



RIDUNAJ
Repositorio Institucional
Digital UNAJ



Publicaciones Científicas

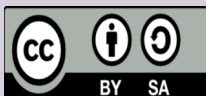
Yeison Harley Mancipe Malagón

Desarrollo de Aplicación para Monitorear el Estado General de las Plantas de Tratamiento de Agua

2024

Evento: III Encuentro Latinoamericano de Experiencias Universitarias

Red internacional de Cooperación Académica



Esta obra está bajo una Licencia Creative Commons.

Atribución – Compartir igual 4.0

<https://creativecommons.org/licenses/by-sa/4.0/>

Documento descargado de RID - UNAJ Repositorio Institucional Digital de la Universidad Nacional Arturo Jauretche

Cita recomendada:

Mancipe Malagón, Y. H. (16-17 de mayo 2024). Desarrollo de Aplicación para Monitorear el Estado General de las Plantas de Tratamiento de Agua [Ponencia]. III Encuentro Latinoamericano de Experiencias Universitarias, Red internacional de Cooperación Académica. <https://rid.unaj.edu.ar/handle/123456789/3336>

3er **Encuentro** *Latinoamericano* *de Experiencias* **Universitarias**

Academia, Investigación y Proyección Social



• RED INTERNACIONAL DE COOPERACIÓN ACADÉMICA •

Desarrollo de Aplicación para Monitorear el Estado General de las Plantas de Tratamiento de Agua

Yeison Harley Mancipe Malagón

yeison.mancipe@est.iudigital.edu.co

Resumen

En este proyecto, se llevará a cabo una investigación sobre las plantas de tratamiento de agua automatizadas mediante sistemas de control, centrándonos en la comunicación que realizan los sistemas automatizados. Estos sistemas utilizan el PLC para llevar a cabo actividades repetitivas durante el proceso de purificación del agua. Abordaremos los sistemas SCADA, PLC y las diversas formas de lectura que se aplican a este tipo de información en la industria. Además, exploraremos cómo estos datos pueden ser transferidos a un computador. Toda esta información nos proporcionará las herramientas necesarias para dar los primeros pasos en el desarrollo de una aplicación que permitirá monitorear los parámetros críticos, como la medida del pH, el cloro y los niveles de agua de los tanques para el proceso de purificación del agua.

Palabras Clave: plc, scada, plantas de tratamiento de agua, aplicación y parámetros

Objetivo Principal:

- Desarrollar una aplicación que muestre información sobre los niveles de pH, cloro e indique el nivel de agua de la planta de tratamiento de agua tanto de los tanque de entrada como de salida. además, que permita notificar sobre posibles fallas en los instrumento o equipos. Como lo son indicaciones del funcionamiento de una bomba en llenado de tanque y este tome más tiempo de lo necesario o nunca se termine de llenar o instrumentos como el medidor de PH que no toma una lectura porque la sonda no se encuentra totalmente sumergida.

Objetivos Específicos:

- El objetivo es identificar las variables clave que deben ser monitoreadas para detectar posibles fallas en la planta de tratamiento de agua. Estas variables pueden incluir caudal, presión, nivel de tanques, calidad del agua, entre otras.
- Investigaremos cómo se establece la comunicación entre el sistema de automatización SCADA (Supervisory Control and Data Acquisition) y el PLC (Controlador Lógico Programable) con el computador. Analizaremos qué información proporcionan y cómo se puede leer posteriormente en un lenguaje de programación.
- Definiremos el método o lenguaje de programación que utilizaremos para desarrollar la interfaz gráfica. Esta interfaz mostrará datos relevantes de la planta, como gráficos, tablas y alarmas.
- Implementar una forma interactiva en la aplicación para presentar las variables de manera cómoda y fácil de entender para los usuarios. Esto facilitará su comprensión y uso.
- Evaluar la aplicación siguiendo las normas ISO 27000 para garantizar un nivel adecuado de seguridad. Esto ayudará a prevenir posibles robos de información y protegerá los datos críticos de la planta.

Introducción

Las plantas de tratamiento de agua desempeñan un papel crucial al abastecer tanto a áreas rurales como urbanas, proporcionando hasta 5000 L/s de agua. En caso de errores, como desequilibrios de pH o presencia de elementos tóxicos, o incluso fallas en los equipos, es vital contar con información en

tiempo real para corregirlos rápidamente. Por ello, se busca desarrollar una aplicación móvil que alerte sobre posibles fallas.

El desafío radica en obtener información en tiempo real del PLC. Por sí solo este no está diseñado para enviar información en tiempo real; su función principal es ejecutar órdenes a los equipos. Por lo tanto, es necesario implementar un sistema de información que permita el monitoreo constante y la notificación de errores.

Esto inclusive es un punto de partida una posterior investigación, pero nosotros nos enfocaremos en la primera.

1. Acceso desde el móvil: la idea principal es que esta información sea accesible desde dispositivos móviles. Así, los operadores pueden recibir notificaciones y acceder a la información en cualquier momento.
2. Planificación de mantenimiento: Con el uso de las notificaciones programar mantenimientos preventivos para programar mantenimientos o calibración de equipos. De esta manera, se optimiza el tiempo.
3. Registro Histórico: Almacenar la información permitiendo un análisis de datos para una revisión posterior y un análisis de incidentes.

Plantas de Tratamiento de Agua

Según (Arenas,2017) en Colombia se cuenta con numerosas plantas para el tratamiento de agua potable (PTAP), la mayoría de gran capacidad que van de 200 a 5.000 L/s, estas manejan sistemas de control para garantizar condiciones de alta calidad en el proceso final donde el agua es apta para el

consumo humano, generando economía y eficiencia en su producción. Las plantas de tratamiento de pequeña (9 a 99 L/s) y mediana capacidad (100 a 199 L/s).

Estas plantas tienen varias etapas en el procesamiento de agua, por lo que requieren equipos que trabajarán en conjunto para aprovechar y mejorar la eficiencia en el tratamiento del agua. En la siguiente ilustración (Figura 17) se presenta sus etapas. Cabe resaltar que en cada proceso se cuenta con equipos que pueden ser manejados de forma automática (mediante un sistema de control) o de forma manual por medio de un operador capacitado.

En cada etapa, hay varios equipos, como bombas, electroválvulas, controladores de pH y cloro, bombas dosificadoras, sensores de nivel medidores de presión, medidores de caudal, entre otros elementos. Todo depende de la capacidad de la planta y de sus necesidades.

Figura 17. Proceso de una planta de tratamiento de agua potable



Fuente: (Rojas, 2017) fecha de consulta 04/04/2024

Sistemas de Control

Anteriormente hablábamos de las plantas son manejadas por sistemas de control.

El sistema de control es un conjunto de elementos (sensores, actuadores y controladores) conectados por un PLC. Este se denomina controlador, se encargará de dar indicaciones a cada uno de los equipos que ejecuta una determinada acción. A los que ejecutan estas acciones se les denominan actuadores. El PLC trabaja en conjunto con los sensores, que le dan información ya sea sobre el nivel del agua o sobre si hay un aumento en la presión en la tubería. Todo esto le ayudara a mantener un control sobre el proceso en todo momento.

(Hernandez,2019) La tecnología para la automatización y la comunicación industrial de los diversos componentes que forman las Redes de Control de Procesos se caracteriza por la interconexión física entre módulos de entrada/salida y lógicas cableadas. El desarrollo de los buses de campo permite aumentar la capacidad de comunicación y de conexión entre los diferentes elementos de campo, como pueden ser sensores, actuadores, controladores...

Estos sistemas pueden trabajar durante horas sin llegar a equivocarse, pero como todo, hay ciertas limitaciones, y este no es una excepción. Estos sistemas de control no tienen la capacidad de comunicar la información que ellos reciben. Además, hay casos en los que se requiere la intervención de un operario para poder evaluar si es necesario corregir algún proceso o si ha habido alguna falla de equipo o un inconveniente que requiera que el PLC se detenga en lugar de seguir funcionando. Sin embargo, el PLC no

comunica ninguna información relevante al operario, por lo que se necesita de un sistema de supervisión que le permita mostrar los datos al operador. Este sistema se le denomina SCADA.

El sistema SCADA es el supervisor del PLC. Este sistema se encargará de tomar los datos de la planta y mostrarlos en una estación central, que en nuestro caso sería el tablero de control ubicado junto a la planta de agua. Además, este sistema actúa como un comunicador entre el operario y la maquina por lo que este presenta la información mediante una interfaz configurable. También lleva un control sobre los equipos que se disponen en la planta.

(Hernandez,2019) Un SCADA es un sistema para la supervisión, control y adquisición de datos de una planta o proceso, utilizando las herramientas de comunicación correspondientes en cada caso [7]. La comunicación se realiza mediante una estación central (MTU Master Terminal Unit), que será la encargada de enviar las órdenes correspondientes a una o varias unidades remotas (RTUs, Receptor Terminal Unit) para que éstas realicen el control y la adquisición de datos demandada por la estación central. El nombre SCADA proviene de las siglas Supervisory Control And Data Acquisition.

Para organizar el orden de importancia que tiene el sistema de control, se mostrará la siguiente imagen (Figura 2). En ella, se representa en la punta de la pirámide al operador, quien estará encargado de tomar decisiones. Por ejemplo, decidir si es necesario detener la planta debido a escasez de agua o si en el lote de agua actual existe una concentración anormal de algún elemento toxico que la planta. También evaluara si algún equipo debe detenerse sin afectar todo el proceso de purificación de la planta.

En el segundo puesto de la imagen (Figura 18) se encuentra SCADA. Como se mencionó anteriormente, SCADA actúa como un comunicador entre el operario y el PLC. Su relevancia radica en que permite tener un seguimiento sobre el estado actual de la planta.

Figura 18 Pirámide de Automatización



Fuente: (Hernández, 2019) fecha de consulta 04/04/2024

El inconveniente con este sistema SCADA es que este sistema muestra la información, pero no la comunica con los demás sistemas. Por lo que, si deseamos tener un histórico de datos de información relevante, ya sea el pH del agua, la concentración de cloro o si se llegó a activar alguna alarma en el proceso (que el operador no lo noto en su momento), es necesario implementar un sistema de comunicación que permita almacenar y gestionar datos.

En la industria se desarrollaron varios sistemas de comunicación para abordar esta necesidad. Actualmente, el OPC (Protocolo de comunicación abierta) ha ganado relevancia debido a su facilidad de ajustarse a la cabina de control su interfaz intuitiva.

(Hernández ,2019) Para facilitar esta tarea de comunicación entre dispo-

sitivos, se ha desarrollado un protocolo de comunicaciones OPC (OLE for Proces Control), [1]. Este protocolo está basado en la tecnología OLE de Microsoft y después ha sido optimizada por la Fundación OPC.

En Tabla 4 se presentan las diferentes interfaces que se han desarrollado para satisfacer las necesidades de la industria. Todas están enfocadas en facilitar el intercambio de información ya sea en tiempo real, almacenamiento de datos o datos históricos.

Para nuestro objetivo nos enfocaremos en las interfaces OPC Data Access Specification(OPC DA) y OPC alarms & Events Specification(OPC A&E).

Tabla 5. Interfaz de comunicación OPC

COMUNICACIÓN OPC	
<i>Requerimientos de la Industria</i>	<i>OPC Foundation</i>
Datos de Tiempo Real	OPC Data Access Specification (OPC DA)
Datos Históricos	OPC Historical Data Access Specification (OPC HAD)
Datos de Alarmas y Eventos	OPC Alarms & Events Specification (OPC A&E)

Fuente: (Hernández, 2019) fecha de consulta 04/04/24

- OPC Data Access Specification La interfaz OPC DA proporciona la lectura, escritura y monitorización de variables del proceso en tiempo real. El principal uso de esta interfaz es captar los datos proporcionados por PLCs, DCSs, y otros dispositivos de control.
- OPC Alarms & Events Specification La interfaz OPC A&E habilita la recepción de notificación de eventos y alarmas. Estos eventos y alarmas están asociados a diferentes situaciones de emergencia que puedan ocurrir durante un proceso, la información se transmite en forma de notificaciones de estado o mensajes de confirmación y respuesta.

Finalmente, con la definición de estas interfaces podemos entender que no es sencillo acceder a la información generada por el sistema SCADA. Además, estas interfaces comunican la información dentro de la misma cabina de control. Sin embargo, no es posible acceder directamente a esta información por una computadora. Para obtener esos datos, se requiere otro software desarrollado por el mismo equipo que creo la interfaz OPC: el software OPC UA. Este software permite comunicar los datos hacia una computadora para poder visualizarlos.

(Hernandez,2019) El cliente OPC UA se utiliza como destino de datos, puede iniciar y controlar la comunicación establecida con el servidor OPC UA. La función principal que realiza el cliente es, crear la petición de comunicación de los datos que el usuario haya solicitado, para poder enviarla al servidor.

Ahora bien, si necesitamos que esta comunicación que proporciona el software OPC UA sea leído por algún lenguaje de programación, se requiere de otros programas que traduzcan esa información y permita una comunicación efectiva con algún lenguaje de programación como lo es Python JAVA o C++.

Estos son algunos softwares que permiten esa comunicación.

- dataFEED OPC UA Client - Softing
- UaExpert - Unified Automation
- OPC UA TOOLBOX - Matlab 2017b

ObjectViewer - Wonderware System Platform R2 2014 SP1 (Hernandez,2019).

Ahora que comprendemos cómo proceder con la conexión de los sistemas de automatización, surge la pregunta de cómo garantizar la seguridad de esta conexión. Dado que todo esto se gestionará a través de Internet y la información manejada es delicada, es crucial mantener un nivel de seguridad aceptable. Para lograrlo, es necesario seguir los parámetros establecidos por una norma del desarrollo de software.

Existe una norma directamente relacionada con la seguridad informática: la ISO 27000. Esta norma establece los requisitos mínimos para mejorar el Sistema de Gestión de la Seguridad de la Información (SGSI). Su objetivo es proteger la información sensible o privilegiada que, de ser obtenida por personas no autorizadas, podría representar un peligro para la planta de tratamiento de agua. (NORMAS ISO,2023)

Además, la ISO 27000 proporciona directrices sobre los controles de seguridad necesarios para prevenir y detectar posibles amenazas, tanto externas como internas. (NORMAS ISO,2023)

Es fundamental comprender que todas estas conexiones necesarias para el monitoreo de la planta de tratamiento de agua involucran varios elementos. Además de la propia aplicación, debemos considerar la conexión del software OPC UA al sistema de control y la interacción entre el software OPC UA y el software de comunicación mediante el lenguaje de programación. Además, es crucial mantener la aplicación conforme a las normas de seguridad ISO 27000. Para gestionar este proceso de conexión, existe una metodología que permite organizar y evaluar cuáles son las actividades críticas durante el desarrollo del aplicativo. Una de estas metodologías es SCRUM.

Scrum es un marco de gestión de proyectos de metodología ágil que ayuda a los equipos a estructurar y gestionar el trabajo mediante un conjunto de valores, principios y prácticas. Al igual que un equipo de rugby (de donde proviene su nombre) cuando entrena para un gran partido, el método scrum anima a los equipos a aprender a través de las experiencias, a autoorganizarse mientras abordan un problema y a reflexionar sobre sus victorias y derrotas para mejorar continuamente. Aunque se considera a menudo un marco de gestión de proyectos ágil, scrum incluye un conjunto de reuniones, herramientas y funciones que, de forma coordinada, ayudan a los equipos a estructurar y gestionar su trabajo. (Atlassian,2024)

La planificación de sprints es un evento en scrum que inicia el sprint. El objetivo de la planificación de sprints es definir lo que se puede entregar en el sprint y cómo se conseguirá ese trabajo. La planificación de sprints se hace en colaboración con todo el equipo de scrum. El sprint es un periodo definido de tiempo en el que se hace todo el trabajo. Sin embargo, antes de poder ponerte manos a la obra tienes que preparar el sprint. Tienes que decidir cuánto tiempo va a durar, el objetivo del sprint y por dónde vas a empezar. La sesión de planificación de sprints inicia el sprint definiendo el orden del día y el punto de atención. (Atlassian,2024)

De acuerdo con la metodología Scrum, en cada reunión se debe establecer un objetivo para las próximas metas a alcanzar. Además, es fundamental organizar las actividades según su relevancia. Durante cada Sprint, se debe generar un informe que detalle el progreso del producto (en este caso, la aplicación). Este informe indicará en qué etapa de desarrollo nos encontramos y señalará los elementos pendientes y los que no se lograron entre las reuniones del Sprint.

Para priorizar las actividades, se establecerá un orden de importancia. Es decir, se determinará qué actividades son críticas y requieren una solución rápida, y cuáles no representan un riesgo inmediato para el proyecto.

Es importante recordar que la duración de cada Sprint puede variar según la complejidad de las tareas. Además, la cantidad de Sprints puede aumentar si surgen cambios o si se requieren ajustes para satisfacer las necesidades del cliente.

Para complementar la información, se requiere presentar dos casos en donde se utilizaron otras herramientas para el acceso a los datos.

El primero consistió en crear una interfaz para una aplicación que controlaba la información que proporcionaba el software OPC UA. En el segundo caso, no se logró la conexión entre el software OPC UA y Python, lo que requirió el uso otras herramientas para poder manipular los datos del proyecto.

Interfaz Gráfica-Caso De Estudio Caso 1

(Matute, 2023)

Monitoreo Y Operación De Un Sistema Industrial De Servo Posicionamiento: Hmi Basada En Python Y Opc Ua:

En este trabajo de grado, se procedió a conectar un servomotor al sistema OPC UA. La idea principal del proyecto se enfocaba en crear una interfaz gráfica que le permitiera obtener información en tiempo real del servomotor, incluyendo datos de su posición, velocidad de rotación y el mismo consumo en Watts. Para diseñar la interfaz, se utilizó otro software llamado QT Designer, en lugar de programar directamente en lenguaje Python. Hay

que destacar que el diseño de QT Designer resulto muy intuitivo por lo que redujo el tiempo de práctica. Para la integración de esta interfaz se procedió a utilizar bibliotecas propias de Python. En la siguiente imagen (Figura 19) se presentan las bibliotecas usadas.

Figura 19. Biblioteca integración de interfaz

```
A0: Importación de Librerías

import sys
import time
import collections
import pyqtgraph as pg

from PyQt5.QtWidgets import QApplication, QMainWindow
from PyQt5.QtCore import QPropertyAnimation, QEasingCurve, QTimer, QTime
from PyQt5.QtGui import QColor
from PyQt5 import QtCore, QtWidgets, QtGui
from PyQt5.uic import loadUi

from opcua import Client
from opcua import ua
```

Fuente: (Matute, 2023) fecha de consulta 04/04/24

Cada una de las librerías es importante para la integración del sistema. Algunas de ellas se pueden reconocer directamente, como:

- from opcua import ua
- from opcua import Client

Estas dos líneas de código nos indican que se encargarán de conectar con el software OPC UA-CLIENT. Este software es crucial para establecer la comunicación entre los sistemas de automatización y el computador, permitiendo la lectura de archivos en tiempo real mediante el software OPC UA.

Si desarrollamos la aplicación en Python, estas líneas serán fundamentales al inicio del proceso de desarrollo.

Además, el autor menciona otras librerías que ya forman parte de la interfaz gráfica, como:

- `from PyQt5.QtWidgets import QApplication, QMainWindow`
- `from PyQt5.QtCore import QPropertyAnimation, QEasingCurve, QTimer, QTime`
- `from PyQt5.QtGui import QColor`
- `from PyQt5 import QtCore, QtWidgets, QtGui`
- `from PyQt5.uic import loadUi`

Esto sugiere que se necesitarán más librerías para configurar la interfaz gráfica. Por ejemplo, se requerirán gráficas para mostrar la velocidad y posición del servomotor. Además, la indicación de `QMainWindow` al inicio del código es necesaria para mostrar la información en una ventana principal.

Para exportar la información en tiempo real que proporcionaba OPC UA. Se necesita de un software que gestione esa entrega de información directamente a Python.

(Matute, 2023) Para acceder al servidor OPC UA a través de Python, se utiliza la aplicación UA Expert en su versión 1.6.3. UA Expert es un cliente OPC UA que permite visualizar en tiempo real todas las variables que se encuentran en el controlador lógico programable (PLC) en el servidor.

Ya definido el programa de conexión. Solo queda el código de integración con Python se puede apreciar (Figura 20)

Figura 20. Biblioteca integración de interfaz

A3: Visualización de la interfaz y vinculación del servidor OPC al programa

```
if __name__ == "__main__":
    client = Client("opc.tcp://192.168.0.1:4841")
    try:
        client.connect()
        app = QApplication(sys.argv)
        mi_app = SplashScreen()
        mi_app.show()
        sys.exit(app.exec_())
    finally:
        client.disconnect()
```

Fuente: (Matute, 2023) fecha de consulta 04/04/24

Si bien es muy importante ser cuidadoso en la integración de estos sistemas informáticos, ya que en este proyecto no se presentaron errores. El autor sugiere mantener actualizados los controladores del computador como los softwares. Esto es fundamental para evitar problemas relacionados con la conectividad, hasta incluso controladores del PLC que normalmente nunca se actualizan con frecuencia.

Interfaz Gráfica de Aplicaciones de Control Desarrolladas con Python Caso II

Este proyecto consistía en comunicar la información que proporcionaba el sistema OPC UA de una planta azucarera. El objetivo era desarrollar un programa que permitiera realizar simulaciones relacionadas con la producción o el consumo de energía para el próximo mes. Para lograr esto, era necesario que estos datos fueran accesibles desde un lenguaje de programación, en este caso Python.

Sin embargo, durante la fase de conexión, surgieron varios inconvenientes que impidieron la conexión como lo fue la conexión con el sistema SCADA con el software OPC UA indicando incompatibilidad y desconexión del sistema. Además, la información que llevaba a proporcionar tomada no siempre era tomada en tiempo real por lo que no eran confiables estos datos. Como resultado se optó por utilizar otra herramienta que proporcionaba el sistema SCADA (Wonderware).

(Hernandez,2019) Se detectaron problemas en la comunicación de datos de tipo "array" y en la configuración de gráficas para la representación de los valores de las variables. Por lo tanto, se decidió utilizar el paquete de integración MS Office que proporciona Wonderware, más concretamente, Microsoft Excel. Esta solución ha permitido introducir el escenario de precios y la llegada de remolacha, utilizando filas y columnas para simular los datos del tipo "array". Estos datos serán los que leerá el servidor de la optimización para realizar los cálculos necesarios. También se ha utilizado este programa para la representación gráfica de los valores aportados en el cálculo de la optimización. Con esto se ha conseguido visualizar la evolución temporal de las entradas del modelo que se comunicarán al servidor de la simulación.

En este caso, podemos comprender que a veces no es tan fácil conectar estos datos a un lenguaje de programación. Sin embargo, se entiende que existen otras herramientas disponibles para este propósito como un archivo Excel, que puede ser leído por los lenguajes de programación que pueden tomar datos específicos para su posterior uso.

METODOLOGÍA:

Para el desarrollo de la aplicación, hemos optado por utilizar la metodología SCRUM. Esta metodología nos permite agilizar y organizar las actividades durante el proceso de desarrollo del software. La idea principal es dividir el proyecto en partes más pequeñas, lo que facilita el control de los avances realizados en cada sprint. Además, SCRUM incorpora las opiniones del cliente durante el Sprint Review, lo que nos permite recibir retroalimentación sobre los avances realizados en la aplicación y abordar rápidamente los cambios necesarios para satisfacer las necesidades del cliente. En la Tabla 5, se detallan las actividades de cada sprint.

Tabla 6. Rint de actividades

	ACTIVIDADES
SPRINT 1	Identificar las variables clave que deben ser monitoreadas para detectar posibles fallas en la planta de tratamiento de agua
SPRINT 2	Establecer la comunicación entre el sistema de automatización y el software OPC UA.
SPRINT 3	Determinar cuál lenguaje de programación es más adecuado para desarrollar el software y la conexión con el software OPC UA. Establecer la comunicación entre el lenguaje de programación y el software OPC UA.
SPRINT 4	Definir el método o lenguaje de programación que utilizaremos para desarrollar la interfaz gráfica. Esta interfaz mostrará datos relevantes de la planta, como gráficos, tablas y alarmas.

SPRINT 5	Implementar una forma interactiva en la aplicación para presentar las variables de manera cómoda y fácil de entender para los usuarios. Esto facilitará su comprensión y uso.
SPRINT 6	Desarrollar el código de programación teniendo en cuenta la norma ISO 27000 de seguridad informática. Sprint review. Crear un protocolo de pruebas para el aplicativo.
SPRINT 7	1) Revisión de errores en el código desarrollado. 2) Sprint review. 3) Crear un video explicando el funcionamiento de la aplicación. 4) Implementar la aplicación en la planta de tratamiento de agua.
SPRINT 8	1) Revisar la información presentada en la aplicación. 2) Seguir el protocolo de pruebas. 3) Corrección de errores
SPRINT 9	Evaluar la aplicación siguiendo las normas ISO 27000 para garantizar un nivel adecuado de seguridad.
SPRINT 10	Presentar el programa finalizado a los clientes

Para aclarar, algunos Sprints pueden completarse en una semana, mientras que otros pueden extenderse entre dos y tres semanas de desarrollo. Por ejemplo, el Sprint 2 se enfoca en establecer la comunicación entre el sistema de automatización y el software OPC UA. Dado que no tenemos un profundo conocimiento de estos programas y considerando posibles componentes de conexión o incompatibilidades entre dispositivos, su implementación podría requerir más tiempo.

En el Sprint Review 6, evaluaremos los avances tanto en el diseño de la apli-

cación como en la corrección de errores de código. Presentaremos cada uno de los logros alcanzados. Además, siguiendo la metodología, elaboraremos un informe que documente lo discutido en la reunión. Esto podría influir en la cantidad de Sprints reflejados en la Tabla 2.

Es importante tener en cuenta que los cambios solicitados por el cliente pueden afectar la planificación. Por lo tanto, debemos incluir un Sprint adicional para verificar que los ajustes realizados cumplan con los requisitos del cliente. La metodología también establece que durante cada reunión se deben definir las actividades, y estas pueden requerir más Sprints según la complejidad del proyecto.

En resumen, la duración de cada Sprint puede variar según la complejidad, y la cantidad de Sprints puede aumentar debido a posibles errores o variables no consideradas inicialmente."

DESARROLLO:

Sprint 1

Para el Sprint 1 se realizó una tabla 3 en la que se presenta las variables con la que se realiza el proceso de potabilización de las plantas de tratamiento de agua. En la cual clasificaremos cada variable según su importancia (si es crítica o no), esto nos permitirá determinar si estas variables deben incluirse en la aplicación que vamos a desarrollar. Dando una descripción del

porque dicha variable es importante. También proporcionaremos observaciones sobre cómo podrían presentarse estas variables en la aplicación de una forma interactiva para que la facilitar la comprensión del usuario.

Tabla 7. Selección de variables críticas de las plantas de tratamiento de agua Sprint 1

Variables Planta	Crítica, Descripción.	Como podría presentarse en la aplicación.
Sensores de Nivel (alto -bajo)	Si, estos sensores son críticos, ya que cualquier error podría resultar en un desperdicio significativo de agua.	Mostrar un tanque de agua con la ubicación de los sensores de nivel. Representar visualmente el nivel de agua según la activación de cada sensor.
Electroválvula	Si, En caso de que se produzca un error, la electroválvula podría quedarse abierta, lo que resultaría en desperdicio de agua. Por otro lado, si permanece cerrada, el tanque podría quedar vacío y las bombas de agua podrían dañarse al funcionar en vacío, lo que dañaría las bombas.	Presentar esta información mediante una notificación de la apertura y cierre sin errores de la electroválvula.
Bombas dosificadoras	No, aunque lleguen a fallar, el proceso sigue manteniendo un control de calidad del agua. Esto se detectaría rápidamente mediante los sensores de pH y cloro	No aplica.
Bombas	No, El operario puede identificar rápidamente en el tablero de control cuando	No aplica.
Válvulas para sistemas de filtración.	Si, Si llegaran a fallar, es probable que el proceso de filtración del agua no se realice correctamente.	Para indicar el estado general de la válvula, se utilizará una bombilla. Si hay una falla, la bombilla se mostrará en color rojo; si la válvula funciona con normalidad, la bombilla será verde. Además, se enviará una notificación para informar sobre cualquier problema
Presostato	Si, El presostato es crucial, ya que indica la presión de la tubería. Si llegara a fallar una sección de la tubería, no solo afectaría su funcionamiento, sino también podría dañar otros equipos	Para indicar el estado general del presostato, utilizaremos una bombilla. Si hay una sobre presión, la bombilla se mostrará en rojo; si el presostato trabaja con normalidad, la bombilla será verde. Además, esta información se presentará mediante una notificación.

Sensor de PH	Si, El sensor de pH es fundamental para el control de calidad del agua.	En la pantalla principal, mostraremos la información del sensor de pH. Además, si el pH es alto o bajo, se generará una notificación para alertar al operario.
Sensor Cloro	Si, El sensor de Cloro es fundamental para el control de calidad del agua	En la pantalla principal, mostraremos la información del sensor de pH. Además, si el pH es alto, se generará una notificación para alertar al operario.
Sensor de Temperatura	Si, El sensor de Temperatura es fundamental para el control de calidad del agua.	En la pantalla principal, mostraremos la información del sensor de Cloro. Además, si el pH es alto o alto, se generará una notificación para alertar al operario.
Presostato diferencial	Si, ya que este instrumento indica la presión de la tubería y más específicamente en la presión de los filtros y lo que permite saber cuándo se debe cambiar un filtro.	Indicar mediante una bombilla el estado general del presostato, esta se presentará en rojo si hay una sobre presión y verde si trabaja con normalidad, además se presentará esta información por medio de una notificación.
Arrancador suave	Si, El arrancador suave tiene la función de encender las bombas y controlar su velocidad. Si llegara a trabajar en vacío, indicaría que las bombas están funcionando sin carga, lo que podría ser perjudicial	En la pantalla principal, mostraremos la velocidad de la bomba. Además, si las bombas están trabajando en vacío, se generará una notificación de alerta para informar al operario
PLC	El PLC supervisa todos los procesos de la planta de tratamiento de agua. Si se produce un error durante su funcionamiento, todos los procesos se detendrán	Indicar por medio de una notificación si el PLC de detuvo por algún error.

Cabe destacar que esta evaluación de variables nos da un panorama general de lo que debería ir en la aplicación. Sin embargo, es probable que en una posterior evaluación se puedan ir cambiando o agregando elementos. Esto podría deberse a que hay equipos que, por diseño, no deban estar

conectados al sistema automatizado o porque algunos equipos no permiten una conexión. Por lo tanto, lo que tenemos hasta ahora es un resultado preliminar.

Continuando con el cronograma presentado en la tabla 2 de la metodología Sprint, se espera que se realice una comunicación entre el sistema de automatización y el software OPC UA. Este proceso será un trabajo de prueba y error en el que se tendrá que verificar de manera física cómo se conectará el Software OPC UA con el Sistema SCADA. Durante esta fase, se explorarán diferentes configuraciones y se evaluará la interoperabilidad entre ambos sistemas. Además, se considerarán aspectos como la seguridad de la comunicación y la integridad de los datos transmitidos. Este paso es crucial para garantizar una comunicación eficiente y confiable entre los componentes automatizados. Una vez completada esta etapa, se podrá avanzar hacia la implementación y monitoreo continuo del sistema.

CONCLUSIONES

Las variables que logramos identificar en la planta nos permiten visualizar cómo debemos presentar esta información al usuario. Aunque se presenten soluciones, es probable que en la realidad no sean tan funcionales o fáciles de implementar. Por lo tanto, se requiere una revisión durante el desarrollo.

Referencias

Arenas Castaño, F. A., & Londoño Giraldo, W. H. (2017). Diseño de un sistema automatizado para una planta de tratamiento de agua potable.

Hernández Sánchez, S. (2019). Interfaz gráfica de aplicaciones de control desarrolladas con Python.

Matute Caguana, J. A., & Zhindon Sumba, M. G. (2023). Monitoreo y operación de un sistema industrial de servo posicionamiento: HMI basada en Python y OPC UA (Bachelor's tesis).

Atlassian. (s/f). Kanban. Recuperado el 29 de abril de 2024, de Atlassian website: <https://www.atlassian.com/es/agile/kanban>

Normas ISO para Desarrollo de Software: Mejora de Calidad. (2023, julio 15). Informática y Tecnología Digital. <https://informatecdigital.com/software/normas-iso-para-desarrollo-de-software-mejora-de-calidad/>