

Ana Carolina Coelho Robl

Desenvolvimento e integração de interfaces em linhas de produção automatizadas

Uberlândia, MG

2024

Ana Carolina Coelho Robl

Desenvolvimento e integração de interfaces em linhas de produção automatizadas

Trabalho de Conclusão de Curso da Engenharia de Controle e Automação da Universidade Federal de Uberlândia - UFU - Campus Santa Mônica, como requisito para a obtenção do título de Graduação em Engenharia de Controle e Automação.

Universidade Federal de Uberlândia - UFU
Faculdade de Engenharia Elétrica - FEELT

Orientador Prof. Dr. Renato Ferreira Fernandes Junior

Uberlândia, MG

2024

Ana Carolina Coelho Robl

Desenvolvimento e integração de interfaces em linhas de produção automatizadas

Trabalho de Conclusão de Curso da Engenharia de Controle e Automação da Universidade Federal de Uberlândia - UFU - Campus Santa Mônica, como requisito para a obtenção do título de Graduação em Engenharia de Controle e Automação.

Trabalho aprovado em 14 de Novembro de 2024.

COMISSÃO EXAMINADORA

Prof. Dr. Renato Ferreira Fernandes Junior
Orientador

Professor Dr. Renato Santos Carrijo
Membro Avaliador

**Professor Dr. Fábio Vincenzi Romualdo da
Silva**
Membro Avaliador

Uberlândia, MG
2024

Agradecimentos

Agradeço à minha mãe, Sandra Regina Coelho, por me apoiar, me inspirar, incentivar e mostrar que com garra e determinação tudo é possível.

Agradeço à minha irmã, Viviane Coelho Robl, me dar confiança, apoio e por me motivar durante toda esta jornada.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) por ter me concedido a bolsa de estudos BRAFITEC e que tornou possível o desenvolvimento deste projeto na França.

Agradeço à equipe da empresa Operamatrix, pelos conhecimentos, conselhos e orientações que fizeram com que a jornada de desenvolvimento deste projeto fosse a melhor possível.

Agradeço aos meus amigos por terem me ajudado, dado suporte e feito desta jornada uma experiência incrível e divertida.

Agradeço ao professor Renato Ferreira Fernandes Junior por todo o conhecimento que me foi ensinado, pela paciência e por ter aberto as portas do mundo da automação industrial para mim.

Por fim, agradeço a todos os professores da minha graduação com quem tive o prazer de interagir, trabalhar e aprender.

Resumo

Sistemas supervisórios são sistemas de controle e monitoramento em tempo real de processos industriais que utilizam os dados coletados de sensores e dispositivos de campo para a transmissão e tratamento dessas informações em um sistema central.

Este projeto busca descrever o desenvolvimento das interfaces de supervisão de uma linha de produção automatizada. O projeto, que será nomeado "Projeto Conhaque", trata do desenvolvimento das interfaces de supervisão de uma parte da linha de produção de conhaque, mais especificamente na parte de filtragem ao final do processo de produção da bebida.

Para o desenvolvimento destas interfaces, o programa Ignition SCADA foi utilizado. Com isto, um estudo sobre a forma de endereçamento de variáveis utilizada no programa CLP foi necessário para a verificação da maneira correta de inclusão destas variáveis no software de desenvolvimento das interfaces de supervisão. Foi então constatado que a inclusão dessas variáveis é feita de forma exaustiva, logo a criação de um código em Python que torne este processo mais eficaz e eficiente foi realizada e é detalhada neste documento.

Este documento também apresenta o resultado das interfaces SCADA que foram entregues ao operador. Estas interfaces foram adaptadas às necessidades de supervisão do processo de forma que o resultado obtido foram interfaces intuitivas e eficazes, ou seja, de fácil utilização e compreensão das informações contidas na mesma.

Palavras-chaves: Supervisão; Linhas de produção automatizadas; SCADA; Produção de Conhaque; Modbus TCP.

Abstract

Supervisory systems are real-time control and monitoring systems for industrial processes that use data collected from sensors and field devices to transmit and process this information in a central system.

This project aims to describe the development of supervisory interfaces for an automated production line. The project, which will be called the “Cognac Project”, deals with the development of supervisory interfaces for part of the cognac production line, specifically the filtering part at the end of the beverage production process.

The Ignition SCADA program was used to develop these interfaces. As a result, a study of how to address the variables used in the PLC program was necessary in order to verify the correct way to include these variables in the software used to develop the supervisory interfaces. It was then found that the inclusion of these variables is done exhaustively, so the creation of a code in Python that makes this process more effective and efficient was carried out and is detailed in this document.

This document also presents the result of the SCADA interfaces that were delivered to the operator. These interfaces were adapted to the needs of process supervision so that the result was intuitive and effective interfaces, i.e. easy to use and understand the information contained therein.

Key-words: Supervision; Automated production lines; SCADA; Cognac production; Modbus TCP.

Lista de ilustrações

Figura 1 – Vantagens da integração TO-TI	12
Figura 2 – Processo de produção do conhaque	15
Figura 3 – Processo de Filtragem Conhaque	16
Figura 4 – Piramide de Automação Industrial	18
Figura 5 – Exemplo de CLP	19
Figura 6 – Estrutura Modbus TCP/IP	21
Figura 7 – Exemplo de troca de dados no Modbus TCP	21
Figura 8 – Principais funções Modbus	22
Figura 9 – Comunicação OPC UA	24
Figura 10 – Exemplo de Supervisão	25
Figura 11 – Estrutura Ignition	26
Figura 12 – Principais vantagens Ignition	27
Figura 13 – Fluxograma do sistema conhaque	31
Figura 14 – Esquema Hidráulico - Recepção do Conhaque Bruto	31
Figura 15 – Esquema Hidráulico - Filtro quente	33
Figura 16 – Esquema Hidráulico - Resfriamento	35
Figura 17 – Esquema Hidráulico - Filtros á terra	37
Figura 18 – Esquema Hidráulico - Rede de Escoamento	38
Figura 19 – Esquema Hidráulico - Aquecimento	40
Figura 20 – Esquema Hidráulico - Expedição do conhaque filtrado quente	41
Figura 21 – Arquitetura do projeto	42
Figura 22 – Criação Alarm Journal Profile	43
Figura 23 – Endereçamento de variáveis	46
Figura 24 – Tratamento de variáveis	49
Figura 25 – Exemplo de UDT criada a partir do .json	50
Figura 26 – Exemplo de uma Tag do tipo de uma UDT criada a partir do .json	50
Figura 27 – Interface de supervisão - Sistema Completo	51
Figura 28 – Interface de supervisão - Tanque	51
Figura 29 – Interface de supervisão - Pop-up Tanque	52
Figura 30 – Interface de supervisão - Válvulas	52
Figura 31 – Interface de supervisão - Pop-up Válvulas	53
Figura 32 – Interface de supervisão - Pop-up Bombas	53
Figura 33 – Interface de supervisão - Regulação das Bombas	54
Figura 34 – Interface de supervisão - Histórico de Alarmes	55
Figura 35 – Interface de supervisão implementada na linha de produção	55

Lista de tabelas

Tabela 1 – Equipamentos utilizados na supervisão - Recepção conhaque bruto . . .	32
Tabela 2 – Equipamentos utilizados na supervisão - Filtro quente	34
Tabela 3 – Equipamentos utilizados na supervisão - Resfriamento	36
Tabela 4 – Equipamentos utilizados na supervisão - Filtros à terra	38
Tabela 5 – Equipamentos utilizados na supervisão - Rede de escoamento	39
Tabela 6 – Equipamentos utilizados na supervisão - Aquecimento	40
Tabela 7 – Equipamentos utilizados na supervisão - Expedição do conhaque filtrado quente	42

Códigos

3.1	Arquivo .xsy mapeamento de variáveis	43
3.2	Arquivo .json criação das UDTs	47
3.3	Arquivo .json criação das tags	48

Lista de abreviaturas e siglas

CLP	Controlador Lógico Programável
SCADA	Supervisory Control and Data Acquisition
TCP/IP	Transmission Control Protocol/Internet Protocol
TI	Tecnologia da Informação
TO	Tecnologia da Operação
UDT	User Defined Type

Sumário

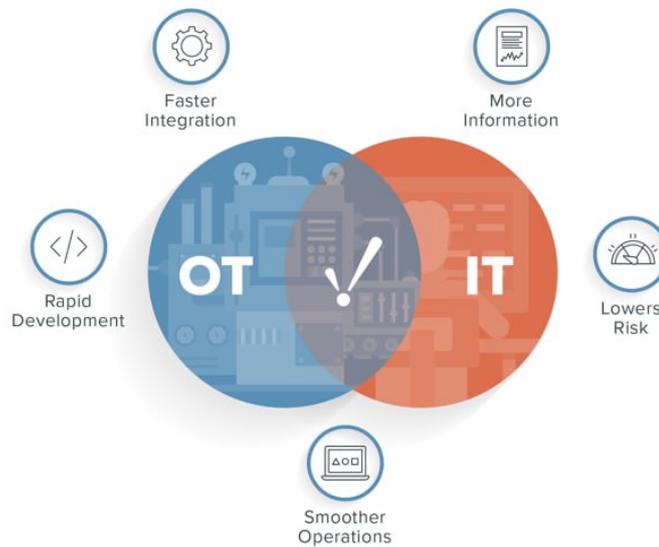
1	INTRODUÇÃO	12
1.1	Justificativas	13
1.2	Objetivos	14
1.2.1	Objetivos Específicos	14
2	REFERENCIAIS TEÓRICOS	15
2.1	Processo de Produção do Conhaque	15
2.1.1	Processo de Filtração do Conhaque	15
2.1.1.1	Recepção do Conhaque	16
2.1.1.2	Filtro quente	16
2.1.1.3	Resfriamento	16
2.1.1.4	Filtros à Terra	17
2.1.1.5	Rede de escoamento	17
2.1.1.6	Aquecimento	17
2.1.1.7	Expedição do conhaque filtrado	17
2.2	Automação Industrial	17
2.2.1	Nível de Controle e de Dispositivo de campo	18
2.2.2	Redes Industriais	19
2.2.2.1	Protocolo Modbus TCP/IP	20
2.2.2.2	Protocolo OPC UA	23
2.2.3	Nível de Supervisão e Gestão da Planta	24
2.2.3.1	Ignition SCADA	25
2.2.3.1.1	Vision e Perspective	27
2.2.3.1.2	Tags	27
2.2.3.1.3	UDT - User Defined Types	28
2.3	Linguagens	28
2.3.1	Javascript + CSS	28
2.3.2	Python	28
3	METODOLOGIA E DESENVOLVIMENTO	30
3.1	Características do Sistema	30
3.1.1	Recepção conhaque bruto	31
3.1.2	Filtro quente	33
3.1.3	Resfriamento	34
3.1.4	Filtros à terra	36
3.1.5	Rede de escoamento	38

3.1.6	Aquecimento	39
3.1.7	Expedição do conhaque filtrado quente	41
3.2	Desenvolvimento das Interfaces	42
4	RESULTADOS E DISCUSSÕES	50
5	CONCLUSÃO	56
	REFERÊNCIAS BIBLIOGRÁFICAS	57

1 Introdução

O crescimento da exigência do mercado nos níveis de produtividade, criou uma necessidade de aumentar o nível tecnológico dos equipamentos, e com isso iniciou-se o estudo da informática industrial que apresentou grandes mudanças nos processos de manufatura e produção. Sendo uma área que combina a tecnologia da informação (TI) com a tecnologia de operação (TO) e se propõe a otimizar e automatizar o meio industrial. A Figura 1 ilustra alguns dos benefícios dessa integração.

Figura 1 – Vantagens da integração TO-TI



Fonte: (AUTIC, 2024).

Assim sendo, com o avanço da Indústria 4.0, a integração de sistemas de controle e supervisão, como SCADA, com tecnologias de comunicação robustas e eficientes, como Modbus TCP, tornou-se essencial para garantir a eficiência, a segurança e a continuidade das operações industriais.

Os sistemas SCADA são fundamentais para o monitoramento e controle dos processos da indústria em tempo real. Eles permitem a coleta, o processamento e a visualização de dados provenientes de sensores e outros equipamentos, possibilitando aos operadores uma visão completa do processo produtivo. Além disto, este sistema, quando integrado com o protocolo de comunicação correto, possui como outra vantagem sua confiabilidade e robustez (DANEELS; SALTER, 1999), características primordiais para o bom desenvolvimento dos processos industriais.

Associados à supervisão, os protocolos de comunicação são essenciais para a transmissão de dados entre os elementos e, portanto, para o bom funcionamento das redes industriais. Estes protocolos são os responsáveis pelo endereçamento das variáveis,

que representam os dispositivos.

Cada protocolo de comunicação industrial possui suas próprias regras e convenções para identificar e acessar as variáveis nos dispositivos conectados. Ou seja, o endereçamento de variáveis varia de acordo com o protocolo de comunicação utilizado. Assim sendo, outro ponto crucial para o desenvolvimento das interfaces SCADA é a garantia da ligação dos dispositivos com a interface, e para isto é preciso entender o protocolo de comunicação aplicado no projeto.

Portanto, é necessário também o estudo do protocolo de comunicação utilizado no ambiente industrial para trocar dados de forma precisa e eficiente na rede industrial. Por exemplo, entender como associar o software utilizado no desenvolvimento da interface de supervisão e as variáveis oriundas do CLP é algo primordial.

A produção de bebidas é uma área muito complexa e minuciosa, assim sendo, qualquer mínimo detalhe pode estragar a bebida final. Por exemplo, no processo de produção de cerveja, a temperatura de fermentação da bebida é algo de extrema importância, onde um desvio de apenas 1°C pode alterar significativamente o perfil de sabor (BRIGGS, 2004). Já para a produção de conhaque, por exemplo, na etapa de filtração da bebida, a temperatura de filtração é um parâmetro essencial, pois caso a bebida não seja filtrada na temperatura ideal, resíduos indesejados podem não ser filtrados do produto final.

A integração de interfaces de supervisão no processo de produção de bebidas proporciona benefícios consideráveis em termos de eficiência operacional, qualidade do produto e manutenção preventiva (ARAÚJO et al., 2023). Isso, pois o monitoramento em tempo real dos dispositivos e variáveis do processo permite uma visibilidade prática e abrangente de cada etapa, o que permite a detecção imediata de falhas, possibilitando ao operador a aplicação rápida de médias corretivas, o que por sua vez garante uma produção mais consistente (ARAÚJO et al., 2023).

1.1 Justificativas

Em ambientes industriais de grande porte como o apresentado neste trabalho, supervisionar e controlar os processos de forma eficaz são cruciais para a garantia de qualidade do produto final.

Por exemplo, para a produção de bebidas, a variação de temperatura anormal pode afetar a qualidade do produto final. Além disto, caso haja um vazamento dos tanques dessas bebidas, acidentes no campo industrial podem ocorrer.

Assim sendo, um sistema SCADA intuitivo que permita uma interação entre o operador e o sistema de forma clara e eficiente é a chave para detectar erros e anomalias em tempo hábil. Possibilitando a adoção ações corretivas antes que o problema se amplie,

evitando a degradação do produto e conseqüentemente perdas financeiras significativas.

Em resumo, tal conceito é fundamental, visto que a correta implementação desses sistemas pode ser a diferença entre uma produção eficiente e um processo repleto de desperdícios e prejuízos.

Este trabalho tem como principal motivação a aplicação dos conhecimentos adquiridos através de matérias do curso de Engenharia de Controle e Automação como Informática Industrial, Redes Industriais, Programação Script, entre outras.

1.2 Objetivos

O objetivo deste trabalho é detalhar o processo de desenvolvimento de interfaces de supervisão SCADA no ambiente industrial, mais especificamente para a produção de conhaque. Este sistema foi projetado e desenvolvido entre fevereiro e julho de 2024 na França. Ao longo destes estudos, também serão detalhadas as estratégias escolhidas para simplificar o desenvolvimento destas interfaces, mais especificamente o endereçamento de variáveis dentro do programa de desenvolvimento de interfaces, e os resultados obtidos.

1.2.1 Objetivos Específicos

Como dito anteriormente este projeto foi desenvolvido para a linha de produção de conhaque, neste contexto o objetivo principal para este projeto foi desenvolver a interface de supervisão do processo de filtração do conhaque. Mais especificamente, os objetivos eram os seguintes:

- Desenvolvimento da interface de supervisão da fábrica de acordo com os requisitos do operador.
- Testes no local para verificar o bom funcionamento.

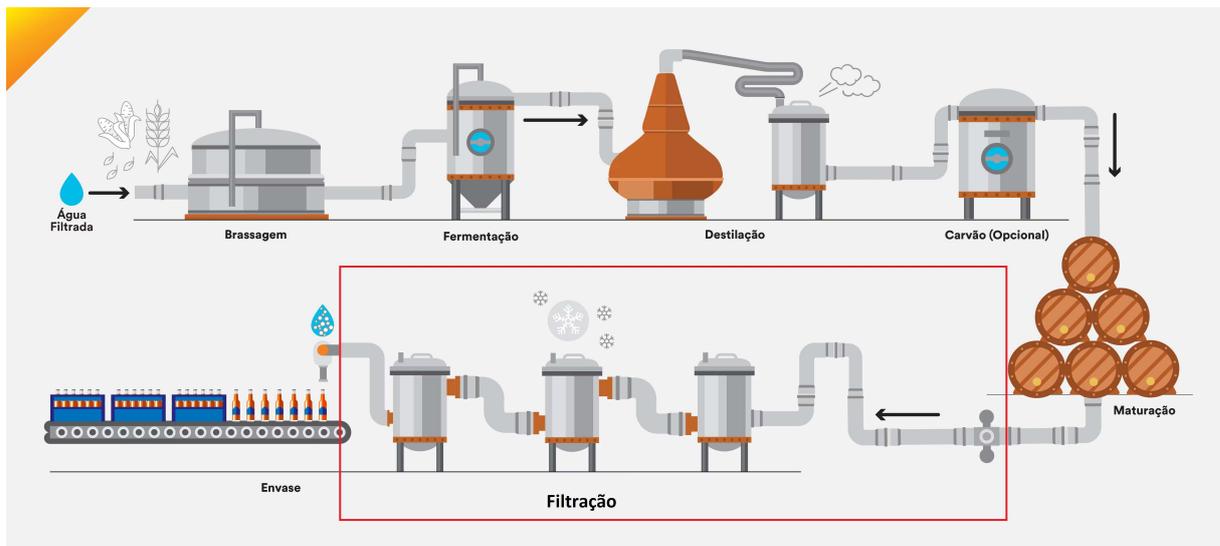
2 Referenciais Teóricos

Neste capítulo serão descritos todos os conceitos necessários para melhor compreensão deste trabalho. Primeiramente será apresentado o conceito geral sobre o processo de produção de conhaque, e posteriormente será abordada toda a teoria de automação industrial relativa ao processo de produção de bebidas.

2.1 Processo de Produção do Conhaque

A produção do conhaque que chega ao consumidor final passa por vários processos, desde a escolha das uvas, até a fermentação, destilação, segunda destilação, envelhecimento/maturação, filtração e embalagem. Um fluxograma resumindo os processos para a produção desta bebida pode ser observado na Figura 2.

Figura 2 – Processo de produção do conhaque



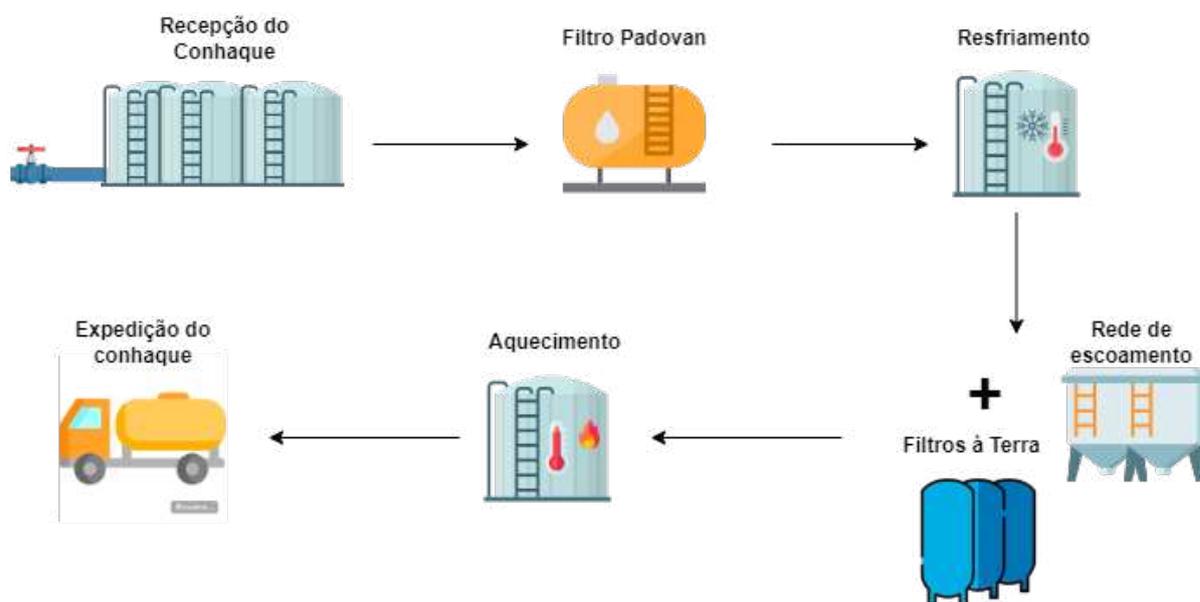
Fonte: (3M, 2024).

Todas estas etapas são cruciais para o desenvolvimento de um produto final satisfatório, desde a escolha das uvas certas até a dupla destilação que aumenta o teor alcoólico da bebida para 70% (STORE, 2024). O projeto descrito neste documento tem o enfoque específico no processo de filtração da bebida antes do engarrafamento da mesma, a parte da filtração destacada em vermelho na Figura 2.

2.1.1 Processo de Filtração do Conhaque

Esta etapa é fundamental para garantir a qualidade e aparência do produto final. A Figura 3 apresenta um resumo das etapas da filtração do conhaque.

Figura 3 – Processo de Filtragem Conhaque



Fonte: Próprio Autor (2024).

2.1.1.1 Recepção do Conhaque

O conhaque é recebido para o começo do processo de filtração. Assim sendo, este conhaque é armazenado em tanques até que o operador ordene que a próxima etapa do processo comece. Neste instante o conhaque se encontra quente (em torno de 23°C).

2.1.1.2 Filtro quente

Nesta etapa, o conhaque é retirado dos tanques de recepção e passado pelo primeiro filtro ainda em temperatura ambiente.

Esta primeira filtragem serve para retirar as partículas sólidas, sedimentos e óleos fusel que são solúveis ou suspensos a altas temperaturas. Após esta primeira filtragem, este conhaque quente é armazenado em outro tanque que contém, agora, o do conhaque quente bruto.

Apesar de já ter passado pelo filtro quente nesta etapa, o conhaque ainda é considerado bruto, pois ainda não passou pela etapa dos filtros à terra, onde será feita a filtragem fina deste líquido.

2.1.1.3 Resfriamento

Após a filtragem do conhaque com a temperatura quente, a próxima etapa é a filtragem com o líquido frio para garantir a máxima pureza e estabilidade do produto final. Logo antes de passar pelos filtros à terra, o conhaque passa por este processo de resfriamento para que compostos que não foram filtrados na temperatura quente possam ser detectados nesta etapa.

Assim sendo, a temperatura da bebida nesta etapa passa a ser - 6 °C, e este é o chamado conhaque bruto frio. Este conhaque é então armazenado em tanques para o aguardo do começo da próxima etapa.

2.1.1.4 Filtros à Terra

Com o conhaque bruto frio e pronto para filtração, o líquido é encaminhado aos filtros a terra. Os filtros a terra são sistemas de filtração que utilizam camadas de materiais granulares, como areia e outros materiais porosos, para remover impurezas de líquidos.

O líquido a ser filtrado passa através de uma camada de material granulado, que é colocada em camadas em um recipiente cilíndrico. O conhaque a ser purificado contém impurezas que são retidas nas diversas partes do filtro. O conhaque de saída deste processo é chamado de conhaque filtrado frio.

2.1.1.5 Rede de escoamento

Após o processo de filtragem, o conhaque de saída é analisado pelo operador. Caso o mesmo acredite que o conhaque não está puro o suficiente, o conhaque é enviado para a rede de escoamento. Nela, o conhaque é aquecido e enviado de volta aos tanques de recepção de conhaque bruto para o recomeço do ciclo.

2.1.1.6 Aquecimento

Caso o conhaque passe pelo teste de pureza do operador após o processo de filtragem à terra, este conhaque, chamado de conhaque filtrado frio, é levado à estação de aquecimento. Nela, o conhaque é aquecido para sua volta à temperatura ambiente (23°C). O conhaque, ao sair da etapa de aquecimento, é chamado de conhaque filtrado quente e é armazenado em tanques até a próxima etapa.

2.1.1.7 Expedição do conhaque filtrado

Como o próprio nome já diz, nesta etapa o conhaque filtrado quente já é o produto final, logo, este líquido é levado para a etapa de engarrafamento do conhaque.

2.2 Automação Industrial

Após a revolução industrial, houve um grande aumento pela busca por novas tecnológicas mais eficientes e poderosas, principalmente por tecnologias que aprimorassem a produção das fabricas, com o isto o termo automação industrial surgiu.

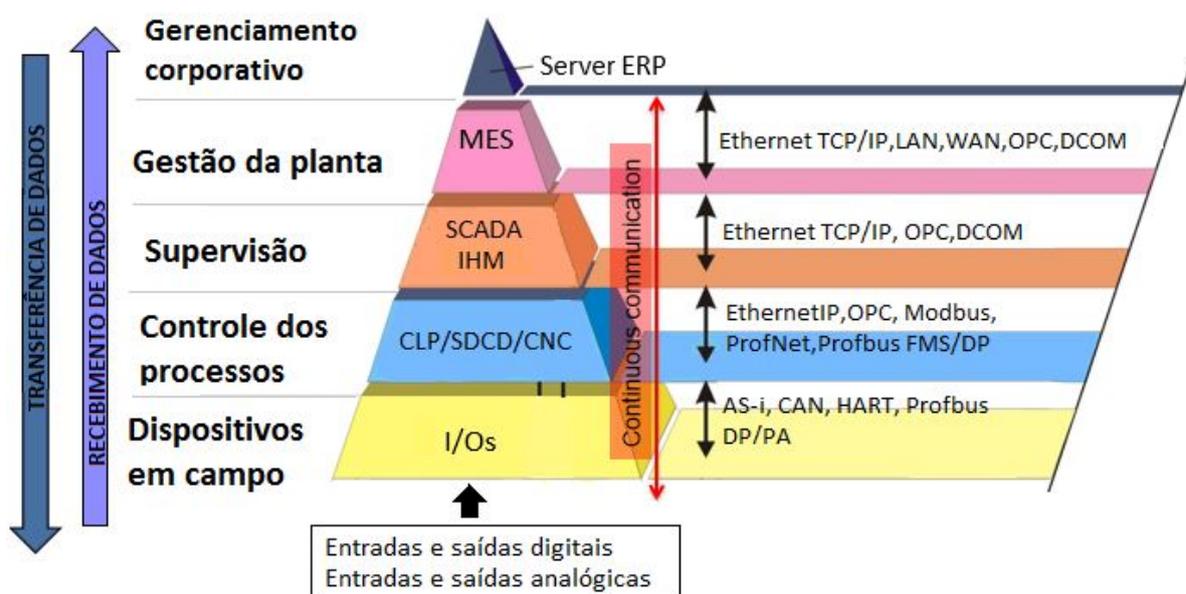
A automação é um sistema de controle pelo qual os mecanismos verificam sua própria operação, fazendo medições e introduzindo correções, sem a necessidade de intervenção

humana (ROGGIA; FUENTES, 2016). A automação industrial, portanto, refere-se a esse mesmo conceito, mas aplicado ao setor industrial.

A automação industrial elimina a possibilidade de erro humano, reduz custos, economiza tempo e alcança desempenho superior (DAKOL, 2024), e é devido a isto que é uma área de grande importância hoje em dia.

Esta área pode ser dividida em 5 níveis de controle que são definidos pela pirâmide de automação industrial representada na Figura 4. Esta pirâmide serve para organizar e otimizar a automação de uma linha de produção, ajudando na compreensão de qual o papel específico de cada nível na automação e no gerenciamento de processos.

Figura 4 – Pirâmide de Automação Industrial



Fonte: (NUNES, 2024).

É possível observar que nesta pirâmide as primeiras camadas são responsáveis por controlar e supervisionar os dispositivos de campo que atuam diretamente no processo. Já nas últimas camadas tratam do gerenciamento dos processos e da planta. Este projeto foi desenvolvido com os conceitos das primeiras 3 camadas de base desta pirâmide.

2.2.1 Nível de Controle e de Dispositivo de campo

A camada de base da pirâmide inclui os dispositivos da linha de produção, como por exemplo sensores, medidores de pressão e temperatura. Já a parte de controle de processos é representada principalmente pelo CLP (Controlador Lógico Programável).

Um CLP é um tipo de computador industrial que pode ser programado para executar funções de controle (AUTOMACAO, 2024b). Ou seja, um dispositivo programável

responsável por comandar os dispositivos da linha de produção, armazenar e coletar os dados da linha, e indicar divergências ou falhas nos dispositivos. A Figura 5 ilustra um CLP comumente utilizado nas indústrias.

Figura 5 – Exemplo de CLP



Fonte: (PETRUZELLA, 2014).

O CLP é um sistema de tempo real, onde a saída do sistema controlado por ele depende das condições da entrada (PETRUZELLA, 2014). O que quer dizer que o CLP monitora o estado do processo (por exemplo, por meio de sensores) e, a partir dessas informações, faz escolhas e cria ações para controlar o processo de acordo com a forma programada.

2.2.2 Redes Industriais

As redes dentro de um processo industrial são classificadas de acordo com seus níveis de automação, vindo desde o nível mais baixo, chamado de nível de entrada e saída ou nível de chão de fábrica ou de campo, onde estão localizados os sensores e atuadores responsáveis pelo andamento do processo produtivo.

As diversas redes são distribuídas dentro da empresa e se interagem constantemente. E estes diversos equipamentos de diferentes fabricantes somente conseguem se comunicar se trabalharem dentro de ambiente padronizado de comunicação, instalação, operação, garantindo assim a melhoria da qualidade de serviços e produtos (DYNAMICS, 2017).

Atualmente, no ambiente industrial existe um híbrido de tecnologias de TO e TI tradicionais operando em conjunto em uma forma de otimizar as operações e os processos. Essa evolução tecnológica conhecida como convergência de TI/TO tem sido construída através de redes comuns baseada em IP. Porém estas redes possuem características e requisitos diferentes, como por exemplo, os requisitos dos sistemas TO são confiabilidade, latência e determinismo, enquanto a rede TI possui requisitos como grande tráfego de dados, segurança e garantia de entrega da informação (DYNAMICS, 2017).

Os protocolos de comunicação são conjuntos de regras que orientam como será a comunicação entre os dispositivos ([AUTOMACAO, 2024c](#)). Os protocolos de redes industriais de campo surgiram por volta da década de 80, com protocolos HART, Modbus, e vem se evoluindo hoje sendo dominado pelos fieldbuses com tecnologia IP. Entre os protocolos ethernet que mais se destacam hoje estão o Profinet, Ethernet/IP, EtherCat e o Modbus TCP, segundo estudo de ([WORLD, 2023](#)).

2.2.2.1 Protocolo Modbus TCP/IP

O protocolo de comunicação Modbus, criado pela Modicon (atualmente Schneider Electric) na década de 1970, foi um dos primeiros protocolos desenvolvidos para comunicação industrial, o que significa que não havia muitas opções de protocolo na época em que ele foi desenvolvido. Além disso, as outras opções disponíveis no mercado não eram de código aberto. Esse último fato é a causa da enorme popularização desse protocolo. Como resultado, quase todos os dispositivos de automação industrial atuais podem operar com base nesse protocolo.

O Modbus também é um dos protocolos de comunicação mais usados devido a sua facilidade de implementação e simplicidade, graças a sua estrutura de mensagens padronizada e de fácil compreensão, a sua documentação e aos seus baixos requisitos de processamento.

O protocolo Modbus TCP/IP é a combinação do protocolo Modbus com os protocolos TCP (*Transmission Control Protocol*) e IP (*Internet Protocol*), conforme mostrado na Figura 6 que ilustra a estrutura de rede do Modbus TCP/IP. As vantagens dessa combinação são:

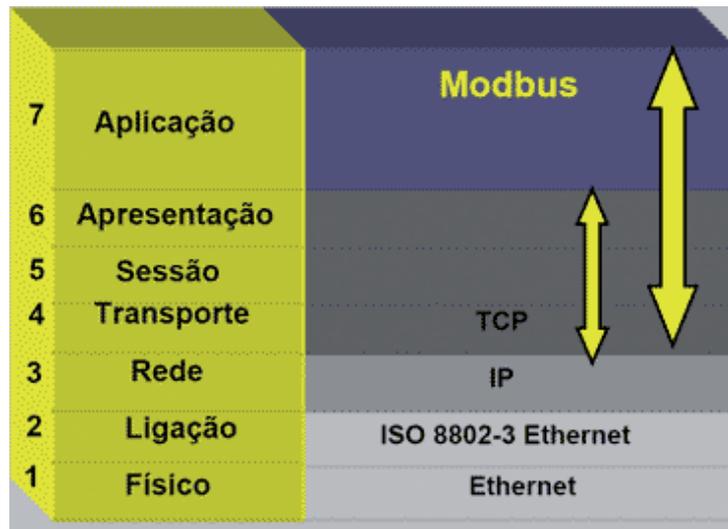
- O TCP garante uma entrega de dados mais confiável ([ROISENBERG, 2024](#)).
- O IP permite a comunicação em redes Ethernet.

O Modbus TCP/IP é baseado na comunicação cliente/servidor. Um cliente Modbus envia uma solicitação a um servidor Modbus que processa a solicitação e responde ao cliente. Nesse contexto, um CLP geralmente atua como servidor Modbus e os dispositivos de campo, ou seja, os dispositivos da linha de produção, atuam como clientes ([CONTROL.COM, 2023](#)).

A Figura 7 mostra um exemplo de troca de dados no modbus TCP. A mensagem Modbus é basicamente o endereço do escravo, código da função e o dado.

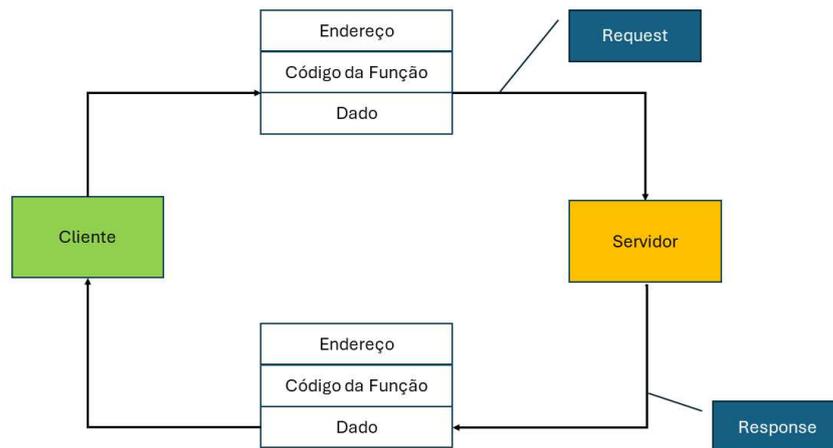
O primeiro byte da mensagem Modbus indica o endereço do escravo. Uma vez que o escravo confirme que o pedido foi para ele, este executa o comando solicitado e responde a mensagem com o código da função e os dados da requisição. O segundo byte é o Código

Figura 6 – Estrutura Modbus TCP/IP



Fonte: (Valete Editora, 2023).

Figura 7 – Exemplo de troca de dados no Modbus TCP



Fonte: Próprio Autor (2024).

de Função, que define o tipo de operação que o mestre deseja realizar (por exemplo, 0x03 para leitura de Holding Registers). A Figura 8, representa uma tabela com as principais funções Modbus (KELLER, 2017).

Figura 8 – Principais funções Modbus

		Código de funções				
		Código	Sub código	Hex		
Acesso de dados	Acesso ao bit	Entradas Discretas Físicas	Read Discrete Inputs	02		01
		Bobinas físicas ou bits internos	Read Coils	01		01
			Write Single Coil	05		05
			Write Multiple Coils	15		0F
	Acesso a 16 Bits	Registros de entrada físicos	Read Input Register	04		04
		Registros de saída física	Read Holding Register	03		03
			Write Single Register	06		06
			Write Multiple Registers	16		10
			Read/Write Multiple Registers	23		17
			Mask Write Register	22		16
Read FIFO queue			24		18	
Acesso a arquivos	Read File record	20		14		
	Write File record	21		15		
Diagnóstico	Read Exception status	07		07		
	Diagnostic	08	00-18,20	08		
	Get Com event Counter	11		0B		
	Get Com Event Log	12		0C		
	Report Server ID	17		11		
	Read device identification	43	14	2B		
Outro	Encapsulated interface Transport	43	13,14	2B		

Fonte: (KELLER, 2017).

Os demais “dados” da mensagem vão depender do tipo de função que se deseja fazer. Por exemplo, para uma leitura de uma função 0x03 (Holding Register), os próximos bytes vão conter o endereço do Registro Inicial e a quantidade de registros desejada na leitura. Os registros são endereços dos parâmetros do escravo e são do tipo inteiro de 2 bytes (KELLER, 2017).

Por exemplo, suponha que o mestre deseja ler 10 registros Analógicos do tipo Holding Register, começando do endereço 1000, do dispositivo escravo 1. Então a solicitação do Cliente (ou Mestre) será:

- Unidade de Endereço: 0x01
- Código de Função: 0x03 (Leitura de registros de retenção)
- Endereço Inicial: 0x03E8 (1000 em hexadecimal)
- Quantidade de Registros: 0x000A (10 registros)

E a resposta do Servidor (ou Escravo) será:

- Unidade de Endereço: 0x01
- Código de Função: 0x03

- Byte Count: 0x14 (20 bytes, pois cada registro ocupa 2 bytes e foram solicitados 10 registros)
- Dados dos Registros: Retorna os valores dos 10 registros, cada um com 2 bytes.

Para enviar e receber essas mensagens, o mestre abre uma conexão TCP com o dispositivo escravo na porta padrão Modbus TCP (porta 502) e envia o pacote de solicitação. O escravo, então, responde com um pacote de resposta contendo os dados dos registros solicitados (KELLER, 2017).

2.2.2.2 Protocolo OPC UA

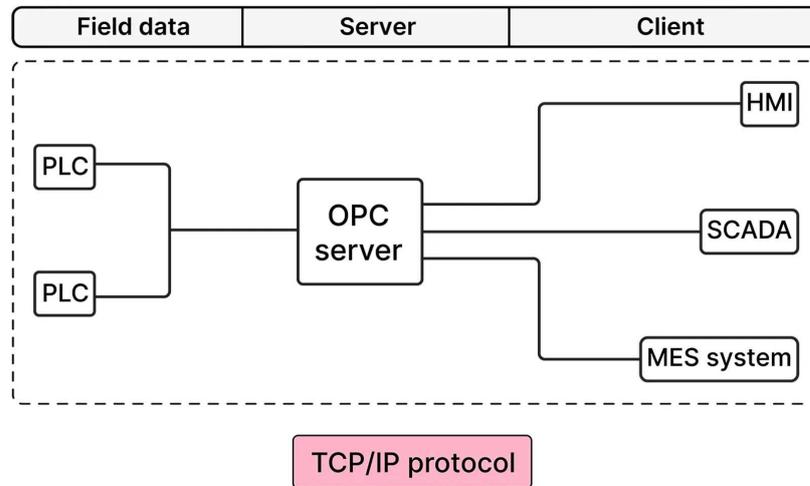
Outro protocolo de comunicação importante na computação industrial é o OPC UA (*Open Platform Communications Unified Architecture*) devido à sua capacidade de fornecer comunicação segura e confiável em ambientes industriais heterogêneos, ou seja, ambientes com equipamentos de diferentes fabricantes. Isso ocorre porque o protocolo não está vinculado a um fabricante específico, o que garante que ele funcione para diferentes fabricantes.

Além disso, a arquitetura de várias camadas do OPC UA oferece uma estrutura “à prova do futuro”. Tecnologias e metodologias inovadoras, como novos protocolos de transporte, algoritmos de segurança ou padrões de codificação, podem ser incorporadas ao OPC UA e, ao mesmo tempo, manter a compatibilidade com versões anteriores dos produtos existentes (FOUNDATION, 2024).

O OPC UA possibilita a troca de dados de processo, alarmes e eventos, além de dados históricos, tudo a partir de um aplicativo cliente e servidor (REALPARS, 2024). O OPC UA não se baseia no DCOM, mas em uma pilha de protocolos construída sobre o conhecido protocolo TCP/IP (REALPARS, 2024). A Figura 9 apresenta uma síntese desta comunicação.

Outras vantagens deste protocolo são sua robustez, segurança, o fato de possuir criptografia, sistema de autenticação de usuário e sistema de browsing de servidores da rede, porém é mais caro, mais novo, mais complexo, precisa de mais software. Ou seja, é um protocolo indicado para grandes projetos, onde a simplicidade do Modbus não é prática.

Figura 9 – Comunicação OPC UA



Fonte: (REALPARS, 2024).

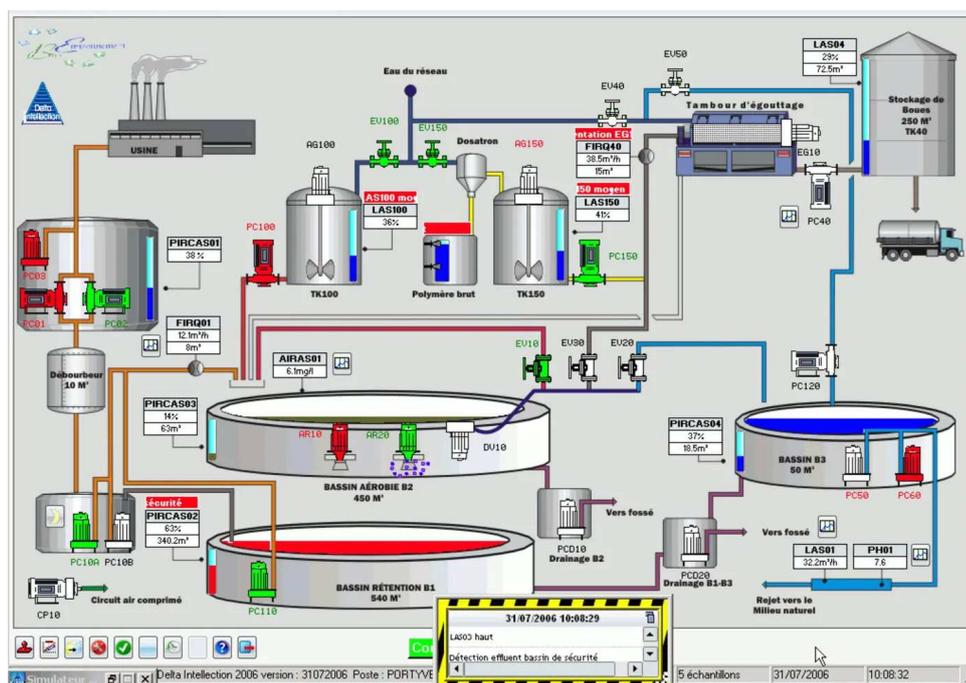
2.2.3 Nível de Supervisão e Gestão da Planta

As interfaces de supervisão são uma parte essencial do processo de automação, pois são usadas para analisar e controlar a evolução dos processos de produção industrial em tempo real. SCADA (Supervisory Control and Data Acquisition) é um termo muito usado no meio industrial quando se fala de interfaces de supervisão.

Os sistemas SCADA são usados para controlar e supervisionar processos industriais em tempo real. Isso envolve a coleta e o processamento de dados de instrumentos e outros sensores, bem como o monitoramento e a exibição dos resultados para os operadores, a fim de obter uma visão completa de como o processo de produção está operando. Outra vantagem desse sistema é sua confiabilidade e robustez, características essenciais para o bom funcionamento dos processos industriais (AUTOMACAO, 2024a).

Uma boa interface de supervisão muda completamente o gerenciamento da linha de produção. Por exemplo, a Figura 10 ilustra uma interface de supervisão e a análise dessa figura facilita a dedução de quais dispositivos estão funcionando, quais estão quebrados e quais não estão em uso, graças à escolha correta das cores exibidas. De fato, o vermelho simboliza o mau funcionamento, o verde o bom funcionamento e a ausência de cor indica o não uso: é um sistema de cores intuitivo.

Figura 10 – Exemplo de Supervisão



Fonte: (BENALLEL, 2024).

2.2.3.1 Ignition SCADA

Um dos principais componentes de software utilizados para o desenvolvimento das interfaces de supervisão, controle e análise de dados em tempo real em ambos os projetos deste estudo é o Ignition SCADA. Este software baseia-se em tecnologias web modernas, projetadas para uma personalização simples e prática de acordo com as necessidades da indústria. Ignition permite a programação semelhante a um site web, com base em linguagens de programação como JavaScript e CSS, o que facilita o desenvolvimento de uma interface ergonômica.

Além disto, o Ignition tem como diferencial o suporte à linguagem de programação Python, utilizada para administrar eventos dentro de uma aplicação. Foi implementado por ser uma linguagem intuitiva e moderna, em vez da Visual Basic for Applications (VBA) utilizada pelos demais softwares já existentes no mercado (Vista Automação, 2023).

Este software possui uma arquitetura modular, operando através dos modelos cliente/servidor e produtor/consumidor, possibilitando a comunicação entre computadores, dispositivos móveis e sistemas. A estrutura do software Ignition pode ser observada na Figura 11.

Figura 11 – Estrutura Ignition



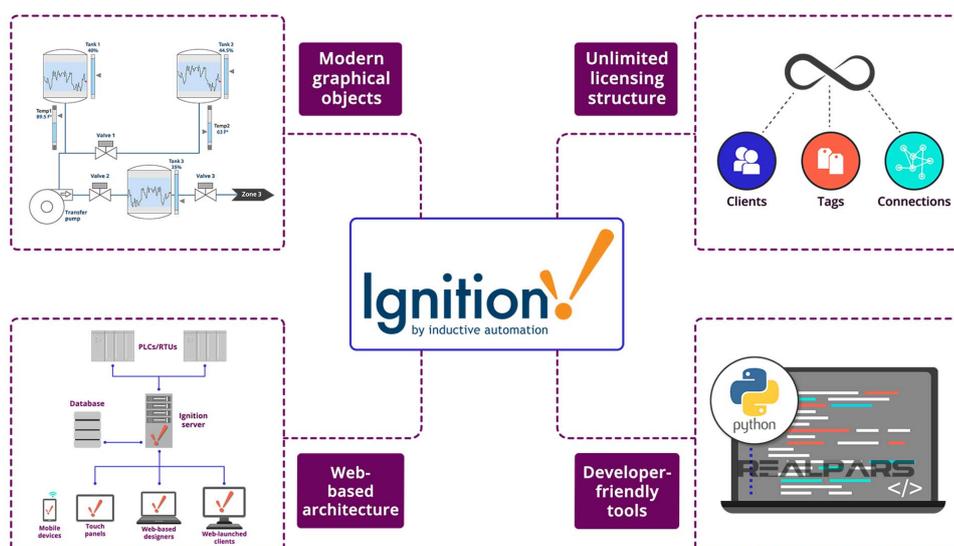
Fonte: (Inductive Automation, 2024b).

O Ignition é altamente escalável, podendo ser utilizado em pequenas instalações ou grandes operações industriais. Sua arquitetura modular permite que módulos adicionais sejam instalados conforme necessário, expandindo suas funcionalidades. Ligado a isto outro fator importante é que este software possui uma licença única feita pelo servidor, ou seja ao adicionar um novo cliente não faz-se necessário uma nova licença.

Por estes fatores para desenvolver uma parte tão importante do processo de automação foi utilizado o software Ignition SCADA, desenvolvido pela Inductive Automation. Esse software é usado por grandes empresas, como Coca-Cola, Johnson-Johnson e Pfizer (Inductive Automation, 2024a), o que prova que sua eficácia já é reconhecida internacionalmente.

Um resumo sobre as vantagens deste software é apresentado na Figura 12.

Figura 12 – Principais vantagens Ignition



Fonte: (RealPars, 2024).

2.2.3.1.1 Vision e Perspective

Dois módulos de visualização extrema importância deste software são o módulo *Vision* e o módulo *Perspective*.

O módulo *Vision* é o software projetado para ser usado em telas de desktop e IHMs tradicionais de chão de fábrica, o que se aproxima mais aos padrões dos softwares mais comuns de desenvolvimento de sistemas SCADA.

Já o módulo *Perspective* é a abordagem nova e moderna da Ignition uma vez que, ele é totalmente web-based. Isto permite a inicialização do sistema de supervisão de qualquer lugar em um navegador web, desde que esteja conectado com segurança à rede da fábrica (RealPars, 2024). Devido à forma como foi projetado, o *Perspective* é facilmente implementável em dispositivos móveis, como tablets e smartphones.

2.2.3.1.2 Tags

Um conceito de extrema importância dentro deste software são as Tags. São chamadas Tags as etiquetas que armazenam os dados brutos dos dispositivos presentes na rede. Por exemplo em um tanque equipado com captadores de nível, estes captadores enviam sinais de dados brutos ao PLC que são armazenado como uma tag. Da mesma forma no caso de um motor, a tag pode ser utilizada para escrita/leitura de valores do mesmo.

Essa etiqueta é o que é usado e exibido em um aplicativo SCADA. Dependendo da aplicação industrial, podem estar presentes centenas ou até milhares de tags.

2.2.3.1.3 UDT - User Defined Types

UDTs (*User Defined Types*) são tipos definidos pelo usuário que oferecem a capacidade de aproveitar os princípios de design de dados orientados a objetos no software de supervisão (AUTOMATION, 2024). As UDT são como classes em orientação a objeto, ou seja, são modelos de dados para gerar instâncias.

Então, seguindo esta analogia aos conceitos de programação orientada a objeto, tags representam as variáveis ou objetos e as UDTs as classes. A construção das UDT é feita usando tags simples dos tipos padrões, como por exemplo BOOL, INT, FLOAT. Após a criação da UDT, é possível gerar Tags que serão instâncias desta UDT, ou seja, Tags do tipo desta UDT e que seguem todas as características definidas na mesma.

2.3 Linguagens

Nesta seção será apresentado as linguagens utilizadas para o desenvolvimento das interfaces de supervisão.

2.3.1 Javascript + CSS

JavaScript é uma Linguagem de Programação, criada por Brendan Eich, a pedido da empresa Netscape, em meados de 1995 (Centro de Produções Técnicas, 2024). Atualmente a grande maioria dos sites modernos usam JavaScript e todos os navegadores modernos possuem interpretadores JavaScript, o que faz desta linguagem a linguagem de programação mais onipresente da história (FLANAGAN; MATILAINEN, 2007).

É uma linguagem de programação web, interpretada e orientada a objetos, que roda no navegador do usuário (*client-side*), permitindo a integração de páginas web dinâmicas, interativas e responsivas. O fato desta linguagem ser *client-side* permite a validação de formulários diretamente no navegador, podendo por exemplo verificar se uma senha atende os requisitos antes de enviar ao servidor. Isto tudo pois o comportamento da página pode ser baseado em eventos.

CSS é uma linguagem também utilizada para o desenvolvimento web, porém mais utilizada para definir a aparência e do layout de uma página web. A combinação dessas duas linguagens permite a criação de interfaces mais dinâmicas e intuitivas. Uma vez que o JavaScript controla a lógica e o comportamento, enquanto o CSS cuida da apresentação.

2.3.2 Python

Python foi criado em 1990 por Guido van Rossum, no Instituto Nacional de Pesquisa para matemática e Ciência da computação da Holanda (CWI) (BORGES, 2014), é uma

linguagem que foi desenvolvida com o intuito de possuir uma sintaxe clara e legível, para facilitar a aprendizagem e o desenvolvimento de projetos complexos. Ou seja, é uma linguagem que possui diversas estruturas de alto nível, como listas e dicionários, mas proporciona a fácil compreensão do seu código fonte.

O python é uma linguagem que conta com diversas bibliotecas padrões e diversas bibliotecas de terceiros. O que facilita ainda mais a implementação de projetos utilizando esta linguagem. Outra grande vantagem é que esta linguagem é dinamicamente tipada, o que significa que não há necessidade de declaração explícita do tipo das variáveis.

Por fim, esta linguagem também suporta a programação orientada a objetos. Todas essas vantagens fazem com que o python seja hoje uma das linguagens mais utilizadas para a automação de tarefas repetitivas.

Dentro do contexto do software Ignition, esta linguagem é utilizada para a programação de scripts visando a automação de processos industriais, manipulação de dados, e interações com dispositivos e interfaces de operação.

3 Metodologia e Desenvolvimento

O presente tópico aborda todas as etapas realizadas para o desenvolvimento de interfaces de supervisão em uma parte da linha de produção automatizada de conhaque. O projeto tem efeitos imediatos na eficiência e segurança do sistema.

O projeto descrito neste documento teve seu desenvolvimento iniciado em fevereiro de 2024 e se prolongou até o fim de julho de 2024, na França. Para abordar o desenvolvimento desse projeto de forma detalhada, além de incluir uma revisão bibliográfica, o documento será organizado em dois tópicos: Características do Sistema e Desenvolvimento das Interfaces.

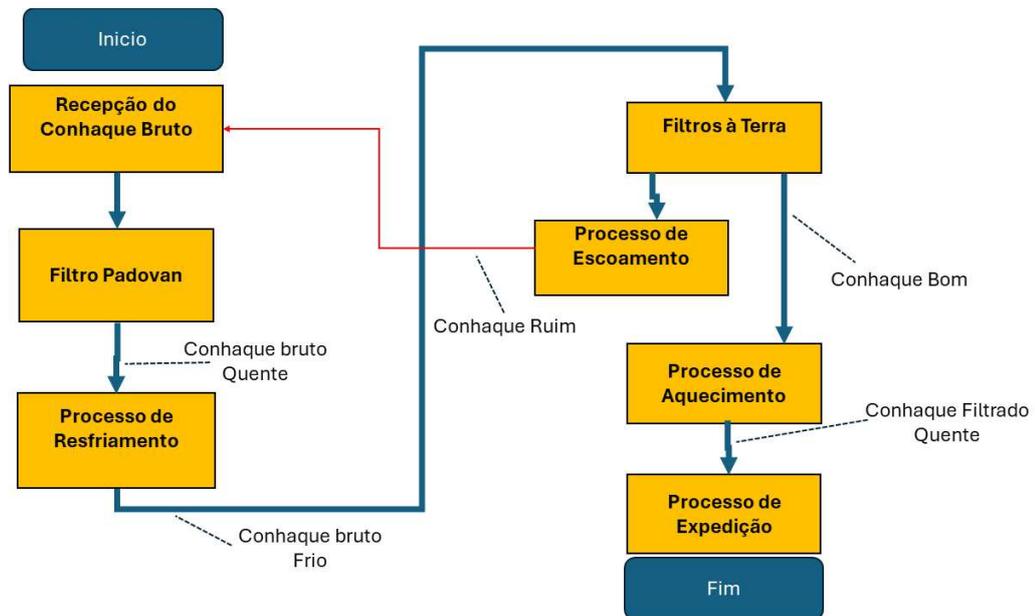
3.1 Características do Sistema

Como apresentado na introdução deste documento, o projeto conhaque se trata do desenvolvimento das interfaces SCADA da linha de produção da bebida conhaque. Isto ocorre pois as interfaces de supervisão que estavam presentes na empresa estavam obsoletas e, com a introdução de um novo armário elétrico para controle desta linha, veio a necessidade de desenvolvimento de novas interfaces de supervisão.

Além disto, a linha de produção contava apenas com um local para a visualização das interfaces de supervisão no escritório do chefe de linha. Como se trata uma grande linha de produção viu-se a necessidade de possibilitar a movimentação dentro da linha com a interface de supervisão em mãos para analisar cada processo de forma isolada e prática. Por isto, veio a proposta de implementação de um tablet com a interface de supervisão.

Para o desenvolvimento do projeto proposto, é necessário o estudo do sistema e de seus dispositivos. Um fluxograma geral do sistema pode ser observado na Figura 13, e nos próximos tópicos serão abordados de forma detalhada as partes importantes para o desenvolvimento da supervisão do mesmo.

Figura 13 – Fluxograma do sistema conhaque

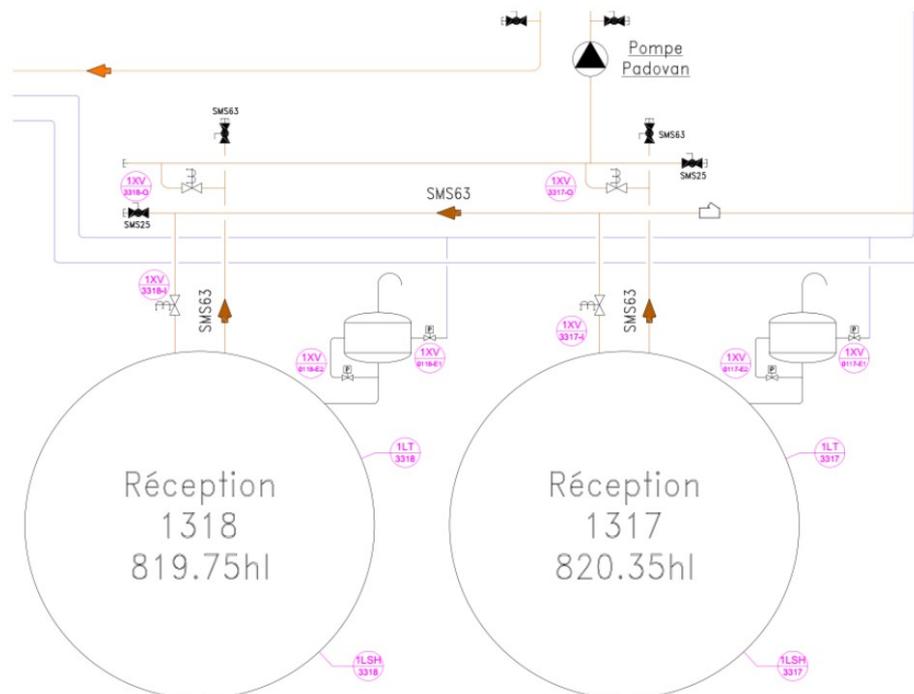


Fonte: Próprio Autor (2024).

3.1.1 Recepção conhaque bruto

Uma representação do esquema hidráulico de recepção do chamado *conhaque bruto quente* pode ser observada na Figura 14, bem como os dispositivos deste sistema que são utilizados para interface podem ser observados na tabela 1.

Figura 14 – Esquema Hidráulico - Recepção do Conhaque Bruto



Fonte: Próprio Autor (2024).

Tabela 1 – Equipamentos utilizados na supervisão - Recepção conhaque bruto

Referência	Designação
Entradas Digitais	
1LSH3317	Nível de transbordamento do tanque de recepção 1317
1LSH3318	Nível de transbordamento do tanque de recepção 1318
Entradas analógicas	
1LT3317	Radar de nível do tanque 1317
1LT3318	Radar de nível do tanque 1318
Rede	
1XV3317I	Válvula de entrada do tanque de recepção 1317
1XV3318I	Válvula de entrada do tanque de recepção 1318
Equipamento não gerenciado pelo PLC	
Bomba de recepção do conhaque bruto	

Fonte: Autoria própria (2024).

Assim sendo, para esta parte do sistema, a interface a ser desenvolvida funcionará da seguinte forma:

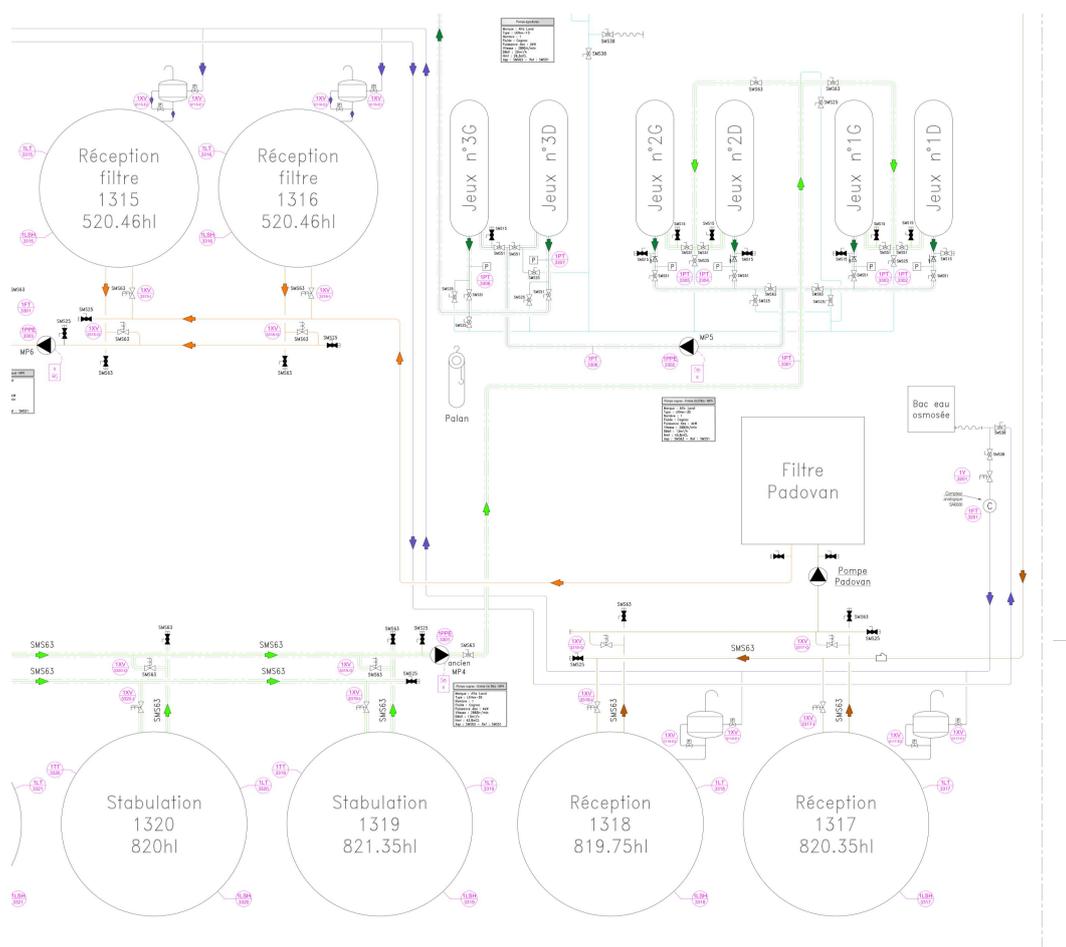
- O operador escolhe entre 3 modos de funcionamento para as válvulas de entrada: Fechada, Manual e Automático. O primeiro corresponde ao estado em que a válvula se encontra fechada, o segundo corresponde à abertura e fechamento de válvulas feitos exclusivamente para o operador, podendo este ser na interface de supervisão ou fisicamente no dispositivo. O modo automático das válvulas de entrada corresponde ao mesmo funcionamento do modo manual e ao fechamento automático da válvula de enchimento ao atingir o nível de transbordamento, sendo a abertura da mesma ainda dependente do operador.
- O operador ajusta na interface de supervisão em quantos hectolitros correspondem aos níveis: muito baixo, baixo, alto e muito alto de enchimento de cada tanque.
- Com as informações dos sensores de nível de transbordamento, do radar de nível de cada tanque e os ajustes feitos anteriormente pelo operador, a interface apresentará o estado em tempo real do enchimento do tanque, podendo ou não agir automaticamente no fechamento da válvula de enchimento do tanque, a depender da escolha do modo de funcionamento feito pelo operador. Além disto ao atingir os limites baixo ou alto um aviso de atenção deve aparecer na interface, e para os limites muito alto ou muito baixo um alarme em vermelho deve estar disparar na tela para alertar o operador. Já para o nível de transbordamento um alarme visual e piscando é necessário para demonstrar a urgência ao operador.

Os equipamento não gerenciados pelo CLP não serão equipamentos gerenciados na supervisão, assim sendo a bomba de entrada do conhaque bruto nos tanques não será comandada diretamente pela supervisão. Porém no processo a bomba está ligada as válvulas de forma que quando as válvulas são abertas a bomba começa a funcionar e quando estão fechadas ela para.

3.1.2 Filtro quente

Após a recepção do conhaque bruto quente, o líquido passa pelo filtro quente e é armazenado em dois outros tanques, os chamados tanques de recepção filtrada. O esquema hidráulico desta parte pode ser observado na Figura 15, bem como os dispositivos relevantes para a supervisão podem ser averiguados na tabela 2.

Figura 15 – Esquema Hidráulico - Filtro quente



Fonte: Próprio Autor (2024).

De acordo com a Figura 15 e tabela 2 aos tanques 1315 e 1316 são similares aos dispositivos da etapa anterior para os tanques 1317 e 1318, isto se deve ao fato de que todos os tanques deste projeto possuem o mesmo comportamento esperado. Ou seja, todas as válvulas de entrada de tanque possuem o mesmo modo de funcionamento descrito

Tabela 2 – Equipamentos utilizados na supervisão - Filtro quente

Referência	Designação
Entradas Digitais	
1LSH3315	Nível de transbordamento do tanque de recepção filtrada 1315
1LSH3316	Nível de transbordamento do tanque de recepção filtrada 1316
Entradas Analógicas	
1LT3315	Radar nível tanque recepção filtrada 1315
1LT3316	Radar nível tanque recepção filtrada 1316
Rede	
1XV3315I	Válvula entrada tanque recepção filtrada 1315
1XV3316I	Válvula entrada tanque recepção filtrada 1316
1XV3317O	Válvula saída tanque recepção 1317
1XV3318O	Válvula saída tanque recepção 1318
Equipamento não gerenciado pelo PLC	
Bomba filtro quente	

Fonte: Autoria própria (2024).

anteriormente e o comportamento/definições de níveis na interface também permanecem os mesmos.

Além dos dispositivos similares aos da etapa anterior, para permitir a saída do conhaque dos tanques de recepção, esta etapa apresenta as válvulas de saída dos tanques de recepção. As válvulas de saída dos tanques 1317 e 1318 funcionam da mesma forma que as válvulas de entrada para o modo "Fechado", "Manual", porém não possuem modo automático.

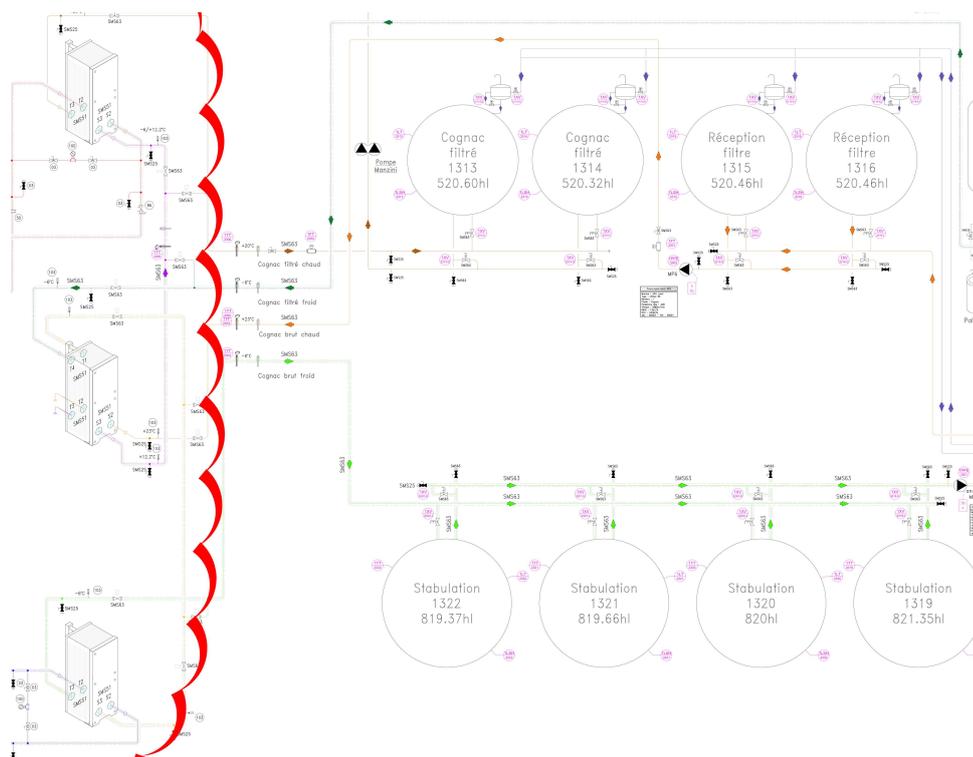
3.1.3 Resfriamento

Após a passagem pelo filtro quente, o conhaque bruto quente é conduzido dos tanques de recepção filtrada 1315/1316 para o resfriamento e, após isto, o agora chamado *conhaque bruto frio*, é armazenado nos tanques de estabulação, 1319/1320/1321/1322. O esquema hidráulico desta parte é ilustrado na Figura 16, bem como os dispositivos importantes para o desenvolvimento da interface são apresentados na tabela 3.

Assim sendo, para que este processo possa ocorrer, o operador terá na interface:

- A escolha do modo de funcionamento da bomba MP6, bomba que faz parte do processo de condução do conhaque do tanque de recepção filtrada ao resfriamento. Podendo ser automático, parado ou manual. Se a escolha for o modo manual, a bomba funcionará ou deixará de funcionar apenas com o comando do operador na interface. Já em modo automático, o programa feito no CLP faz o comando de

Figura 16 – Esquema Hidráulico - Resfriamento



Fonte: Próprio Autor (2024).

quando a bomba estará funcionando ou parada.

- O estado de funcionamento das bombas deverá ser mostrado na IHM. Podendo esta variar entre: Parada, Em funcionamento ou com Defeito.
- A interface deve permitir ao operador escolher a regulação da bomba entre os modos: Regulação T° saída conhaque bruto frio, Misto e Regulação débito de entrada conhaque frio. A informação com a escolha do modo de funcionamento é enviada para o CLP para que o mesmo regule a bomba de acordo com os parâmetros programados no mesmo.
- A interface deve conter as informações de temperatura do conhaque e a vazão de saída de conhaque bruto frio e deve possibilitar a escolha destas informações pelo operador, as quais serão passadas para o CLP para possibilitar a regulação da bomba.
- As válvulas de saída dos tanques de recepção filtrada 1315/1316 funcionam igualmente às outras válvulas de saída descritas na seção 3.1.2.
- As válvulas de entrada dos tanques de estabulação 1319/1320/1321/1322 possuem o mesmo funcionamento das válvulas de entrada descritas na seção 3.1.1, e a definição dos níveis dos tanques e comportamento da interface esperados também são os mesmos.

Tabela 3 – Equipamentos utilizados na supervisão - Resfriamento

Referência	Designação
Entradas Digitais	
1LSH3319	Nível de transbordamento do tanque de estabulação 1319
1LSH3320	Nível de transbordamento do tanque de estabulação 1320
1LSH3321	Nível de transbordamento do tanque de estabulação 1321
1LSH3322	Nível de transbordamento do tanque de estabulação 1322
Entradas Analógicas	
1TT3319	Sensor de temperatura do tanque de estabulação 1319
1TT3320	Sensor de temperatura do tanque de estabulação 1320
1TT3321	Sensor de temperatura do tanque de estabulação 1321
1TT3322	Sensor de temperatura do tanque de estabulação 1322
1LT3319	Radar de nível do tanque de estabulação 1319
1LT3320	Radar de nível do tanque de estabulação 1320
1LT3321	Radar de nível do tanque de estabulação 1321
1LT3322	Radar de nível do tanque de estabulação 1322
1FT3301	Medidor de vazão de entrada de conhaque quente bruto
1TT3303	Sensor de temperatura de saída de conhaque bruto frio
Rede	
Bomba MP6	Controlador da bomba 1PPE3303 MP6
1XV3315O	Válvula saída tanque recepção filtrada 1315
1XV3316O	Válvula saída tanque recepção filtrada 1316
1XV3319I	Válvula entrada tanque de estabulação 1319
1XV3320I	Válvula entrada tanque de estabulação 1320
1XV3321I	Válvula entrada tanque de estabulação 1321
1XV3322I	Válvula entrada tanque de estabulação 1322

Fonte: Autoria própria (2024).

3.1.4 Filtros à terra

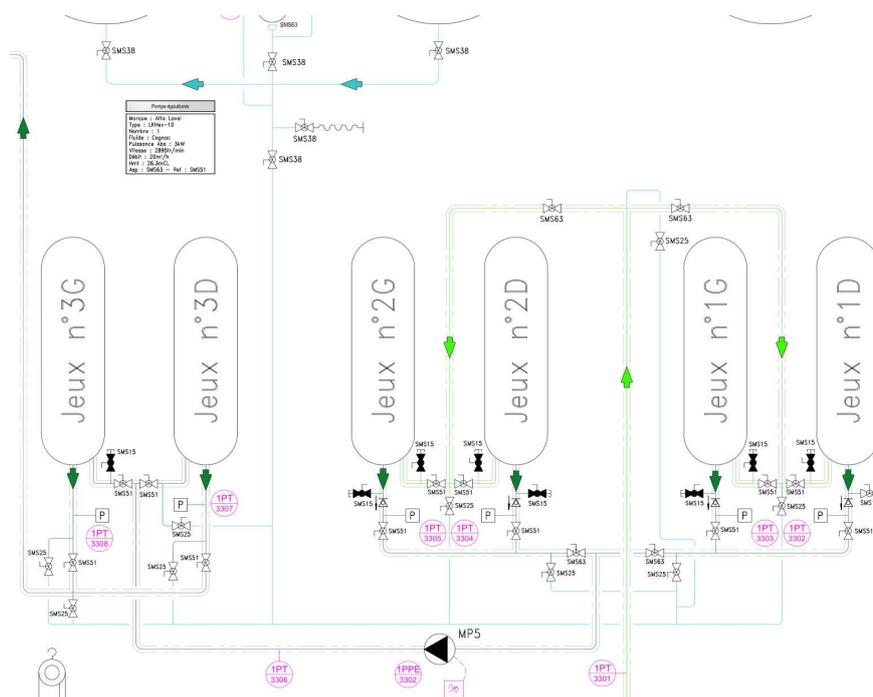
Nesta etapa o conhaque bruto frio sai dos tanques de estabulação 1319/1320/1321/1322 para um primeira filtragem nos filtros 1G/1D e 2G/2D e após isto para a segunda filtragem no filtro 3G/3D e assim obtemos o chamado *conhaque filtrado frio*. A Figura 17 ilustra a parte do sistema hidráulico que corresponde a esta etapa e a tabela 4 mostra os dispositivos importantes para o desenvolvimento da supervisão da mesma.

Para a parte de filtragem, alguns dos pontos desejados na interface de supervisão são:

- A escolha do modo de funcionamento das bombas 1PPE3301 e 1PPE3302. Podendo ser automático, parado ou manual.

Se a escolha for o modo manual, a bomba funcionará ou deixará de funcionar apenas com o comando do operador a partir da interface. Já em modo automático,

Figura 17 – Esquema Hidráulico - Filtros á terra



Fonte: Próprio Autor (2024).

o programa feito no PLC que faz o comando de quando a bomba será aberta ou fechada.

- O estado de funcionamento das bombas será mostrado na IHM. Podendo esta variar entre: Parada, Funcional ou com Defeito.
- A interface deve conter as informações de temperatura do conhaque após a filtragem. Bem como deve possibilitar a escolha da temperatura final do conhaque (após o aquecimento 3.1.6) e a vazão de saída de conhaque filtrado quente pelo operador, as mesmas serão passadas para o CLP para possibilitar a regulação das bombas.
- A interface deve permitir ao operador regular as bombas entre os modos: Regulação T° saída conhaque filtrado quente, Misto e Regulação debito de saída do Conhaque quente. A informação com a escolha do modo de funcionamento é enviada para o CLP para que o mesmo regule a bomba de acordo com os parâmetros programados no mesmo.
- As válvulas de saída funcionam da mesma forma que explicado na seção anterior.
- Cada filtro possui um medidor de pressão que o que permite o calculo da pressão diferencial entre eles. O valor deste pressão e da pressão diferencial deve estar presente na interface de supervisão. Isso permite que o operador monitore o acúmulo de partículas nos filtros e programe os ciclos de limpeza ou substituição.

Tabela 4 – Equipamentos utilizados na supervisão - Filtros à terra

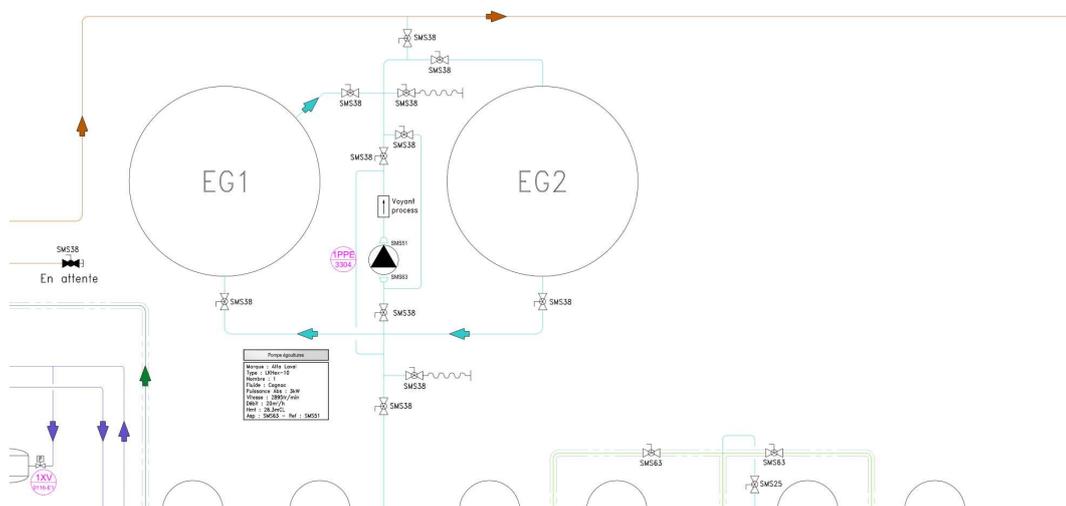
Referência	Designação
Entradas Analógicas	
1PT3302	Sensor de pressão do filtro 1D
1PT3303	Sensor de pressão do filtro 1G
1PT3304	Sensor de pressão do filtro 2D
1PT3305	Sensor de pressão do filtro 2G
1PT3307	Sensor de pressão do filtro 3D
1PT3308	Sensor de pressão do filtro 3G
1PT3301	Sensor de pressão de entrada da filtragem
1PT3303	Sensor de pressão de entrada da segunda filtragem
1TT3304	Sensor de temperatura de saída do conhaque filtrado
Rede	
1PPE3301	Bomba de entrada de conhaque no filtro 1G/1D e 2G/2D
1PPE3302	Bomba de entrada na filtro secundário 3G/3D
1XV3319O	Válvula saída tanque de estabulação 1319
1XV3320O	Válvula saída tanque de estabulação 1320
1XV3321O	Válvula saída tanque de estabulação 1321
1XV3322O	Válvula saída tanque de estabulação 1322

Fonte: Autoria própria (2024).

3.1.5 Rede de escoamento

Caso o operador não esteja satisfeito com a pureza do líquido após a filtragem, o conhaque é enviado para esta etapa, responsável por encaminhar este líquido de volta ao começo do ciclo, para os tanques de recepção do conhaque bruto. Para isto ele é equipado com dois tanques de escoamento EG1 e EG2. A Figura 18 representa esta parte do processo e a tabela 5 contém os dispositivos utilizados na interface de supervisão para a mesma.

Figura 18 – Esquema Hidráulico - Rede de Escoamento



Fonte: Próprio Autor (2024).

Tabela 5 – Equipamentos utilizados na supervisão - Rede de escoamento

Referência	Designação
1PPE3304	Rede Bomba Rede de escoamento

Fonte: Autoria própria (2024).

Os pontos desejados na interface de supervisão para a rede de escoamento são:

- A escolha do modo de funcionamento da bomba da rede de escoamento na interface. Podendo ser parado ou manual.
- O estado de funcionamento da bomba será mostrado na IHM. Podendo esta variar entre: Parada, Funcional ou em Defeito.

3.1.6 Aquecimento

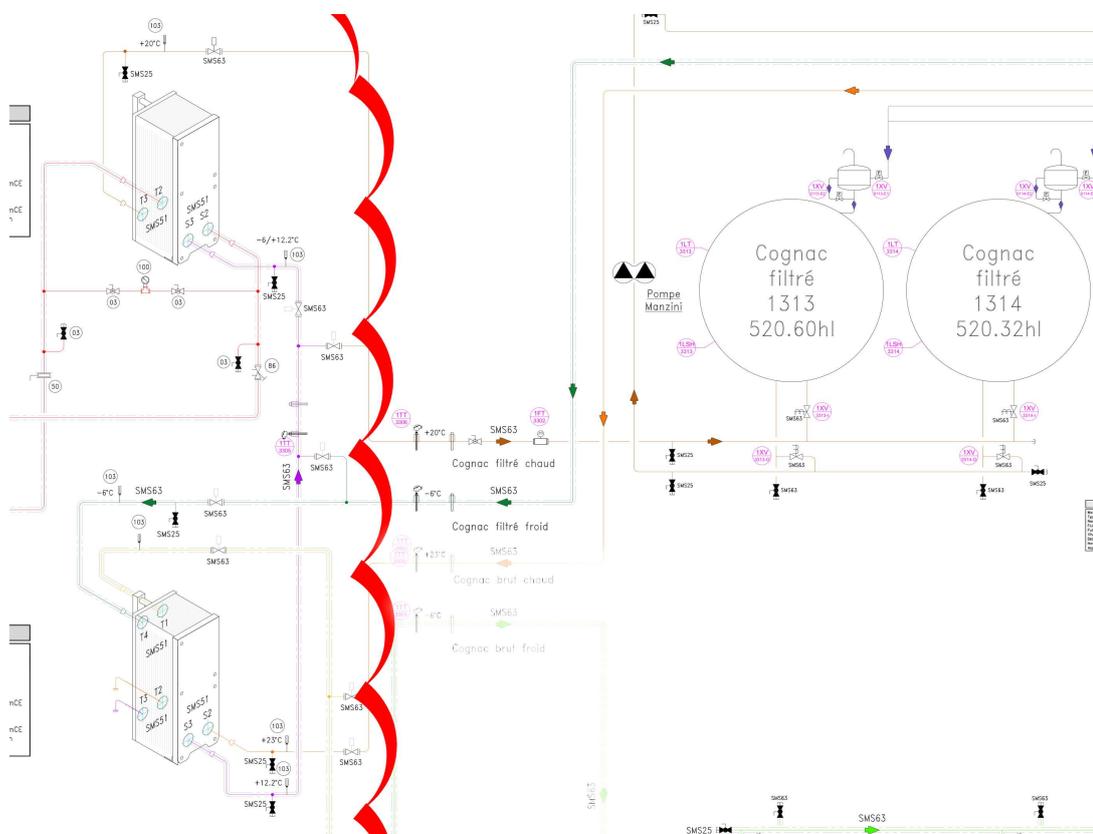
Após o processo de filtragem do conhaque bruto frio, caso este conhaque passe no teste de pureza do operador, o mesmo passa pela etapa de aquecimento e armazenamento nos tanques de conhaque filtrado 1313 e 1314, e agora passa a se chamar *conhaque filtrado quente*. O esquema hidráulico desta etapa é representado na Figura 19.

Na etapa de filtragem, já foram definidos pelo operador na interface qual a temperatura final desejada para o conhaque e qual a vazão de saída de conhaque filtrado quente. Da mesma forma que anteriormente, estes valores são enviados para o PLC para comandar as bombas PPE3301 e PPE3302. Estas bombas levam o conhaque até a estação de aquecimento. Logo, os dispositivos importantes para a interface de supervisão relacionados às bombas e sua regulagem são os mesmos da etapa de filtragem. No entanto, os novos dispositivos importantes para esta etapa são os indicados na tabela 6.

E para esta parte temos na interface:

- As mesmas representações e regulagens de válvulas de entradas apresentadas na etapa 3.1.1, utilizando as informações dos sensores da mesma forma que as etapas anteriores.
- Display na tela de todas as temperaturas medidas antes e depois do aquecimento e do valor do medidor de vazão de saída do conhaque quente.

Figura 19 – Esquema Hidráulico - Aquecimento



Fonte: Próprio Autor (2024).

Tabela 6 – Equipamentos utilizados na supervisão - Aquecimento

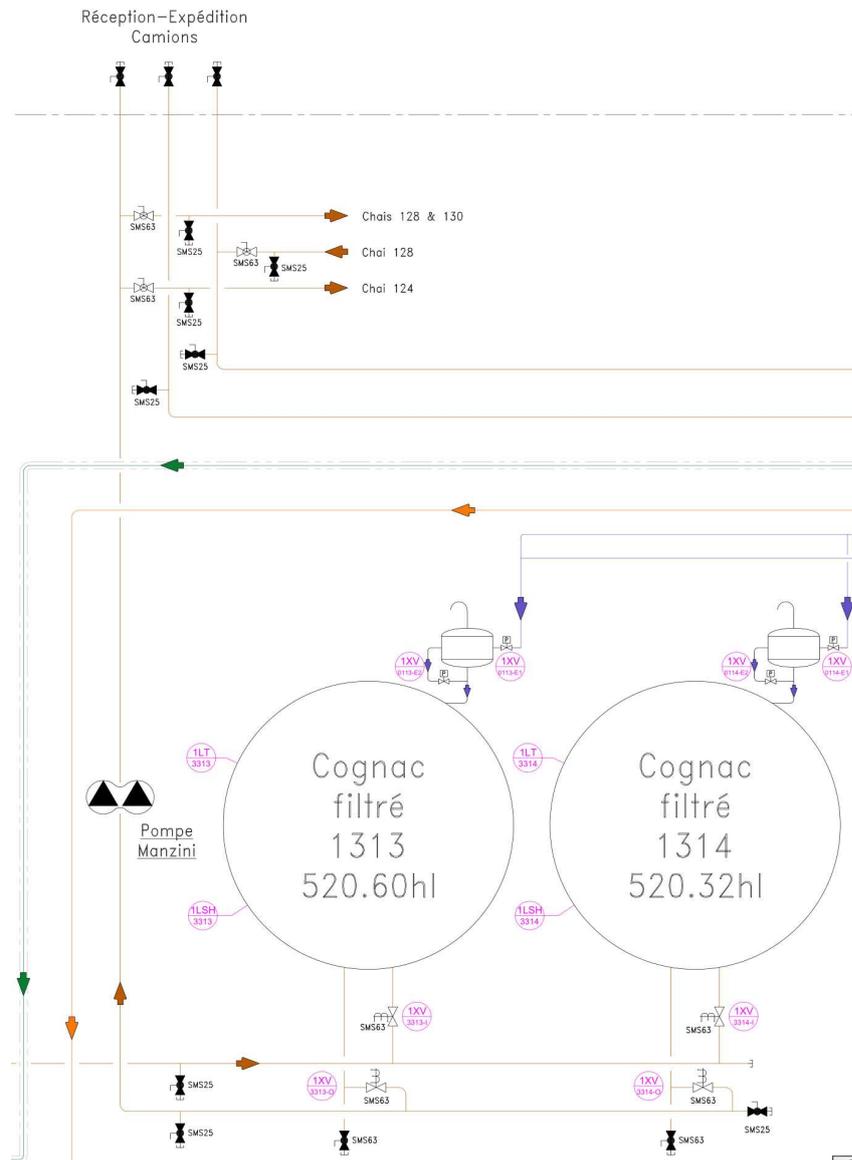
Referência	Designação
Entradas Digitais	
1LSH3313	Nível de transbordamento do tanque de conhaque filtrado 1313
1LSH3314	Nível de transbordamento do tanque de conhaque filtrado 1314
Entradas Analógicas	
1FT3302	Medidor de vazão de saída de conhaque filtrado quente
1TT3304	Sensor de temperatura de saída do conhaque filtrado frio
1TT3306	Sensor de temperatura de saída do conhaque filtrado quente
1LT3313	Radar de nível do tanque de conhaque filtrado 1313
1LT3314	Radar de nível do tanque de conhaque filtrado 1314
Rede	
1XV3313I	Válvula entrada tanque de conhaque filtrado 1313
1XV3314I	Válvula entrada tanque de conhaque filtrado 1314

Fonte: Autoria própria (2024).

3.1.7 Expedição do conhaque filtrado quente

Para finalizar o projeto, temos a etapa de expedição do conhaque filtrado quente, o diagrama hidráulico desta parte pode ser observado na Figura 20 e os dispositivos importantes para a supervisão estão presentes na tabela 7.

Figura 20 – Esquema Hidráulico - Expedição do conhaque filtrado quente



Fonte: Próprio Autor (2024).

Os elementos esperados da supervisão nesta parte do processo são: - O operador pode escolher entre os modos "Fechada" e "Manual" para as válvulas de saída. - O operador pode escolher entre os modos "Parada" e "Funcionando" para a bomba Manzini. Sendo o modo funcional para quando o operador deseja que a bomba esteja ligada e o modo parada para quando o operador deseja que a bomba esteja parada.

Tabela 7 – Equipamentos utilizados na supervisão - Expedição do conhaque filtrado quente

Referência	Designação
	Rede
1XV3313O	Válvula saída tanque de conhaque filtrado 1313
1XV3314O	Válvula saída tanque de conhaque filtrado 1314
	Bomba de Manzani

Fonte: Autoria própria (2024).

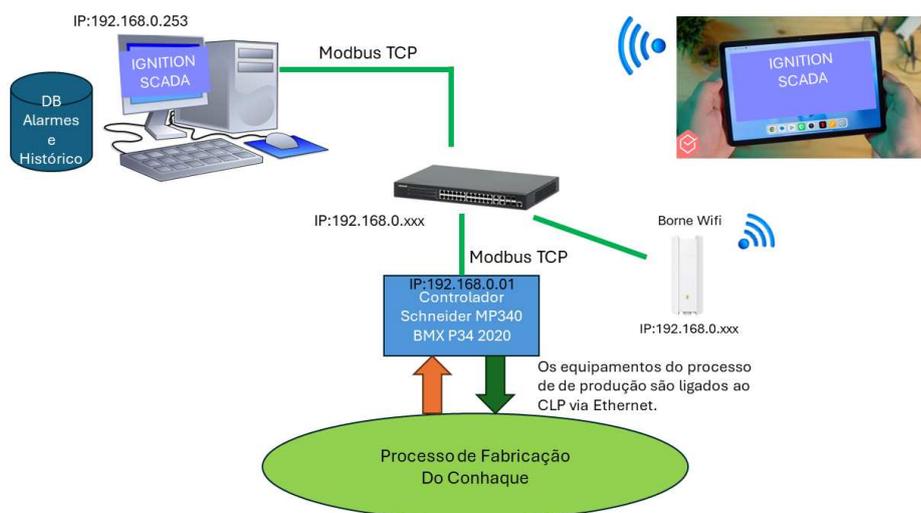
3.2 Desenvolvimento das Interfaces

As interfaces deste projeto foram desenvolvidas utilizando o software Ignition, isto se deve ao fato deste programa permitir o desenvolvimento de interfaces mais modernas e intuitivas para o usuário, uma vez que possibilita a criação destas interfaces utilizando linguagens de programação modernas como javascript, Python e CSS.

Além disto, para facilitar o monitoramento do processo de forma que o operador possa olhar os dados na interface de supervisão e no processo real ao mesmo tempo, um tablet móvel com a interface é desejado, além da interface no escritório do chefe de projeto. O Ignition possui módulo para dispositivos móveis, além de uma licença única para todos os dispositivos do cliente, o que favoreceu muito na escolha deste software para o desenvolvimento da supervisão.

Assim sendo, a arquitetura do projeto pode ser analisada na Figura 21.

Figura 21 – Arquitetura do projeto

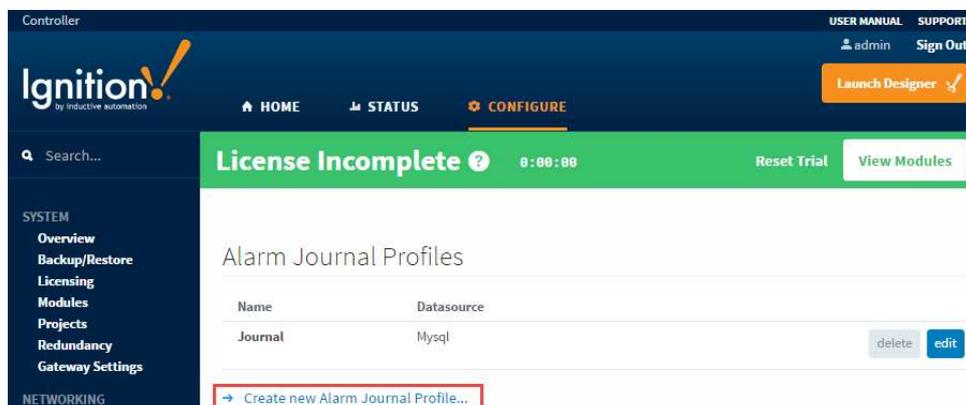


Fonte: Próprio Autor (2024).

O banco de dados, MySQL, que armazena o histórico de alarmes, já havia sido configurado no software Ignition, sendo feita apenas a criação do chamado "Alarm Journal

Profile”, responsável por armazenar informações históricas sobre alarmes no banco de dados. A Figura 22 representa um exemplo de criação de um Alarm Journal Profile.

Figura 22 – Criação Alarm Journal Profile



Fonte: (AUTOMATION, 2023).

Para este projeto, já existia toda a parte de instrumentação e controle através de um CLP Schneider M340 BMXP342020 que utiliza o software de configuração EcoStruxure TM Control Expert. O CLP somente possui o drive de comunicação no protocolo Modbus TCP.

Para mapear os parâmetros do processo no sistema Supervisório através do Modbus TCP é uma tarefa árdua pois eram uma centena de parâmetros e teria de fazer um a um colocando o endereço de cada variável do sistema.

Considerando que o endereçamento das variáveis mudava substancialmente cada vez que uma alteração importante era feita no programa do CLP durante as fases de teste na fabricante de conhaque, tornou-se um processo ainda mais repetitivo, cansativo e longo.

Deste modo, estudando as formas de exportação das variáveis e suas informações pelo programa Control Expert, foi averiguado que este processo poderia ser feito, gerando como resultado um arquivo do tipo .xsy com o mapeamento dessas variáveis. Um exemplo da descrição de variáveis neste arquivo pode ser observado no fragmento do código 3.1 .

Fragmento de Código 3.1 : Arquivo .xsy mapeamento de variáveis

```

1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <VariablesExchangeFile>
3   <fileHeader company="Schneider Automation" product="Control Expert V15.0
   ↪ -SP1 - 210706C" dateTime="date_and_time#2024-4-22-15:25:6" content="
   ↪ Fichier source variables" DTDVersion="41"></fileHeader>
4   <contentHeader name="Projet" version="0.0.5" dateTime="date_and_time#2024
   ↪ -4-4-15:29:49"></contentHeader>
5   <dataBlock>
6     ...

```

```
7      <variables name="MESURES" typeName="ARRAY[0..65] OF MEASURE"
8          ↪ topologicalAddress="%MW4800">
9          <instanceElementDesc name="[5]">
10             <attribute name="Alias" value="EXT_1LT3313"></attribute>
11         </instanceElementDesc>
12         ...
13 <DDTSource DDTName="MEASURE" version="1.16" dateTime="dt#2024-04-02-19:46:32
14     ↪ ">
15     <comment>Struct Version 1.0: MEASURE</comment>
16     <attribute name="TypeSignatureChecksumString" value="8300"></attribute>
17     <structure>
18         ...
19         <variables name="EN_HH" typeName="BOOL">
20             <comment>Activation du dfaut HH</comment>
21         </variables>
22         <variables name="EN_H" typeName="BOOL">
23             <comment>Activation du dfaut H</comment>
24         </variables>
25         <variables name="EN_L" typeName="BOOL">
26             <comment>Activation du dfaut L</comment>
27         </variables>
28         <variables name="EN_LL" typeName="BOOL">
29             <comment>Activation du dfaut LL</comment>
30         </variables>
31         <variables name="EN_HG" typeName="BOOL">
32             <comment>Activation du dfaut HG</comment>
33         </variables>
34         <variables name="MES" typeName="REAL">
35             <comment>Mesure</comment>
36         </variables>
37         ...
38     </structure>
39 </DDTSource>
```

Fonte: Próprio Autor (2024).

Como podemos observar no exemplo de código.xsy 3.1, este arquivo possui uma UDT chamada 'MEASURE' (linha 12), que significa medida em português. Esta UDT é utilizada como base para todos os instrumentos de medida deste projeto, assim sendo, esta UDT carrega as TAGs relacionadas aos medidores, como, por exemplo:

- EN_HH (linha 17): Variável do tipo booleano responsável por ativar o alarme que indica que o tanque está em estado muito alto de líquido.

- EN_L (linha 23): Variável do tipo booleano responsável por ativar o alarme que indica que o tanque está em estado baixo de líquido.
- MES (linha 32): Variável do tipo REAL responsável por passar para a interface de supervisão qual o nível de líquido no tanque.

Logo, um instrumento de medida como, por exemplo, o radar de nível LT3313 é uma tag do tipo `MESURE`, que carrega consigo as tags `EN_HH`, `EN_L` e `MES`, por exemplo. Assim sendo, o que é importante saber para o desenvolvimento da supervisão é que, caso seja necessária uma informação relativa ao nível do tanque LT3313, esta informação se encontra na tag do tipo `MESURE` "LT3313".

Além do estudo do funcionamento das tags e UDTs, foi necessário a compreensão da forma de endereçamento das mesmas. Para começar, foi essencial o entendimento de que as tags do tipo:

- Booleana ocupam meio registro (8 bits).
- Float ocupam 2 registros (4 bytes) e só podem estar em endereços pares.
- Int ocupam 1 registro (2 byte).

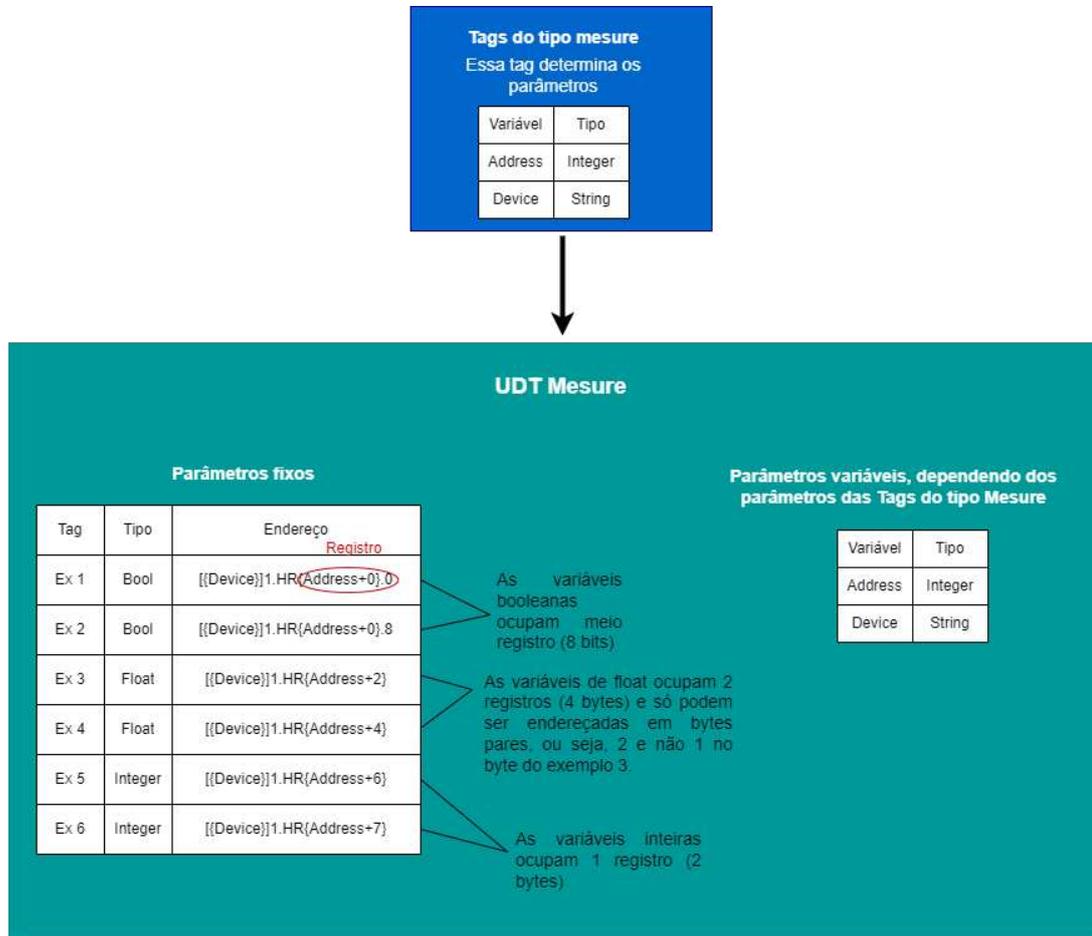
Assim sendo, o endereçamento das tags internas às UDTs deve seguir esta mesma lógica. Outro ponto importante é que as tags do tipo de uma UDT possuem um endereço de começo, por exemplo, é possível observar na linha 7 do código 3.1 que o endereçamento das tags do tipo da UDT `Mesure` começa no endereço 4800. Então suponho que as tags de dentro da UDT `Mesure` ocupem no total 500 bytes, a primeira tag do tipo `Mesure` terá endereço 4800, a segunda tag do tipo `Mesure` terá endereço 5300 e assim sucessivamente. O endereço de uma tag do tipo `Mesure` é usado como base para o endereçamento das tags internas à mesma.

Resumindo, o radar de nível LT3313 será uma Tag do tipo `Mesure` chamada "EXT_1LT3313", podemos observar na linha 8 do código 3.1 que ele é o quinto elemento declarado do tipo `Mesure`, logo, seguindo a lógica explicada anteriormente, seu endereço seria $4800 + 5 * 500 = 7300$. Assim, se considerarmos que a primeira Tag deste elemento de medida seria a tag 'EN_HH' que é booleana, esta teria o endereço 7300.0, se a segunda tag for 'EN_H' também do tipo booleana, a mesma teria o endereço 7300.8 e assim sucessivamente para todas as tags internas da tag "EXT_1LT3313".

A Figura 23 apresenta uma explicação visual do funcionamento deste endereço de variáveis. É possível observar que os parâmetros fixos das UDTs são os parâmetros que não mudam em todos os elementos do mesmo tipo, ou seja, as tags internas e o endereçamento destas tags. Já os parâmetros variáveis são os parâmetros que diferem entre os dispositivos

que são do mesmo tipo de UDT, por exemplo, o endereçamento de base para as tags internas às UDTs.

Figura 23 – Endereçamento de variáveis



Fonte: Próprio Autor (2024).

Após o entendimento da organização e do funcionamento do endereçamento das variáveis, foi feito um estudo para o endereçamento direto destas variáveis no software Ignition utilizando como base o arquivo .xsy. Sendo constatado que o Igniton aceita como arquivo de entrada para criação direta das tags e UDTs um arquivo do tipo .json. Logo, caso houvesse uma forma de converter diretamente o arquivo .xsy para um arquivo .json, este endereçamento poderia ser feito de forma direta.

Assim sendo, após o estudo dos arquivos.json e buscando resolver o problema de exaustão do endereçamento destas variáveis, foram desenvolvidos dois códigos em Python (ROBL, 2024) que têm como entrada o arquivo .xsy e como saída dois arquivos do tipo .json diferentes. O primeiro código tem enfoque em criar um arquivo .json com as informações das UDTs e o segundo para criar o arquivo .json contendo as informações das tags que utilizam estas UDTs. Dois arquivos diferentes foram necessários, pois para a criação de tags do tipo de uma UDT no programa Ignition as UDT já devem estar definidas dentro do escopo da plataforma. Por isso, primeiro o arquivo gerado com as informações das

UDTs é integrado à plataforma e, após isso, o segundo arquivo é utilizado para a criação das tags no Ignition.

O código 3.2 apresenta o arquivo .json de saída com as informações para a criação da UDT Measure e o código 3.3 mostra uma parte do segundo arquivo gerado contendo as informações para a criação de tags do tipo Measure.

Fragmento de Código 3.2 : Arquivo .json criação das UDTs

```
1 {
2 "tags": [
3 ...
4 {
5   "name": "MESURE",
6   "parameters": {
7     "Address": {
8       "dataType": "Int4",
9       "value": 0
10    },
11    "Device": {
12      "dataType": "String",
13      "value": "PLC"
14    }
15  },
16  "tagType": "UdtType",
17  "tags": [
18    ...
19    {
20      "opcItemPath": {
21        "bindType": "parameter",
22        "binding": "[{Device}]1.HR{Address+28}.8"
23      },
24      "valueSource": "opc",
25      "dataType": "Boolean",
26      "name": "EN_HH",
27      "tagType": "AtomicTag",
28      "opcServer": "Ignition OPC UA Server"
29    },
30    {
31      "opcItemPath": {
32        "bindType": "parameter",
33        "binding": "[{Device}]1.HR{Address+29}.0"
34      },
```

```
35     "valueSource": "opc",
36     "dataType": "Boolean",
37     "name": "EN_H",
38     "tagType": "AtomicTag",
39     "opcServer": "Ignition OPC UA Server"
40   },
41   {
42     "opcItemPath": {
43       "bindType": "parameter",
44       "binding": "[{Device}]1.HR{Address+29}.8"
45     },
46     "valueSource": "opc",
47     "dataType": "Boolean",
48     "name": "EN_L",
49     "tagType": "AtomicTag",
50     "opcServer": "Ignition OPC UA Server"
51   },
52   ...
53   {
54     "opcItemPath": {
55       "bindType": "parameter",
56       "binding": "[{Device}]1.HRF{Address+8}"
57     },
58     "valueSource": "opc",
59     "dataType": "Float4",
60     "name": "MES",
61     "tagType": "AtomicTag",
62     "opcServer": "Ignition OPC UA Server"
63   },
64   ...
```

Fonte: Próprio Autor (2024).

Fragmento de Código 3.3 : Arquivo .json criação das tags

```
1 {
2   "tags": [
3     ...
4     {
5       "name": "EXT_1LT3313",
6       "typeId": "MESURE",
7       "tagType": "UdtInstance",
8       "parameters": {
9         "Address": {
```

```

10         "value": "5010"
11     }
12 }
13 },
14 ...

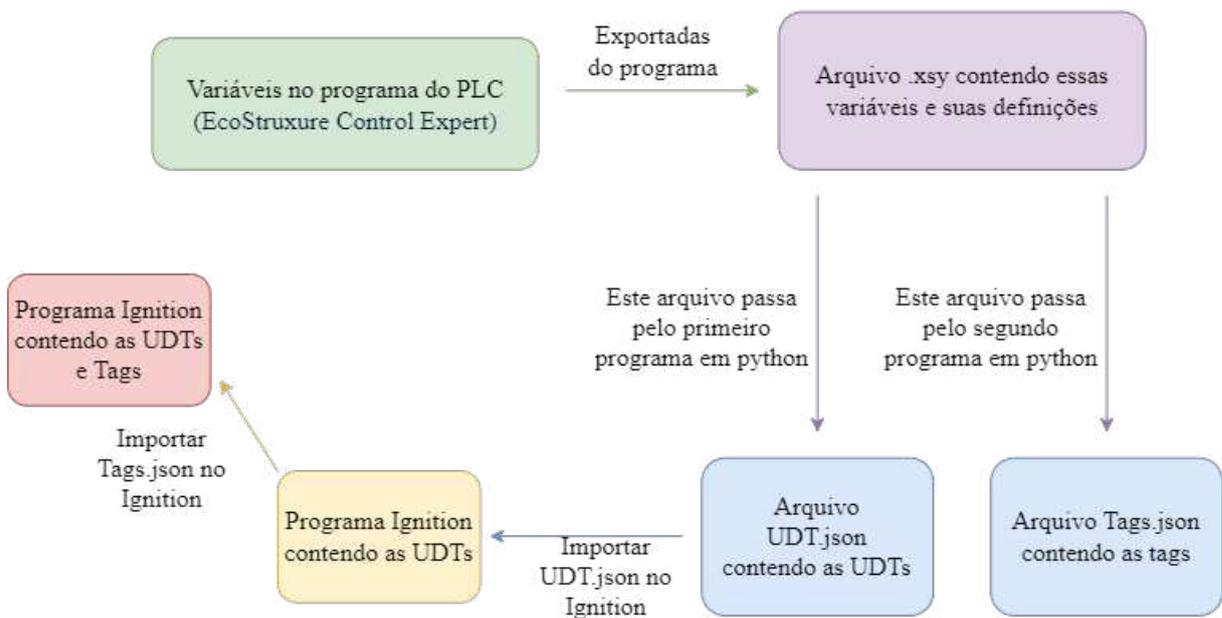
```

Fonte: Próprio Autor (2024).

Finalmente, analisando os fragmentos de código 3.2 e 3.3, é possível concluir que estes criam a UDT Measure com as Tags EN_HH, EN_H, EN_L e MES e após isto a tag EXT_1LT3313 do tipo Measure também é criada, exatamente como o esperado.

Assim sendo, com as variáveis já disponíveis no programa Ignition, o desenvolvimento das interfaces segue utilizando as instruções detalhadas na seção [Características do Sistema](#). Um resumo de todo este processo pode ser observado na Figura 24.

Figura 24 – Tratamento de variáveis



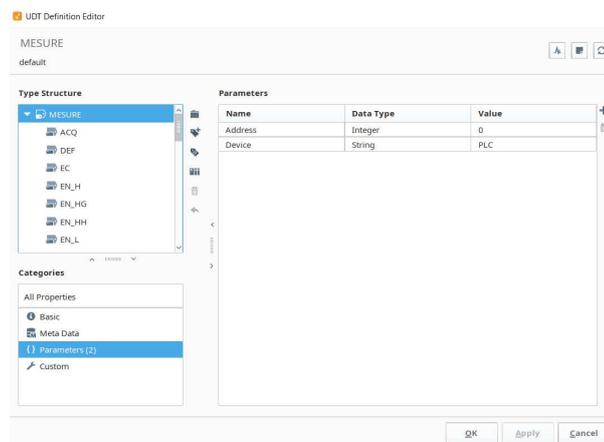
Fonte: Próprio Autor (2024).

Toda a interface de supervisão foi desenvolvida utilizando o módulo *Perspective* do software Ignition. Entendendo o funcionamento do sistema e com as tags disponíveis na plataforma de criação da supervisão, as interfaces foram desenvolvidas com a escolha dos elementos de ilustração e a ligação dos mesmos às suas respectivas tags, utilizando elementos das linguagens Javascript e CSS. O resultado pode ser observado na seção [Resultados e Discussões](#) deste documento.

4 Resultados e Discussões

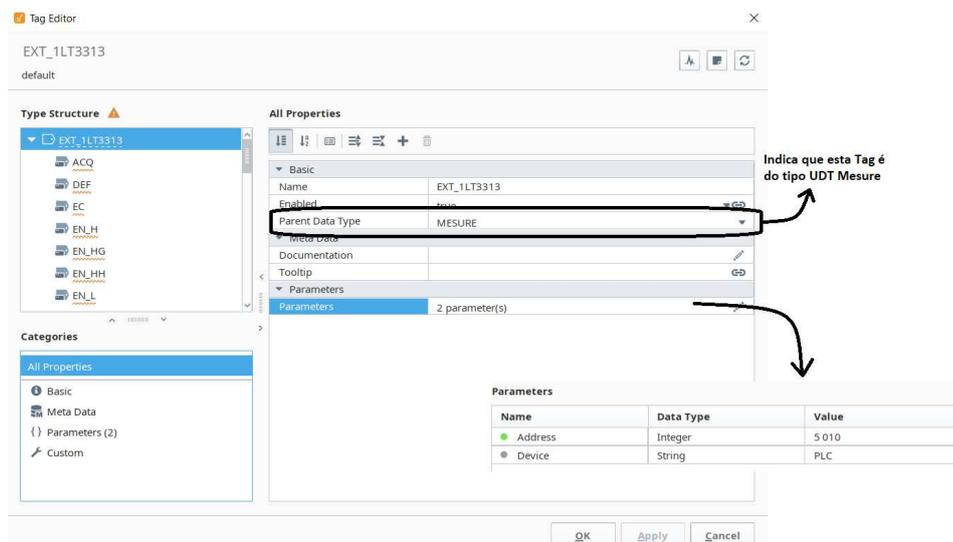
Com a utilização dos códigos em Python, a implementação e o endereçamento de variáveis no software Ignition foram feitos de forma mais eficiente, e as mesmas estavam disponíveis para conexão com a interface de supervisão, como ilustrado nas Figuras 25 e 26.

Figura 25 – Exemplo de UDT criada a partir do .json



Fonte: Próprio Autor (2024).

Figura 26 – Exemplo de uma Tag do tipo de uma UDT criada a partir do .json

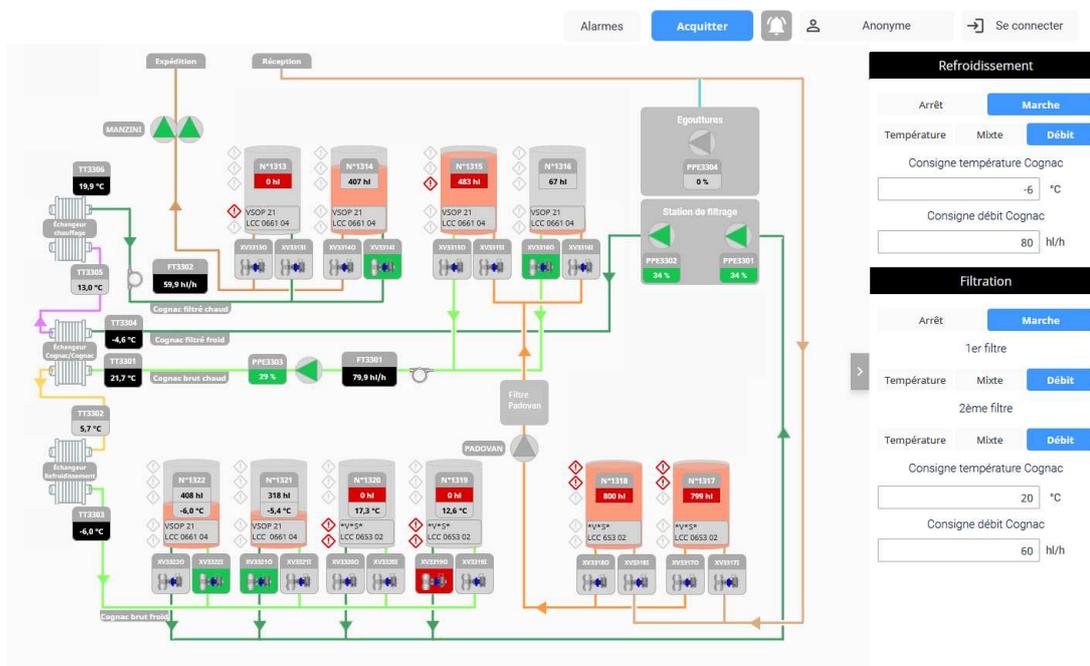


Fonte: Próprio Autor (2024).

A Figura 25 representa uma UDT, a UDT MESURE, que foi criada automaticamente a partir do arquivo.json gerado pelo código Python. Já a Figura 26 representa uma Tag, "EXT_1LT3313" do tipo MESURE, criada a partir de uma UDT, utilizando o segundo código.json gerado pelo programa Python.

Com os dispositivos disponíveis e os meios de comunicação com o CLP estabelecidos, a interface de supervisão final foi implementada. A tela que contém o processo de supervisão do processo de filtragem do conhaque pode ser observada na Figura 27.

Figura 27 – Interface de supervisão - Sistema Completo



Fonte: Próprio Autor (2024).

Para melhor entendimento desta interface final, ela será analisada por partes. Em relação aos tanques, esta interface permite que o usuário escolha o nome do tanque, e que o usuário seja alertado em caso de algum alarme relacionado ao tanque, como é possível analisar na Figura 28.

Figura 28 – Interface de supervisão - Tanque



Fonte: Próprio Autor (2024).

Além disso, quando o usuário clica em cima do ícone do tanque na supervisão, uma pop-up de configuração do tanque é aberta. Esta pop-up pode ser observada na Figura 29.

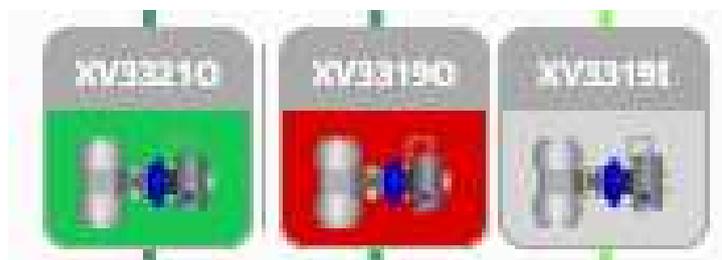
Figura 29 – Interface de supervisão - Pop-up Tanque



Fonte: Próprio Autor (2024).

Em relação às válvulas, a interface foi feita para que, quando houvesse qualquer tipo de defeito, a válvula ficasse rodeada por um campo vermelho, caso ela esteja em pleno funcionamento, a mesma fica cercada por um campo verde e, caso não esteja em utilização, a cor cinza é utilizada. A Figura 30 representa estes diferentes comportamentos em diferentes válvulas do circuito.

Figura 30 – Interface de supervisão - Válvulas



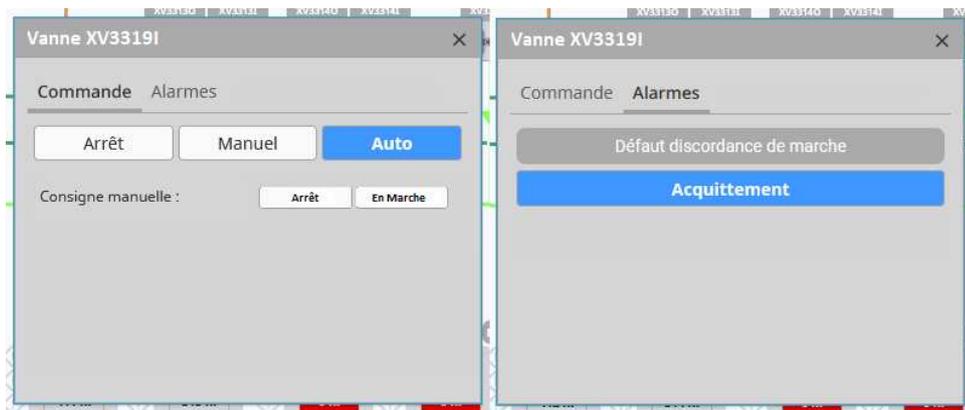
Fonte: Próprio Autor (2024).

Caso o operador clique em cima do ícone de qualquer válvula, uma pop-up, Figura 31 é aberta, esta permite a escolha do modo de funcionamento da válvula clicada e apresenta os alarmes.

A válvula mostrada é uma válvula de entrada, por isso possui a opção de funcionamento automático. No caso de válvulas de saída, a pop-up é a mesma, mas sem a opção de funcionamento automático. Outro ponto importante é que o alarme de discordância é para alertar ao operador que existe uma divergência entre o que foi pedido e o que foi efetuado, por exemplo, caso seja comandada a abertura de uma válvula, mas ela continue fechada, o problema de discordância será sinalizado. Este alarme está presente tanto para as válvulas quanto para as bombas.

As bombas no sistema são representadas por círculos com triângulos dentro, que é a representação comumente utilizada em diagramas e esquemas. Para retratar que a

Figura 31 – Interface de supervisão - Pop-up Válvulas

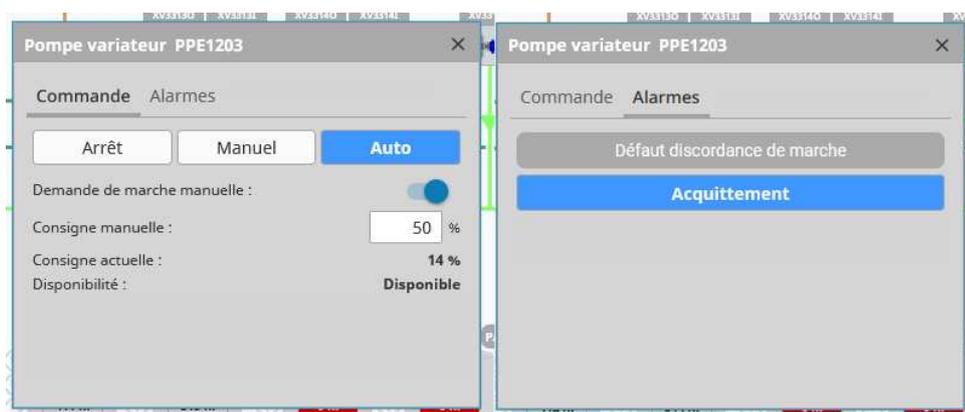


Fonte: Próprio Autor (2024).

bomba está em uso e sem defeitos, o triângulo fica verde; caso haja defeitos, o triângulo fica vermelho; e caso não esteja em uso, a cor cinza é apresentada. Caso a bomba não seja controlada pelo CLP, Bomba Padovan e Bomba Manzini, esta bomba terá apenas duas cores, verde para quando ela está em funcionamento e cinza para quando não está.

Da mesma forma que para os tanques e válvulas, ao clicar em um elemento que representa uma bomba controlada pelo CLP, PPE3301/ PPE3302/PPE3303, uma pop-up, Figura 32, é aberta. Esta permite escolher o modo de funcionamento da bomba e analisar o alarme.

Figura 32 – Interface de supervisão - Pop-up Bombas



Fonte: Próprio Autor (2024).

Além disto, o canto direito da supervisão, Figura 33, possui uma barra que permite regular as informações para funcionamento das bombas do sistema de resfriamento e do sistema de filtragem à terra.

No mais, é possível observar na Figura 27 da supervisão completa o botão "Alarmes", este botão leva o usuário à página com o histórico de alarmes, representada na Figura 34.

Além disto, é possível observar o botão "Acquiter". Este botão permite apagar

Figura 33 – Interface de supervisão - Regulação das Bombas

The image shows a mobile application interface for pump regulation, divided into two main sections: 'Refroidissement' (Cooling) and 'Filtration' (Filtration). Each section has a black header with white text. Below the headers, there are control buttons for 'Arrêt' (Stop) and 'Marche' (Start). The 'Refroidissement' section includes a 'Température' (Temperature) control with a 'Mixte' (Mixed) button and a 'Débit' (Flow) button. It features two input fields: 'Consigne température Cognac' (Cognac temperature setpoint) with a value of -6 °C, and 'Consigne débit Cognac' (Cognac flow setpoint) with a value of 80 hl/h. The 'Filtration' section also has 'Arrêt' and 'Marche' buttons. It includes two filter-specific controls: '1er filtre' (1st filter) and '2ème filtre' (2nd filter), each with 'Température' and 'Débit' buttons. Below these are two more input fields: 'Consigne température Cognac' with a value of 20 °C, and 'Consigne débit Cognac' with a value of 60 hl/h. A grey arrow button is visible on the left side of the interface.

Fonte: Próprio Autor (2024).

todo o histórico de alarmes e retornar só aqueles que ainda estão presentes depois de 10 segundos que o histórico foi apagado.

Finalmente, esta interface foi implementada no escritório do chefe de linha e no tablet, como pode ser observado na Figura 35.

A implementação desta interface moderna, que possibilita até mesmo andar no chão de fábrica com a interface em mãos de forma prática, facilitou a supervisão do processo. De forma que, o operador tem até mesmo a possibilidade de, ao mesmo tempo que verifica os equipamentos da linha, verificar as informações na supervisão.

Além disto, a organização da interface de forma intuitiva facilita a detecção de problemas na linha de filtragem e possibilita o total controle dos dispositivos de forma eficaz e rápida. O que, por consequência, diminui as perdas do processo e aumenta o lucro do fabricante.

Figura 34 – Interface de supervisão - Histórico de Alarmes

Active Time	Display Path	Priorité	State
09/27/2024 10:37:55	XV3318I/MSEC/Defaut débordement de cuve	Critique	Active, Acknowledged
09/27/2024 10:37:55	XV3317I/MSEC/Defaut débordement de cuve	Critique	Active, Acknowledged
09/27/2024 10:32:58	Niveau haut cuve 1317	Medium	Active, Acknowledged
09/27/2024 10:32:58	LT3317/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/27/2024 09:11:23	DP3302/HI/Alarme niveau haut	Medium	Active, Acknowledged
09/27/2024 09:11:23	DP3302/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/27/2024 09:07:22	DP3303/HI/Alarme niveau haut	Medium	Active, Acknowledged
09/27/2024 09:07:22	DP3303/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/26/2024 14:21:15	Niveau bas cuve 1313	Medium	Active, Acknowledged
09/26/2024 14:21:15	LT3313/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/26/2024 10:33:49	XV3319O/DISC_F/Defaut discordance à la fermeture	High	Active, Acknowledged
09/26/2024 10:32:40	Niveau très bas cuve 1319	High	Active, Acknowledged
09/26/2024 10:21:59	Niveau bas cuve 1319	Medium	Active, Acknowledged
09/26/2024 10:21:59	LT3319/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/25/2024 15:31:54	Niveau haut cuve 1318	Medium	Active, Acknowledged
09/25/2024 15:31:54	LT3318/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/25/2024 10:01:58	Niveau très bas cuve 1320	High	Active, Acknowledged
09/25/2024 09:52:22	Niveau bas cuve 1320	Medium	Active, Acknowledged
09/25/2024 09:52:22	LT3320/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/24/2024 10:44:04	Niveau haut cuve 1315	Medium	Active, Acknowledged
09/24/2024 10:44:04	LT3315/DEF/Defaut fonctionnement	High	Active, Acknowledged
09/11/2024 10:39:24	XV3320O/MSEC/Defaut débordement de cuve	Critique	Active, Acknowledged

Fonte: Próprio Autor (2024).

Figura 35 – Interface de supervisão implementada na linha de produção



Fonte: Próprio Autor (2024).

5 Conclusão

Este trabalho teve como objetivo descrever o desenvolvimento de uma interface SCADA para supervisão da linha de filtragem da produção de conhaque, aumentando a eficiência deste processo e diminuindo as perdas. O sistema composto consistia de sete etapas, que foram mapeadas em diferentes telas no supervisório para cada etapa, e que todo o sistema pode ser acessado em diferentes pontos da planta com sucesso.

Durante a fase de análise e levantamento das interfaces, foi necessário a entrevista com os “operadores” da planta para entender exatamente as necessidades de cada um. Esta fase foi importante para melhorar a qualidade do produto final.

Importante também foi o desenvolvimento do software em python que conseguia gerar a base de parâmetros Modbus para o Sistema Supervisório. Este software foi considerado super útil, pois conseguiu mitigar o esforço de gerar uma configuração de pontos grande, de forma rápida e confiável, poupando o trabalho manual e possíveis erros na configuração.

A supervisão criada unificou as informações dos diversos dispositivos da linha de produção (sensores, bombas..), em uma única tela, tornando a visualização e o controle da linha de filtração mais simples. A implementação de um tablet no sistema ofereceu mais flexibilidade ao operador.

Esta interface, por ser uma interface SCADA, permite que variáveis como temperatura e vazão possam ser analisadas em tempo real, além da possibilidade de análise de um histórico de alarmes. As representações visuais dos alertas da supervisão permitem também um fácil diagnóstico do sistema. No futuro, pretende-se adicionar alarmes via email/sms e auditivos a esta supervisão para que alertas graves sejam detectados, mesmo se o operador não estiver no escritório ou com a supervisão em mãos. Além disto, caso o tamanho do sistema aumente futuramente, um estudo sobre a mudança do protocolo de comunicação para um protocolo mais moderno, como por exemplo o OPC UA, será realizado.

Ao longo da execução do projeto, foi possível utilizar os conhecimentos adquiridos no curso de Engenharia de Controle e Automação, estes contribuíram para a criação das interfaces e como base para o desenvolvimento de códigos que tornam menos longos, ou seja, menos exaustivos, o processo de endereçamento de variáveis no Igniton. No mais, o entendimento da necessidade do utilizador final e uma comunicação constante foram essenciais para o sucesso da entrega final.

Referências Bibliográficas

3M. *Alimentos e Bebidas*. 2024. <https://www.3m.com.br/3M/pt_BR/alimentos-e-bebidas/aplicacoes/destilados/>. Acessado em: 04 de outubro de 2024. Citado na página 15.

ARAÚJO, J. A. d. et al. Implementação de um sistema provisório scada para controle e otimização da produção em uma indústria de bebidas. Garanhuns, 2023. Citado na página 13.

AUTIC. *Ignition Enterprise*. 2024. <<https://www.autic.se/ignition-enterprise/>>. Consulté le: 04 de novembro de 2024. Citado na página 12.

AUTOMACAO, K. *Sistema SCADA: O que é e como funciona?* 2024. <<https://www.korp.com.br/sistema-scada/>>. Consultado em: 04 de novembro de 2024. Citado na página 24.

AUTOMACAO, L. L. *CLP Controlador Lógico Programável Modicon M221 C/ Ethernet - 24 Entradas, 16 Saídas, 24VCC, 40 E/S - Schneider Electric*. 2024. <<https://www.loja.lysiautomacao.com.br/clp-controlador-logico-programavel-modicon-m221-c-ethernet-24-entradas-16-saidas-24vcc-40-e-s-schneider/18517>>. Consultado em: 05 de outubro de 2024. Citado na página 18.

AUTOMACAO, P. *Protocolos de Redes Industriais: Quais São os Principais?* 2024. <<https://pahcautomacao.com.br/protocolos-de-redes-industriais-quais-sao-os-principais/>>. Consultado em: 05 de outubro de 2024. Citado na página 20.

AUTOMATION, I. *Alarm Journal - Journal Properties and Tables*. 2023. <<https://www.docs.inductiveautomation.com/docs/7.9/alarms/alarms-journal-properties-and-tables>>. Consulté le: 01 de novembro de 2024. Citado na página 43.

AUTOMATION, I. *Creating User-Defined Types (UDTs)*. 2024. <<https://www.docs.inductiveautomation.com/docs/7.9/tags/creating-user-defined-types-UDTs>>. Consultado em: 05 de novembro de 2024. Citado na página 28.

BENALLEL, M. *Chapitre 1: Supervision Industrielle*. 2024. <<https://fr.slideshare.net/slideshow/chapitre1-supervision-industriellepdf/266264412>>. Consulté le: 28 mai 2024. Citado na página 25.

BORGES, L. E. *Python para desenvolvedores: aborda Python 3.3*. [S.l.]: Novatec Editora, 2014. Citado na página 28.

BRIGGS, D. E. *Brewing: science and practice*. [S.l.]: Woodhead Publishing, 2004. v. 108. Citado na página 13.

Centro de Produções Técnicas. *Linguagem de Programação JavaScript - Um Breve Histórico*. 2024. <<https://www.cpt.com.br/cursos-informatica-desenvolvimentodesoftwares/artigos/linguagem-de-programacao-javascript-um-breve-historico>>. Acessado em: 17 ago. 2024. Citado na página 28.

CONTROL.COM. *Introduction to Modbus*. 2023. <<https://control.com/technical-articles/introduction-to-modbus/>>. Consulté le: 08 juin 2024. Citado na página 20.

DAKOL. *O que é automação industrial?* 2024. <<https://www.dakol.com.br/o-que-e-automacao-industrial/>>. Consultado em: 05 de outubro de 2024. Citado na página 18.

DANEELS, A.; SALTER, W. *What is SCADA?* [S.l.: s.n.], 1999. Citado na página 12.

DYNAMICS, D. *Convergência entre tecnologia da informação e tecnologia operacional é necessária para garantir a continuidade do negócio*. 2017. <<https://www.datacenterdynamics.com/br/opiniões/convergência-entre-tecnologia-da-informação-e-tecnologia-operacional-é-necessária-para-garantir-a-continuidade-do-negócio>>. Consultado em: 05 de outubro de 2024. Citado na página 19.

FLANAGAN, D.; MATILAINEN, P. *JavaScript*. [S.l.]: Anaya Multimedia, 2007. Citado na página 28.

FOUNDATION, O. *OPC UA - Unified Architecture Overview*. 2024. <<https://opcfoundation.org/about/opc-technologies/opc-ua/>>. Consultado em: 13 de dezembro de 2024. Citado na página 23.

Inductive Automation. *Customers*. 2024. <<https://inductiveautomation.com/about/customers>>. Acessado em: 28 de maio de 2024. Citado na página 26.

Inductive Automation. *Ignition Architectures*. 2024. <<https://inductiveautomation.com/ignition/architectures>>. Acessado em: 17 ago. 2024. Citado na página 26.

KELLER, A. L. *Internet das coisas aplicada à indústria: dispositivo para interoperabilidade de redes industriais*. Universidade do Vale do Rio dos Sinos, 2017. Citado 3 vezes nas páginas 21, 22 e 23.

NUNES, M. J. R. *Pirâmide de Automação Industrial*. 2024. <<https://profes.com.br/profmarciojrunes/blog/piramide-de-automacao-industrial>>. Consulté le: 05 de outubro de 2024. Citado na página 18.

PETRUZELLA, F. D. *Controladores lógicos programáveis*. [S.l.]: AMGH Editora, 2014. Citado na página 19.

REALPARS. *OPC UA Basics*. 2024. <<https://www.realpars.com/blog/opc-ua-basics>>. Consulté le: 13 de dezembro de 2024. Citado 2 vezes nas páginas 23 e 24.

RealPars. *What is Ignition SCADA?* 2024. <<https://www.realpars.com/blog/ignition-scada>>. Acessado em: 17 ago. 2024. Citado na página 27.

ROBL, A. C. *EcoStruxure Control Expert to Ignition Variable Addressing*. 2024. <<https://github.com/AnaCarolinaRobl/EcoStruxure-Control-Expert-to-Ignition-Variable-Addressing/tree/main>>. Acesso em: 3 nov. 2024. Citado na página 46.

ROGGIA, L.; FUENTES, R. C. *Automacao industrial. Santa Maria: E-tec Brasil*, 2016. Citado na página 18.

ROISENBERG, L. *Implementação do Modbus RTU versus Modbus TCP/IP*. 2024. <<https://blog.lri.com.br/implementacao-do-modbus-rtu-versus-modbus-tcp-ip/>>. Consulté le: 28 de maio de 2024. Citado na página 20.

STORE, B. *Como é feito o conhaque: o processo de produção*. 2024. <<https://bartenderstore.com.br/como-e-feito-o-conhaque-o-processo-de-producao/>>. Consultado em: 15 de setembro de 2024. Citado na página 15.

Valete Editora. *Controle e Instrumentação: Comunicação em SCADA*. 2023. <https://www.editoravalete.com.br/site_controleinstrumentacao/arquivo/ed_119/special_com.html>. Acessado em: 17 ago. 2024. Citado na página 21.

Vista Automação. *Diferenciais da Plataforma Ignition*. 2023. <<https://vistaautomacao.com.br/2023/08/04/diferenciais-da-plataforma-ignition/>>. Acessado em: 17 ago. 2024. Citado na página 25.

WORLD, A. *Industrial Ethernet Growth Continues as Fieldbus Declines*. 2023. <<https://www.automationworld.com/communication/article/33009016/industrial-ethernet-growth-continues-as-fieldbus-declines>>. Consulté le: 05 de outubro de 2024. Citado na página 20.