipo de interfaz de operador que sea gratuita y que cuente con la mayoría de las aplicaciones que tiene una aplicación de software HMI.

En el presente proyecto se realiza una interfaz de operador web, es decir, una aplicación web que funciona como un HMI. Se desarrolla la página web por medio de código abierto, para que su acceso sea gratuito. Se implementa el control de variables analógicas y digitales, así como el manejo de componentes (máquinas industriales), se almacenan los datos en una base de datos local y se realizará una conexión con una base de datos remota

para leerlos y vincularlos con los datos de las variables locales. Además, se cuenta registros de alarmas, así como gráficas con datos históricos.

1.1 Objetivo general

Diseñar e implementar un interfaz de operador industrial web basado en software de código abierto.

1.2 Objetivos específicos

- Realizar una recopilación bibliográfica sobre las principales características de un interfaz de operador industrial, el mismo que pueda ser accedido por medio de un navegador web.
- Establecer los requerimientos y criterios necesarios para la implementación del interfaz de operador industrial web propuesto.
- Seleccionar el software de código abierto para el desarrollo del interfaz de operador web propuesto.
- Diseñar e implementar el interfaz de operador industrial web acorde a los requerimientos previamente establecidos.
- Realizar la validación del correcto funcionamiento del interfaz de operador en un navegador web.

1.3 Alcance

Se realizará una recopilación bibliográfica y de normativas acerca de los criterios de diseño que se deben tener en cuenta para la elaboración de un interfaz de operador industrial.

Se realizará una búsqueda bibliográfica y de normativas si existieran sobre los componentes que son más frecuentemente implementados en los procesos industriales.

Se realizará una recopilación bibliográfica de los editores de código más actuales para Windows y sobre los lenguajes de programación más utilizados para el desarrollo de páginas web.

Se realizará una recopilación bibliográfica sobre las diferentes bases de datos y los sistemas de gestión de base de datos más populares, en donde se almacenará la información del interfaz de operador industrial web.

Se establecerán los requerimientos del interfaz de operador industrial a desarrollar, en función de la búsqueda bibliográfica de los criterios de diseño, tomando en cuenta que su desarrollo se va a realizar como una página web.

Se realizará una identificación de los componentes digitales y analógicos más usados en la industria, destacando sus principales características, así como los tipos de datos y variables con los que trabajan.

Se seleccionará el editor de código y el lenguaje de programación que permita crear páginas web y que sea de código abierto, para el desarrollo de un interfaz de operador acorde a los lineamientos establecidos a partir de la información recolectada.

Según la recopilación bibliográfica se seleccionará el sistema de gestión de la base de datos y la base de datos que va a ser conectada con la programación interna de la página web, el back-end, que se va a desarrollar.

Se diseñarán los códigos bases de cada componente análogo y digital, definidos como los más utilizados en un proceso industrial.

En base a los requerimientos establecidos previamente para el interfaz de operador, se diseñará una página web, en donde, se podrá visualizar tanto las variables como los componentes digitales y analógicos del interfaz de operador web, así como su respectiva información, además, la página web tendrá acceso a la base de datos local.

Se diseñará una base de datos local, en donde se recopilará y almacenará la información de un proceso industrial proveniente desde la página web creada.

Se implementarán los códigos de los componentes bases, análogos y digitales, así como el módulo de variables con los cuales se irá desarrollando el interfaz de operador industrial tipo web.

Se implementará una página web previamente diseñada, que sea capaz de trabajar de la mano con la recopilación y almacenamiento de información en una base de datos local.

Se implementará una base de datos, en donde se podrá recibir y guardar información de cada componente proveniente de un proceso industrial, así como de la página web desarrollada.

Se realizará la verificación del funcionamiento del interfaz de operador desarrollado en un navegador web, utilizando como base de pruebas la emulación de un proceso industrial que este compuesto de los dispositivos básicos que comúnmente se localizan en los sistemas de automatización y control industrial prácticos.

1.4 Marco teórico

1.4.1 Interfaz de operador industrial

Un HMI, Interfaz Hombre-Máquina, es un software de automatización industrial que brinda una herramienta de visualización gráfica para el monitoreo y control de un proceso industrial. Gráficamente representa cada máquina (componente) de forma visual, tendiendo a simular en algunos casos hasta sus propias características y comportamientos [3].

Los componentes están tan desarrollados que muestran y simulan características y comportamiento de los dispositivos que representan, tales como motores, válvulas, amortiguadores y actuadores analógicos [3].

Para supervisar un proceso desde un HMI se puede dividir la información de un componente en varias categorías, tales como interfaces de entrada, entradas o salidas de hardware, valores de proceso, todos estos representan el estado del dispositivo en un momento determinado y pueden ser guardados periódicamente en una base de datos [3].

1.4.1.1 Ventajas de usar un interfaz de operador industrial

A continuación, se presentan algunos de los beneficios de implementar un sistema HMI.

- a) **Alarmas/Advertencias:** Un HMI permitirá que los operadores de la planta conozcan las alarmas y ubiquen el mal funcionamiento en la maquinaria, permitiéndoles reaccionar más rápidamente. Las alarmas pueden ser preventivas, por lo que el operador recibirá una alerta antes de que el sistema alcance un nivel de emergencia [4].
- b) **Mensajería automática:** Los operadores pueden confiar en la mensajería automática mostrada en un HMI. Por ejemplo, cuando una máquina deba ser recargada debido a un bajo nivel de combustible, el operador será alertado automáticamente [5].
- c) **Gestión general más sencilla de la planta:** Con la tecnología HMI, la empresa también puede administrar y ejecutar tareas fácilmente, mejorando su productividad. Los gráficos de alta calidad de los sistemas HMI brindan vistas realistas de las operaciones del proceso de una planta para que los operadores puedan controlar la instalación desde ubicaciones centrales [5].

- d) Seguridad:** Un sistema HMI brinda el acceso al control de las máquinas de forma remota, evitando que los operadores tengan un contacto físico con la misma, esto impide posibles accidentes dentro del sector industrial [4].
- e) Reducción de costos:** Una interfaz hombre-máquina puede reducir los costos operativos al reemplazar cientos de selectores, botones, luces indicadoras y más. Esto significa que la necesidad de cables, paneles y consolas adicionales también se reduce considerablemente [5].
- f) Registro de información:** Con la implementación de un HMI se pueden ir registrando todos los datos que se deseen en una base de datos comercial, de esta forma se contaría con un almacenamiento de datos [5].
- g) Manejo fácil:** Un HMI al presentar un diseño amigable para el usuario, permite que el operador se pueda guiar fácilmente en su maniobra, esto evita que necesiten de capacitaciones a futuro para el manejo de este [4].

1.4.1.2 Criterios de diseño para un interfaz de operador industrial

Lo primero que se debe tener en cuenta para el diseño de una interfaz es la presentación visual de la información, las imágenes que hacen referencia a los componentes del proceso, entre otros. Para ello se van definiendo pantallas/ventanas:

- **Ventana Resumen:** Muestra un contenido general de todo el proceso.
- **Ventana de Grupo:** Muestra toda la instrumentación presente en un área específica de la planta.
- **Ventana de Lazo:** Muestra un instrumento en específico, proporcionando su información técnica.
- **Ventana de alarmas:** Indica una lista de alarmas según un orden determinado. Existen dos tipos de registros, un registro histórico de alarmas el que a pesar de que la alarma se haya resuelto, el registro de la misma se puede seguir visualizando en la ventana, por otro lado, existe el registro de alarmas activo el cual una vez que se haya reconocido la existencia de la alarma y se haya resuelto, ésta ya no podrá ser visualizada en la ventana de alarmas.
- **Ventana de Eventos:** Indica las actividades que el operador ejecutó en el HMI.
- **Ventana de tendencias:** Representación visual del valor de una variable que se ejecuta en función del tiempo [6].

Los principios básicos a tener en cuenta para el diseño de un interfaz de operador son:

- Al ser una herramienta visual, la interfaz debe ser clara para cualquier usuario que la vaya a manejar. Debe contener mensajes precisos y entendibles para el usuario. Las imágenes deben ser legibles y solo se debe colocar información importante para no agobiar visualmente al lector.
- Se debe tener un sistema de retroalimentación, es decir, se debe implementar una confirmación antes de realizar alguna acción, de esta forma se evita que el usuario realice operaciones inadecuadas en el HMI.
- Las gráficas que se utilicen en el HMI deben tener coherencia con el componente y proceso en el que se trabaje [7].

Finalmente, se presentan algunas consideraciones a tener en cuenta:

- Se debe tener en cuenta el número de variables que se van a vincular con los componentes del HMI, debe ser un número que no llene visualmente la interfaz.
- Se debe contar con una gran variedad de componentes (representación gráfica de máquinas e instrumentación), de esta forma el usuario podrá tener varias opciones al elegir un componente.
 - Deben contener el valor del proceso de las variables vinculadas.
 - Su tamaño debe ser proporcional.
- Se debe agrupar información similar.
- Se debe ubicar uniformemente los componentes dentro de la interfaz.
- El fondo de la interfaz de preferencia debe ser gris, de esta forma se evita la fatiga visual del usuario.
- Se deben colocar botones en la interfaz para que el usuario pueda navegar entre las ventanas existentes.
- No se deben usar colores fosforescentes para las alarmas, se deben usar colores de advertencia como amarillo.
 - Los datos numéricos se deben colocar con un color oscuro.
- En las tendencias, las escalas se deben ajustar de forma automática.

- La visualización de los datos numéricos decimales se debe colocar con punto y los mismos deben contar con sus unidades [8].

1.4.2 Componentes más usados en la industria

Los componentes que frecuentemente son usados en los procesos industriales y considerados en la implementación de un interfaz de operador industrial son: tanques, calderas, bombas, motores, bandas de transportadoras, mezcladoras, válvulas, tolvas, hornos, entre otros. En la tabla 1.4.2 se aprecian las variables con las que usualmente trabajan cada uno de estos componentes.

Tabla 1.4.2. Componentes industriales.

Componente	Variable
Tanque	<ul style="list-style-type: none"> • Nivel • Temperatura • Presión
Caldera	<ul style="list-style-type: none"> • Flujo de vapor • Presión • Temperatura • Nivel
Bomba	<ul style="list-style-type: none"> • Caudal • Velocidad • Presión
Motor	<ul style="list-style-type: none"> • Velocidad de rotación • Presión de aceite • Temperatura • Vibración
Banda transportadora	<ul style="list-style-type: none"> • Velocidad
Mezclador	<ul style="list-style-type: none"> • Velocidad
Válvula	<ul style="list-style-type: none"> • Presión • Porcentaje de apertura o cierre • Caudal
Tolva	<ul style="list-style-type: none"> • Nivel • Peso
Horno	<ul style="list-style-type: none"> • Temperatura

	<ul style="list-style-type: none"> • Flujo de gas y aire
--	---

1.4.3 Editores de código

Actualmente los editores de código son utilizados como una herramienta de soporte para la escritura, lectura y visualización de código de programación. De esta manera es más fácil encontrar errores en la sintaxis del código, lo que hace un ambiente agradable para el usuario que se encuentre programando.

En la tabla 1.4.3 se presentan los editores de código más usados en la actualidad por los programadores, dando a conocer sus principales características, su costo, el sistema operativo con el que son compatibles y los lenguajes de programación con los que trabajan.

Tabla 1.4.3. Comparación entre editores de código.

Editor de código	Lenguajes de programación que soporta	Sistema operativo	Características	Costo
UltraEdit	JavaScript, HTML, PHP, C++, Perl, Java, entre otros.	Windows, Mac o Linux	<ul style="list-style-type: none"> • El inicio y carga de archivos lo hace con mucha rapidez. • Gran potencia y rendimiento. • Compara archivos entre local y remoto, existe una versión de origen y una de copia se seguridad. • Cuenta con motor de búsqueda, por lo que el reemplazo de código es fácil de realizar. • Administrador XML. • Listado de funciones, fácil acceso a importaciones, variables, clases, etc. • Herramientas HTML. • Ejecución de comandos por medio de un servidor remoto. • Autocompletado de código dinámico. • Selección múltiple de caracteres [9]. 	\$79.95 por año
Atom	HTML, CSS, Java, C/C++, PHP, JavaScript	Mac, Windows o Linux	<ul style="list-style-type: none"> • Es de código abierto. • Cuenta con teletype, colaboración de usuarios para escribir código en tiempo real. • Compatible con Git y GitHub. • Autocompletado inteligente. • Encuentra código y lo reemplaza fácilmente. 	Gratis

			<ul style="list-style-type: none"> • Capaz de dividir su interfaz en múltiples paneles. • Permite edición multiplataforma para Mac. • Tiene navegador de sistemas de archivos. • Cuenta con resalte de sintaxis [10]. 	
Sublime text	Python, C++, PHP, Scala, Java.	Mac, Windows o Linux	<ul style="list-style-type: none"> • Renderiza con su GPU, teniendo resolución de hasta 8K. • Cuenta con la función de TAB multi selección. • Motor de autocompletado. • Resaltado de sintaxis. • API de Python actualizada, lo que le hace compatible con otras versiones. • Maneja proyectos grandes y codificación pesada. • Cuenta con la función de cursor múltiple. • Es multiplataforma [11]. 	\$ 99
Visual Studio Code	JavaScript, Python, HTML, CSS, JSON, Java, TypeScript, C/C++, PHP, PowerShell.	Windows, Mac o Linux.	<ul style="list-style-type: none"> • IntelliSense, además del resalte de sintaxis, brinda un autocompletado del tipado de variables, importación de librerías y definición de funciones. • Cuenta con depuración de código. • Compatible con comandos GIT. • Cuenta con una librería de extensiones, para cambiar el tema, idioma y conectarse a otros servicios. • Es de código abierto. • Es un editor multiplataforma. • Tiene un terminal integrado, permite ejecutar comandos desde el propio editor. • Tiene soporte para TypeScript, JavaScript y Node.js. • Es liviano [12]. 	Gratis
Notepad++	JavaScript, PHP, C++, HTML, CSS, XML y Swift.	Windows	<ul style="list-style-type: none"> • Resalte de sintaxis. • Autocompletado de palabras y funciones. • Gran velocidad de ejecución. • Es liviano y no utiliza muchos recursos. • Cuenta con plegado de código en bloques. 	Gratis

			<ul style="list-style-type: none"> • Implementa un mapa de documento, necesario para trabajar en código extensos. • Tiene la función de sincronización para realizar un desplazamiento vertical. • Permite dividir, eliminar, unir y clasificar líneas de código [13]. 	
Brackets	HTML, CSS, JavaScript, Ruby.	Windows, Mac o Linux.	<ul style="list-style-type: none"> • Editor en línea, permite abrir ventanas de código. • Conexión en tiempo real con el navegador. • Es potente y liviano. • Cuenta con funciones de corrección de errores. • Es de código abierto. • Especialmente diseñado para desarrolladores Front-end [14]. 	Gratis

1.4.4 Lenguajes de programación más utilizados

Los 6 lenguajes más utilizados son los presentados en la tabla 1.4.4 donde también se muestra una comparación de estos lenguajes, en donde se indica el nivel que un usuario debe tener para empezar a trabajar con esta tecnología, la plataforma en la que puede operar y los usos que generalmente tiene [15].

Tabla 1.4.4. Lenguajes de programación [16], [17].

Lenguajes de programación	Nivel	Plataforma	Aplicaciones
Python	Principiante	Web y escritorio	<ul style="list-style-type: none"> • Desarrollo de Back-end. • Ciencia de datos y servicios financieros. • Desarrollo de aplicaciones. • Inteligencia artificial y Machine Learning.
Java	Intermedio	Web, móvil y escritorio	<ul style="list-style-type: none"> • Desarrollo de aplicaciones cliente servidor. • Comercio electrónico. • Finanzas.

JavaScript	Principiante a intermedio	Web, móvil y escritorio	<ul style="list-style-type: none"> • Desarrollo web (Back-end y Front-end). • Desarrollo de juegos. • Aplicaciones móviles. • Construcción de servidores web.
HTML	Principiante	Web, móvil y escritorio	<ul style="list-style-type: none"> • Desarrollo de páginas web. • Estructura de páginas web.
CSS	Principiante	Web, móvil y escritorio	<ul style="list-style-type: none"> • Desarrollo de páginas web. • Diseño de páginas web.

1.4.5 Base de datos

Una base de datos se denomina como aquella que es capaz de almacenar información de todo tipo, desde datos numéricos hasta audio. Tiene la capacidad de compartir información cuando se realicen consultas a la misma. Las ventajas que presentan son las siguientes: los datos pueden o no estar relacionados entre ellos, los datos repetidos se pueden controlar para que no exista redundancia, de esta forma también se reduce espacio de almacenamiento, se puede limitar el acceso a la información, al trabajar con copias de seguridad se tiene integridad de datos, se facilita el acceso a los datos, entre otros [18].

1.4.5.1 Tipos de bases de datos

Los dos tipos de bases que comúnmente se usan se presentan en la tabla 1.4.5.1. En donde se muestran sus principales características junto con algunos ejemplos.

Tabla 1.4.5.1. Tipos de base de datos [19].

Base de datos	Características	Ejemplos
Relacional (SQL)	<ul style="list-style-type: none"> • Excelente solución de almacenamiento. • Permite la consulta con el lenguaje SQL de toda la base de datos. • Permite relaciones entre tablas. • Puede ser accedido por muchos usuarios al mismo tiempo • Son las más utilizadas en producción. • Maneja una escritura simple. 	<ul style="list-style-type: none"> • MySQL • MariaDB • PostgreSQL • SQLite • Oracle

	<ul style="list-style-type: none"> • Trabaja con datos estructurados (formato estándar). • Los datos se organizan por medio de identificadores. 	
NoSQL	<ul style="list-style-type: none"> • Se utiliza para datos no estructurados en donde se requiera la capacidad de la base de datos para crear u obtener grandes cantidades de datos rápidamente. • Procesamiento grande de datos a alta velocidad. • No necesita esquemas de tablas definidos. • Se puede escalar de forma horizontal. • Los datos pueden tener identificadores para su clasificación, más no para la relación entre datos. • Se usa para trabajar con datos que luego no necesiten ser vinculados con otros. • No cuenta con esquema de almacenamiento, más bien utiliza documentos para guardar la información. 	<ul style="list-style-type: none"> • MongoDB • Redis • Couchbase • Casandra • Firebase

1.4.6 Sistemas de gestión de base de datos

Las bases de datos necesitan ser administradas, de allí nace el sistema de gestión de base de datos, el cuál es un software que permite al usuario interactuar con la base de datos. Por lo que cuenta con varias herramientas, tales como manejo de copias de seguridad, manipulación de datos permitiendo actualizar, crear, eliminar u obtener los mismos, construcción, definición y recuperación de datos [18].

Al trabajar con un sistema de gestión de base de datos se cuenta con el control de datos repetidos, se restringe el acceso a usuarios no deseados, se representan los datos vinculados entre sí, existe una recuperación ante fallos, se crean copias de seguridad, en sí, permite controlar el manejo de los datos [20].

1.4.7 Arquitectura de software cliente – servidor

Arquitectura de software que comprende dos partes, como su nombre lo indica, un cliente y un servidor, el primero de ellos manda solicitudes en tanto que el servidor responde a las mismas. Tiene la finalidad de proporcionar una comunicación entre procesos, ya que

implica el intercambio de datos tanto del cliente como del servidor, por lo que cada uno de ellos realiza diferentes funciones [21].

Por lo general, esta arquitectura se compone de; servidor de aplicaciones, servidor de base de datos y PC. Las dos arquitecturas principales son la de 2 y 3 niveles:

- **Arquitectura de 2 niveles:** Involucra solo la base de datos, el servidor y una PC cliente. Los usuarios ejecutan aplicaciones en su PC (Cliente), que se conectan a través de una red al servidor. El cliente ejecuta tanto la codificación como la lógica, y luego muestra la salida al usuario.
 - Se considera cuando el cliente tiene acceso a la base de datos directamente sin intervención de ningún intermediario [21].
- **Arquitectura de 3 niveles:** Involucra la PC del cliente, el servidor de la base de datos y un Servidor de aplicaciones. Se puede extender a N niveles, por lo que involucra más servidores de aplicaciones. El cliente contiene solo la interfaz que se le mostrará al usuario, por lo que no se utilizan muchos recursos ni codificación para el mismo.
 - Admite que un servidor esté a cargo de muchos clientes y proporciona más recursos en el servidor.
 - Se trata de un intermediario (servidor de aplicaciones) [21].

1.4.7.1 Comunicación cliente – servidor

Usualmente el cliente realiza una solicitud al servidor y éste responde, para ello necesitan comunicarse entre sí. Existen varios tipos de comunicación entre un cliente y servidor, a continuación, se mostrarán los más usados:

- **Protocolo HTTP:** Se implementa para obtener documentos HTML, imágenes, videos, audios, entre otros. Cuando el cliente realiza una solicitud al servidor, éste le envía una respuesta y automáticamente se cierra la comunicación entre ambos, por lo que es un proceso cíclico. En este caso, los programadores son responsables de decidir con qué frecuencia realizar solicitudes de datos al servidor. Se necesita recargar al cliente para obtener la respuesta del servidor [22].
- **AJAX:** Esta es una técnica en la que un cliente, por lo general un navegador web, manda una solicitud a un servidor web y éste responde. Antes de que se puedan enviar solicitudes, el cliente debe establecer una conexión con el servidor web. Al

recibir una solicitud, un servidor realiza algún procesamiento y devuelve una respuesta, después de lo cual se cierra la conexión. Este proceso se repite continuamente para intercambiar datos entre el cliente y el servidor. El proceso siempre lo inicia el cliente. Esto significa que incluso si el servidor tiene actualizaciones en las que el cliente podría estar interesado, las actualizaciones no se pueden enviar a menos que el cliente las solicite explícitamente. Puede trabajar en segundo plano y no necesita que la página se recargue para obtener respuestas del servidor [22].

- **Websockets:** Es una conexión bidireccional, el cliente y el servidor pueden mandar y recibir mensajes al mismo tiempo en la misma conexión, el envío de información se realiza en tiempo real. Los Websockets son adecuados para aplicaciones de chat y juegos en línea [22].

2. METODOLOGÍA

2.1 Definición de Requerimientos

2.1.1 Requerimientos de la interfaz de operador web

Para que el desarrollo de este sistema opere como una interfaz de operador web, debe cumplir con los requerimientos que se presentan en la tabla 2.1.1:

Tabla 2.1.1 Requerimientos para la interfaz de operador web

Requerimientos	Descripción
Desarrollo de un área específica dentro de la interfaz para el manejo de variables digitales y analógicas.	Pestaña en la página web para crear, actualizar y eliminar las variables analógicas y digitales.
Desarrollo de un área específica dentro de la interfaz para el manejo de componentes digitales y analógicos.	Pestaña en la página web para crear, actualizar y eliminar los componentes analógicos y digitales.
Implementación de un área dentro de la interfaz para la visualización de todos los componentes creados (HMI).	Pestaña principal de visualización, presenta dos opciones de mesa de trabajo, una estática en donde se visualizan todos los componentes con sus respectivas variables y otra dinámica en donde se puede colocar cada componente de acuerdo al gusto del usuario.

Implementación de un área dentro de la interfaz para el monitoreo de alarmas.	Permite observar las alarmas que se han producido en el proceso industrial a lo largo del tiempo.
Desarrollo de un área específica dentro de la interfaz para la configuración general de la página web, tal como la conexión a la base de dato remota, creación de grupos, visualización de mesa de trabajo y vinculación de variables locales con externas.	Permite elegir entre las dos mesas de trabajo de la ventana principal, dinámica o estática, la configuración de la base de datos remota con la que se va a conectar la página web, un apartado para la vinculación entre variables locales y externas y finalmente un área para la creación, modificación y eliminación de grupos (Números de PLCs).
Implementación de un área dentro de la interfaz para la visualización de históricos.	Permite obtener los históricos a lo largo del tiempo, mientras se encuentre activado el servidor.
Conexión a una base de datos local, así como a su servidor.	Permite almacenar la información que se encuentra en la página web, como: variables, componentes, alarmas, históricos y grupos.
Conexión a una base de datos remota.	Permite leer la información de las variables que tiene la base de datos remota, la misma que guarda los datos presentes en el proceso industrial.

2.1.2 Requerimientos de funcionamiento de la interfaz de operador web

Para que el servidor que se encuentra trabajando con el Back-end opere adecuadamente y no llegue a colapsar, se han definido los requerimientos de la tabla 2.1.2:

Tabla 2.1.2 Requerimientos para el funcionamiento de la interfaz de operador web

Requerimientos	Descripción
Capacidad máxima de 16 variables analógicas y de 16 variables digitales	Para que el servidor actúe de forma correcta, debe trabajar con un máximo de 32 variables en total.
Tiempo de guardado de históricos a cada minuto.	Los históricos se irán almacenando en la base de datos minuto a minuto.

Tiempo de actualización de datos de la base remota a la local cada 5 segundos.	La página web se actualiza cada cinco segundos.
Capacidad máxima de 5 variables por vinculación con cada componente ya sea análogo o digital.	Cada componente análogo o digital podrá tener 5 variables analógicas y 5 digitales vinculadas al mismo.

2.2 Arquitectura de la aplicación web

En el presente proyecto se trabajará con la arquitectura cliente servidor, como se puede observar en la figura 2.2, como cliente se tiene al Front-end, es decir, la página web que será consultada por el usuario, es toda la parte visual del presente proyecto, mientras que como servidor se encontrará el desarrollo del Back-end junto con la base de datos local y remota, se realizará el intercambio de información entre el Front-end, Back-end y la base de datos, por ende, la comunicación entre ellos será bidireccional.

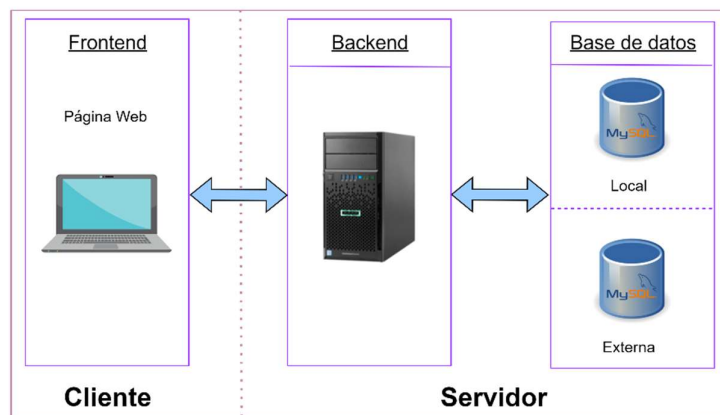


Figura 2.2. Arquitectura de la aplicación web

2.3 Selección de herramientas de software

Las tecnologías usadas en las distintas partes del proyecto se encuentran resumidas en un Diagrama que representa el stack tecnológico de toda la aplicación web en la figura 2.3. En el Front-end se han seleccionado las tecnologías más usadas por los programadores en la actualidad, además, que cuentan con soporte de la comunidad y librerías que constantemente son actualizadas por sus creadores. Por otro lado, en el Back-end se optaron tecnologías que sean compatibles con las anteriormente mencionadas y que, además, cuenten con librerías para realizar la conexión con las bases de datos. Finalmente, en la base de datos se optó por utilizar el sistema de gestión MySQL, ya que

presenta la visualización y administración de la base de datos y cuenta con el servidor en el mismo programa. Se presenta la justificación de su utilización en las tablas 2.3.1, 2.3.2, 2.3.3 y 2.3.4.

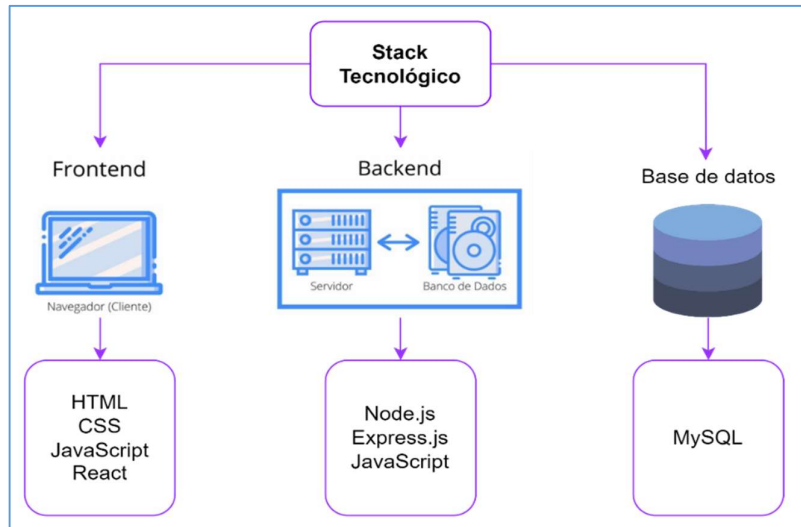


Figura 2.3. Stack de Tecnologías

Tabla 2.3.1 Lenguajes de Programación

Lenguajes de Programación	Justificación
CSS	Da el estilo de la página web.
HTML	Crea la estructura de la página web.
JavaScript	Brinda funcionalidad a la página web, es compatible con React y cuenta con una gran variedad de librerías.

Tabla 2.3.2 Frameworks y Librerías

Frameworks/Librerías	Justificación
React JS	Cuenta con varias librerías para trabajar con los históricos, las alarmas, la representación gráfica de la página web y la comunicación entre Back-end y Front-end.
Express.js	Facilita el enrutamiento y las peticiones HTTP en la parte del Back-end.
AXIOS	Permite convertir una URL (API del Back-end) en una instancia, de esta forma será más fácil trabajar el Front-end.
API REST	Permite la comunicación entre el Front-end y Back-end por medio de métodos HTTP.
Sequelize	Permite una fácil conexión con una base de datos local.

Tabla 2.3.3 Base de datos

Base de datos	Justificación
MySQL	Al ser una base de datos relacional, permite que los datos se encuentren relacionados entre sí por medio de tablas.
MySQL Workbench	Es el sistema de gestión de base de datos MySQL que permite manejar la base de datos antes seleccionada.

Tabla 2.3.4 Herramientas adicionales

Herramientas	Justificación
Visual Studio Code	Editor de código que cuenta con varias extensiones para mejorar el ambiente del programador.

2.4 Diseño de la aplicación web

A continuación, se explicará la lógica de programación que será implementada para cada una de las pestañas de la aplicación web.

2.4.1 Diseño de la pestaña de configuración

2.4.1.1 Configuraciones locales

Aquí se permitirá elegir la mesa de trabajo con la que se desea manipular la pestaña inicial (HMI). Se va a trabajar con un check, en donde si éste se encuentra en la opción Mostrar, se podrá ver la mesa de trabajo dinámica, mientras que, si está en la opción Ocultar, se visualizará la mesa de trabajo estática. Para ello se trabajará con el diagrama de flujo de la figura 2.4.1.1.



Figura 2.4.1.1. Diagrama de flujo de configuraciones locales

2.4.1.2 Base de datos remota

En esta configuración se permitirá ingresar todos los parámetros para hacer posible la conexión con la base de datos remota, la misma que se encuentra en un servidor externo (otra computadora). Para ello se colocará un formulario para que el usuario ingrese los datos de configuración de la base de datos remota, si los datos se encuentran correctos la conexión se realizará y la información será enviada al Back-end para poder realizar las peticiones HTTP de todos los datos de la base de datos remota, caso contrario la conexión no se realizará y no habrá intercambio de información. Se puede apreciar en la figura 2.4.1.2.

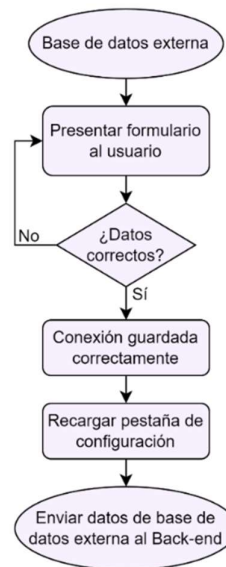


Figura 2.4.1.2. Diagrama de flujo de base de datos remota.

2.4.1.3 Relación de datos

Para que las variables locales se puedan vincular a las variables externas, debe existir un formulario para crear este vínculo, se optará por mostrar al usuario todas las variables locales que se creen y todas las variables externas que se lean. De esta forma, se podrá seleccionar variable por variable, de acuerdo con el nombre de cada una de ellas.

Se obtendrán todas las variables externas por medio de peticiones SQL puras, lenguaje de programación que sirve para intercambiar información con la base de datos, en donde se leerán 3 tablas, las variables de tipo entero, flotante y booleano. Mientras que, las variables locales se llamarán desde la tabla tag de la base de datos local. La lógica de programación se presenta en la figura 2.4.1.3.

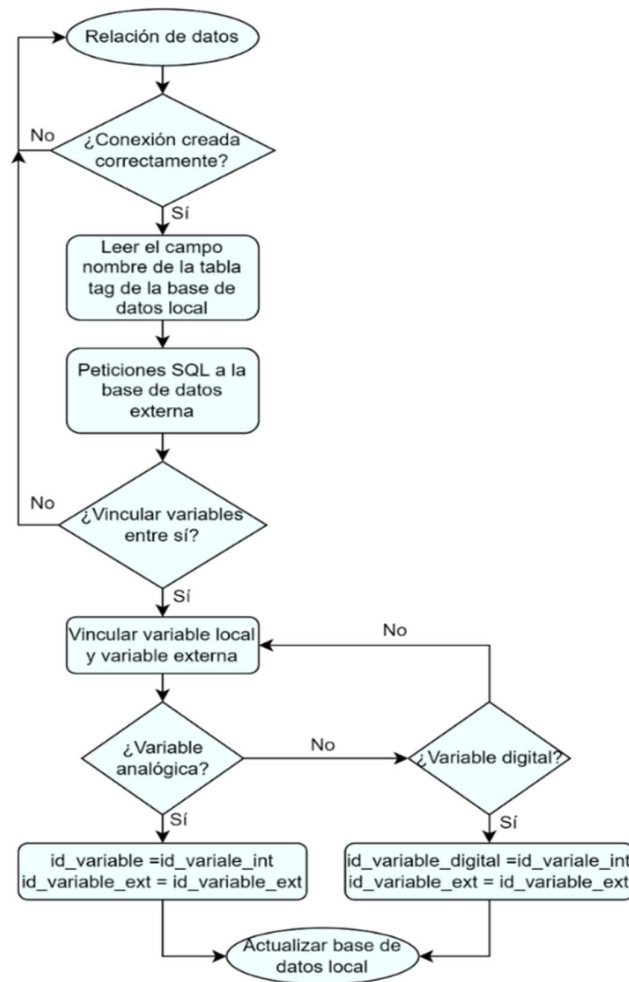


Figura 2.4.1.3. Diagrama de flujo de relación de datos

2.4.1.4 Creación de grupos

Para que las variables se encuentren ordenadas, se ha optado por clasificarlas en grupos, que harán referencia al número de PLCs que se desee incluir en el HMI web, ver figura 2.4.1.4. De esta forma, se trabajará con la creación, actualización y eliminación de grupos. En este caso, solo se trabajará con el nombre del grupo, por lo que se le pedirá al usuario que ingrese al formulario el mismo. Aquí solo se validará que este campo no se encuentre vacío, caso contrario se podrá crear o actualizar un grupo. De la misma forma, si desea eliminar alguno de ellos, si será factible. Luego de que el usuario haya elegido una acción se actualizará la tabla de grupos que se encuentra en la página web, así como la base de datos local.

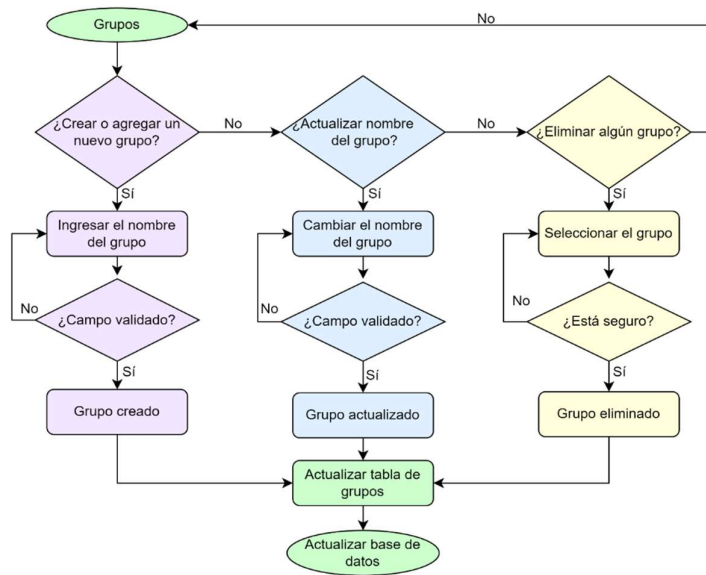


Figura 2.1.4.1. Diagrama de flujo de grupos

2.4.2 Diseño de las pestañas de variables analógicas y digitales

Tanto en la pestaña de variables analógicas como digitales se podrán crear, actualizar o eliminar las variables, una vez que el usuario llene los datos, estos se validarán, si la información es correcta se realizará la acción que el usuario eligió. Posterior a esto se actualizará la tabla de variables en la página web y la base de datos local. Esto se puede apreciar en la figura 2.4.2.

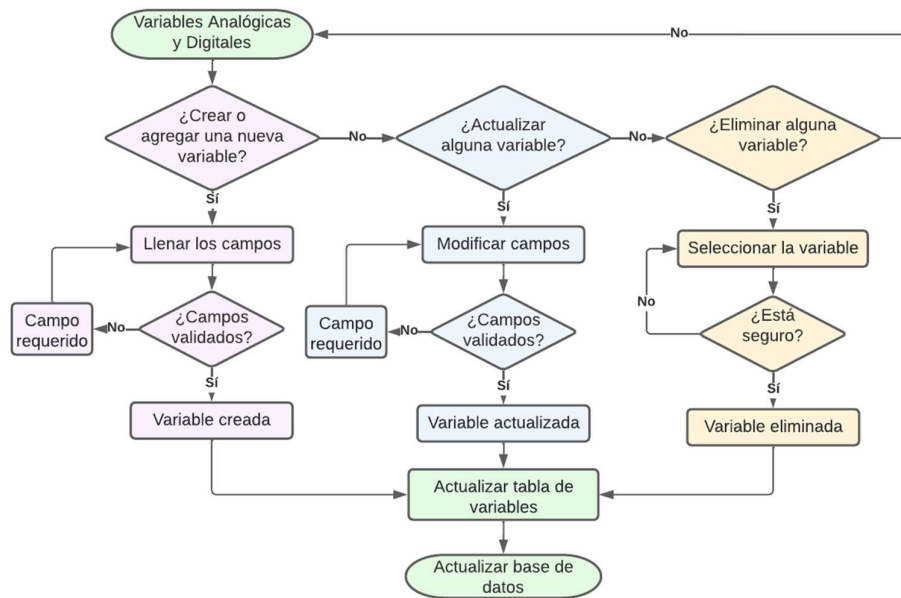


Figura 2.4.2. Diagrama de flujo para variables analógicas y digitales

Las variables analógicas, así como las digitales presentan diferentes campos a nivel de la base de datos local, esto se puede apreciar en la sección 2.4.7, en el diagrama de identidad relacional de la figura 2.4.7.1.

2.4.3 Diseño de las pestañas de componentes analógicos y digitales

En la pestaña de componentes, ya sean analógicos o digitales, se implementó la misma lógica de programación en cuanto a su creación, actualización y eliminación, esto se puede visualizar en el diagrama de flujo de la figura 2.4.3. En este caso, ambos componentes van a trabajar con los campos de: nombre, tamaño y tipo. Aquí se utilizará el campo tipo, para poder clasificar cada componente y así ubicarlos de forma conjunta en la página web principal. Al igual que la pestaña de variables, los componentes trabajan con la opción de crear, actualizar y eliminar. La diferencia es que, en esta pestaña se va a agregar la opción de vincular, esto servirá para que en cada componente se puedan vincular variables tanto analógicas como digitales.

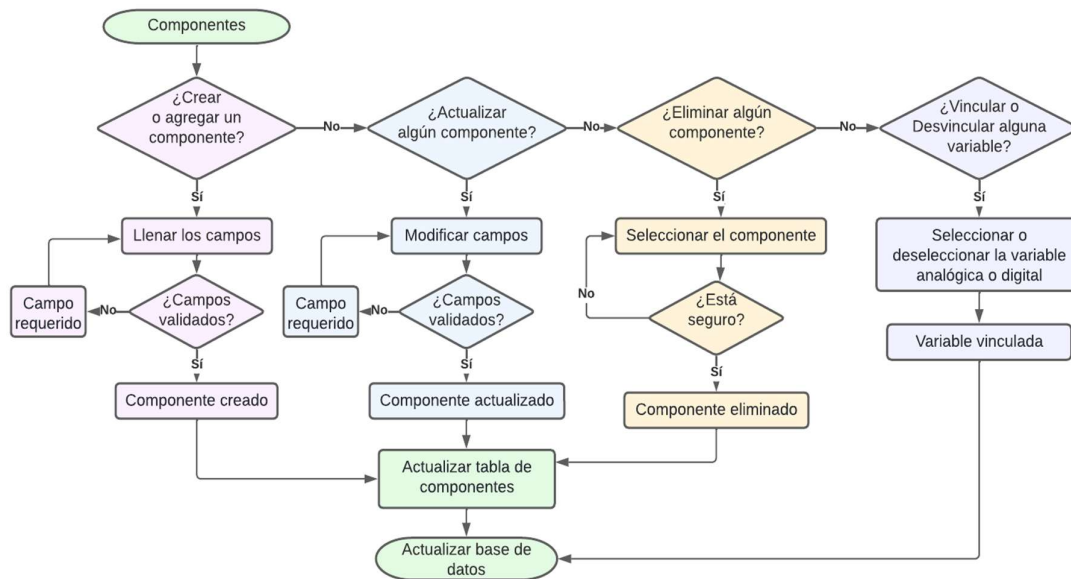


Figura 2.4.3. Diagrama de flujo de componentes analógicos y digitales

2.4.4 Diseño de la pestaña de alarmas

La tabla de alarmas trabajará con los siguientes campos: id_variable, fecha, nombre, hora, estado, tipo, valor, limite y date.

Para mostrar las alarmas se deberá hacer un cálculo según lo que se lea desde la base de datos del valor actual de la variable y los campos de los límites de las alarmas que serán ingresados por el usuario. Esto se puede observar en el diagrama de flujo de la figura 2.4.4.

Se realiza el cálculo de alarmas, dando como resultado el tipo de alarma en que se activó la alarma de la variable, por ejemplo, si es mayor que el límite alto (HA) o límite alto alto (HHA) o menor que el límite bajo (LA) o el límite bajo bajo (LLA).

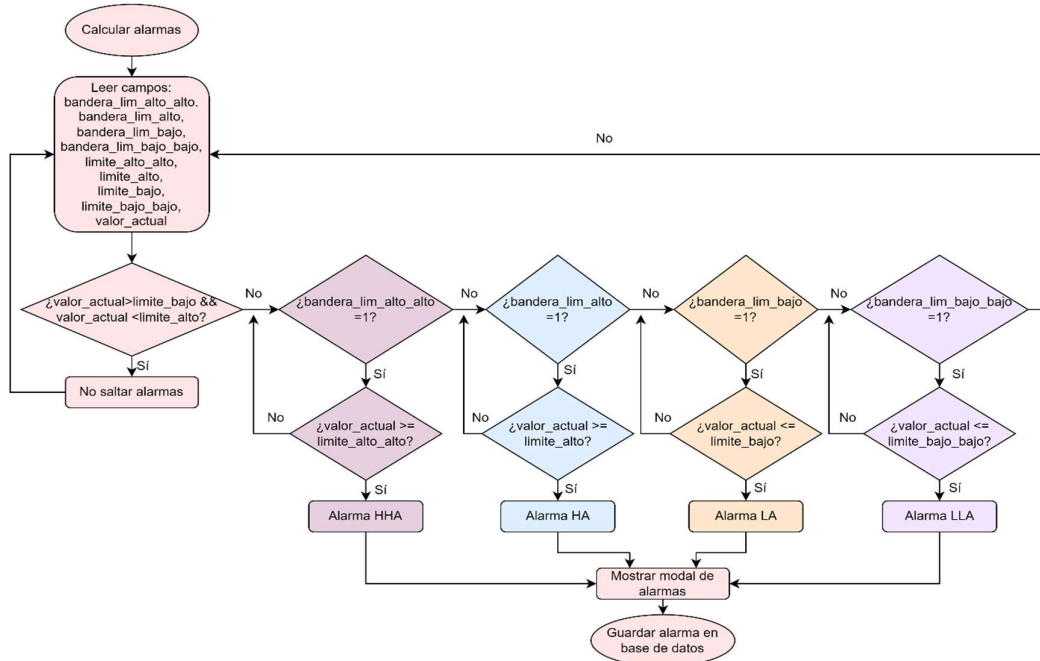


Figura 2.4.4. Diagrama de flujo del cálculo de alarmas

2.4.5 Diseño de la pestaña de históricos

La tabla de históricos contará con los campos: id_tag, valor y fecha. Para saber si el usuario desea ir registrando los históricos, se colocará la opción de elegir o no este campo en las pestañas de variables. Ya que, cada variable se puede ir clasificando por grupo, se optará por graficarlas de la misma forma, en donde se tendrá la opción de agregar o no a la variable en la gráfica general. Los históricos se irán registrando cada un minuto en la base de datos local. La lógica de históricos se encuentra en el diagrama de flujo de la figura 2.4.5.

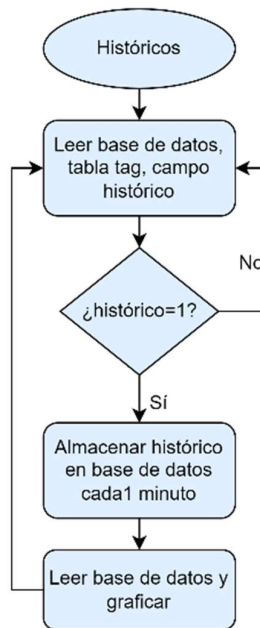


Figura 2.4.5. Diagrama de flujo de históricos

2.4.6 Diseño de la pestaña de inicio

2.4.6.1 Diseño de la pestaña con mesa de trabajo estática

Como la mesa de trabajo es estática, los componentes se leerán desde la base de datos local y si existen componentes creados, ya sean analógicos o digitales, lo primero que se hará es agruparles según el tipo de componente e irles apilando de forma horizontal en la página web. En la figura 2.4.6.1.1, se observa el orden de cada elemento que tendrán los componentes. Además, si se tiene más de un componente del mismo tipo, se creará un paginado para que el usuario pueda recorrer entre ellos.

Dado que, es la página principal, se creará un modal para que el usuario pueda configurar desde esta parte el nombre y el tamaño del componente, esto se aprecia en la figura 2.4.6.1.2.

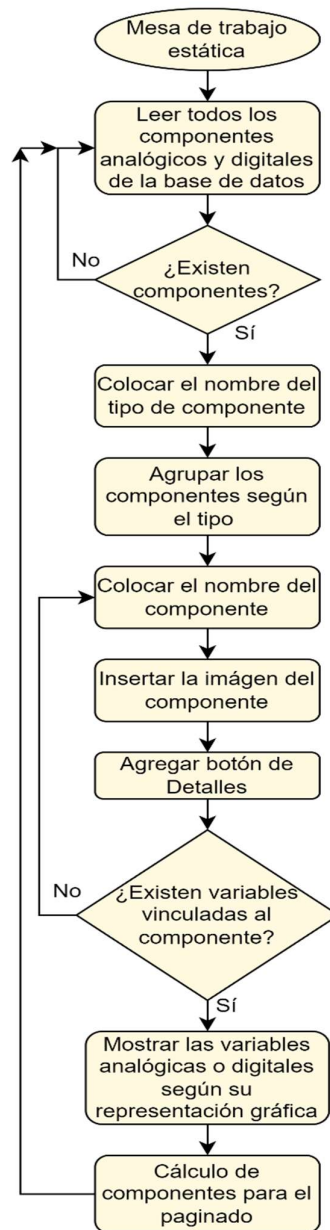


Figura 2.4.6.1.1. Diagrama de flujo de mesa de trabajo estática

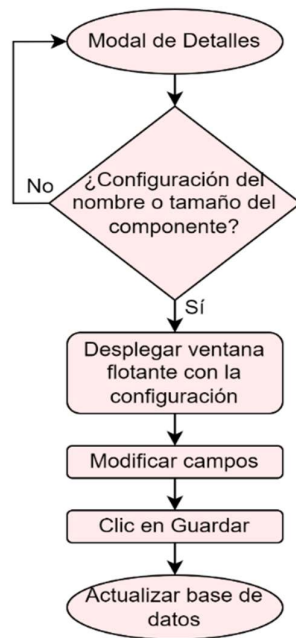


Figura 2.4.6.1.2. Diagrama de flujo del modal Detalles

2.4.6.2 Diseño de la pestaña con mesa de trabajo dinámica

La diferencia de esta mesa de trabajo es que, se utiliza la librería React Flow, lo que permitirá que cada componente principal se pueda mover por toda la interfaz, según el usuario lo requiera. Trabaja con la misma lógica que se observa en las figuras 2.4.6.1.1 y 2.4.6.1.2.

2.4.7 Diseño de la base de datos Local

Para realizar la conexión con la base de datos local se utilizará un ORM lo que facilita la comunicación entre el Back-end y la base de datos local, en este caso se utilizará Sequelize, el cual permite conectarse directamente a la base de datos, al colocar los datos de ésta, tales como el nombre de la base de datos, el usuario y contraseña, en la figura 2.4.7, se puede apreciar un ejemplo.

```

const sequelize = new Sequelize('database', 'username', 'password', {
  host: 'localhost',
  dialect: /* one of 'mysql' | 'mariadb' | 'postgres' | 'mssql' */
});
  
```

Figura 2.4.7. Conexión a base de datos con Sequelize

En la base de datos se crearán 10 tablas, las cuales se van a describir en la tabla 2.4.7.

Tabla 2.4.7. Tablas de la base de datos local

Nombre de la tabla	Descripción
variable	Almacenará los datos de las variables analógicas de tipo entero y flotantes.
variable_digital	Almacenará las variables digitales de tipo booleano.
analogo	Almacenará los componentes analógicos.
digital	Almacenará los componentes digitales.
detalles_analogo	Permitirá la vinculación de variables analógicas con componentes analógicos.
detalles_digital	Permitirá la vinculación de variables digitales con componentes analógicos.
alarma	Almacenará todas las alarmas producidas.
grupo	Almacenará todos los grupos creados.
tag	Almacenará todas las variables ya sean analógicas o digitales
historico	Almacenará todos los históricos que se produzcan cuando el servidor este encendido.

El diagrama entidad relación se puede apreciar en la figura 2.4.7.1. En dicho diagrama se puede observar que algunos campos están declarados como clave foránea (foreign key), esto se da cuando alguna tabla requiere información de otra tabla, esto hace que entre ambas se puedan compartir información de ese campo.

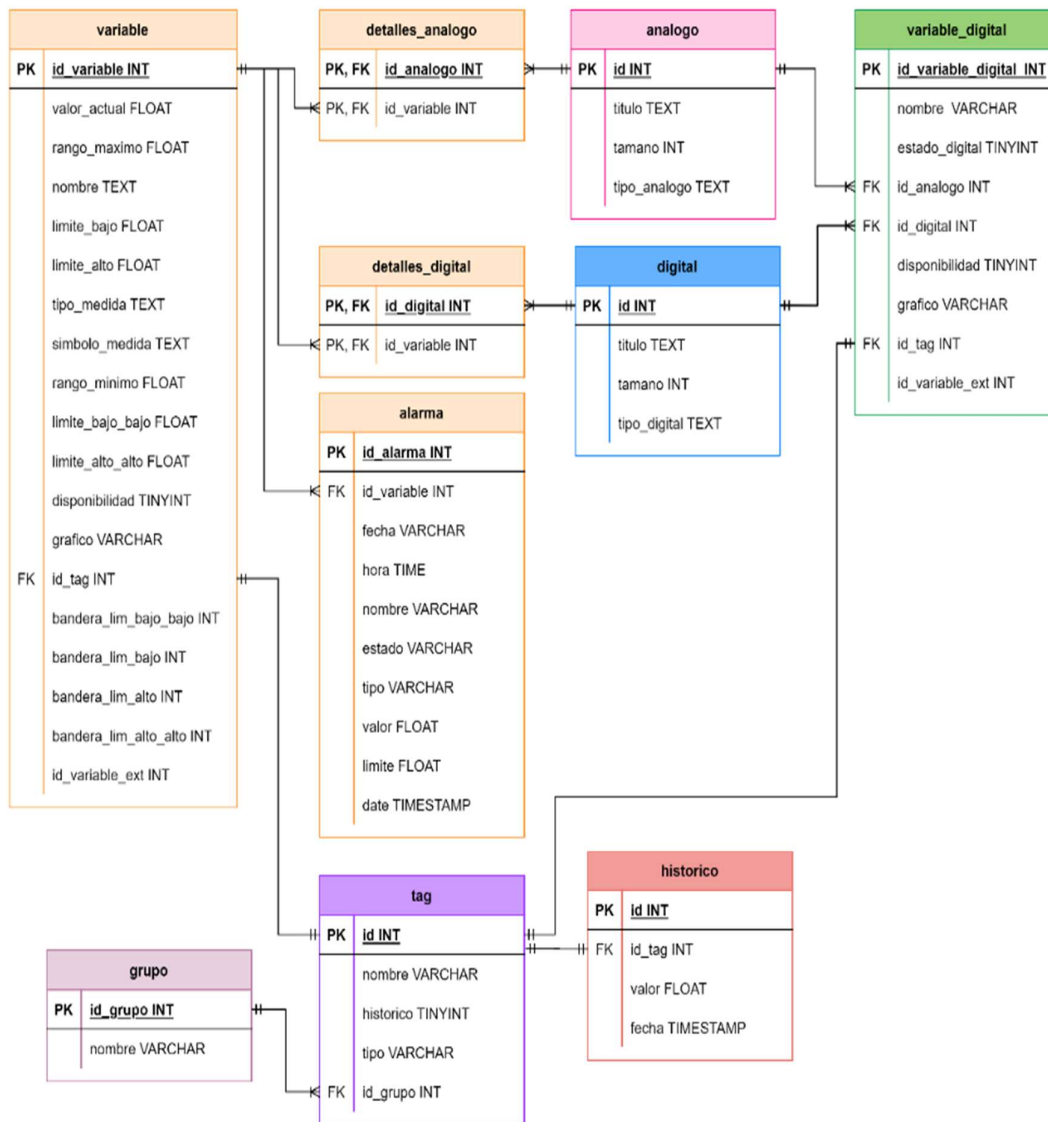


Figura 2.4.7.1. Diagrama entidad relación de la base de datos local

2.5 Implementación de la aplicación web

2.5.1 Implementación de la pestaña de configuración

2.5.1.1 Configuraciones locales

Como se observa en la figura 2.5.1.1, las configuraciones locales sirven para mostrar o no la mesa de trabajo dinámica, tiene las siguientes partes:

1. Representa la pestaña de configuraciones
2. Check para mostrar u ocultar la mesa de trabajo dinámica

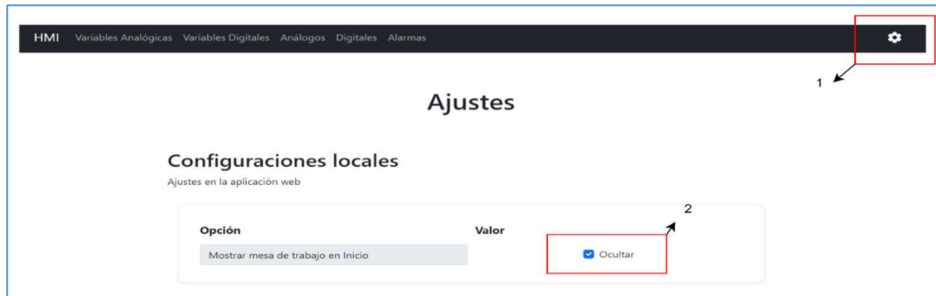


Figura 2.5.1.1. Configuraciones locales

2.5.1.2 Base de datos remota

La base de datos remota, como se puede visualizar en la figura 2.5.1.2, servirá para ingresar los datos que usualmente son configurados en MySQL Workbench de la computadora externa.

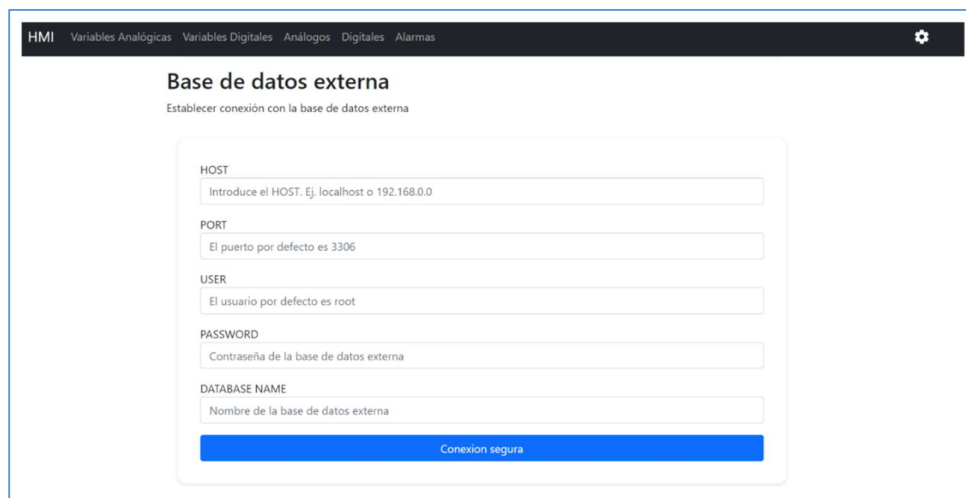


Figura 2.5.1.2. Base de datos remota.

2.5.1.3 Relación de datos

Como se observa en la figura 2.5.1.3, se tiene una tabla con dos campos, uno para cada variable, en este caso, la tabla se encuentra vacía dado que no existen variables locales creadas. En esta sección se podrá realizar el vínculo de las variables locales con las externas.

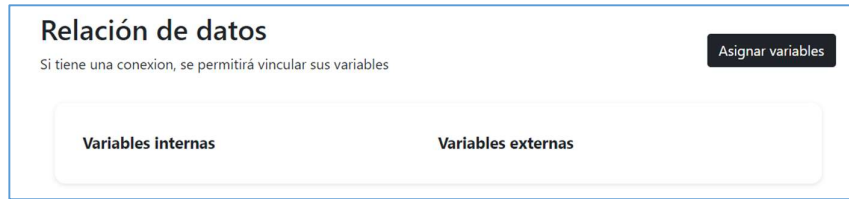


Figura 2.5.1.3. Relación de datos

El botón asignar variables desplegará un modal para realizar este vínculo, ver figura 2.5.1.3.1.

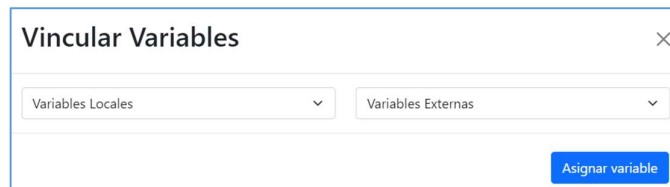


Figura 2.5.1.3.1. Modal de vincular variables

2.5.1.4 Creación de grupos

El área de la figura 2.5.1.4 se desarrolló para crear, actualizar o eliminar grupos, número de PLCs. Si se desea crear un grupo, se observará el modal de la figura 2.5.1.4.1.

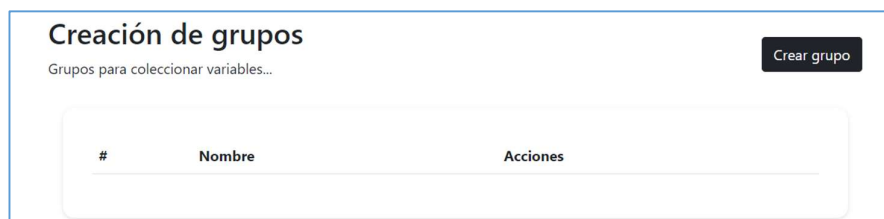


Figura 2.5.1.4. Creación de grupos

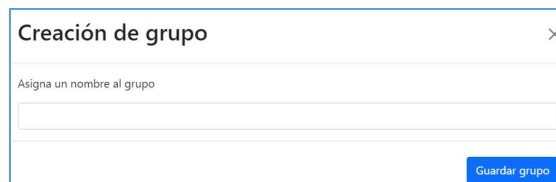


Figura 2.5.1.4.1. Modal de creación de grupo

2.5.2 Implementación de la pestaña de variables analógicas

En la presente pestaña se va a trabajar con las variables analógicas. Como se observa en la figura 2.5.2, consta de las siguientes partes:

1. Pestaña de variables analógicas
2. Botón para pausar el tiempo real, sirve para que el servidor no este leyendo los datos cada 5 segundos.
3. Tabla en donde se encontrarán las variables creadas con sus respectivos campos.
4. Botón Nuevo para crear las variables.

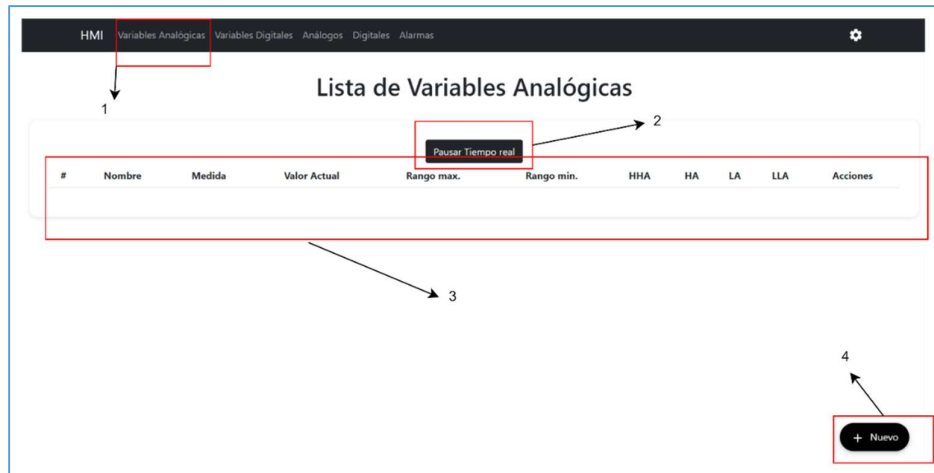


Figura 2.5.2. Pestaña de variables analógicas

El botón nuevo desplegará un modal de formulario, el que servirá para llenar todos los campos con los que va a trabajar la variable, se puede visualizar en la figura 2.5.2.1.

Figura 2.5.2.1. Modal de formulario para variables analógicas.

El presente formulario opera con los siguientes campos:

1. **Nombre:** Nombre de la variable
2. **Medida:** Medida con la que va a trabajar la variable. Este campo se puede visualizar desplegado en la figura 2.5.2.2.

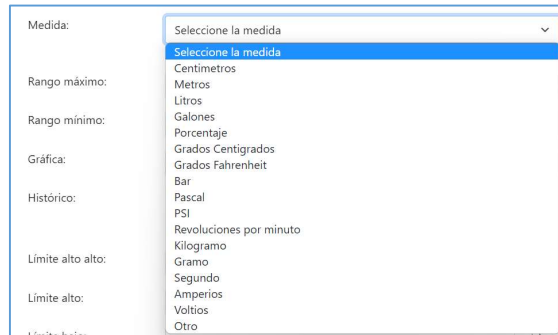


Figura 2.5.2.2. Campo: Medida.

3. **Rango máximo y mínimo:** Rangos en donde operará la variable
4. **Gráfica:** Representación visual de la variable analógica en la pestaña principal (HMI). Se presentan 4 opciones, ver figura 2.5.2.3.

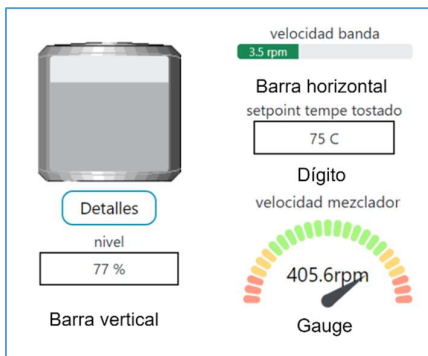


Figura 2.5.2.3. Visualización gráfica de variables analógicas.

5. **Histórico:** Registros históricos para la variable.
6. **Límites de alarmas:** Límites en donde la variable puede o no operar.
7. **Grupo:** Grupo en donde se clasificará a la variable.

Al crear una nueva variable, la tabla se va llenando con sus propios datos, como se ve en la figura 2.5.2.4. En este caso, se activarán tres botones más, un lápiz para modificar los campos, un basureo para eliminar la variable y un botón de histórico para mostrar la gráfica de la variable a través del tiempo.

Lista de Variables Analógicas										
Pausar Tiempo real										
#	Nombre	Medida	Valor Actual	Rango max.	Rango min.	HHA	HA	LA	LLA	Acciones
*** Grupo: plc1										
1	Temperatura tostado	C	0	100	0	95	90	0	0	Historico ✎ ✖

Figura 2.5.2.4. Variable analógica creada

2.5.3 Implementación de la pestaña de variables digitales

Al igual que la pestaña de variables analógicas, la de digitales, consta de las mismas partes, ver figura 2.5.3.

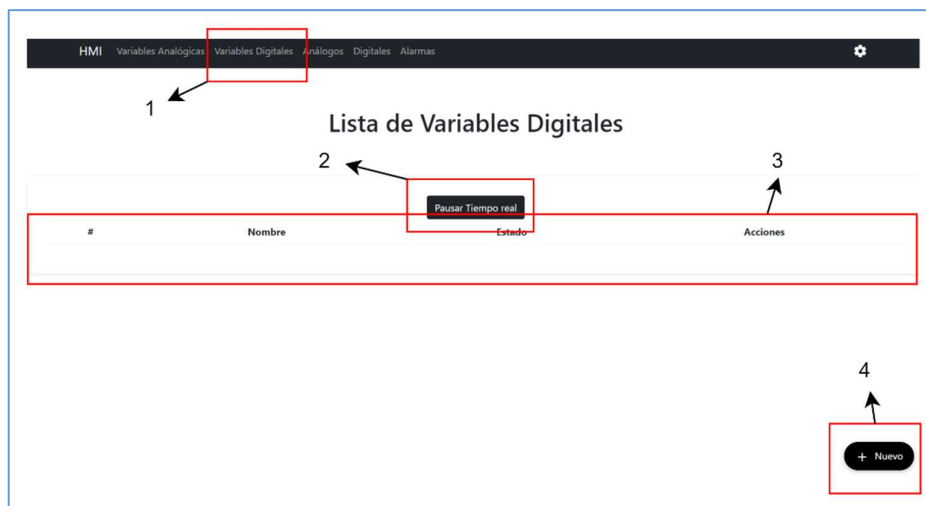


Figura 2.5.3 Pestaña de variables digitales

El botón Nuevo permitirá desplegar el modal de Formulario, correspondiente a las variables digitales, se visualiza en la figura 2.5.3.1.

Formulario ✕

Datos

Nombre:

Gráfica:

Historico: No Sí

Asignación de grupo

Grupo:

Figura 2.5.3.1. Formulario de variables digitales.

Este tipo de variable consta de los siguientes campos:

1. **Nombre:** Nombre de la variable.
2. **Gráfica:** Se muestran las opciones para la representación visual en la figura 2.5.3.2.

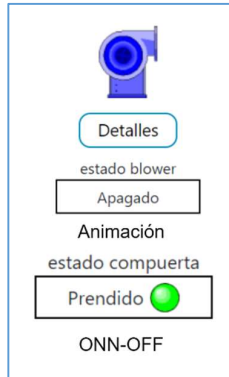


Figura 2.5.3.2. Campo: Gráfica.

3. **Histórico:** Activación de registros históricos para la variable.
4. **Grupo:** Grupo al que se le quiere clasificar a la variable.

De la misma forma que en la variable analógica, al crear una nueva variable digital se activarán 3 botones adicionales, estos servirán para la actualización, eliminación e históricos.

2.5.4 Implementación de la pestaña de componentes analógicos

La presente pestaña controla todos los componentes de este tipo, sirve para crear, actualizar y eliminar el componente que se desee.

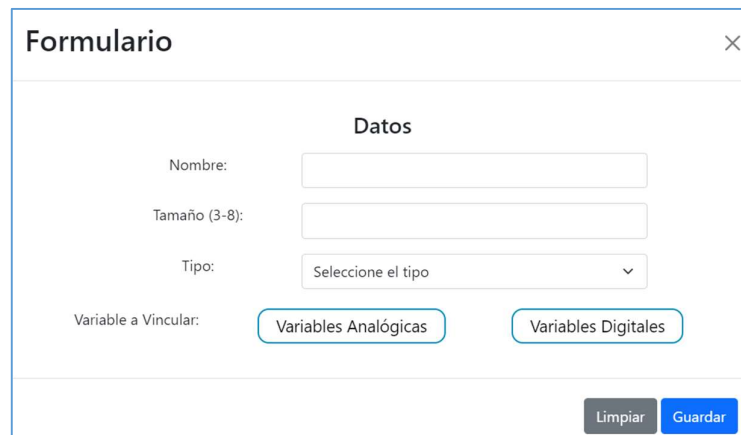


Figura 2.5.4. Pestaña de componentes analógicos.

En la figura 2.5.4 se muestra las partes por las que está compuesta, que son las siguientes:

1. Pestaña de componentes analógicos.
2. Tabla para ir registrando todos los componentes creados.
3. Botón nuevo para crear un componente desde cero.

Este último elemento, despliega un modal que contiene un formulario para la creación de un componente analógico, como se muestra en la figura 2.5.4.1.



Formulario

Datos

Nombre:

Tamaño (3-8):

Tipo:

Variable a Vincular:

Figura 2.5.4.1. Formulario para componentes analógico.

En esta pestaña se maneja el estilo del componente, por lo que, cuenta con los siguientes campos:

1. **Nombre:** Nombre que se le va a dar al componente
2. **Tamaño:** Dimensión que va a tener la imagen del componente en la pestaña principal.
3. **Tipo:** Clasificación que va a tener el componente, para luego ser agrupado por sus semejantes. En la figura 2.5.4.2, se visualizan los tipos disponibles.

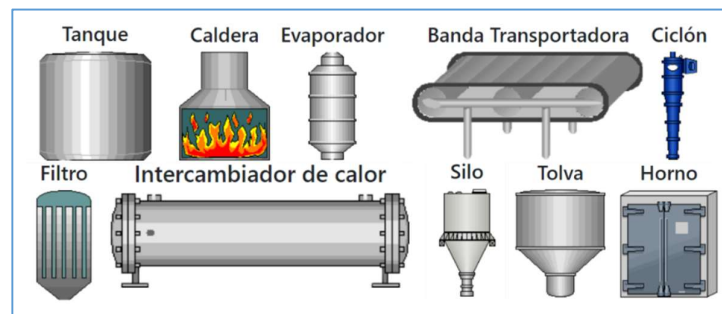


Figura 2.5.4.2. Componentes analógicos.

4. **Variable a vincular:** La lista de variables analógicas como digitales se presentan al pulsar sus correspondientes botones. El modal de variables aparecerá como se muestra en la figura 2.5.4.3.



Figura 2.5.4.3. Lista de variables analógicas

Una vez creado el componente analógico, se registrará en su tabla, ver figura 2.5.4.4, en donde al igual que en las anteriores pestañas aparecen dos botones, el primero para modificar el componente y el segundo para eliminarlo.



Figura 2.5.4.4. Componente analógico creado.

2.5.5 Implementación de la pestaña de componentes digitales

La pestaña de componentes digitales que se aprecia en la figura 2.5.5, consta de las mismas partes que la de analógicos.



Figura 2.5.5. Pestaña de componentes digitales.

El botón Nuevo desplegará un modal, que contiene un formulario para ingresar los datos del nuevo componente digital, tiene el mismo aspecto que la figura 2.5.4.1. Lo diferente de la pestaña anterior, es que el campo tipo tiene diferentes opciones, como se ve en la figura 2.5.5.1.

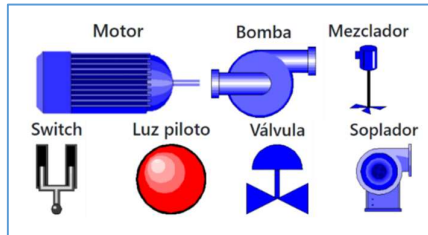


Figura 2.5.5.1. Componentes digitales

De la misma forma, consta de dos botones en donde se encuentran las variables analógicas y digitales. En cada una de ellas se encuentra un check para seleccionarla o no, de esta forma, se le indica a la base de datos que dicha variable queda o no disponible para la vinculación con los demás componentes.

2.5.6 Implementación de la pestaña de alarmas

En este proyecto se tiene una sección de alarmas, en donde se registrarán cada una de ellas según la fecha. Como se puede apreciar la figura 2.5.6, esta pestaña esta compuesta de dos partes:

1. Pestaña de alarmas
2. Tabla de almacenamiento de alarmas.

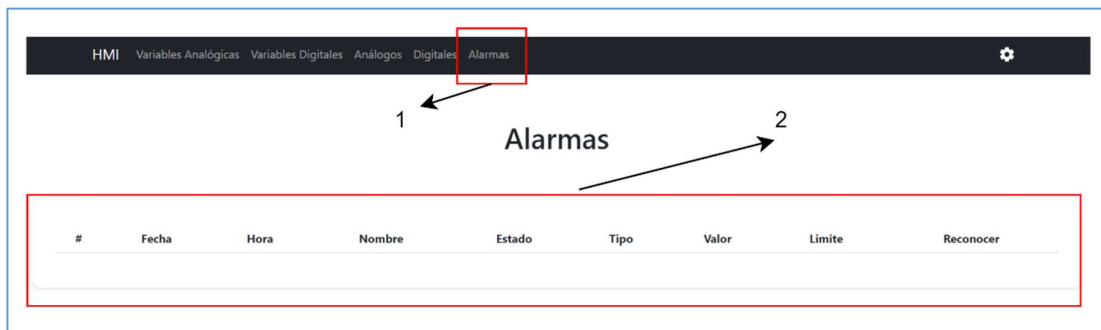


Figura 2.5.6. Pestaña de alarmas.

En la figura 2.5.6.1 se puede ver una alarma existente, en donde se muestra el límite en el que se activó, los datos más relevantes de la variable analógica y un botón de reconocer.

Alarmas									
#	Fecha	Hora	Nombre	Estado	Tipo	Valor	Limite	Reconocer	
1	domingo, 5 jun. 2022	20:43:00	Nivel	sin reconocer	HHA	98	95	<input type="button" value="Reconocer"/>	

Figura 2.5.6.1. Alarma sin reconocer.

2.5.7 Implementación de la pestaña de históricos

Esta pestaña constará de dos partes, se aprecian en la figura 2.5.7, cuando no se han creado variables, que son:

1. Fechas de elección de registros.
2. Mensaje o gráfica de históricos.



Figura 2.5.7. Pestaña de históricos

Una vez que se accede a los históricos, se mostrará una gráfica que contendrá todas las variables que se encuentren asignadas por grupo, dichas variables solo aparecerán si han sido guardadas como históricos. Los registros se irán almacenando cada minuto en la base de datos.

2.5.8 Implementación de la pestaña de mesa de trabajo estático

En la pestaña HMI, se tiene un único elemento que es el nombre de la pestaña, como se observa en la figura 2.5.8. Como esta es la mesa de trabajo estática, no se tiene que modificar nada en esta parte, dado que mientras se van creando los componentes en sus respectivas pestañas, automáticamente se van agregando en la pestaña principal, de la

misma forma, las variables que se vinculen a los componentes se irán apilando debajo de cada uno de ellos.

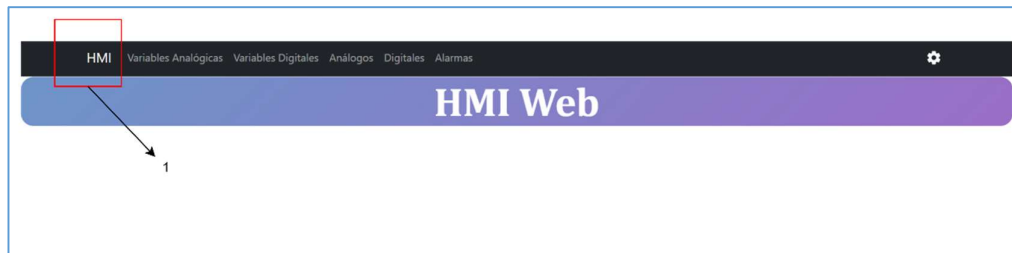


Figura 2.5.8. Pestaña principal (HMI): Mesa de trabajo dinámica.

2.5.9 Implementación de la pestaña de mesa de trabajo dinámico

En la misma pestaña principal se puede observar la mesa de trabajo dinámica, esta se podrá visualizar una vez seleccionada en la pestaña de configuraciones. En la figura 2.5.9 se muestran sus partes, que son las siguientes:

1. Pestaña principal
2. Zoom y Bloque

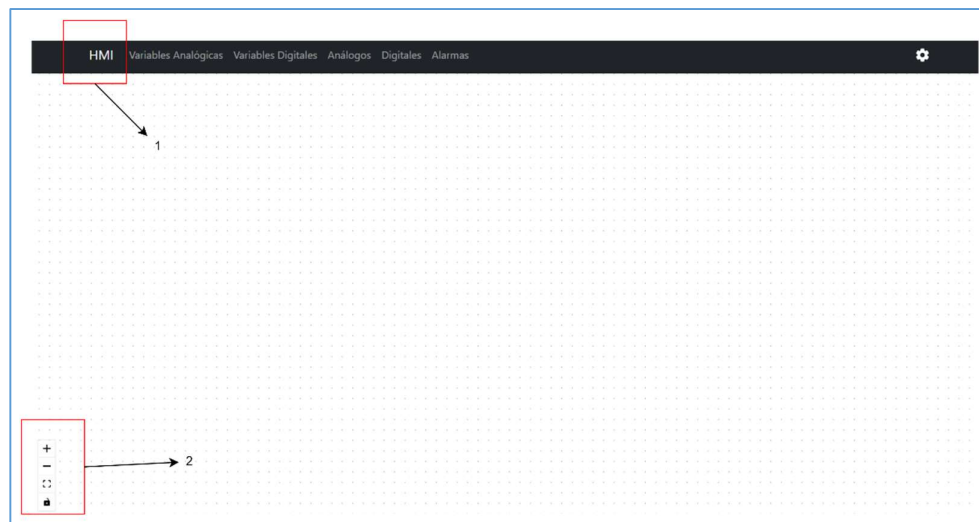


Figura 2.5.9. Pestaña principal (HMI): Mesa de trabajo estática.

Una vez que se encuentren creados todos los componentes, se podrá ingresar a esta pestaña para ubicarlos de acuerdo con la distribución que desee el usuario.

3. RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Resultados

En el presente apartado se muestran las pruebas realizadas en la aplicación web desarrollada, junto con los resultados obtenidos. Para la validación se trabajó con dos computadoras, una que representará el Front-end (aplicación), el Back-end y el servidor local, mientras que en la otra computadora se encontraba la base de datos remota la que opera con un proceso industrial real. El proceso con el que se van a realizar dichas pruebas es la elaboración de chocolate, que opera con dos PLCs, uno para el subproceso de tostado y el otro para el subproceso de mezclado.

3.1.1 Pestaña de Configuración

3.1.1.1 Base de datos remota

Al correr el programa, la primera configuración que se debe hacer en la página web es la conexión de la base de datos remota, para ello se colocó la dirección IP de la otra computadora, como se observa en la figura 3.1.1.1.

The image shows a web form titled "Base de datos externa" with the subtitle "Establecer conexión con la base de datos externa". The form contains the following fields and values:

- HOST: 192.168.10.66
- PORT: 3306
- USER: root
- PASSWORD:
- DATABASE NAME: bd_driver_eip

At the bottom of the form is a blue button labeled "Conexión segura".

Figura 3.1.1.1. Conexión a la base de datos remota.

3.1.1.2 Creación de grupos

Como en este proceso se va a trabajar con dos PLCs, este es el número de grupos que se deben crear, ver figura 3.1.1.2, de esta forma se podrán clasificar las variables según el número de grupos.



Figura 3.1.1.2. Creación de grupos.

3.1.1.3 Configuraciones locales

En la figura 3.1.1.3 se muestra cómo se puede seleccionar la mesa de trabajo dinámica, caso contrario se debería seleccionar otra vez el check, de esta forma se presentaría en la pestaña de inicio la mesa de trabajo estática.

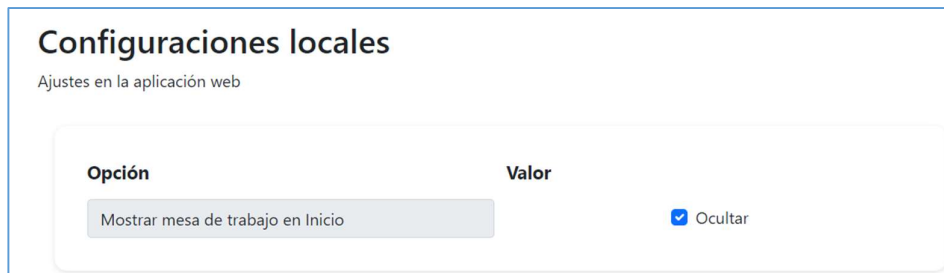


Figura 3.1.1.3. Configuraciones locales

3.1.1.4 Relación de datos

En la figura 3.1.1.4 se observa como las variables locales (internas) han sido vinculadas a las externas, de esta forma se podrá leer el valor actual de cada una de ellas que vienen directamente desde campo y están siendo almacenados en la base de datos remota.

Relación de datos

Si tiene una conexión, se permitirá vincular sus variables Asignar variables

Variables internas	Variables externas
setpoint tempe tostado (A)	sp_temp_tostado
temperatura tostado (A)	temp_tostado
velocidad horno (A)	vel_horno
velocidad molinos (A)	vel_molinos
nivel (A)	nivel
setpoint velocidad mezclador (A)	sp_velmezclador
velocidad mezclador (A)	velmezclador
nivel leche (A)	nivel_leche
nivel azucar (A)	nivel_azucar
velocidad banda (A)	vel_banda
estado horno (D)	status_horno
Estado molino 1 (D)	status_molino1
estado blower (D)	status_blower
Estado molino 2 (D)	status_molino2
Estado mezclador (D)	statusmezclador
salida azucar (D)	salida_azucar
salida leche (D)	salida_leche
Estado banda (D)	status_banda
estado compuerta (D)	status_compuerta1

Figura 3.1.1.4. Variables vinculadas.

3.1.2 Pestaña de Variables analógicas

Cada subproceso opera con diferentes variables, en esta pestaña se crearon las variables de tipo entero y flotantes que corresponden a las analógicas, como se observa en la figura 3.1.2, además se relacionaron a cada PLC (grupo), 5 variables analógicas por cada uno. Se configuró cada variable para que algunas cuenten con límites de alarmas y presenten registros históricos. Para este caso se va a demostrar que las variables temperatura tostado y nivel de leche cuentan con los mismos valores obtenidos desde el PLC físico.

Lista de Variables Analógicas

Pausar Tiempo real

#	Nombre	Medida	Valor Actual	Rango max.	Rango min.	HHA	HA	LA	LLA	Acciones
*** Grupo: PLC 1										
1	Setpoint tempe tostado	C	75	100	70	0	0	0	0	[Historico] [Editar] [Eliminar]
2	Temperatura tostado	C	75.1	100	70	100	95	75	0	[Historico] [Editar] [Eliminar]
3	Velocidad horno	rpm	30.5	50	0	0	0	0	0	[Historico] [Editar] [Eliminar]
4	Velocidad molinos	rpm	460.2	600	200	550	500	0	0	[Historico] [Editar] [Eliminar]
5	Nivel	%	75	100	0	95	90	0	0	[Historico] [Editar] [Eliminar]
*** Grupo: PLC 2										
1	Setpoint velocidad mezclador	rpm	0	500	0	0	0	0	0	[Historico] [Editar] [Eliminar]
2	Velocidad mezclador	rpm	0.1	500	0	0	0	0	0	[Historico] [Editar] [Eliminar]
3	Nivel leche	%	51	100	0	95	90	5	2	[Historico] [Editar] [Eliminar]
4	Nivel azucar	%	81	100	0	95	90	0	0	[Historico] [Editar] [Eliminar]
5	Velocidad banda	rpm	3.2	10	0	0	7	0	0	[Historico] [Editar] [Eliminar] + Nuevo

Figura 3.1.2. Lista de variables analógicas

Se comparan los resultados obtenidos en la página web, presentada en la figura 3.1.2, frente a los datos que vienen desde la base de datos remota, las variables de tipo entero se muestran en la figura 3.1.2.1, mientras que las de tipo flotante se visualizan en la figura 3.1.2.2. Como se aprecian en las imágenes, los datos han sido leídos correctamente.

	idENTERO	valor	idTagE	nombreTag
	3	0	25	sp_vel_mezclador
	4	51	27	nivel_leche
	5	81	28	nivel_azucar
	6	75	33	nivel

Figura 3.1.2.1. Tabla de datos de variables de tipo entero.

	idFLOTANTE	valor	idTagF	nombreTag
	1	75	2	sp_temp_tostado
	2	75.1	3	temp_tostado
	3	30.5	4	vel_horno
	4	460.2	22	vel_molinos
	5	0.1	26	vel_mezclador
	6	3.2	32	vel_banda

Figura 3.1.2.2. Tabla de datos de variables tipo flotante.

Por otro lado, se muestran imágenes de las variables externas, provenientes de un driver, en la figura 3.1.2.3 se presentan las variables correspondientes al PLC 1, mientras que, en la figura 3.1.2.4 se observan las variables del PLC 2.

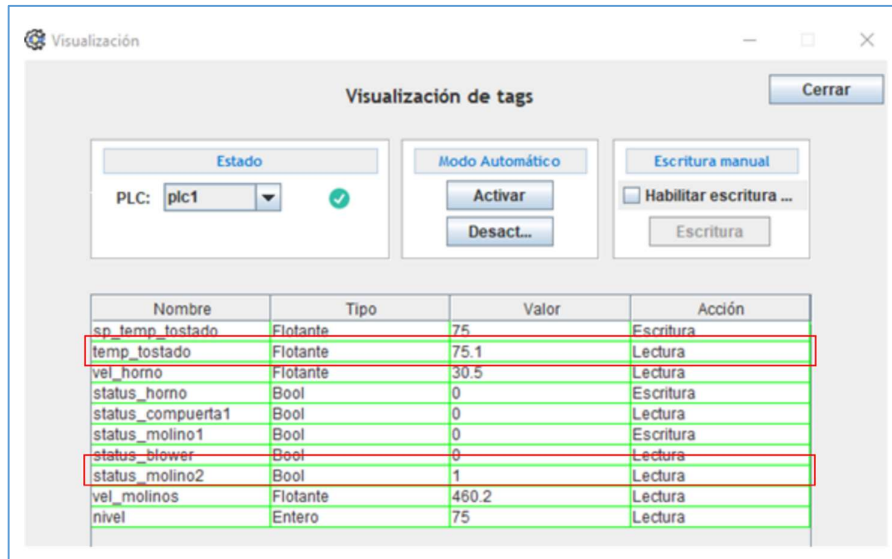


Figura 3.1.2.3. Variables del PLC 1.

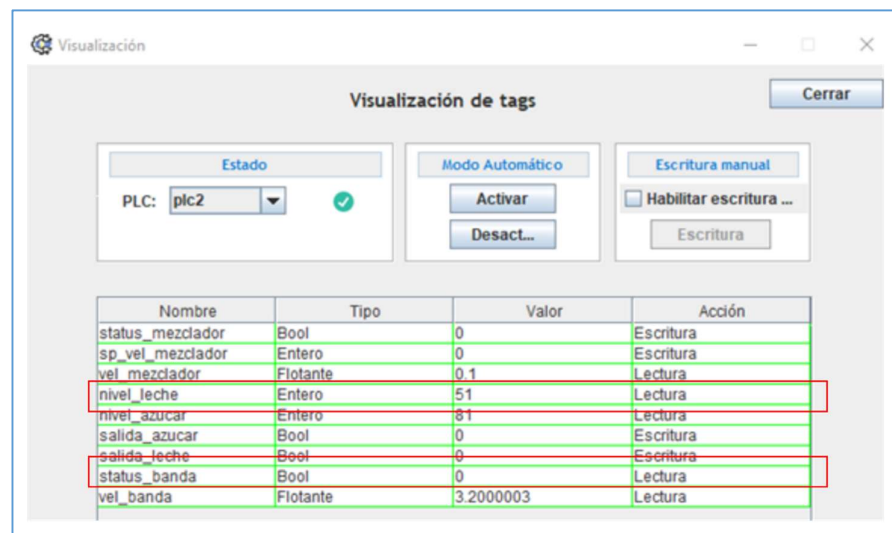


Figura 3.1.2.4. Variables del PLC 2.

Para comprobar que los resultados obtenidos son los mismos que presentan los PLCs, se presentan las figuras 3.1.2.5 y 3.1.2.6, imágenes que han sido captadas desde el programa RSLogix que comanda a los PLCs. Se aprecia que el valor de temperatura tostado se encuentra en 75.1, mientras que el de nivel de leche está en 51, los mismos valores que se presentan tanto en la base de datos externa como en la página web.

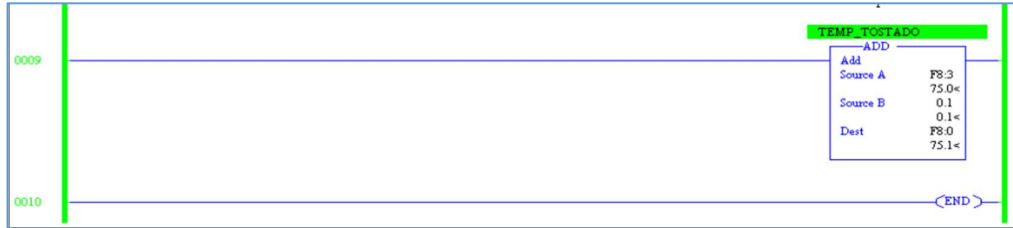


Figura 3.1.2.5. Variable temperatura tostado.

Offset	0	1	2	3	4	5	6	7	8	9
N7:0	0	51	81	0	3	46	23	85	0	0
N7:10	0	0	0	0	0	0	0	0	0	0

Symbol: N7:1
Desc: NIVEL LECHE
Radix: Decimal
Columns: 10

Figura 3.1.2.6. Variable nivel de leche.

En la figura 3.1.2.7 se muestra el formulario desplegado para la configuración de una de las variables, en este caso, para la variable temperatura tostado, que cuenta con guardado de registros históricos, tiene una representación visual de barra horizontal, esta agregada al PLC 1 y cuenta con tres límites de alarma.

Formulario

Datos

Nombre: temperatura tostado

Medida: Grados Centigrados

Rango máximo: 100

Rango mínimo: 70

Gráfica: Barra horizontal

Histórico: Sí No

Alarmas

Límite alto: 100

Límite alto: 95

Límite bajo: 75

Límite bajo bajo: 0

Asignación de grupo

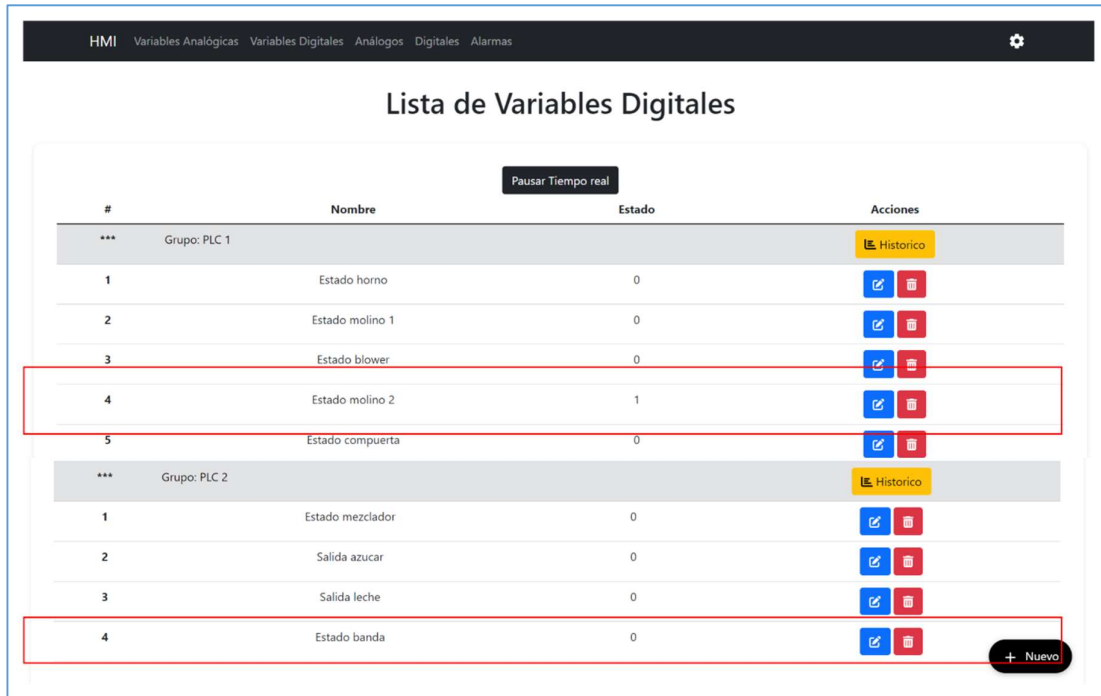
Grupo: PLC 1

Limpiar Actualizar

Figura 3.1.2.7. Formulario de la variable temperatura tostado, con histórico

3.1.3 Pestaña de Variables digitales

Este proceso también opera con variables de tipo booleano, entonces éstas han sido creadas en la pestaña de variables digitales, como se aprecia en la figura 3.1.3. De la misma forma, han sido agregadas a cada grupo (números de PLCs). Para verificar que los datos se están tomando correctamente, en este caso se va a trabajar con las variables estado molino 2 y estado banda.



#	Nombre	Estado	Acciones
*** Grupo: PLC 1			
			Historico
1	Estado horno	0	Editar Eliminar
2	Estado molino 1	0	Editar Eliminar
3	Estado blower	0	Editar Eliminar
4	Estado molino 2	1	Editar Eliminar
5	Estado compuerta	0	Editar Eliminar
*** Grupo: PLC 2			
			Historico
1	Estado mezclador	0	Editar Eliminar
2	Salida azucar	0	Editar Eliminar
3	Salida leche	0	Editar Eliminar
4	Estado banda	0	Editar Eliminar

Figura 3.1.3. Lista de variables digitales

Se comparan los resultados obtenidos en la figura 3.1.3, con las variables vinculadas desde la base de datos remota de tipo booleano que se muestra en la figura 3.1.3.1. Se aprecia que los valores de las variables han sido leídos correctamente.

	idBOOLEAN	valor	idTagB	nombreTag
	1	0	5	status_horno
	2	0	6	status_compuerta1
	3	0	9	status_molino1
	4	0	10	status_blower
	5	1	11	status_molino2
	6	0	24	status_mezclador
	7	0	29	salida_azucar
	8	0	30	salida_leche
	9	0	31	status_banda

Figura 3.1.3.1. Tabla de variables booleanas.

Para corroborar que los valores del PLC están siendo leídos correctamente, se muestra en las figuras 3.1.3.2 y 3.1.3.3 el estado de cada variable que proviene desde el programa RSLogix. Como se visualiza, el valor del estado del molino 2 es “1”, mientras que el valor de la variable de estado de banda es “0”, por lo que se comprueba el correcto funcionamiento.

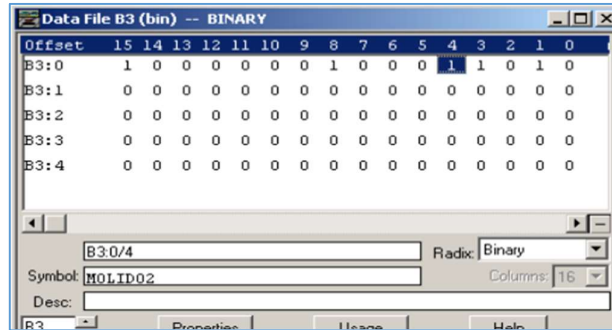


Figura 3.1.3.2. Variable estado molino 2.



Figura 3.1.3.3. Variable estado banda.

3.1.4 Pestaña de Componentes analógicos

Se han creado 6 componentes analógicos, ver figura 3.1.4, en esta pestaña se puede configurar el tamaño con el que se presentará cada componente en la pestaña inicial (HMI). Aquí se vinculó a cada componente con cada variable analógica o digital que se desee que se encuentre dentro del mismo.

#	Nombre	Tipo	Acciones
1	Horno	horno	[Icono de editar] [Icono de borrar]
2	Tanque 1	tanque	[Icono de editar] [Icono de borrar]
3	Tanque de leche	tanque	[Icono de editar] [Icono de borrar]
4	Tanque de azucar	tanque	[Icono de editar] [Icono de borrar]
5	Banda transportadora	banda	[Icono de editar] [Icono de borrar]
6	Tolva	tolva	[Icono de editar] [Icono de borrar]

Figura 3.1.4. Lista de componentes analógicos.

3.1.5 Pestaña de Componentes digitales

Para los componentes digitales se crearon 4, esto se aprecia en la figura 3.1.5, la diferencia de estos con los analógicos es que aquí se le puede cambiar la animación principal según el estado de la variable vinculada, es decir, si la variable digital se encuentra en estado '0' el color del componente se mantiene en azul, mientras que si cambia a '1' el color del componente cambia a verde.






#	Nombre	Tipo	Acciones
1	Molino 1	motor	 
2	Soplador	soplador	 
3	Molino 2	motor	 
4	Mezclador	mixer	 

Figura 3.1.5. Lista de componentes digitales.

3.1.6 Pestaña de alarmas

Como se configuraron algunas de las variables para que tengan alarmas, en el momento en el que éstas se activaron han sido registradas en la pestaña de alarmas. Como se observa en la figura 3.1.6, las alarmas se van ordenando según la fecha en la que se activan.

#	Fecha	Hora	Nombre	Estado	Tipo	Valor	Limite	Reconocer
1	viernes, 10 jun. 2022	11:40:00	temperatura tostado	sin reconocer	HA	84	80	<input type="button" value="Reconocer"/>
2	miércoles, 8 jun. 2022	20:39:00	temperatura tostado	sin reconocer	HA	84	80	<input type="button" value="Reconocer"/>
3	domingo, 5 jun. 2022	14:40:00	temperatura tostado	Reconocida	HHA	95	90	
4	lunes, 30 may. 2022	12:27:00	velocidad molinos	Reconocida	HA	501	500	
5	lunes, 30 may. 2022	12:24:00	nivel	Reconocida	HHA	99	95	
6	lunes, 30 may. 2022	12:17:00	velocidad banda	Reconocida	HA	8	7	
7	domingo, 29 may. 2022	22:20:00	nivel leche	Reconocida	LA	8	10	

Figura 3.1.6. Lista de alarmas

3.1.7 Pestaña de históricos

Si las variables se encuentran configuradas como históricos, los datos del valor actual de cada una de ellas se van almacenando en la base de datos local, cada minuto, si se dirige a la pestaña de históricos se muestran los últimos registros guardados, como se muestran en la figura 3.1.7. En este caso se ha decidido tachar la variable velocidad molinos, para mostrar cómo se elimina una gráfica si no se la desea visualizar.

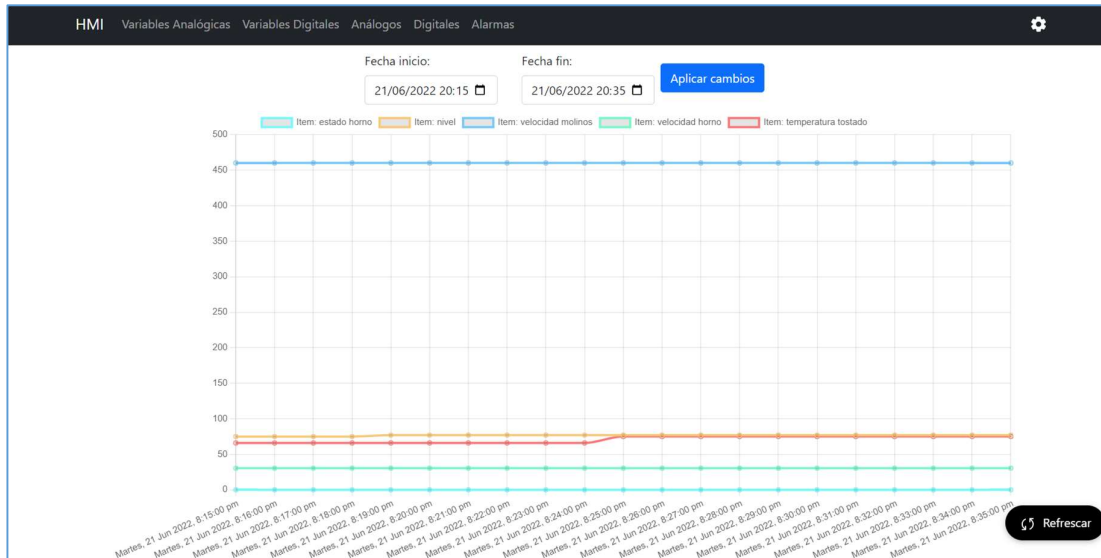


Figura 3.1.7. Pestaña de históricos.

3.1.8 Pestaña de inicio

3.1.8.1 Pestaña de mesa de trabajo estática

Al seleccionar esta mesa de trabajo, se muestra cada componente creado junto con las variables que fueron vinculadas al mismo. Se van apilando de forma vertical, uno debajo del otro, como se visualiza en la figura 3.1.8.1. Es estática, ya que los componentes no pueden ser movidos por el usuario, sin embargo, cada componente si puede cambiar de tamaño al seleccionar detalles y guardar los cambios.

HMI Web

Tanques

Tanque 1
PLC 1



Detalles

nivel

75 %

Tanque de leche
PLC 2



Detalles

nivel leche

51 %

salida leche

Apagado

3

Tanque de azucar
PLC 2



Detalles

nivel azucar

81 %

salida azucar

Apagado

Bandas

Banda transportadora

PLC 2



Detalles

velocidad banda

3.5 rpm

Estado banda

Apagado

3

Hornos

Horno

PLC 1



Detalles

velocidad horno

30.5 rpm

temperatura tostado

75

setpoint tempe tostado

75 C

estado horno

Apagado

3

Tolvas

Tolva

PLC 1



Detalles

estado compuerta

Apagado

3

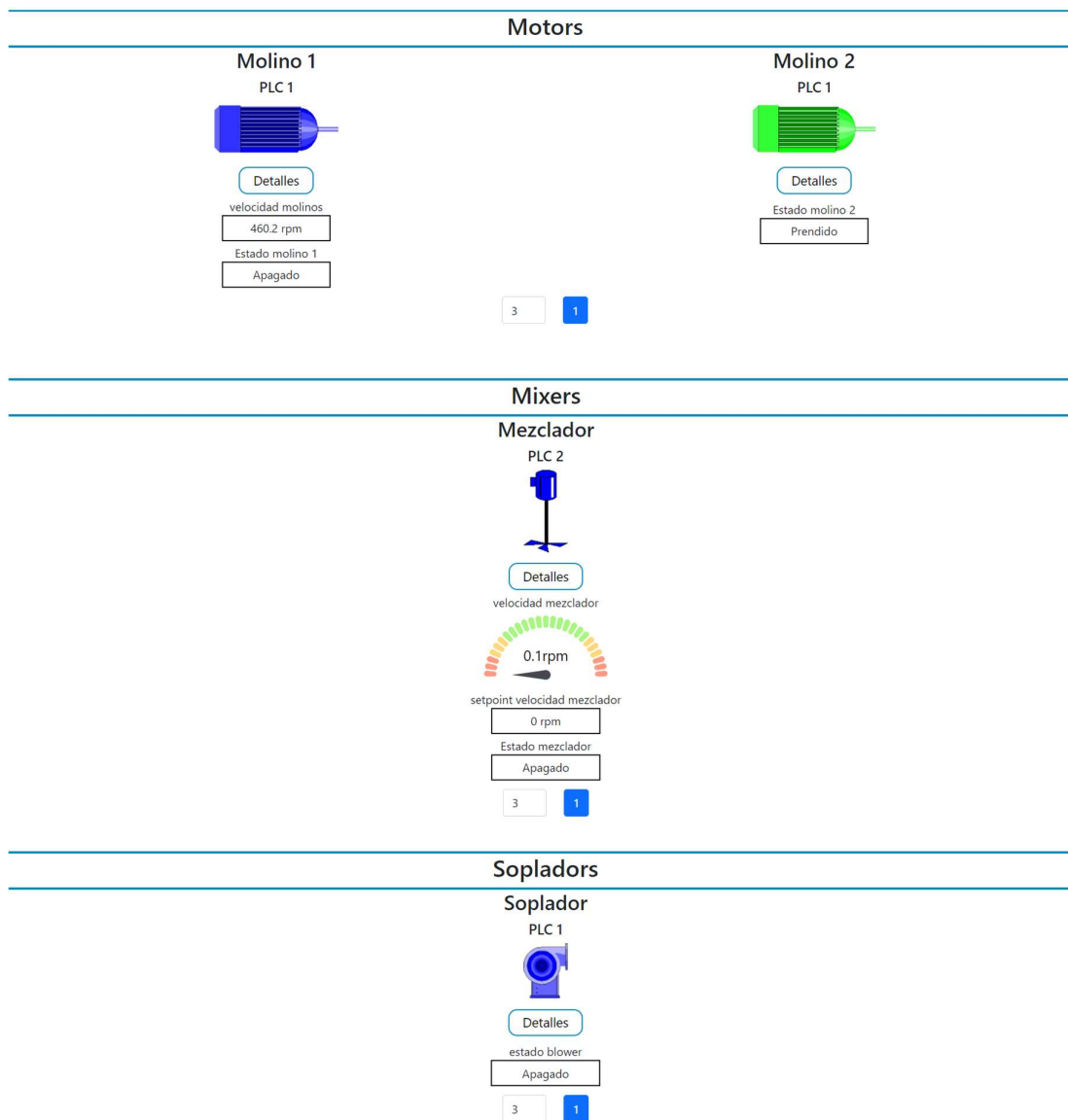


Figura 3.1.8.1. Pestaña de inicio (mesa de trabajo estática)

3.1.8.2 Pestaña de mesa de trabajo dinámico

Al seleccionar esta mesa de trabajo el usuario es quien tiene la potestad de ir ubicando cada componente según desee, esto se puede apreciar en la figura 3.1.8.2. En este caso, los componentes de la izquierda pertenecen al PLC 1, mientras que los de la derecha pertenecen al PLC 2.

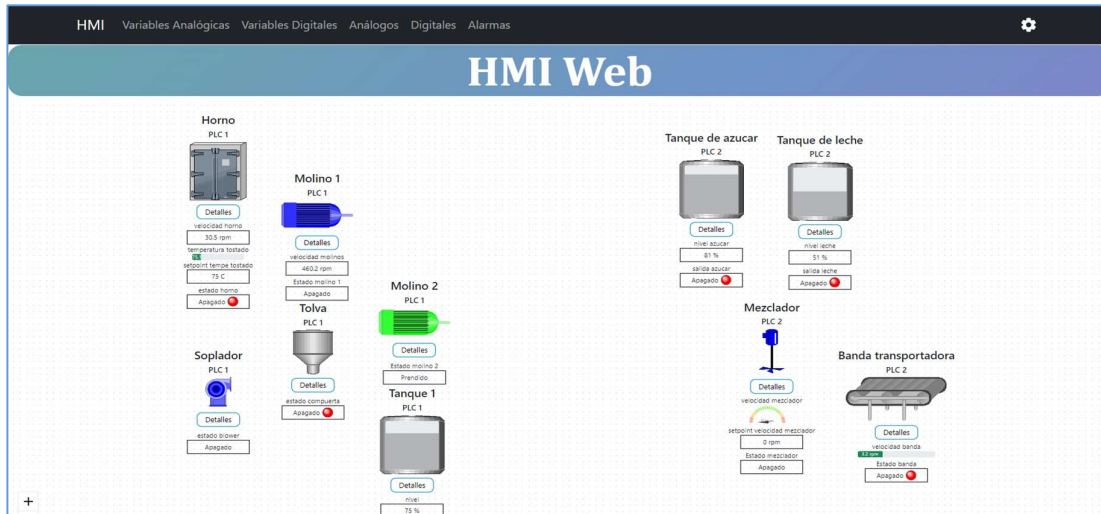


Figura 3.1.8.2. Mesa de trabajo dinámica, segunda vista.

3.2 Conclusiones

- Se diseñó e implementó una interfaz de operador industrial web basado en software de código abierto, el cual permite el monitoreo de variables analógicas como digitales provenientes de una vinculación con variables que se localizan en una base de datos remota de un proceso industrial.
- Se analizaron los criterios de diseño más importantes que debe tener un interfaz de operador funcional, teniendo en cuenta cuáles de ellos pueden ser implementados en una página web por medio de programación con JavaScript en el Front-end, ya que esta es la parte visual que se le presentará al usuario.
- Se analizaron los componentes que se utilizan con mayor frecuencia en los procesos industriales, teniendo en cuenta las variables con las que trabajan y que se deben controlar y monitorear en un HMI.
- Se seleccionó el editor de código Visual Studio Code, ya que cuenta con varias extensiones, las mismas que hacen el ambiente más agradable para programar y este editor es compatible con el sistema operativo Windows.
- Se seleccionó el lenguaje principal de programación JavaScript, ya que tiene más soporte de la comunidad, es decir, que el creador realiza constantes actualizaciones, brindando constantemente una mejora a este lenguaje, por otro lado, es uno de los más antiguos y usados por programadores. Además, este lenguaje puede ser aplicado tanto al Back-end como al Front-end, facilitando la aplicación de este en el desarrollo de una página web.

- Se seleccionó una base de datos relacional, ya que la mayoría de los datos con los que se trabajó estaban vinculados entre sí, por otro lado, tener plasmado los datos en forma de tablas, presenta una mejor organización y estructura entre ellos.
- Se seleccionó como sistema de gestión de base de datos a MySQL, ya que trabaja con base de datos relacionales, permite administrar los usuarios y contraseñas de ésta, permite el control de privilegios para usuarios, da una visualización de datos ordenados según su identificador único y permite conectarse a bases de datos que no se encuentren en la misma computadora que el servidor.
- Se diseñó e implementó una pestaña dentro de la página web capaz de controlar y monitorear las variables analógicas y digitales de un proceso industrial, en donde se tienen diferentes campos, tales como nombre, valor, unidades, límites de alarma, representación gráfica, registro de históricos y rangos en los que trabaja. Otra pestaña denominada analógicos y digitales, en donde se puede controlar la creación, actualización y eliminación de componentes dentro de un proceso industrial. Una pestaña con el nombre de alarmas, que trabaja como un registro histórico de las mismas. Una pestaña de históricos en la cual se muestran las gráficas de las variables que se han seleccionado con este tipo de registro. Una pestaña principal, en donde se pueden visualizar todos los componentes creados que son parte de un proceso industrial y finalmente una pestaña de ajustes para realizar la configuración de la conexión con la base de datos remota, que además cuenta con la vinculación de variables locales con las variables externas de dicha base de datos.
- Se diseñó e implementó una base de datos local capaz de almacenar los datos de las variables y componentes cada cinco segundos, además de guardar los datos del valor de las variables minuto a minuto. Es decir, la base de datos puede leer y escribir sobre la página web creada.
- Se diseñó e implementó la lógica de programación para extraer los datos de variables que se encuentran alojados en una base de datos remota, es decir, que se encuentra en otra computadora, esto se logró mediante el lenguaje SQL que permite realizar peticiones desde el Back-end directamente a la base de datos.
- Se comprobó el funcionamiento de la interfaz de operador web por medio de la conexión de una base de datos remota, la misma que operó en tiempo real, ésta guardaba datos de dos PLCs, uno por cada subproceso de chocolate, mezclado y tostado, respectivamente.
- Este proyecto al ser programado con software de código abierto reduce los costos de implementación, además al usar un editor de texto y un sistema de gestión de

base de datos para su uso, hace que no se necesite de gran capacidad de software a diferencia de un programa de escritorio, por lo que es asequible para cualquier usuario.

3.3 Recomendaciones

- Se recomienda avanzar en el mundo de la programación, desarrollando interfaces de operador que se ejecuten por medio de un navegador web, ya que con este tipo de proyecto se reducen significativamente los costos.
- Se recomienda programar el Back-end como el Front-end con el mismo lenguaje de programación de ser posible, como lo permite JavaScript, de esta forma es más fácil acoplar ambos mundos en uno solo.
- Se recomienda adaptar las interfaces de operador web a bases de datos remotas, de esta forma, la computadora local se encontraría en un sitio seguro sin exposición a riesgos ambientales, mientras que la computadora externa se podría encontrar en cualquier otro lado, como cerca de campo.
- Se recomienda desarrollar la conexión a una base de datos remota con la interfaz de operador web que no dependa de una red interna, podría depender de la conexión a internet, de esta forma se podrá encontrar la computadora externa en otro país, independiente de donde se encuentra la local.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] P. Papcun, "Human Machine Interface in Concept of Industry 4.0", Octubre, 2018, IEEE. DOI: 10.1109/DISA.2018.8490603
- [2] Inductive Automation, "What is HMI?", Agosto, 2018. [En línea]. Disponible en: <https://www.inductiveautomation.com/resources/article/what-is-hmi>
- [3] M. Raafay, "Human machine interface using OPC (OLE FOR PROCESS CONTROL)", Enero, 2006, IEEE. DOI: 10.1109/SCONES.2004.1564766
- [4] Atvise, "¿Qué Es Un Sistema HMI Y Para Qué Sirve?", Abril, 2021. [En línea]. Disponible en: <https://atvise.vesterbusiness.com/news/sistema-hmi/>
- [5] Sicma21, "Qué es un HMI y cómo funciona", Octubre, 2021. [En línea]. Disponible en: https://www.sicma21.com/que-es-un-hmi-y-como-funciona/#Ventajas_de_un_sistema_HMI
- [6] M. Larrea, "Diseño de Interfaces Industriales", Dpto. de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Argentina.
- [7] J. Rodríguez, "Buenas prácticas para diseño de HMI de alto rendimiento", Departamento de ingeniería electrónica, Universidad tecnológica de bolívar, Colombia, Febrero, 2012.
- [8] P. Rodríguez, "Diseño de Interfaces Hombre - Máquina (HMI)", Instituto de Electricidad y Electrónica, Universidad Austral de Chile, Chile.
- [9] UltraEdit, "UltraEdit (Windows) feature highlights". [En línea]. Disponible en: <https://www.ultraedit.com/products/ultraedit/ultraedit-feature-map.html>
- [10] Atom. [En línea]. Disponible en: <https://atom.io>
- [11] Sublime Text. [En línea]. Disponible en: <https://www.sublimetext.com>
- [12] Visual Studio Code. [En línea]. Disponible en: <https://code.visualstudio.com>
- [13] Notepad++. [En línea]. Disponible en: <https://notepad-plus-plus.org>
- [14] Brackets. [En línea]. Disponible en: <https://brackets.io>
- [15] Stack Overflow, "2020 Developer Survey". [En línea]. Disponible en: <https://insights.stackoverflow.com/survey/2020#technology>
- [16] Berkeley, "11 Most In-Demand Programming Languages in 2022". [En línea]. Disponible en: <https://bootcamp.berkeley.edu/blog/most-in-demand-programming->

languages/#:~:text=According%20to%20Stack%20Overflow%27s%202020,(PDF%2C%202.4%20MB)

[17] S. Veeraraghavan, "Best Programming Languages to Learn in 2022", Simplilearn, Junio, 2022. [En línea]. Disponible en: <https://www.simplilearn.com/best-programming-languages-start-learning-today-article>

[18] J. Sánchez, "Diseño Conceptual de Bases de Datos", guía de aprendizaje, 2004.

[19] Geeks for Geeks, "Types of Databases", Septiembre, 2021. [En línea]. Disponible en: <https://www.geeksforgeeks.org/types-of-databases/>

[20] T. Garzón, "Sistemas gestores de bases de datos", informática, Mayo, 2010.

[21] S. Haroon- Sulyman, "Client Server Model", IOSR Journal of Computer Engineering, Enero, 2014. DOI: 10.9790/0661-16195771

[22] A. Bawa, "Types of Client Server Communication". [En línea]. Disponible en: <https://iq.opengenius.org/types-of-client-server-communication/>

5. ANEXOS

ANEXO A: Manual de Instalación

1. Instalación de MySQL

- Acceder a la página oficial de MySQL Downloads (<https://www.mysql.com/downloads/>), en la parte inferior de esta pestaña se encontrará [MySQL Community \(GPL\) Downloads](#) », elegir la opción de MySQL Community Server, dar clic en MySQL Installer for Windows, lo que le llevará a la siguiente página:

<https://dev.mysql.com/downloads/windows/installer/8.0.html>

- Seleccionar la segunda opción de descarga, como se muestra en la figura 1.

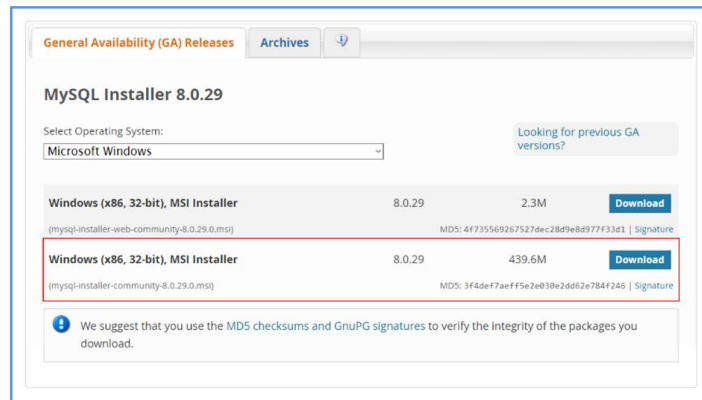


Figura 1. Segunda opción de descarga.

- Si no se cuenta con una cuenta en esta página, dar clic en: No gracias, solo empezar mi descarga (No thanks, just Start my download).

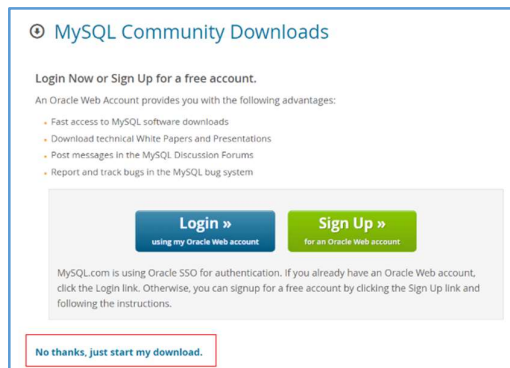


Figura 1.1. Empezar descarga.

- Una vez que se haya descargado el programa, dar doble clic sobre él para empezar la instalación.



Figura 1.2. Programa descargado.

- Esperar que se recargue la configuración de instalación. Si el sistema pide confirmación para realizar cambios en el dispositivo, dar clic en Sí, para que continúe este proceso.

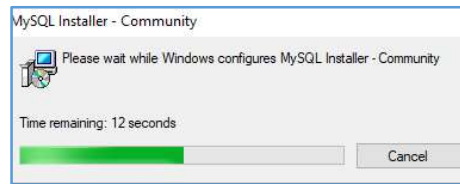


Figura 1.3. Preparación para la instalación.

- Seleccionar la primera opción de instalación, de esta forma se instalarán todos los productos de este programa.

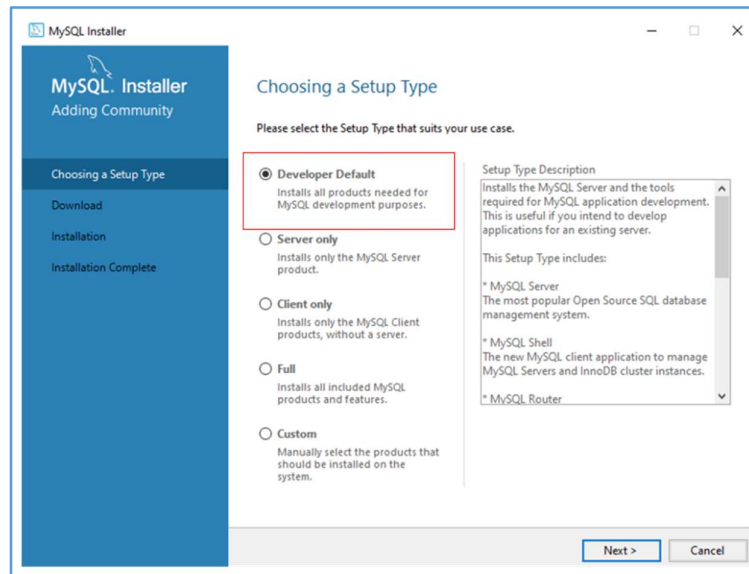


Figura 1.4. Instalación por defecto.

- Si el sistema operativo del computador en donde se está realizando la instalación no contiene todos los programas necesarios, el instalador le indicará los requerimientos que le hacen falta, en este caso dar clic en Execute para que se instalen los requerimientos.

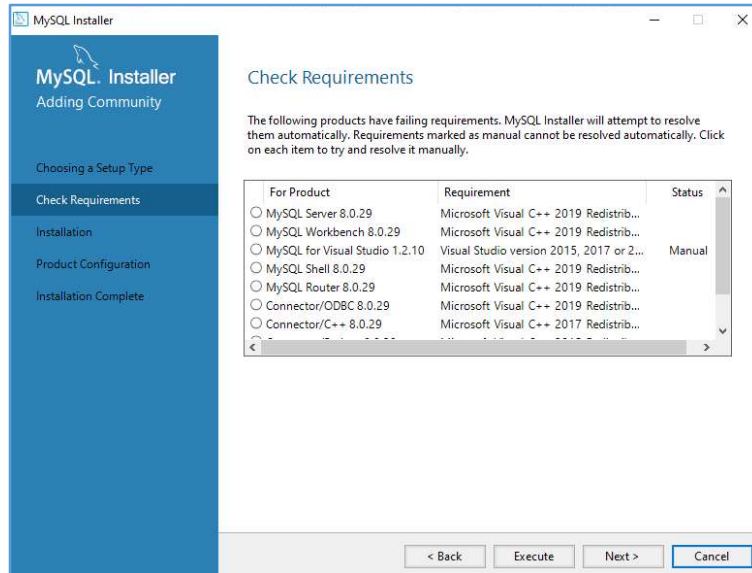


Figura 1.5. Requerimientos de instalación.

- En algunos requerimientos se necesitará que se acepten los términos y condiciones de la licencia, dar check en esta opción y dar clic en instalar.

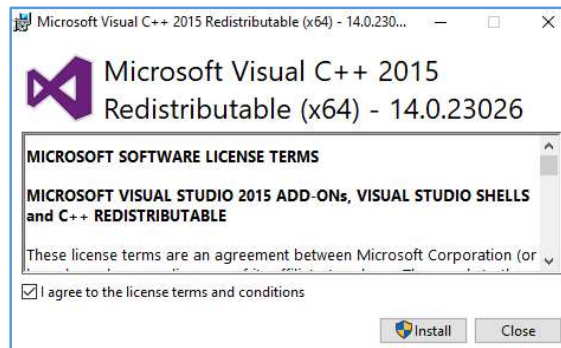


Figura 1.6. Aceptar términos y condiciones de licencia

- Cuando ya se encuentren instalados todos los requerimientos, dar clic en Next.
- Dar clic en Execute para que comience con la instalación.

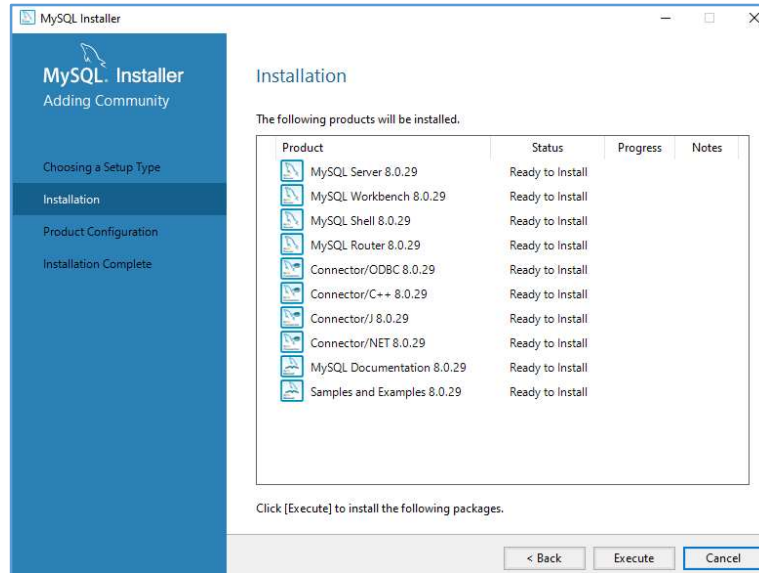


Figura 1.7. Programas listos para instalar.

- Cuando se terminen de instalar todas las herramientas, se mostrará la interfaz como se muestra en la figura 1.8. Luego dar clic en Next.

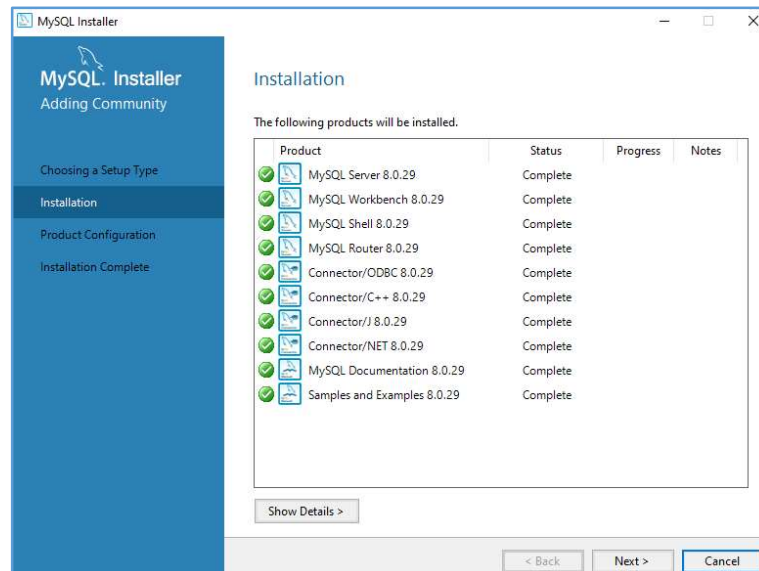


Figura 1.8. Herramientas instaladas.

- Después de esto se mostrarán las herramientas que necesitan ser configuradas, dar clic en Next.

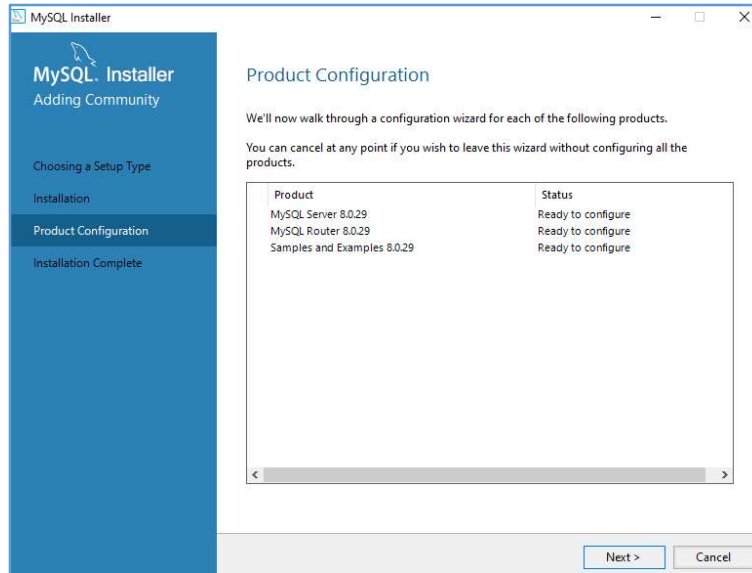


Figura 1.9. Herramientas para configurar.

- Como se muestra en la figura 1.10, dejar esas configuraciones por defecto, ya que, en el presente proyecto se utilizará como puerto el 3306. Dar clic en Next.

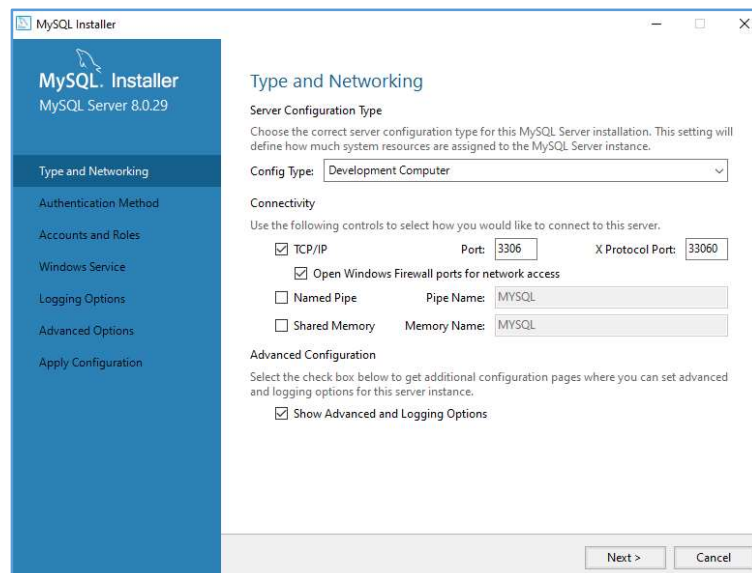


Figura 1.10. Configuración del servidor.

- Dejar la opción por defecto en el método de autenticación y dar clic en Next.

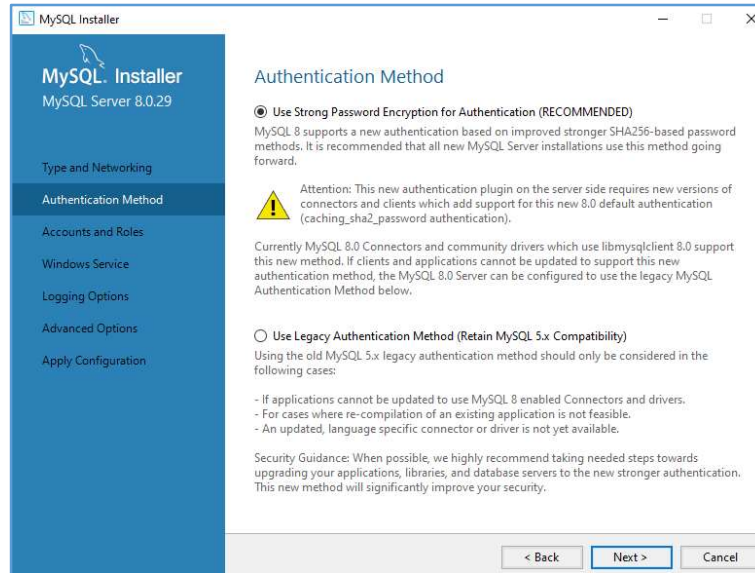


Figura 1.11. Método de autenticación.

- Ingresar la contraseña para el usuario principal que es root, para este proyecto se ha utilizado la contraseña 345199. Dar clic en Next.
 - **Nota:** La contraseña será utilizada en otras ocasiones, ingresar una que sea recordada con facilidad.

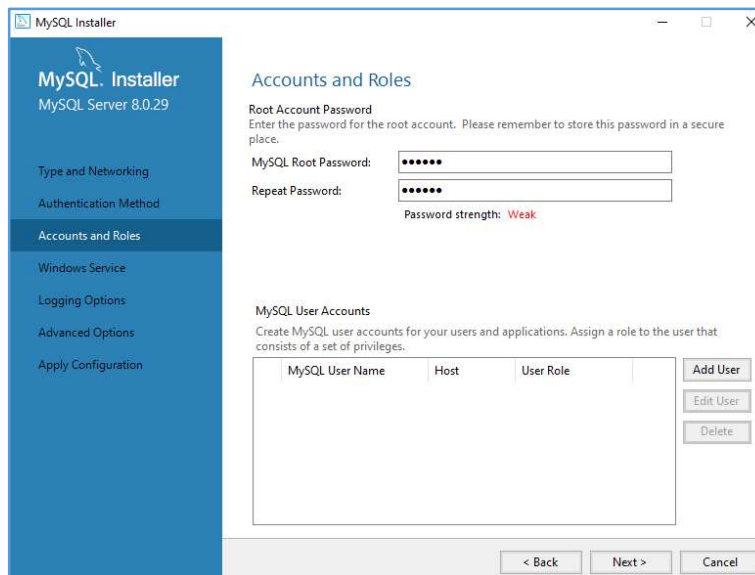


Figura 1.12. Cuentas y Roles.

- Para que el servidor MySQL quede como un servicio de Windows, que en este caso eso es lo que se requiere, dejar las opciones por defecto y dar clic en Next.

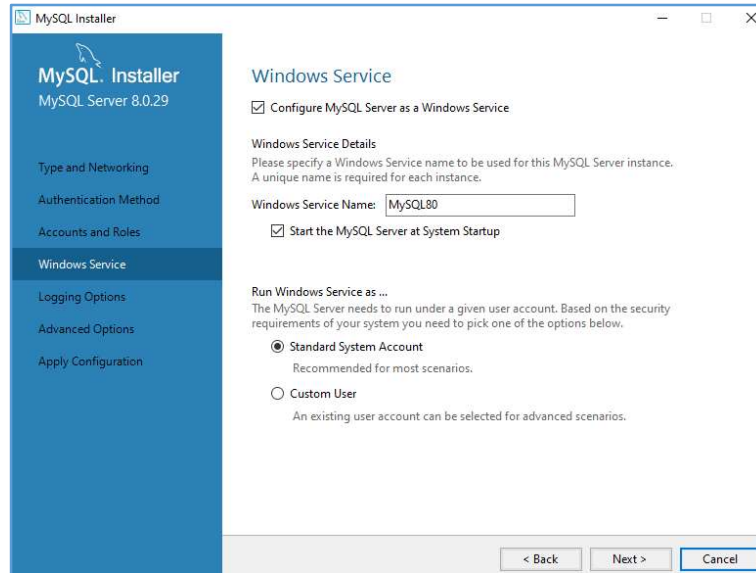


Figura 1.13. Servicios de Windows.

- De igual forma en las opciones de inicio de sesión, figura 1.14, y en las opciones avanzadas, figura 1.15, dejar las opciones por defecto y dar clic en Next.

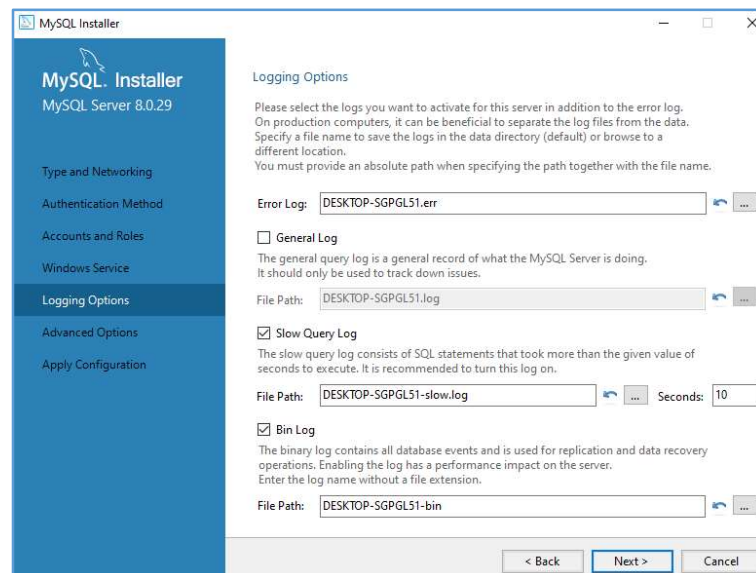


Figura 1.14. Opciones de inicio de sesión.

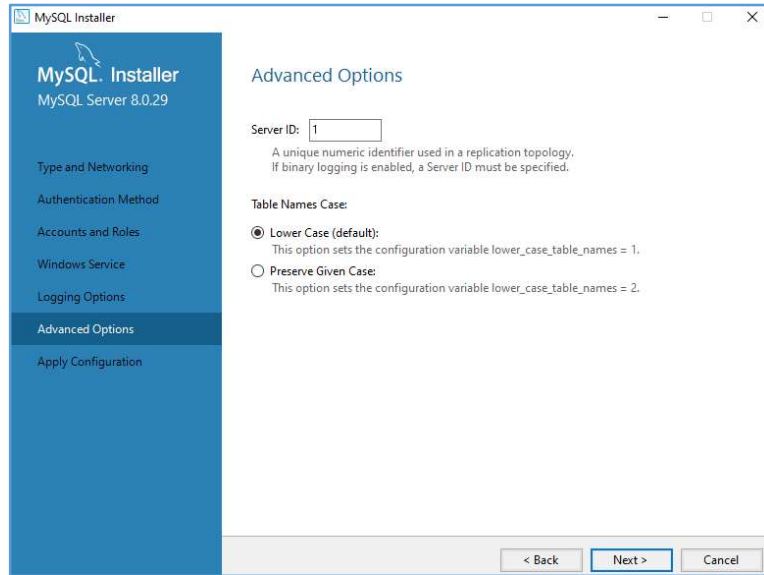


Figura 1.15. Opciones avanzadas.

- Una vez que se apliquen todas las configuraciones como se observa en la figura 1.16, dar clic en Finish.

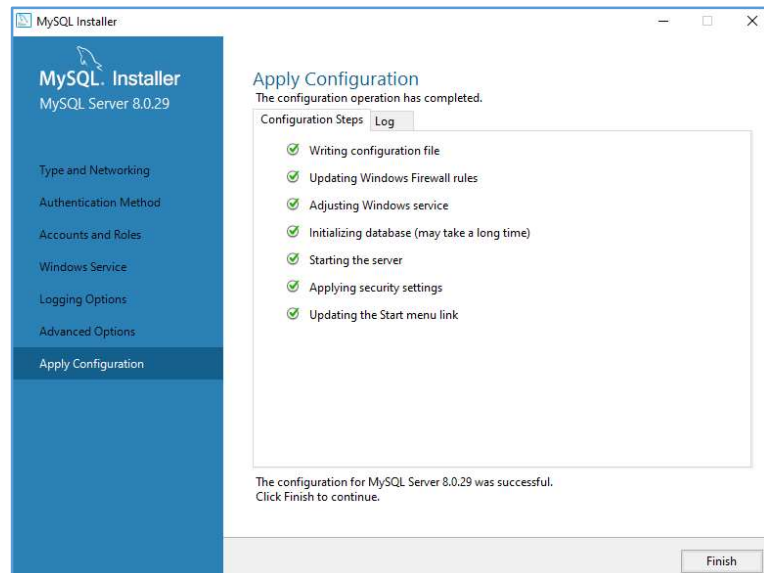


Figura 1.16. Configuraciones realizadas.

- Para realizar la conexión con el servidor, seleccionarlo e ingresar sus datos, en este caso el usuario es root y la contraseña 345199, dar clic en check para verificar los datos. Como se muestra en la figura 1.17 la conexión ha sido exitosa. Luego dar clic en Next.

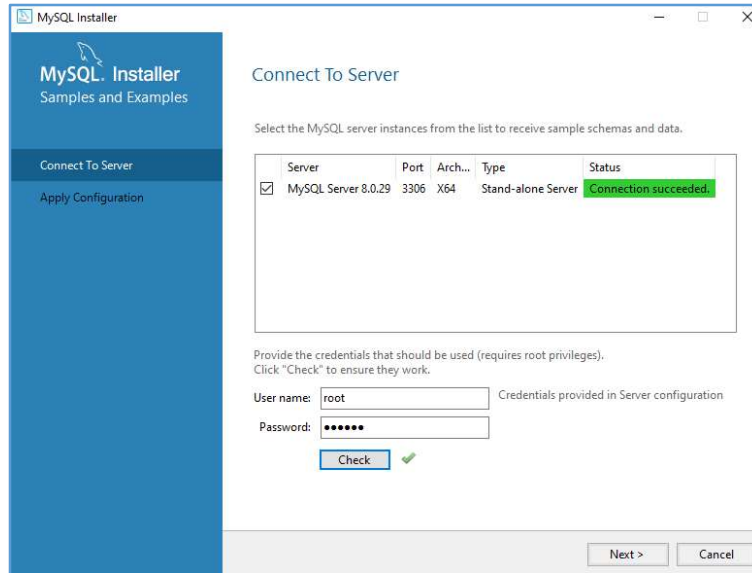


Figura 1.17. Conexión al servidor.

- Una vez que la configuración haya terminado como se ve en la figura 1.18. Dar clic en Finish.

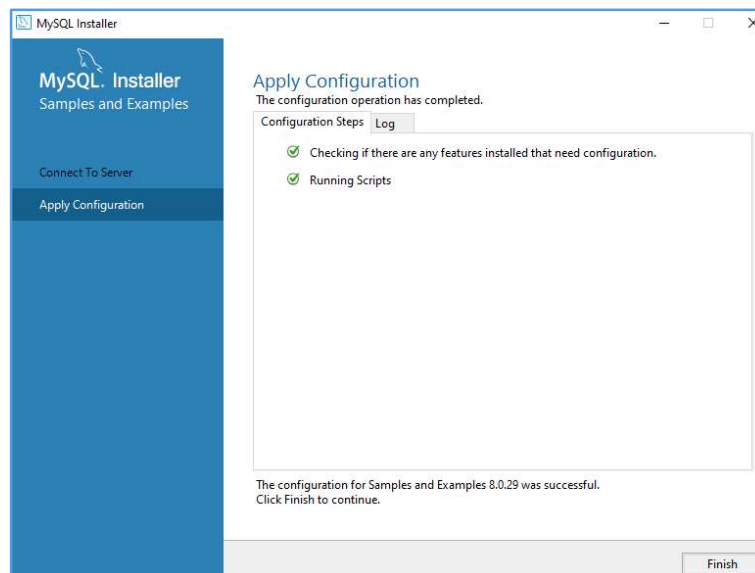


Figura 1.18. Configuración del servidor terminada.

- Cuando la configuración del servidor se encuentra completa, dar clic en Next.

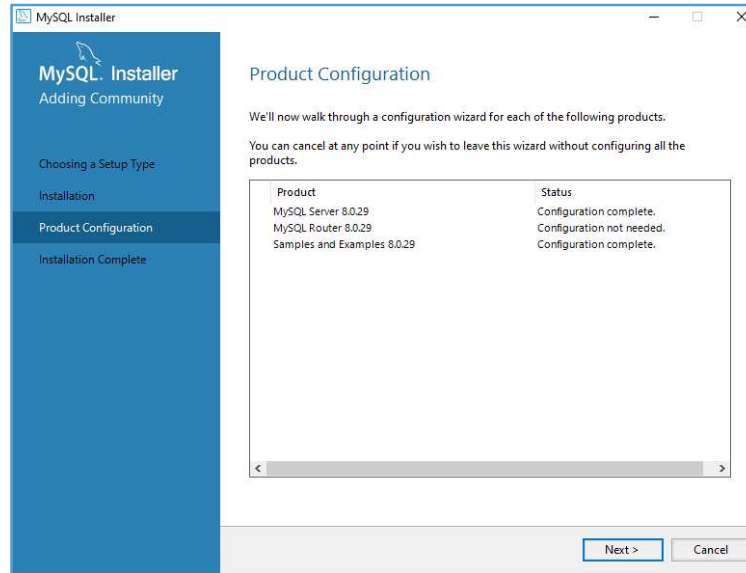


Figura 1.19. Configuración del producto.

- Si se desea que se ejecute MySQL Workbench una vez finalizada la instalación, dejar las opciones marcadas por defecto y dar clic en Finish.

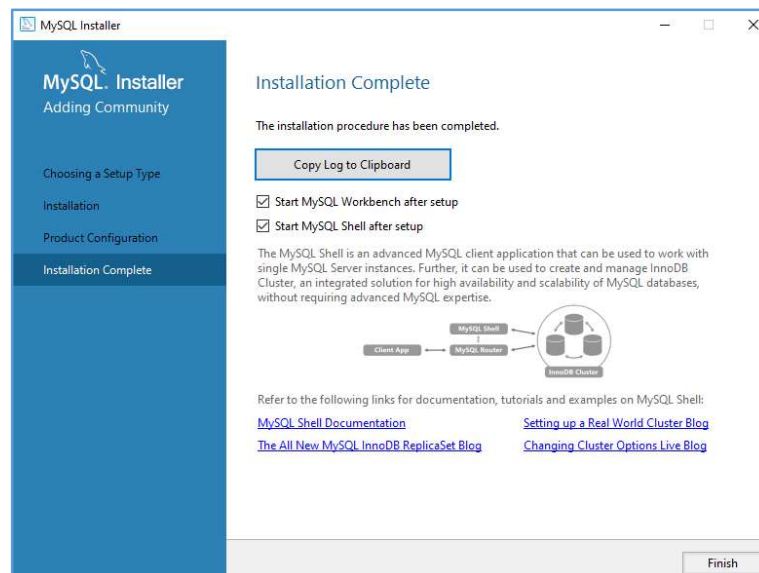


Figura 1.20. Instalación completada.

- Como se muestra en la figura 1.21 MySQL Workbench ha sido instalado correctamente.

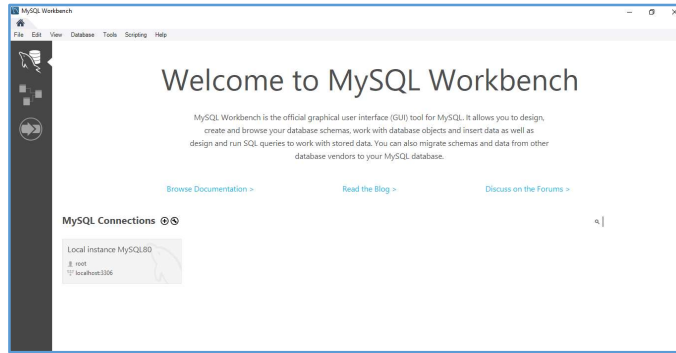


Figura 1.21. MySQL Workbench.

- Se realiza la verificación de su funcionamiento, para ello se da clic en el más, para agregar una nueva conexión, se desplegará una ventana como la que se muestra en la figura 1.22. Se ingresa cualquier nombre de conexión, si es una computadora local en nombre de host se deja la opción por defecto, en nombre de usuario se usará root y en contraseña se ingresará la anteriormente configurada que es 345199.

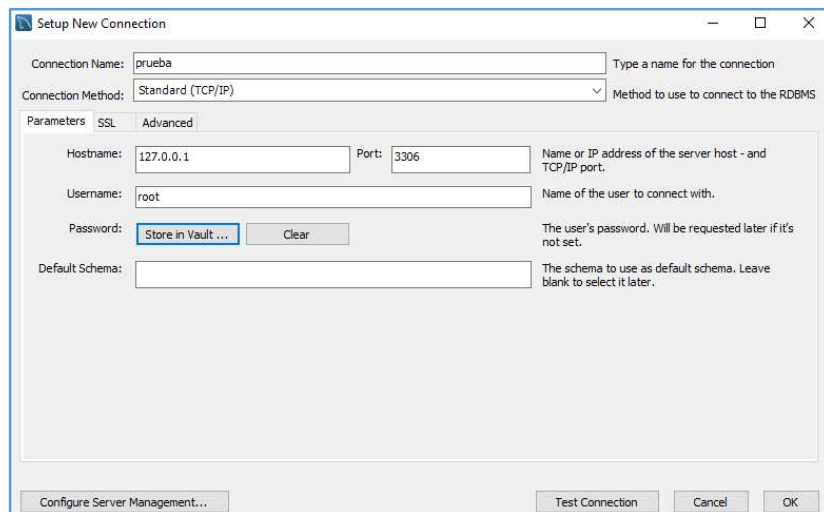


Figura 1.22. Nueva conexión.

- Se da clic en Test Connection, si los datos fueron ingresados correctamente le aparecerá un mensaje de conexión exitosa, como se ve en la figura 1.23.

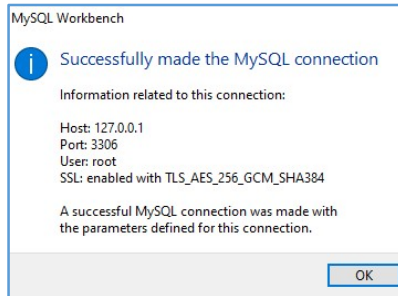


Figura 1.23. Conexión exitosa.

- Al dar clic en OK, se mostrará en el programa un recuadro pequeño con el nombre de la conexión creada, como se visualiza en la figura 1.24.

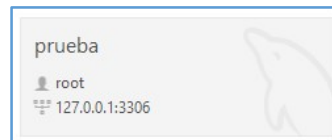


Figura 1.24. Conexión creada.

- Al realizar esta conexión de prueba, ya se puede empezar a trabajar con la base de datos que se requiera.

2. Instalación de Visual Studio Code

- Acceder al siguiente enlace:

<https://code.visualstudio.com/download>

- Seleccionar el sistema operativo de la computadora en donde se realizará la instalación y dar clic sobre el mismo.

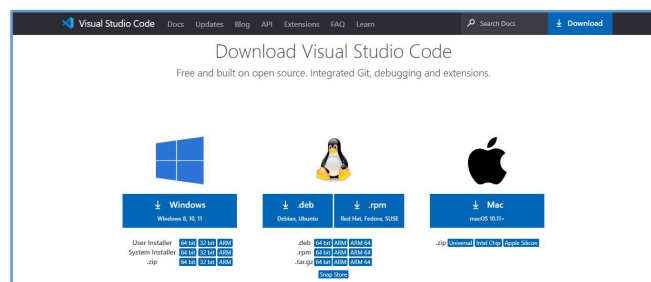


Figura 2. Descarga de Visual Studio Code.

- Una vez descargado el programa dar doble clic sobre el mismo, para empezar con la instalación.

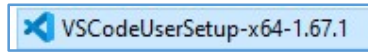


Figura 2.1. Programa descargado.

- Aceptar el acuerdo de licencia y dar clic en Siguiente.

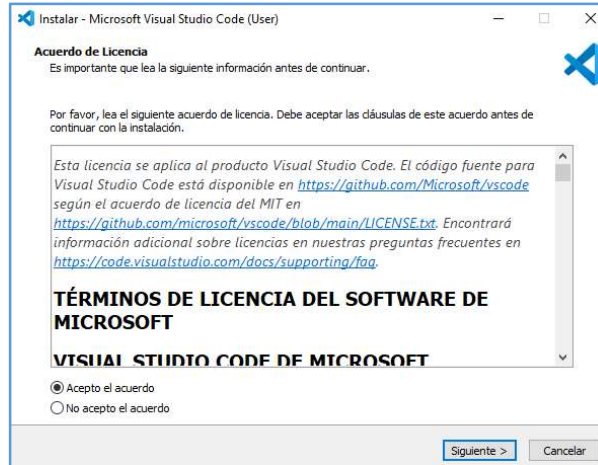


Figura 2.2. Aceptar acuerdo de licencia.

- Seleccionar la ubicación en donde se instalará el programa. Dar clic en Siguiente.

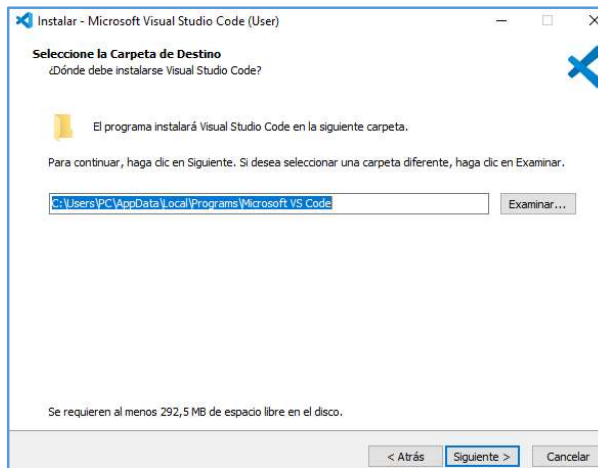


Figura 2.3. Carpeta de destino.

- Seleccionar la carpeta de acceso directo y dar clic en Siguiente.

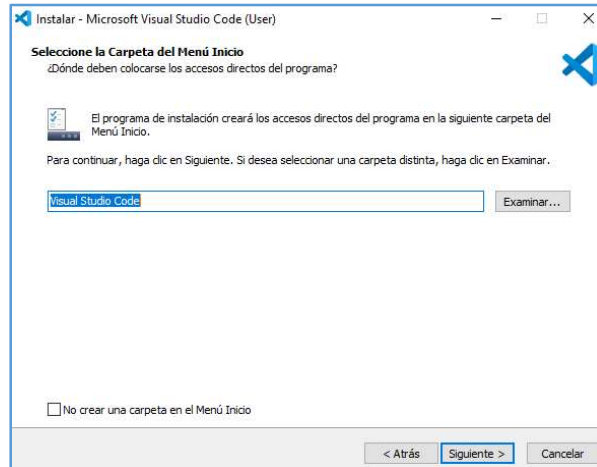


Figura 2.4. Carpeta del menú inicio.

- Seleccionar las opciones adicionales que desee y dar clic en Siguiete.

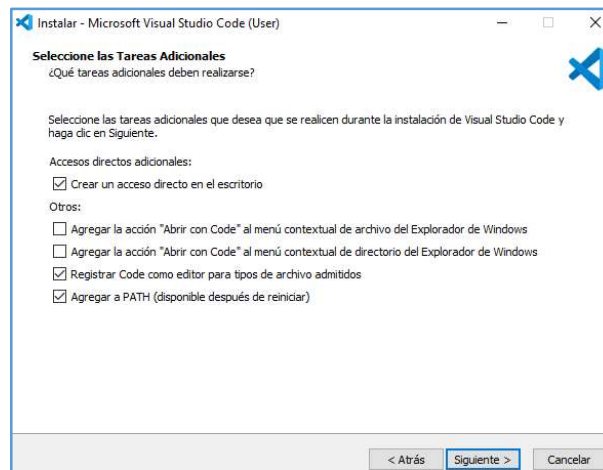


Figura 2.5. Opciones adicionales.

- Si desea realizar la instalación dar clic en Instalar.

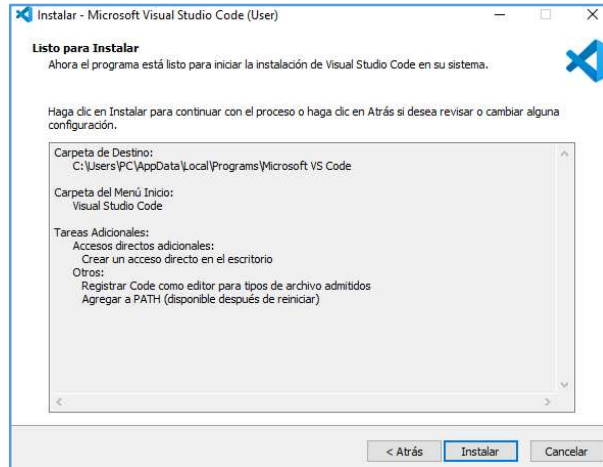


Figura 2.6. Listo para instalar.

- Una vez que la instalación se haya completado, se mostrará una ventana como se muestra en la figura 2.7. Dar clic en Finalizar.

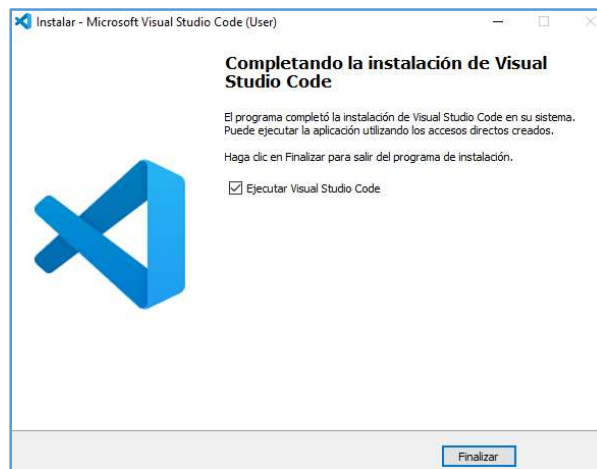


Figura 2.7. Instalación finalizada.

- Visual Studio Code tendrá una vista como se muestra en la figura 2.8. Se puede configurar a gusto del usuario.

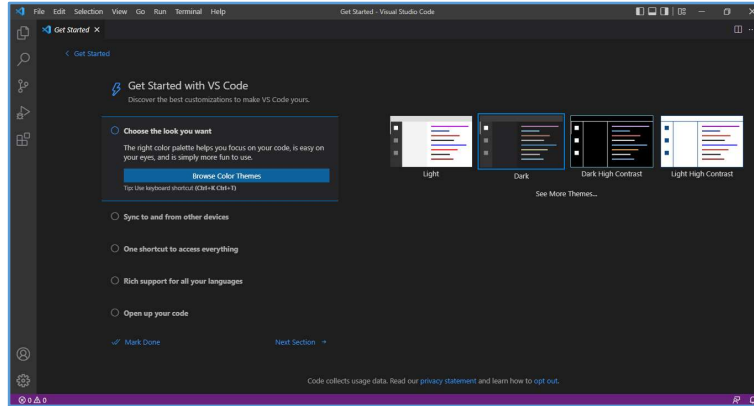


Figura 2.8. Visual Studio Code.

- Dirigirse a extensiones, como se muestra en la figura 2.9.



Figura 2.9. Acceso a extensiones.

- Se recomienda instalar:
 - Activitus Bar: Para conocer el estado de la barra de herramientas.
 - Terminal: Para acceder directamente al terminal desde la barra inferior de Visual Studio Code.

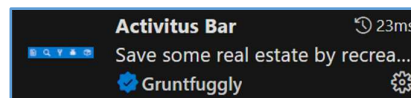


Figura 2.10. Activitus bar.

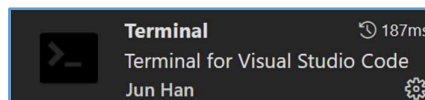


Figura 2.11. Terminal.

- Extensiones para diseño de código y carpetas (opcionales):

- Material Icon Theme: Muestra un diseño predeterminado para las carpetas del proyecto.

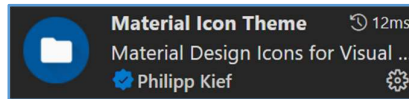


Figura 2.12. Material Icon Theme.

- Monokai Night Theme: Tema de color negro y minimalista para Visual Studio Code.

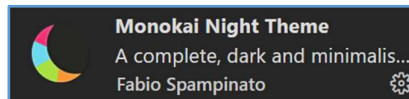


Figura 2.13. Monokai Night Theme.

3. Instalación de Node.js

- Verificar en el terminal de la computadora si cuenta con node.js, para ello se ingresa el siguiente comando: `npm -version`, si cuenta con alguna versión no es necesario seguir este paso de instalación.
- Caso contrario, dirigirse al siguiente enlace:

<https://nodejs.org/es/download/>

- Elegir el sistema operativo de la computadora y descargar el instalador, se recomienda la versión LTS.
- Dar doble clic en el instalador

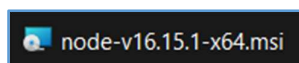


Figura 3. Instalador de node.js.

- Dar clic en Next para continuar con la instalación.



Figura 3.1. Confirmación de instalación.

- Aceptar los términos de la licencia y dar clic en Next.

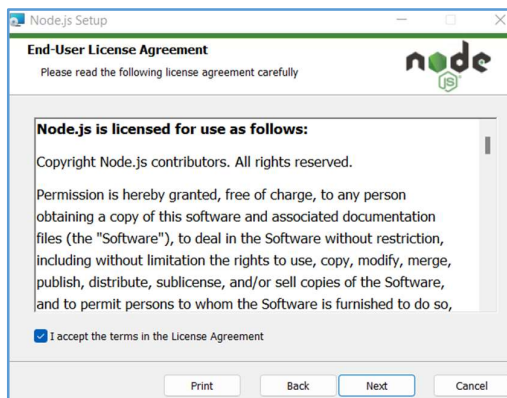


Figura 3.2. Aceptar términos de licencia.

- Seleccionar la carpeta de instalación y clic en Next.

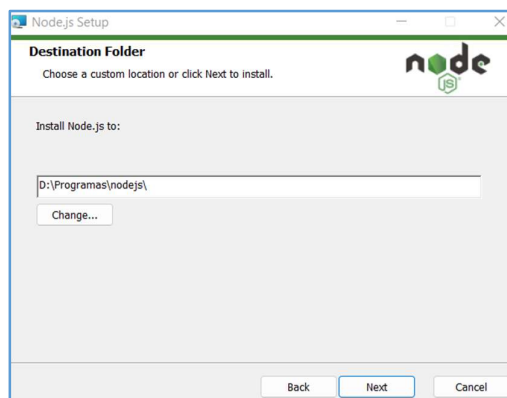


Figura 3.3. Ubicación de carpeta de instalación.

- Dejar la configuración por defecto y dar clic en Next

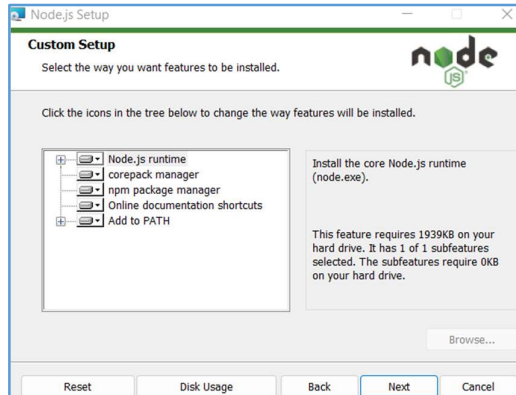


Figura 3.4. Configuración personalizada.

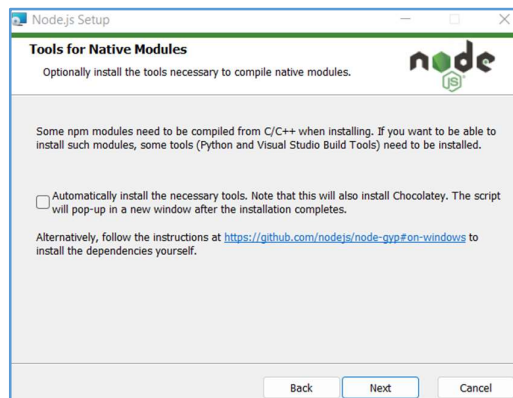


Figura 3.5. Herramientas opcionales.

- Dar clic en Install

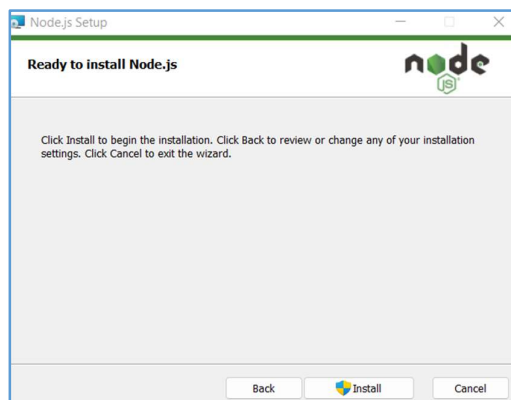
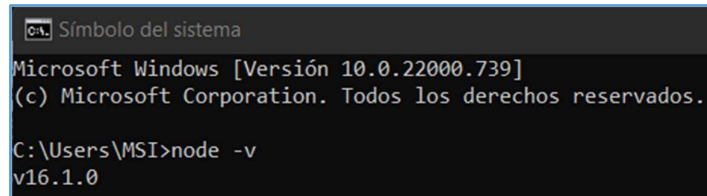


Figura 3.6. Instalación lista para comenzar.

- Finalmente, dar clic en Finish

- Comprobar que se haya instalado correctamente, para ello, abrir el terminal de Windows (Símbolo de sistema) e ingresar el comando: `node -v`. Si le sale alguna versión quiere decir que ya se encuentra instalado.



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22000.739]
(c) Microsoft Corporation. Todos los derechos reservados.
C:\Users\MSI>node -v
v16.1.0
```

Figura 3.7. CMD.

ANEXO B: Manual del interfaz de operador industrial web

El interfaz de operador industrial web es una aplicación web que sirve para el monitoreo de un proceso industrial a través de la lectura de información de variables que se encuentran en una base de datos externa almacenando datos desde campo, cuenta con registro de históricos, minuto a minuto, monitoreo de alarmas, visualización de componentes y control de variables analógicas y digitales.

Para trabajar con esta página web se recomienda instalar lo siguiente:

- MySQL Workbench 8.0 (versión 8.0.25): Sirve como sistema de gestión para la base de datos local, se podrán visualizar todos los datos con mayor facilidad.
- MySQL Server: Es el servidor para poder administrar la base de datos.
- Visual Studio Code: Es el editor de código que permitirá lanzar la página web.

Requisitos básicos que una computadora debe tener para manejar estos programas:

- Sistema operativo: Windows 10 o superior
- RAM: 4 GB
- Procesador: 2 núcleos

Nota: Para utilizar este proyecto, tanto la computadora que tiene la base de datos local como la que tiene la base de datos externa deben estar conectadas en la misma red interna. Para la realización de las pruebas la dirección IP de la computadora local se utilizó 192.168.10.41, mientras que, para la computadora externa se trabajó con la dirección IP 192.168.10.66.

1 CONFIGURACIÓN DE LA BASE DE DATOS LOCAL

Luego de haber instalado MySQL Workbench y su servidor se crea la conexión con la base de datos local, para ello se accede al programa y se da clic en el ícono más, como se observa en la figura 1.1.



Figura 1.1 Crear conexión

Se llenan todos los campos acordes a la configuración de instalación que se realizó, ver figura 1.2, a continuación, se presentan dichos campos:

1. **Connection Name:** Se ingresa el nombre que el usuario desee.
2. **Hostname:** Al ser una conexión local, se deja la dirección IP que viene por defecto.
3. **Puerto:** Puerto por defecto para trabajar con la base de datos es 3306.
4. **Username:** Nombre de usuario que se configuró en la instalación.
5. **Store in Vault:** Ingresar la contraseña configurada en la instalación.

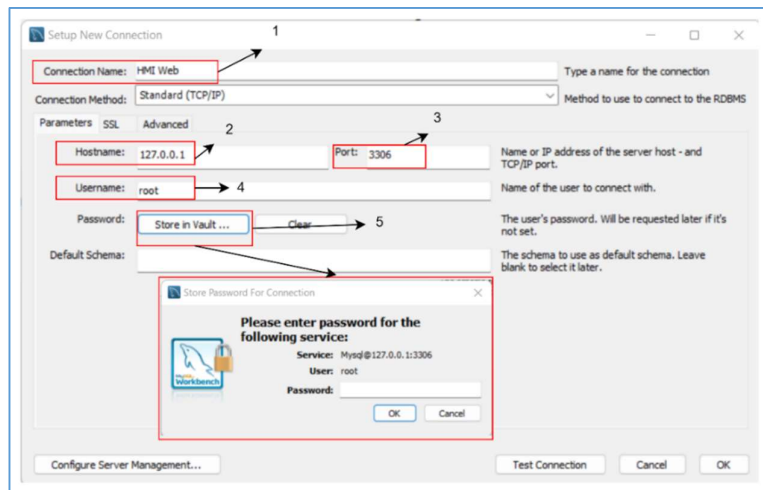


Figura 1.2. Campos para establecer una conexión.

Nota: Para este proyecto, la instalación se realizó con el usuario root y la contraseña 345199. Cuando el usuario realice la instalación puede cambiar estos datos, pero deberá seguir una configuración adicional en el código del programa, esto se explicará más adelante.

Una vez que se llenaron todos los campos, se da clic en Test Connection, si los datos son correctos le saldrá un mensaje de conexión exitosa, caso contrario, deberá llenar nuevamente los datos y reintentar.

Si la conexión fue exitosa se da clic en OK para tener la nueva conexión, un ejemplo se presenta en la figura 1.3.

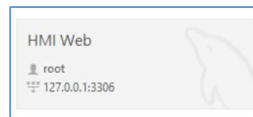


Figura 1.3. Conexión exitosa.

Para acceder a la conexión se da doble clic sobre ella, como se requiere una base de datos con un modelo previamente diseñado se procede a importar el esquema de la base de datos con el que va a trabajar la aplicación web. Se dirige a Server y se selecciona Data Import como se observa en la figura 1.4.

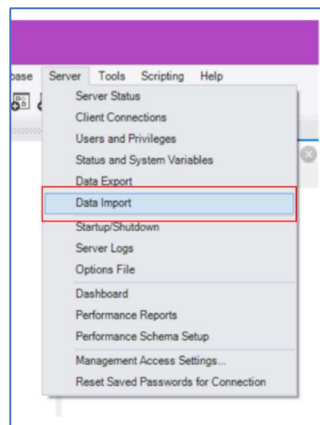


Figura 1.4. Importar esquema.

Luego se seleccionan los siguientes datos, como se muestra en la figura 1.5:

1. Se selecciona la carpeta que contiene la base de datos, en este caso db_hmi
2. Se elige el esquema db_hmi
3. Se escoge sola la estructura de la base de datos
4. Se da clic en Star Import

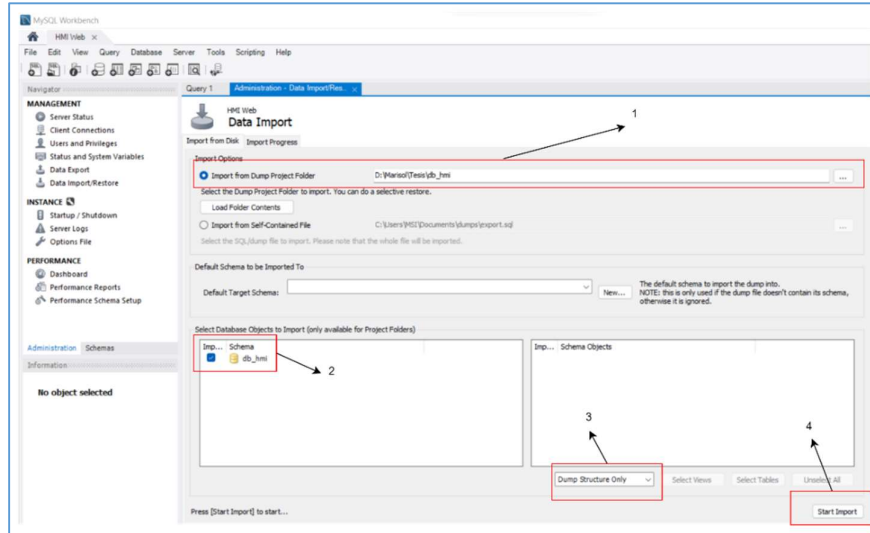


Figura 1.5. Empezar importación de base de datos.

Cuando la importación haya culminado se dirige a esquemas y se da clic en refrescar, esto se aprecia en la figura 1.6. Con esto se podrá visualizar la nueva base de datos importada, ver figura 1.7.

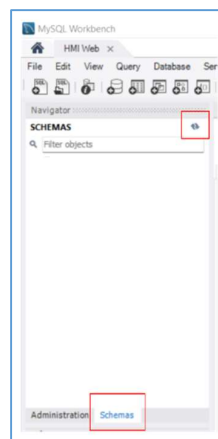


Figura 1.6. Refrescar navegador de base de datos.

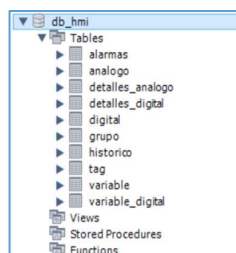


Figura 1.7. Base de datos db_hmi importada.

2 CONFIGURACIÓN DE LA BASE DE DATOS EXTERNA

Para que el presente proyecto pueda acceder a la base de datos externa, ésta debe contar con permisos y privilegios. Entonces, se dirige a MySQL Workbench de la computadora externa y se selecciona la base de datos con la que se va a trabajar.

Se dirige a administración y se selecciona usuarios y privilegios, como se aprecia en a figura 2.

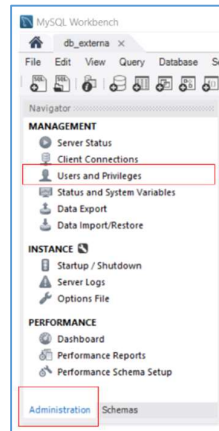


Figura 2. Usuarios y privilegios.

Luego se da clic en agregar cuenta, en Nombre se ingresa 'root', el nombre de la base de datos externa, en Host se ingresa la dirección IP de la computadora local, en contraseña se coloca la misma de la base de datos de la computadora externa en este caso 'root' y se da clic en Apply. Estos campos se pueden ver en la figura 2.1.

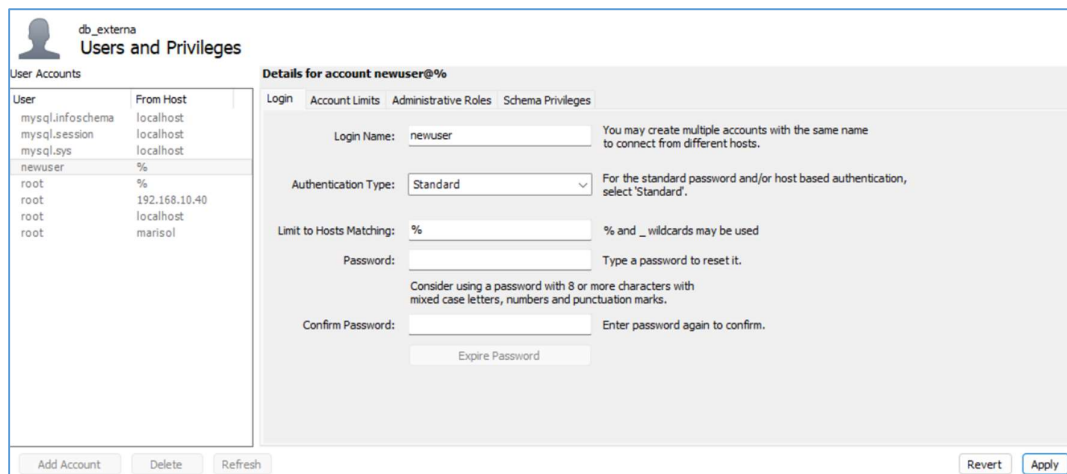


Figura 2.1. Agregar cuenta.

Luego se selecciona la pestaña de privilegios de esquema y se da clic en agregar entrada, se selecciona el esquema de la base de datos externa como se ve en la figura 2.2. Y se da clic en OK.

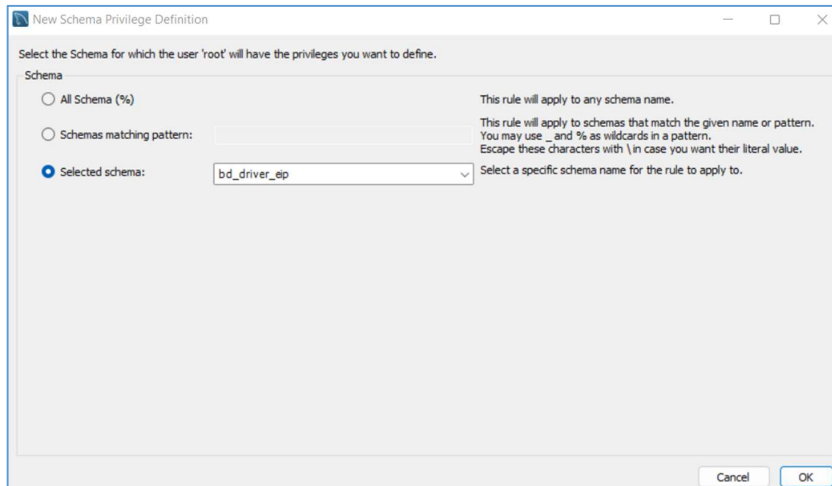


Figura 2.2 Agregar entrada.

Se seleccionan todos los campos y se da clic en Apply, deben quedar las opciones seleccionadas como se muestra en la figura 2.3.

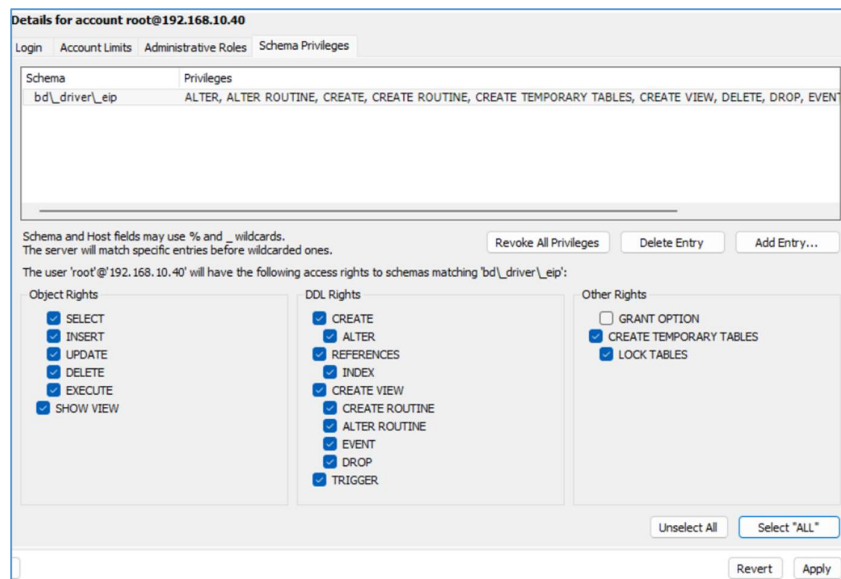


Figura 2.3. Privilegios de esquema.

Finalmente, se dirige a la pestaña de roles administrativos y se da clic en DBA, como se observa en la figura 2.4. Para guardar los cambios se da clic en Apply.

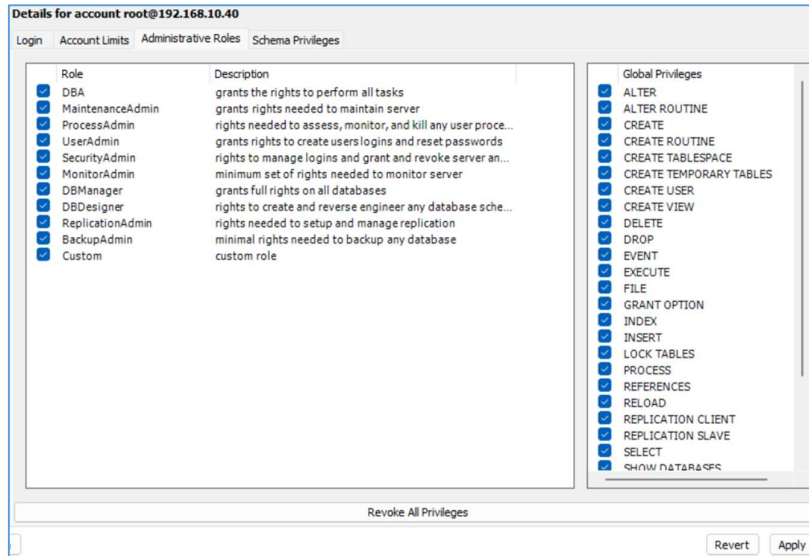


Figura 2.4. Roles administrativos.

3 EJECUTAR APLICACIÓN WEB

Para lanzar la aplicación web se utiliza Visual Studio Code, por ende, se abre este programa y se selecciona open folder como se visualiza en la figura 3, se dirige a la ubicación de la carpeta llamada HMI_WEB y se da clic en seleccionar carpeta.

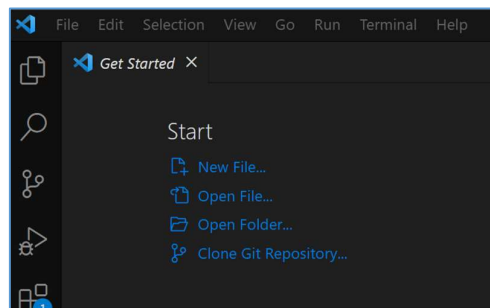


Figura 3. Open folder en Visual Studio Code.

Como se aprecia en la figura 3.1 se visualiza el contenido del proyecto en la parte izquierda, por otro lado, en la parte derecha se ve una flecha, este es el atajo para dirigirse al terminal de la computadora desde este programa.

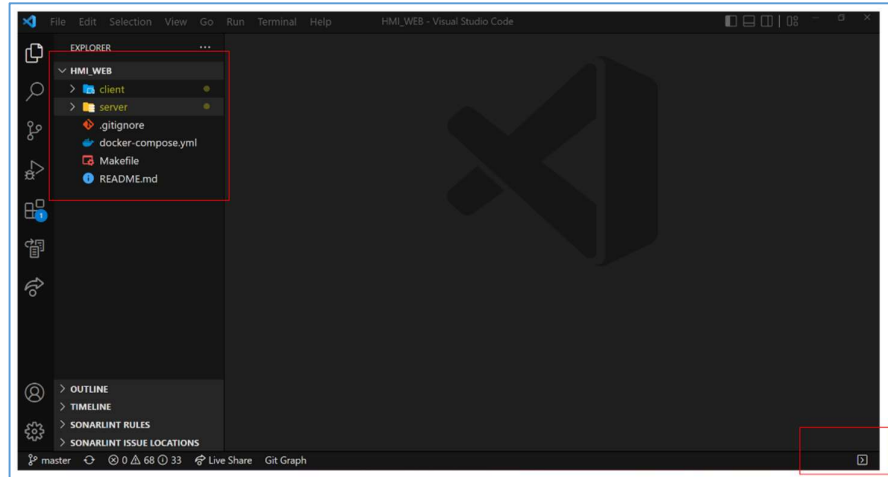


Figura 3.1. Proyecto HMI_WEB.

Lo primero que se debe hacer en el proyecto es dirigirse a la carpeta server, luego a Base_de_Datos y se selecciona el archivo db.js. En la línea 2, como se observa en la figura 3.2, se deben colocar los datos de la base de datos local, esta configuración solo se realiza si no se siguió el manual de instalación de MySQL Workbench, dado que, al instalar, la configuración de la base de datos puede cambiar según el gusto del usuario.

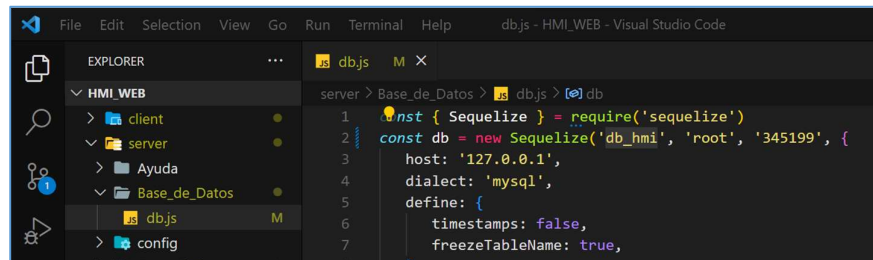


Figura 3.2. Configuración de base de datos local.

Abrir el terminal, al dar clic en la flecha como se indica en la figura 3.1, debe tener una pestaña abierta como se muestra en la figura 3.3.

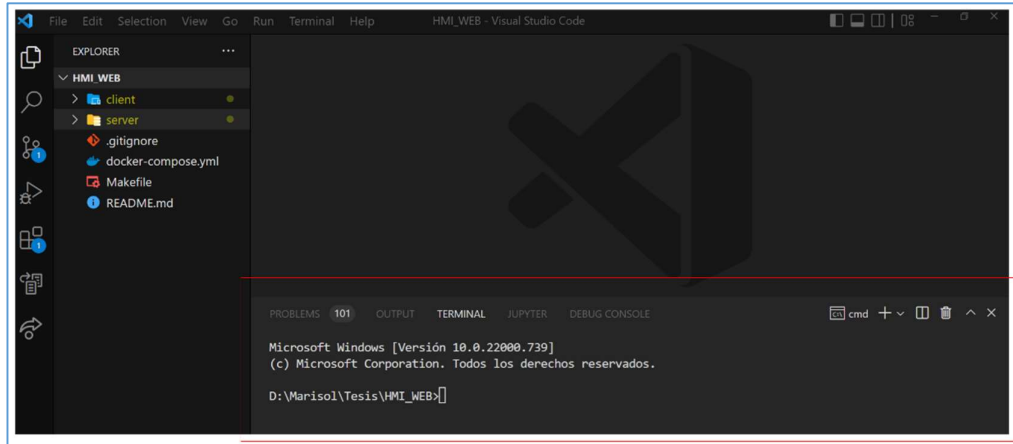


Figura 3.3. Terminal abierto.

Una vez que se confirma que en el terminal aparece la dirección en donde está ubicado nuestro proyecto, en este caso se muestra la figura 3.4.

```
D:\Marisol\Tesis\HMI_WEB>
```

Figura 3.4. Ubicación del archivo.

Para acceder a una carpeta se presiona el comando `cd` seguido del nombre de la carpeta y para retroceder de carpeta se ingresa el comando `cd..`.

En este caso se debe acceder a la carpeta del cliente y del servidor, primero se debe levantar el servidor, entonces se ingresa `cd server` y se presiona `enter`, como se observa en la figura 3.5.

```
D:\Marisol\Tesis\HMI_WEB>cd server
```

Figura 3.5. Ingresar al servidor.

Si es la primera vez que se va a ejecutar el servidor y el cliente, el primer comando que se debe ejecutar en el terminal es `npm install`, esto se lo realiza solo una vez. Luego de esto, se debe ingresar `npm run dev`, que es el comando para levantar al servidor, este último se realiza cada vez que se desea trabajar con la página web, esto se aprecia en la figura 3.6.

```
D:\Marisol\Tesis\HMI_WEB\server>npm run dev
```

Figura 3.6. Levantar servidor.

Ahora se procede a levantar el cliente, para ello se abre otro terminal, se da clic en el más, como se ve en la figura 3.7.

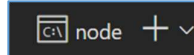


Figura 3.7. Abrir un nuevo terminal.

De la misma forma que en el servidor, se accede a la carpeta de cliente con `cd client`, se da enter, si es la primera vez en ejecutar la aplicación web se ingresa el comando `npm install`, luego se ingresa el comando `npm run dev` y se da enter, como se visualiza en la figura 3.8.

```
D:\Marisol\Tesis\HMI WEB\client>npm run dev
```

Figura 3.8. Levantar cliente.

En el terminal donde se encuentra el cliente luego de ejecutar el último comando, se visualiza un link, dar clic en el mismo para abrir la página web en el navegador, este enlace también puede ser agregado manualmente en una nueva pestaña de cualquier navegador web. El enlace se presenta en la figura 3.9.

```
D:\Marisol\Tesis\HMI_WEB\client>npm run dev
> hmiv2-app@0.0.0 dev
> vite

vite v2.8.4 dev server running at:
> Local: http://localhost:3000/
> Network: use `--host` to expose

ready in 487ms.
```

Figura 3.9. Enlace de cliente.

4 CONFIGURACIÓN DE LA APLICACIÓN WEB

4.1 Configuración de la pestaña de ajustes

Para dirigirse a esta pestaña se da clic en la tuerca de configuraciones, como se observa en la figura 4.1.

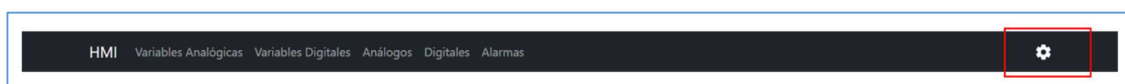


Figura 4.1. Pestaña de configuraciones.

4.1.1 Configuraciones locales

Esta configuración permite mostrar o no la mesa de trabajo dinámica de la pestaña principal. Si se deja el check por defecto (Mostrar), ver figura 4.1.1, se mostrará la mesa de trabajo estática, aquí se irán apilando los componentes junto con las variables vinculadas a ellos de forma vertical, en un orden preestablecido. Por otro lado, si se da clic al check, el estado de este cambia a Ocultar, por lo que, se mostrará la mesa de trabajo dinámica, en este espacio se podrán mover los componentes como se desee.



Opción	Valor
Mostrar mesa de trabajo en Inicio	<input checked="" type="checkbox"/> Mostrar

Figura 4.1.1 Configuraciones locales.

4.1.2 Base de datos externa

En esta configuración se debe ingresar toda la información de la base de datos externa, un ejemplo de ello se presenta en la figura 4.1.2.

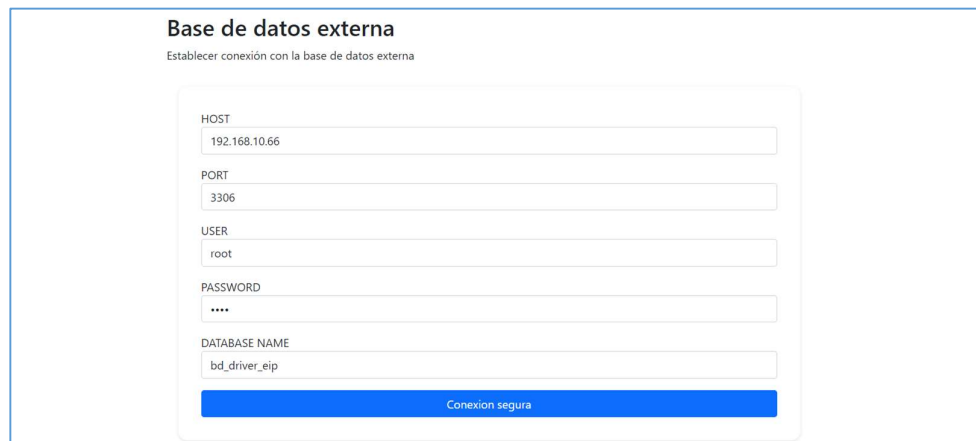


Figura 4.1.2. Base de datos externa.

1. **Host:** Ingresar la dirección IP de la otra computadora, en donde se encuentra la base de datos externa (192.168.10.66).
2. **Port:** Ingresar 3306, puerto por defecto.
3. **User:** Ingresar el usuario que se encuentra configurado en MySQL Workbench de la otra computadora (root).

- 4. Password:** Ingresar la contraseña que se encuentra configurado en MySQL Workbench de la otra computadora (root).
- 5. Database name:** Ingresar el nombre de la base de datos externa con la que se va a trabajar (bd_driver_eip).

Nota: Los datos que se encuentran entre paréntesis, son con los que se trabajaron para realizar las pruebas de la aplicación web.

Si los datos ingresados pertenecen a una base de datos externa real, se mostrará el mensaje de la figura 4.1.2.1, lo que indica que se ha realizado la conexión exitosamente, caso contrario le aparecerá el mensaje de la figura 4.1.2.2. En este último caso se debe reiniciar el servidor, es decir, se debe levantar el Back-end nuevamente.

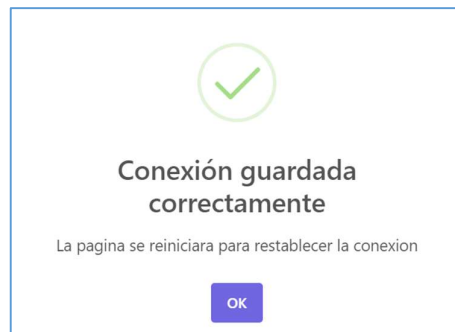


Figura 4.1.2.1 Conexión exitosa

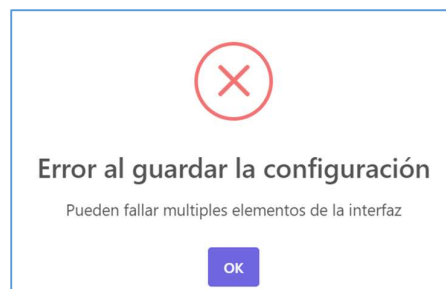


Figura 4.1.3.2 Conexión fallida

4.1.3 Relación de datos

Una vez que la conexión con la base de datos externa se realizó, se observa en la figura 4.1.3 como se realiza la vinculación entre variables de las distintas bases de datos.

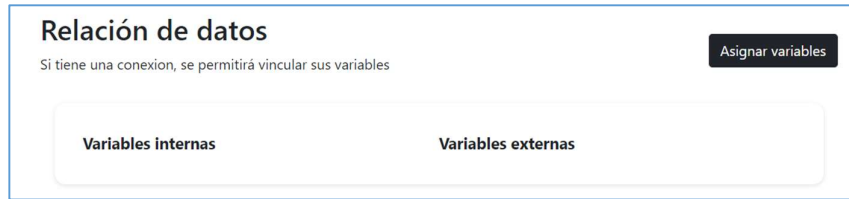


Figura 4.1.3 Relación de datos.

Siempre y cuando se tenga una conexión, se podrán vincular las variables entre sí, para ello se da clic en el botón Asignar variables y se despliega el modal de la figura 4.1.3.1.

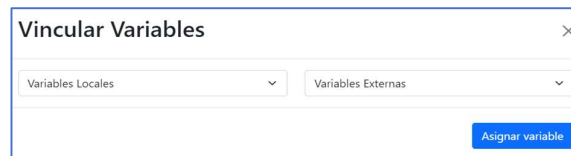


Figura 4.1.3.1. Modal de vincular variables.

Para observar las variables internas se da clic en Variables Locales, como se observa en la figura 4.1.3.2. La variable de interés para ser vinculada se selecciona al dar clic sobre la misma. En este caso se ha colocado un identificador al final del nombre de la variable, se reconoce como variable analógica la que tiene una letra (A) y digital la que tiene una (D).

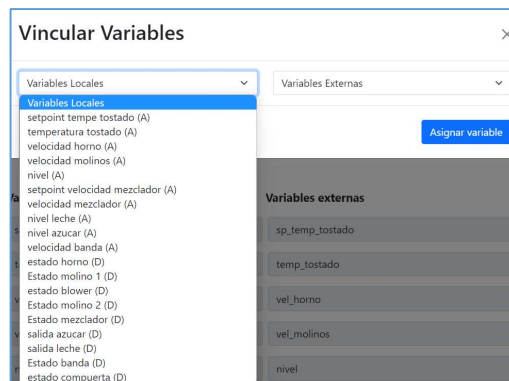


Figura 4.1.3.2. Variables locales.

Por otro lado, si se desean seleccionar las variables de la base de datos externa, se da clic en Variables externas, como se visualiza en la figura 4.1.3.3. De la misma forma, se selecciona cualquier variable al dar clic sobre ella.

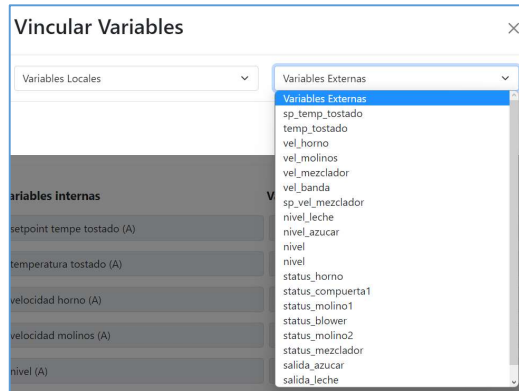


Figura 4.1.3.3. Variables Externas.

Una vez que se selecciona tanto la variable local como la externa, para que ambas se vinculen se debe dar clic en el botón Asignar Variable, ver figura 4.1.3.4. Al realizar esta acción la pestaña se actualiza y se van guardando las vinculaciones.

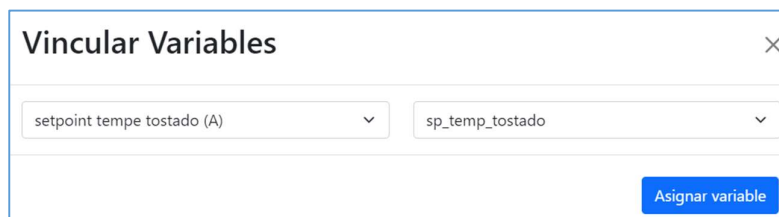


Figura 4.1.3.4. Asignar variable.

Cabe recalcar que, si no se desea que alguna de las variables locales se encuentre vinculada con alguna de las externas, pero ya se realizó dicha vinculación, en la opción de variables externas debe seleccionar Variables Externas, de esta forma se borra la asignación de variables. Un ejemplo de esto se aprecia en la figura 4.1.3.5, se realizó con la variable temperatura tostado (A), por lo que, al finalizar todo el proceso se visualiza un mensaje de Sin vincular, por otro lado, las demás variables como sí han sido asignadas se encuentran junto a su par vinculado.



Figura 4.1.3.5. Variables vinculadas.

4.1.4 Creación de grupos

Si se desea empezar con la creación de variables, primero se debe ir a la sección de configuración y dirigirse a la parte final, aquí se realiza el manejo de grupos. Antes de crear una variable se debe crear el grupo en donde se la va a clasificar, para que no suceda una inconsistencia de datos. En la figura 4.1.4 se aprecia la sección de creación de grupos.

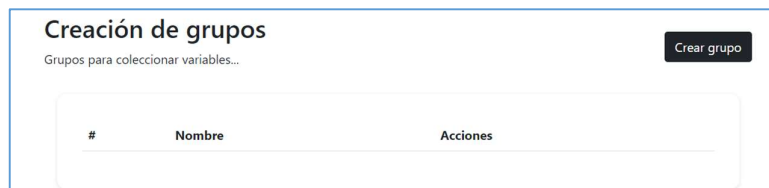


Figura 4.1.4. Creación de grupos

Si se desea crear un grupo, se debe dar clic en el botón Crear grupo, en donde se desplegará un nuevo modal como se observa en la figura 4.1.4.1, se ingresa el nombre del grupo y al dar clic en Guardar grupo, se validará que este campo no este vacío y se procede a almacenar la información en la base de datos, así como en la tabla de grupos.

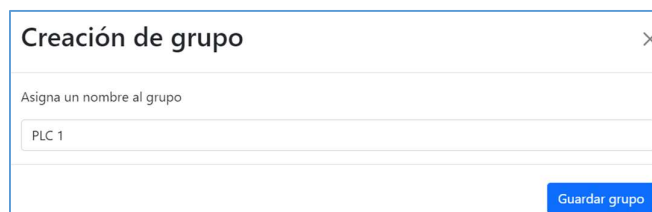


Figura 4.1.4.1 Modal de creación de grupo.

Si el grupo se creó exitosamente, se mostrará un mensaje como en la figura 4.1.4.2. En este caso se realiza una validación para que el nombre del grupo no se encuentre vacío, si se detecta este error se presenta el mensaje de la figura 4.1.4.3.

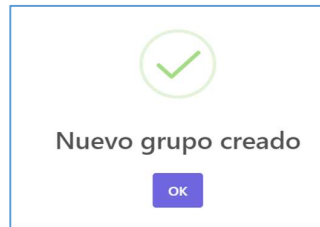


Figura 4.1.4.2. Grupo creado.

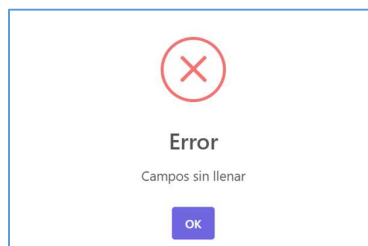


Figura 4.1.4.3. Error de creación de grupo.

Si se desea actualizar el nombre del grupo, se da clic en el ícono de lápiz del grupo seleccionado, como se ve en la figura 4.1.4.4, se actualiza su nombre, se da clic en Actualizar, se valida que no este vacío este campo y se procede a refrescar la información, esta acción se confirma con un mensaje de Buen trabajo.

#	Nombre	Acciones
1	PLC 1	 

Figura 4.1.4.4 Íconos de lápiz y basurero.

Si se desea eliminar algún grupo, se da clic en el ícono de basurero del grupo seleccionado, se confirma la eliminación de este, como se muestra en la figura 4.1.4.5 y se actualiza tanto la tabla de grupos como la base de datos. De la misma forma, para confirmar esta acción se le muestra al usuario un mensaje de Eliminado.

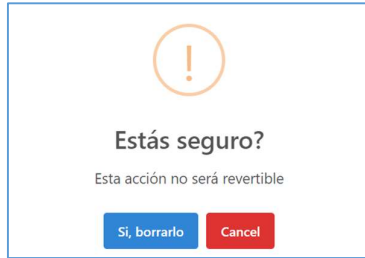


Figura 4.1.4.5 Confirmación de eliminación del grupo.

4.2 Configuración de la pestaña de variables analógicas

Para acceder a esta pestaña se da clic como indica la figura 4.2. Aquí se podrán crear variables que sean de tipo entero o flotante.

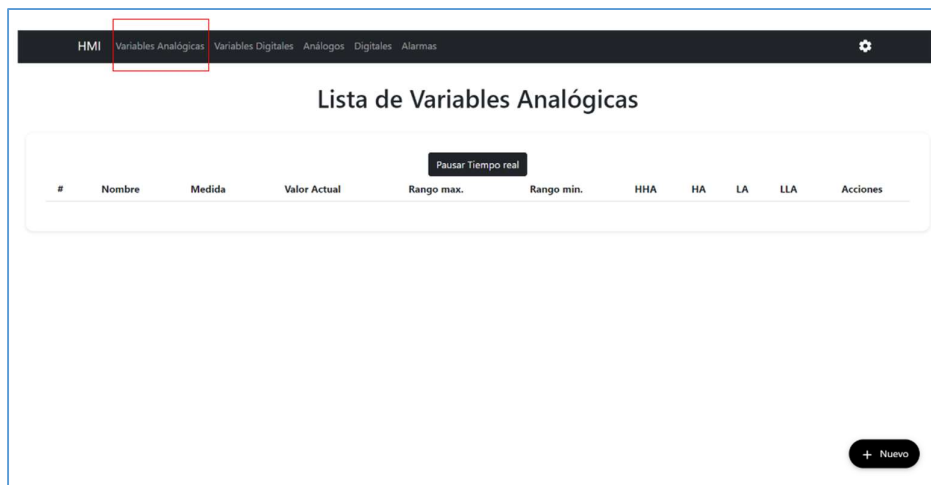


Figura 4.2 Pestaña de variables analógicas.

Si se desea crear una nueva variable, primero se debe seleccionar el botón Nuevo para que se despliegue el formulario de la figura 4.2.1, se llenan los campos acordes a la información de cada variable. En este caso, si algún campo necesario se encuentra vacío se muestra un mensaje como en la figura 4.2.2. Al aplastar Guardar, si todos los campos han sido validados se le mostrará un mensaje de Buen trabajo al usuario, automáticamente esa variable se mostrará en la página web y se almacenará en la base de datos local.

Formulario [X]

Datos

Nombre:

Medida:

Rango máximo:

Rango mínimo:

Gráfica:

Histórico: No Sí

Alarmas

Límite alto:

Límite alto:

Límite bajo:

Límite bajo bajo:

Asignación de grupo

Grupo:

Figura 4.2.1. Modal de formulario para variables analógicas.

Nombre:

Campo requerido

Figura 4.2.2. Campo requerido.

En el presente formulario se encuentran los siguientes campos:

1. **Nombre:** Se ingresa el nombre de la variable
2. **Medida:** Es la medida con la que va a trabajar la variable. Se han colocado algunas de las medidas más utilizadas. Se ha colocado la opción de otro para representar alguna medida que el usuario desee, pero no se encuentre dentro de las opciones. Este campo se puede visualizar desplegado en la figura 4.2.3.

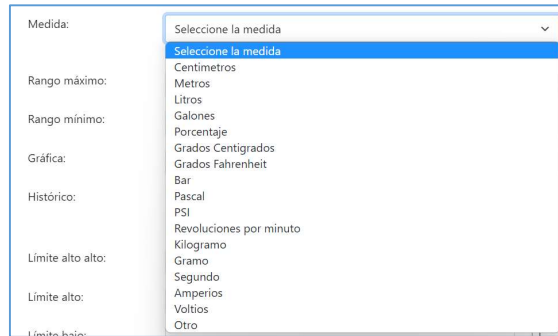


Figura 4.2.3. Campo: Medida.

3. **Rango máximo y mínimo:** Rangos en donde opera la variable
4. **Gráfica:** Representación visual de la variable analógica en la pestaña principal (HMI). Se presentan 4 opciones, ver figura 4.2.4. La opción Gauge se aprecia en la figura 4.2.5. La opción barra horizontal se observa en la figura 4.2.6. La opción vertical, es mostrada en la figura 4.2.7, esta gráfica solo es válida para el componente analógico tipo tanque, ya que, se ha buscado brindar una representación para la variable nivel. Finalmente, se presenta en la Figura 4.2.8, la opción dígito.

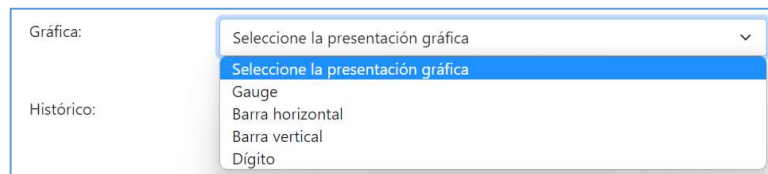


Figura 4.2.4. Campo: gráfica.

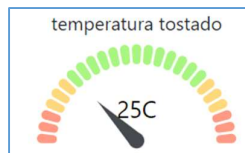


Figura 4.2.5. Gráfica: Gauge.

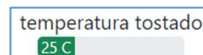


Figura 4.2.6. Gráfica: Barra horizontal.



Figura 4.2.7. Gráfica: Barra vertical.

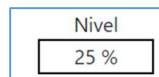


Figura 4.2.8. Gráfica: Dígito.

5. **Histórico:** Se registra la variable para que se vayan guardando los datos del valor actual cada minuto en la base de datos. Si selecciona esta opción podrá acceder a la gráfica de históricos del grupo en donde se encuentra la variable.
6. **Límites de alarmas:** Representan los límites en donde la variable puede o no operar. Si desee trabajar con alarmas y que la variable trabaje dentro de cierto rango, se debe colocar el límite de la alarma que desee, en este caso se han colocado 4 opciones y se debe seleccionar el check de cada límite.
7. **Grupo:** Se elige el grupo en donde se clasificará a la variable. En este campo se desplegarán todos los grupos que se hayan creado anteriormente.

Si se desea modificar algún campo de estas variables, se debe seleccionar el ícono de lápiz se modifica el campo que se requiera y se da clic en Actualizar. Se muestra un mensaje de confirmación cuando se ha actualizado la variable.

Finalmente, si se desea eliminar una de las variables, se selecciona el ícono de basurero de la variable correspondiente, se confirma la acción mediante un mensaje al usuario y de igual forma, se muestra un mensaje de variable eliminada.

Nota: Esta pestaña cuenta con un botón de pausar tiempo real, sirve para ya no tomar datos de la base de datos externa, ya que, si en algún caso el monitoreo no es necesario, se pueda dar clic en este botón para no gastar los recursos de la base de datos local.

4.3 Configuración de la pestaña de variables digitales

Para acceder a la presente pestaña se da clic en el recuadro que muestra la figura 4.3. Esta pestaña sirve para la creación, actualización y eliminación de variables de tipo booleano.

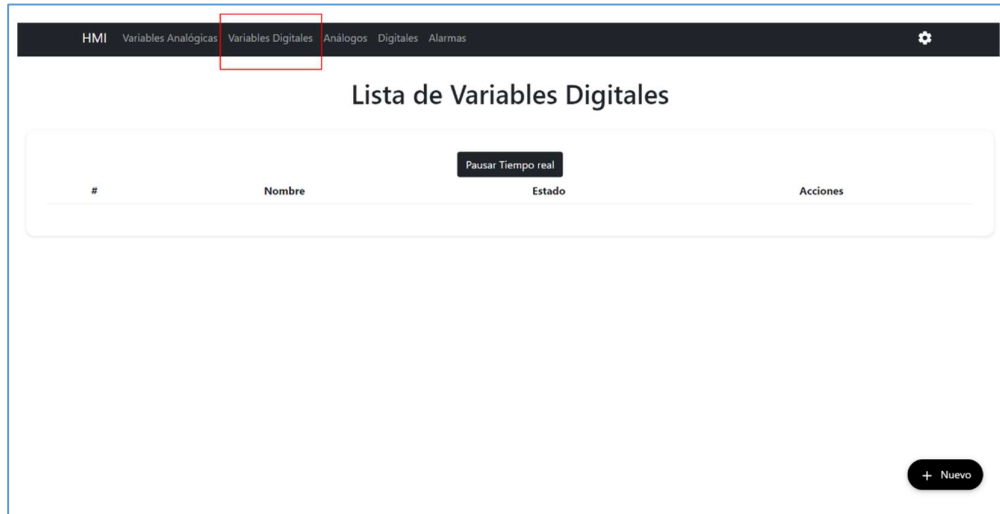


Figura 4.3. Pestaña de variables digitales.

Esta pestaña trabaja de la misma forma que la pestaña de variables analógicas, como se puede notar, tiene las mismas partes, sin embargo, lo que se diferencia de esta pestaña son los campos presentes en el formulario, al dar clic en el botón Nuevo, se observa dicho formulario en la figura 4.3.1.

Figura 4.3.1. Formulario de variables digitales.

Consta de los siguientes campos:

5. **Nombre:** Se ingresa el nombre de la variable digital.

6. **Gráfica:** Se muestran las opciones para la representación visual en la figura 4.3.2. La opción ON-OFF se aprecia en la figura 4.3.3, este tipo de representación presenta las letras de prendido o apagado junto con un indicador que muestra su estado, mientras que la opción de animación se visualiza en la figura 4.3.4, esta opción permite vincular al componente digital asignado, es decir, el estado que presente esta variable también mostrará como estado el componente, si la variable se encuentra prendida '1' el componente se verá de color verde, mientras que, si se encuentra apagado '0' se mostrará de color azul.

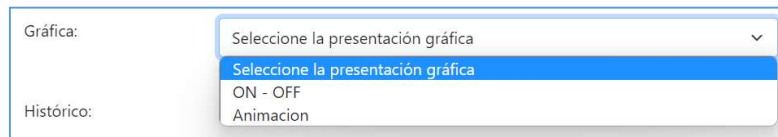


Figura 4.3.2. Campo: Gráfica.

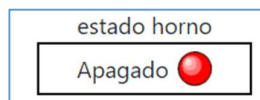


Figura 4.3.3. Gráfica: ON-OFF.

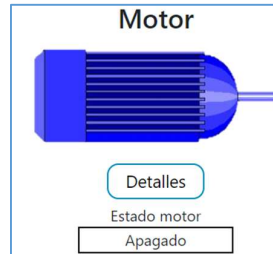


Figura 4.3.4. Gráfica: Animación, componente digital.

7. **Histórico:** Campo para que la variable se vaya registrando cada minuto en la base de datos.
8. **Grupo:** Selección de grupo al que se le quiere clasificar a la variable.

4.4 Configuración de la pestaña de componentes analógicos y digitales

En la figura 4.4 se muestra como acceder a la pestaña de componentes analógicos, mientras que, en la 4.4.1 se muestra como acceder a la de digitales. Para el usuario estas pestañas son prácticamente iguales, sin embargo, para el programador son diferentes, ya que, la de analógicos al querer presentar una imagen estática su componente está

conformado solo por una imagen, mientras que, el componente digital presenta dos imágenes montadas, esto para dar el efecto de animación de las variables digitales, como se explicó en la sección anterior.

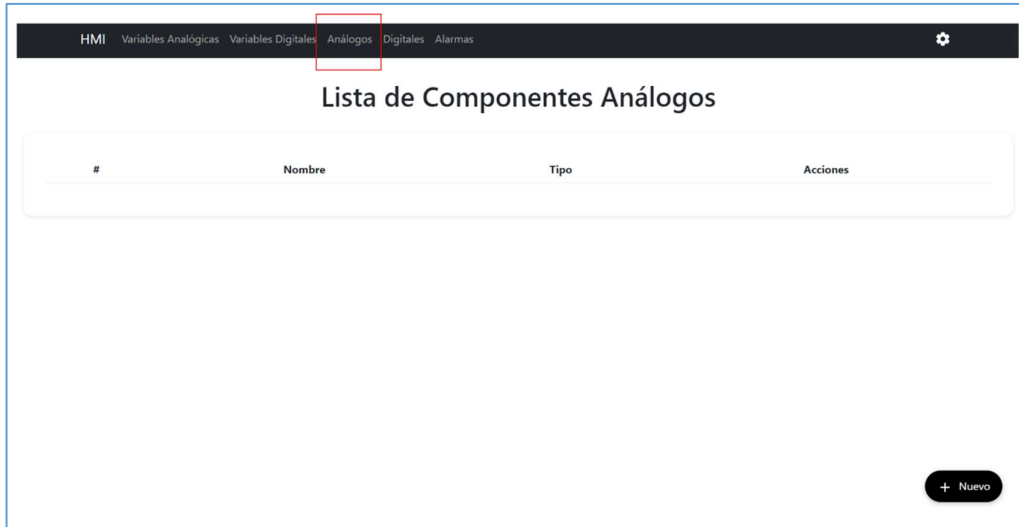


Figura 4.4. Pestaña de componentes analógicos.

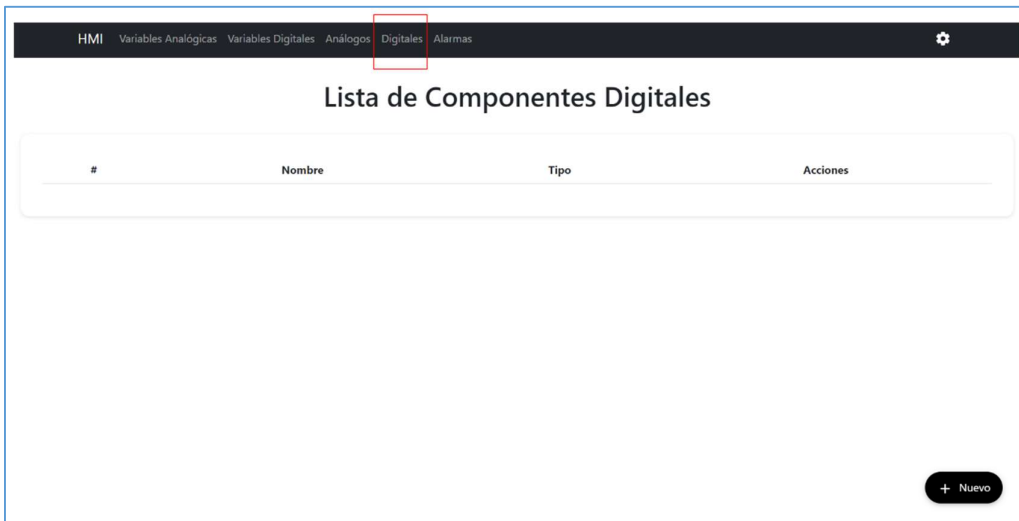


Figura 4.4.1. Pestaña de componentes digitales.

Si se desea crear un nuevo componente se da clic en el botón Nuevo, con esto se desplegará un formulario para obtener los datos del componente, esto se muestra en la figura 4.4.2.

Figura 4.4.2. Formulario de componentes analógicos y digitales.

En este caso, ambos componentes cuentan con los siguientes campos de:

5. **Nombre:** Nombre que se le va a dar al componente
6. **Tamaño:** Dimensión que va a tener la imagen del componente en la pestaña principal.
7. **Tipo:** Clasificación que va a tener el componente, para luego ser agrupado por sus semejantes.
8. **Variable a vincular:** La lista de variables analógicas como digitales se presentan al pulsar sus correspondientes botones. Se mostrarán solo las que se encuentren disponibles en ese momento, es decir, que no hayan sido vinculadas a otros componentes.

En el campo tipo se muestran diferentes casos para los componentes analógicos y digitales. Para los analógicos se han seleccionado los que se ven en la figura 4.4.3, para los digitales se han seleccionado los que se muestran en la figura 4.4.4.

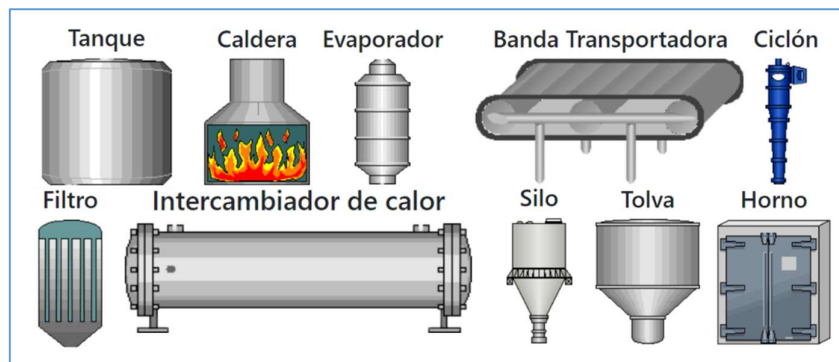


Figura 4.4.3. Componentes analógicos.

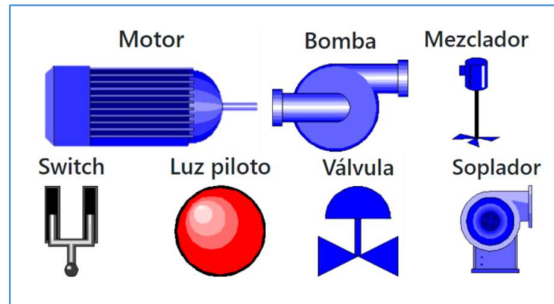


Figura 4.4.4. Componentes digitales.

Aquí se utiliza el campo tipo, para poder clasificar cada componente y así ubicarlos de forma conjunta en la página web principal. Además, estos componentes cuentan con botones de variables analógicas y digitales, aquí se puede realizar la vinculación de cada variable que se desea en cada componente, teniendo en cuenta que como máximo se aceptan 5 variables de cada tipo. En la figura 4.4.5 se presenta un ejemplo del componente analógico con variables analógicas. Las que se encuentran con el check en la sección de selección quiere decir que dicha variable ya se encuentra vinculada al componente elegido, caso contrario, serían las variables disponibles y que todavía no se han vinculado con ningún componente.

Lista de Variables Analógicas				
#	Nombre	Medida	Valor	Selección
1	velocidad horno	rpm	30.5	<input checked="" type="checkbox"/>
2	temperatura tostado	C	66	<input checked="" type="checkbox"/>
3	setpoint tempe tostado	C	0	<input checked="" type="checkbox"/>

Figura 4.4.5. Lista de variables analógicas.

En la figura 4.4.6 se presenta un ejemplo de componente digital con variables digitales vinculadas. En este caso, la variable estado compuerta presenta la selección en blanco, quiere decir que se encuentra disponible para ser vinculada en cualquier componente. Cabe recalcar que solo las variables disponibles se muestran en el componente seleccionado, además de las variables vinculadas al mismo.

Lista de Variables Digitales			
#	Nombre	Estado	Selección
1	Estado molino 2	1	<input checked="" type="checkbox"/>
1	estado compuerta	1	<input type="checkbox"/>

Figura 4.4.6. Lista de variables digitales.

Solo para el caso de los componentes, la información de los campos de nombre y tamaño se pueden modificar tanto en la parte de componentes como en la pestaña principal, ya que, estos parámetros son mostrados para una mejor orientación del usuario.

Si se desea crear un nuevo componente, se debe seleccionar el botón Nuevo, se llenan todos los campos requeridos, al dar clic en Guardar se validan que los campos no se encuentren vacíos y se procede a almacenar la información en la base de datos y se actualiza la tabla de componentes.

Si se desea modificar algún componente, se seleccionar el ícono de lápiz del componente correspondiente, se modifican los campos necesarios, al dar clic en Actualizar se corrobora que los campos no estén vacíos y se actualiza tanto la base de datos como la tabla de la pestaña.

Si se desea vincular o desvincular una o varias variables (máximo 5) en alguno de los componentes, se da clic en el ícono de lápiz se selecciona la variable analógica o digital, se da clic en el recuadro de la variable requerida y se pone en Guardar. Con esto se manda a refrescar la base de datos, así como la tabla.

Nota: Primero se debe crear el componente, luego se regresa a la pestaña de componentes y se da clic en el lápiz de actualizar, de esta forma se podrán vincular las variables, no en el momento de la creación del componente.

Finalmente, si se desea eliminar algún componente se da clic en ícono del basurero, se confirma la acción por el usuario, en el caso de que el componente tenga variables vinculadas a él no se permite la eliminación, ya que, el usuario no debe tener ninguna variable vinculada a él, se presente un mensaje como se ve en la figura 4.4.7.

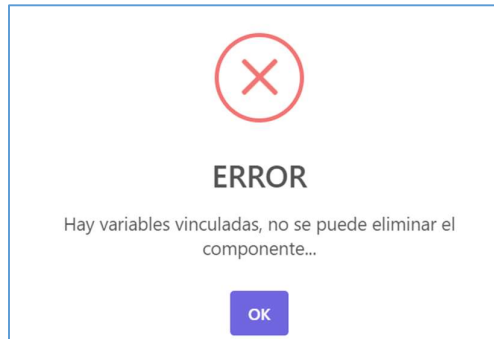


Figura 4.4.7. Mensaje de error al eliminar componente con variables vinculadas.

4.5 Configuración de la pestaña de alarmas

Para acceder a esta pestaña se dirige a la sección que se muestra en la figura 4.5. En esta tabla se mostrarán todas las alarmas que salten de las variables analógicas. Si se desea trabajar con las alarmas se debe realizar su configuración en la pestaña de variables analógicas, dirigirse a la sección 4.2.

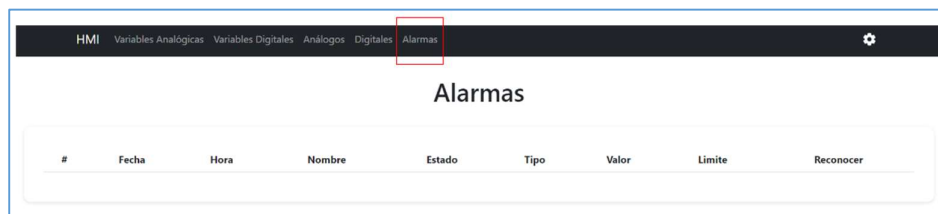


Figura 4.5. Pestaña de alarmas

Una vez que la alarma salte se irá apilando verticalmente en la tabla de alarmas, el orden es por fecha y como primera alarma se presenta la más actual. En la figura 4.5.1 se muestra un ejemplo, se colocó un botón de reconocer, esto es para que el usuario se dé cuenta que ya se han percatado del problema.

#	Fecha	Hora	Nombre	Estado	Tipo	Valor	Limite	Reconocer
1	viernes, 10 jun. 2022	11:40:00	temperatura tostado	sin reconocer	HA	84	80	<input type="button" value="Reconocer"/>
2	miércoles, 8 jun. 2022	20:39:00	temperatura tostado	sin reconocer	HA	84	80	<input type="button" value="Reconocer"/>
3	domingo, 5 jun. 2022	14:40:00	temperatura tostado	Reconocida	HHA	95	90	
4	lunes, 30 may. 2022	12:27:00	velocidad molinos	Reconocida	HA	501	500	
5	lunes, 30 may. 2022	12:24:00	nivel	Reconocida	HHA	99	95	

Figura 4.5.1. Tabla de alarmas

Ha pesar de que el problema con dicha variable se reconozca o deje de existir, se seguirán almacenando las alarmas en la base de datos, es decir, el sistema se ha creado como un registro de alarmas, en donde la alarma perdura en la tabla de alarmas mientras la variable analógica se encuentre creada, si esta variable se elimina, por ende, la alarma también desaparece del registro.

Mientras la alarma perdure se presentará un mensaje en la pestaña principal (HMI), en donde se muestra el nombre de la variable con la que saltó la alarma y el límite de la alarma por la cual salto. Un ejemplo se ve en la figura 4.5.2.

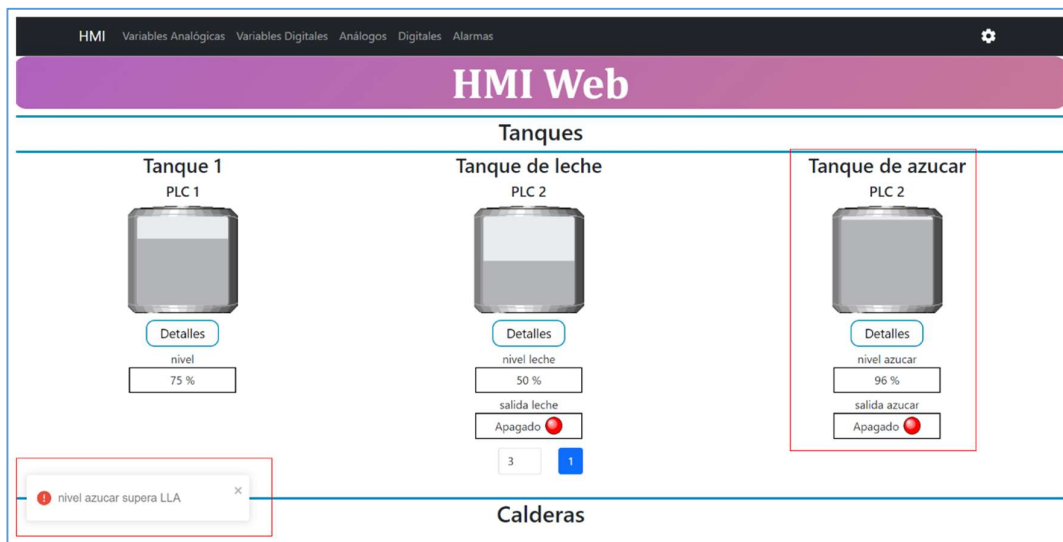


Figura 4.5.2. Mensaje de alarma en la pestaña principal.

4.6 Configuración de la pestaña de históricos

El acceso a esta pestaña es por medio de las pestañas de variables analógicas o digitales, siempre y cuando estén creadas y las mismas se hayan vinculado algún grupo. Esto se aprecia en la figura 4.6. Al dar clic al botón amarillo con el nombre de histórico se podrán visualizar las gráficas.

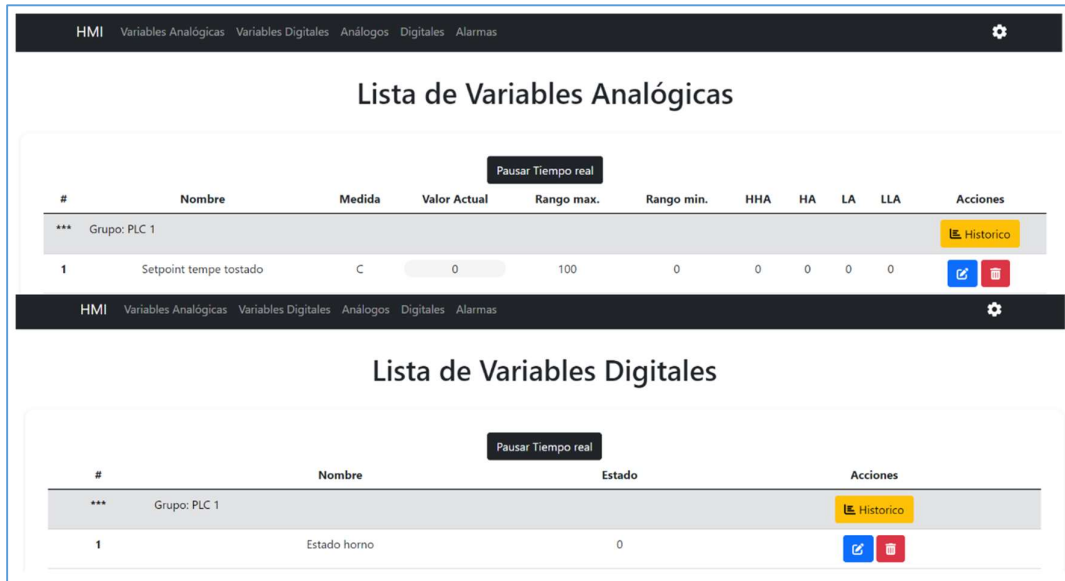


Figura 4.6. Acceso a pestaña de históricos.

Para ir registrando los históricos, primero se debe seleccionar este campo en cada una de las variables en sus respectivas pestañas, revisar sección 4.2, ya sean analógicas o digitales. Las gráficas se han ido agrupando según el grupo de la variable, es decir, en cada grupo se presenta una gráfica global de todas las variables que se encuentran dentro del grupo, solo sí, las variables han sido guardadas como históricos en su configuración. Sin embargo, si no desea ver todas las variables que pertenecen al grupo, se da clic dentro del recuadro que contiene el color de la variable, de esta forma se le tacha a la variable y se la elimina de la gráfica global. Ver figura 4.6.1.



Figura 4.6.1. Pestaña de históricos.

Aquí también se puede visualizar el histórico el día que seleccione el usuario, siempre y cuando la variable a monitorear haya sido guardada como histórico y el servidor haya estado activado en dicha fecha.

Dado que los registros históricos se guardan minuto a minuto en la base de datos, se ha colocado un botón de refrescar, de esta forma se presentarán los datos actuales de las variables.

4.7 Configuración de la pestaña de inicio

4.7.1 Configuración de la pestaña con mesa de trabajo estática

Para acceder a esta pestaña se da clic en la parte de HMI como se muestra en la figura 4.7.1. Para tener la vista de esta mesa de trabajo, revisar la configuración que se muestra en la sección 4.1.1.

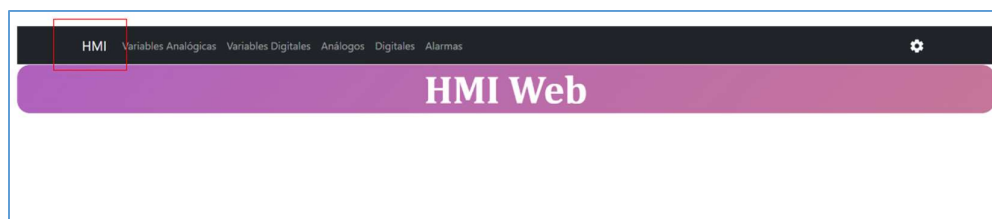


Figura 4.7.1. Pestaña de inicio.

Esta pestaña en sí no tiene una configuración de la página, ya que, los componentes se van mostrando en un orden preestablecido por el programador. Sin embargo, si se desea modificar el nombre o el tamaño de cada uno de los componentes, se da clic en el botón Detalles del componente deseado, como se muestra en la figura 4.7.1.1 se despliega un modal con esta información, se elige la información a modificar y se da clic en guardar. Se presenta un mensaje de confirmación y el cambio se presenta al instante en la pestaña principal.

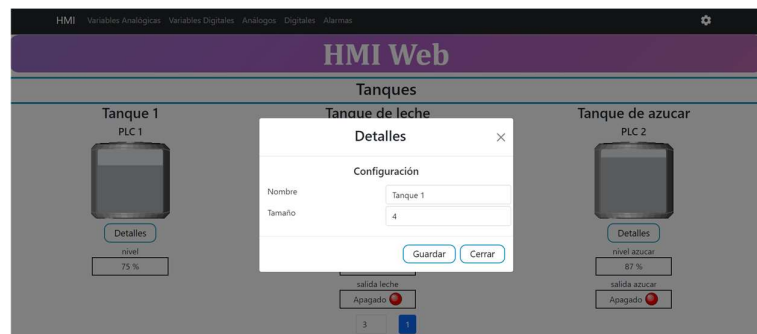


Figura 4.7.1.1. Detalles de configuración.

4.7.2 Configuración de la pestaña con mesa de trabajo dinámica

Revisar la configuración de la sección 4.1.1 para ingresar a la vista de esta pestaña, el acceso es el mismo que se muestra en la figura 4.7.1. Si es la primera vez que se tiene el acceso a esta página web y recién se han creado todos los componentes y variables, se tiene una primera vista como se muestra en la figura 4.7.2.

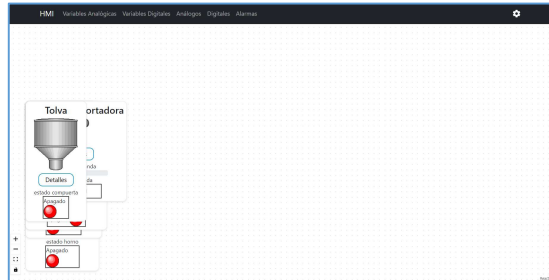


Figura 4.7.2. Mesa de trabajo dinámica, primera vista.

En esta mesa de trabajo cada componente puede ser arrastrado y colocado en cualquier posición de la página web que desee el usuario. Aquí se tiene la capacidad de bloquear la pantalla en la que se está trabajando, es decir, se queda guardado la ubicación de cada componente que fue movido, esto se logra si se da clic en el candado que se muestra en la figura 4.7.2.1. De igual forma se tienen opciones para hacer zoom a la página web. Un ejemplo del proceso de chocolate se presenta en la figura 4.7.2.2.



Figura 4.7.2.1. Configuración de mesa dinámica.

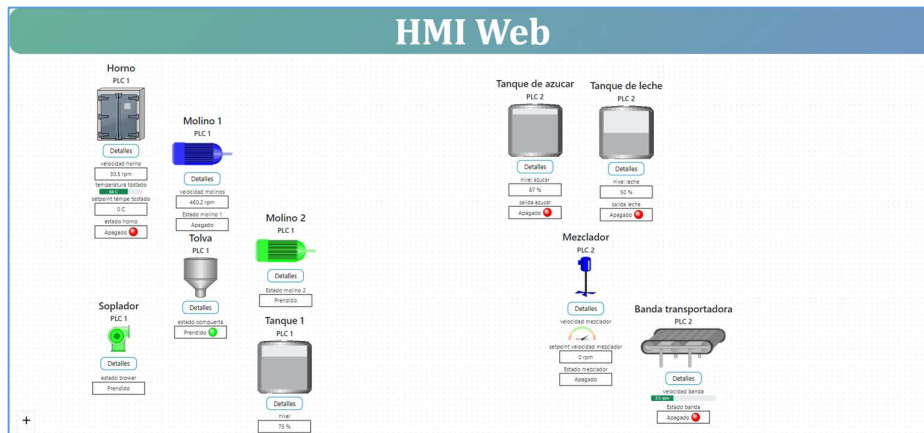


Figura 4.7.2.2. Mesa de trabajo dinámica.