A Major Project Report

On

ENHANCING THE SECURITY OF SCADA SYSTEM FOR MILK FACTORY

Submitted in partial fulfilment of the requirements for the Award of the Degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS & INSTRUMENTATION ENGINEERING

By

PAGADALA NAGA SRI SAI TEJ	**(218W1A1035)**
ARIGE DHANESH	**(218W1A1004)**
VADDI GOUTHAM KUMAR	**(218W1A1051)**

Under the guidance of

G.JALALU M. Tech

ASSISTANT PROFESSOR

DEPARTMENT OF ELECTRONICS AND INSTRUMENTATION ENGINEERING

VELAGAPUDI RAMAKRISHNA
SIDDHARTHA ENGINEERING COLLEGE

(Affiliated to JNTUK, Kakinada)

Sponsored by SAGTE, Kanuru, Vijayawada-520007

(Approved by AICTE, Accredited by NBA and NAAC A+ GRADE)

2024-2025

DEPARTMENT OF ELECTRONICS AND INSTRUMENTATION ENGINEERING

V.R. SIDDHARTHAENGINEERING COLLEGE

(AUTONOMOUS)

CERTIFICATE

This is to certify that the major project titled **"ENHANCING THE SECURITY OF SCADA SYSTEM FOR MILK FACTORY"** is a bonafide record of work done by **P. NAGA SRI SAI TEJ (218W1A1035), A. DHANESH (218W1A1004), V. GOUTHAM KUMAR (218W1A1051)** under my guidance and supervision and is submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Electronics and Instrumentation Engineering, V. R. Siddhartha Engineering college, (Autonomous, Affiliated to JNTUK) during the academic year 2024-2025.

(Mr. G. JALALU)
M. Tech.
Assistant professor,
Dept. of EIE.

(Dr. G. N. SWAMY)
M. Tech., Ph.D.
Professor & Head,
Dept. of EIE.

ACKNOWLEDGEMENTS

We would like to express our heartfelt gratitude to our guide **Mr. G. Jalalu, Assistant Professor, Department of Electronics and Instrumentation Engineering**, **VR Siddhartha Engineering College, Kanuru, Vijayawada.** She has given us tremendous support in both technical and moral front. Without her support and encouragement, we would never have been able to complete the Major Project successfully.

We are profoundly grateful to **Dr.G.N.Swamy, Professor, Head of the department, Electronics and Instrumentation Engineering, VR Siddhartha Engineering College, Kanuru, Vijayawada,** for his kind and whole hearted support and valuable guidance throughout the Major Project.

We are also grateful to **Project Review Committee members** for their supervision and valuable suggestions and other teaching and non-teaching staff for their kind co-operation.

We are thankful to **Dr. A.V.Ratna Prasad, Principal, VR Siddhartha Engineering College Kanuru, Vijayawada,** for his support during the completion of the Major Project.

We would like to express our gratitude to our project coordinator, **Dr. N. Swathi, Assistant Professor,** for his utmost support and encouragement and regular monitoring of our work throughout the Major Project.

We would also like to express our special gratitude and thanks to **Dr. P. Srinivas, Associate Professor** (and any other person) for providing the information to us.

P. NAGA SRI SAI TEJ

A. DHANESH

V. GOUTHAM KUMAR

CONTENTS

TOPIC **PAGE NO**

LIST OF FIGURES

ABSTRACT

Supervisory Control and Data Acquisition (SCADA) systems play a critical role in managing and automating industrial processes in modern milk factories. However, these systems are vulnerable to security threats that can compromise sensitive operations and data. This project focuses on enhancing the security of a SCADA system used in a milk factory by implementing advanced security measures using Siemens TIA (Totally Integrated Automation) Portal. The project involves securing key modules such as user administration, recipe management, and alarm systems to prevent unauthorized access and ensure operational integrity.

The **User administration** module was fortified by implementing role-based access control (RBAC), ensuring that only authorized personnel could access critical system functions. **Recipe management** was secured through encrypted data storage and controlled access, protecting proprietary process configurations from tampering. The **Alarm system** was enhanced to detect anomalies and generate real-time alerts, ensuring swift response to potential security breaches.

Siemens TIA Portal was leveraged to configure and monitor these security improvements, utilizing built-in security features such as user authentication, audit trails, and secure communication protocols. The result is a more robust SCADA system that ensures the confidentiality, integrity, and availability of critical data, ultimately enhancing the overall security and operational efficiency of the milk factory.

INTRODUCTION

In today's industrial landscape, Supervisory Control and Data Acquisition (SCADA) systems are essential for automating and monitoring critical processes in various industries, including milk processing plants. These systems manage key operations such as controlling machinery, monitoring temperature, and managing recipe configurations to ensure product quality and efficiency. However, with the increasing reliance on digital technologies, SCADA systems have become prime targets for cyber threats, making it imperative to strengthen their security.

This project focuses on enhancing the security of a SCADA system used in a milk factory by addressing potential vulnerabilities through secure user administration, recipe management, and alarm systems. Unauthorized access, data tampering, and delayed incident responses can lead to compromised production quality and financial losses. Therefore, implementing robust security measures is crucial to safeguard sensitive data and maintain operational reliability.

Siemens TIA (Totally Integrated Automation) Portal, a comprehensive engineering framework, was used to integrate these security enhancements. Through the implementation of role-based access control (RBAC), secure recipe management, and an enhanced alarm system, the project aims to establish a secure and resilient SCADA environment. The improved system ensures that only authorized personnel can modify critical configurations, recipes remain protected, and anomalies trigger timely alerts, thereby preventing potential disruptions.

This report outlines the methodologies, security measures, and results achieved through the integration of advanced security protocols in the SCADA system, ultimately contributing to the safety and efficiency of the milk factory's operations.

Moreover, as milk factories operate in a highly regulated environment, ensuring compliance with industry standards and cybersecurity best practices is essential. A compromised SCADA system can not only disrupt production but also lead to non-compliance with quality and safety regulations, resulting in significant reputational and financial damage. By leveraging Siemens TIA Portal's advanced features, such as encrypted communication, secure access management, and audit logging, this project aims to create a secure operational framework that mitigates these risks. The implementation of these security measures not only enhances system reliability but also provides real-time visibility into system activity, enabling prompt detection and response to any security incidents. Through these improvements, the milk factory’s SCADA system is better equipped to withstand cyber threats while maintaining uninterrupted and secure operations.

1.1 MOTIVATION

The motivation for enhancing the security of the SCADA system in a milk factory stems from the increasing threat landscape faced by industrial control systems (ICS). As milk processing involves highly sensitive operations—such as maintaining precise temperatures, managing ingredient recipes, and ensuring seamless production—any compromise in system security can result in significant quality control issues, production downtime, and financial losses. Cyberattacks targeting SCADA systems can lead to unauthorized access, data manipulation, and operational disruption, posing serious risks to product safety and regulatory compliance.

Additionally, with the growing adoption of Industry 4.0 and the Internet of Things (IoT), SCADA systems are becoming more interconnected, which further exposes them to cyber threats. This increased connectivity creates additional vulnerabilities that need to be addressed through proactive security measures. Ensuring the security of user administration, recipe management, and alarm systems is critical to preventing unauthorized modifications and ensuring the integrity of the production process.

Moreover, compliance with industry standards such as ISA/IEC 62443 and other cybersecurity frameworks has become mandatory for maintaining a secure production environment. By leveraging Siemens TIA Portal’s advanced security features, this project aims to strengthen system defences, protect sensitive data, and ensure operational continuity. Ultimately, the motivation behind this project is to safeguard the milk factory’s SCADA system against potential threats, ensuring product quality, regulatory compliance, and long-term operational efficiency.

1.2 PROBLEM DESCRIPTION

Milk processing factories rely on SCADA (Supervisory Control and Data Acquisition) systems to automate and control critical processes such as pasteurization, temperature regulation, and recipe management. However, as these systems become increasingly interconnected and digitized, they are exposed to various cybersecurity threats that can compromise the integrity, availability, and confidentiality of industrial processes. Unauthorized access, data tampering, and system disruptions can lead to severe consequences, including compromised product quality, regulatory non-compliance, and operational downtime.

One of the major vulnerabilities in SCADA systems is **User administration**, where weak or improperly managed access controls can allow unauthorized personnel to gain access to critical system functions. **Recipe management** is another potential risk, as manipulation of process configurations can alter production parameters, resulting in defective products. Additionally, the **Alarm system** is critical for detecting and responding to anomalies, but if it is not properly secured or configured, delays in detecting security breaches can result in significant losses.

Despite the availability of advanced security features, many SCADA systems in milk factories lack robust protection mechanisms that can mitigate these risks. Siemens TIA Portal, an industry-standard engineering platform, provides security tools that can enhance user management, protect recipe data, and ensure real-time anomaly detection through improved alarm systems. This project addresses these vulnerabilities by integrating advanced security protocols into the SCADA system, ensuring a secure and resilient operating environment for the milk factory.

.

LITERATURE SURVEY

Amarjit R and Hussain LR (2019) proposed a novel method for denoising impulse noise in colour images [1]. By combining fuzzy logic and Support Vector Machine (SVM) techniques, the approach effectively preserves image details while removing noise. Traditional denoising methods struggle with impulse noise due to its sudden and intense pixel value variations. The method integrates fuzzy logic to handle uncertainties in image data and SVM for accurate pixel differentiation. Experimental results likely demonstrate superior denoising performance, possibly evaluated using metrics like PSNR, SSI, or MSE.

The paper presents a denoising method leveraging neighbouring wavelet coefficients to preserve image features while removing noise [2]. Wavelet transform's ability to decompose signals into different frequency components is utilized, likely discussing in-depth methodology including coefficient selection and noise parameter estimation. Experimental results are provided, possibly evaluating denoising performance using metrics like PSNR, SSI, or MSE, showcasing effectiveness compared to other techniques. This contribution enhances image denoising by effectively utilizing spatial correlation between wavelet coefficients.

The paper discusses soft-thresholding as a denoising technique for signals corrupted by additive noise [3]. Soft-thresholding selectively attenuates coefficients below a threshold in the signal's transform, preserving crucial details. It likely explores theoretical aspects such as threshold function selection and optimal threshold determination, along with practical considerations like implementation and computational complexity. Experimental findings probably showcase soft-thresholding's efficacy through metrics like SNR, MSE, or visual comparisons, offering a robust denoising solution in signal processing.

The paper introduces wavelet shrinkage for adaptive denoising of signals or images by selectively thresholding coefficients [4]. It uses a soft-thresholding approach, preserving vital features while attenuating coefficients below a threshold. Theoretical discussions likely include threshold function selection and optimal threshold determination. Practical aspects, such as implementation details and computational efficiency, are expected, with experimental results demonstrating effectiveness via metrics like SNR or MSE.

Elad and Aharon's paper [5] proposes denoising with sparse representations and dictionaries to retain image features while removing noise. Sparse representations, from learned dictionaries, approximate images with few coefficients. It includes training dictionaries for better representation and finding sparse patches for denoising. Theoretical aspects cover dictionary learning, sparse coding, and practical considerations, with experimental results likely evaluating effectiveness using metrics like PSNR, SSI, or MSE.

The paper [6] reviews denoising methods in image processing, including traditional and modern techniques. Traditional methods like Gaussian smoothing or median filtering work on pixel values directly, while modern ones, such as wavelet-based or deep learning, use advanced algorithms. Modern techniques often offer superior denoising performance by exploiting complex image statistics models. The paper may cover hybrid approaches and method selection considerations based on image characteristics and denoising needs, serving as a valuable resource for understanding image denoising advancements.

Fodor and Kamath's study [7] likely investigates wavelet shrinkage's efficacy for denoising signals/images affected by additive noise. It probably explores various aspects like wavelet families, thresholding strategies, and parameters to assess denoising performance. Experimental results are expected with metrics like SNR, MSE, and visual comparisons against other techniques. Practical considerations such as computational complexity and parameter selection may also be discussed, enhancing understanding of wavelet shrinkage's applicability in denoising.

Image denoising, vital in image processing, involves noise removal while preserving key features, as discussed in the review [8]. Classical methods like Gaussian smoothing and median filtering offer simplicity and computational efficiency, while modern techniques utilize advanced algorithms and machine learning for improved performance. The review likely covers principles, merits, and limitations of these methods, aiding in holistic understanding, along with emerging trends and evaluation metrics. Overall, it serves as a comprehensive guide for researchers, practitioners, and enthusiasts in the field.

The paper provides a thorough overview of image denoising techniques, covering classical methods like spatial filtering and wavelet-based approaches, as well as modern advancements such as machine learning and deep learning [9]. It discusses the strengths, limitations, and applicability of different methods, considering factors like noise characteristics, image attributes, and computational resources.

Hasan and El-Sakka enhance the Wiener filtering step in the BM3D algorithm with an SSIM-optimized filter [10], leveraging the Structural Similarity Index (SSIM) for better local image structure and noise estimation. Experimental findings show enhanced denoising performance, particularly in complex texture regions, evaluated using metrics like PSNR and SSIM against original BM3D and alternative denoising methods.

The adaptive digital ridge let transform [11] adjusts parameters based on local image features, involving ridge detection and transform computation to identify linear features and decompose images. It adapts to ridge orientations and scales, aiding in denoising by preserving vital image features. Theoretical frameworks, including ridge detection algorithms and parameter adaptation, are discussed. Experimental results likely show effective denoising performance, evaluated with metrics like PSNR and SSIM.

The paper introduces a parallel version of the NLM denoising algorithm for remote sensing images on the Intel Xeon Phi platform [12]. It optimizes the algorithm for parallel computing, distributing workload across multiple cores. Techniques like data decomposition and task scheduling are employed for efficient parallelization. Experimental results show enhanced denoising quality and computational speedup, assessed using metrics like PSNR and processing time.

The Artificial Bee Colony algorithm, inspired by honeybees, iteratively seeks optimal solutions [13]. It comprises employed, onlooker, and scout bees, exploring the search space for solutions. The paper likely discusses its theoretical foundation, practical aspects, and experimental results. Performance metrics evaluate its effectiveness in optimization across domains.

The paper discusses image denoising [14] via iterative average filtering, aiming to remove noise while preserving image features. Iterative average filtering involves applying a spatial averaging filter repeatedly to refine the denoised image, reducing noise progressively. Theoretical discussions may include convergence properties, computational complexity, and practical considerations for different noise types and image characteristics. Experimental results likely demonstrate effectiveness through metrics like PSNR or SSIM, contributing to image processing with a computationally efficient denoising method.

The paper proposes a method to enhance ECG signal quality using empirical mode decomposition (EMD) and the non-local mean (NLM) technique [15]. EMD decomposes the signal into intrinsic mode functions (IMFs) to isolate noise components. NLM exploits signal similarities for denoising. Implementation details, including parameter selection and computational complexity, are likely discussed. Experimental results likely demonstrate effectiveness using metrics like SNR or MSE, contributing to biomedical engineering by enhancing ECG signal accuracy.

BM3D (Block-Matching 3D) is an image denoising algorithm with collaborative and Wiener filtering stages [16]. It groups similar patches into 3D blocks for denoising and employs Wiener filtering for final image denoising. The paper likely delves into BM3D's principles, including block matching, transform domain denoising, and Wiener filtering. It may cover theoretical foundations, practical implementation, optimization techniques, and experimental results demonstrating effectiveness.

Li and Suen propose a variant of Non-local Means (NLM) for image denoising, integrating principles from grey theory [17]. The method likely adapts NLM by incorporating grey relational analysis or grey prediction, enhancing its handling of uncertainty. The paper discusses theoretical foundations, algorithmic details, and experimental results, demonstrating improved denoising performance. By combining NLM with grey theory, the proposed method offers potential advancements in denoising effectiveness and robustness.

Li et al. propose a method to address mixed noises [18], offering theoretical foundations and algorithms for noise identification and separation. The paper likely discusses noise removal methodologies and integration for clean signal/image reconstruction. Experimental results showcasing effectiveness through metrics like SNR, MSE, or visual comparisons are expected, contributing to improved denoising accuracy and robustness.

Liu and Guo propose enhancements to median filtering for better denoising performance [19]. They modify filter arithmetic, adjust neighborhood selection, and dynamically change window size. Experimental results likely demonstrate effectiveness through metrics like PSNR, SSIM, or visual comparisons, contributing to improved noise reduction in image processing.

Liu and Tan propose an SVD-based watermarking scheme [20] to embed imperceptible information into digital media for ownership assertion. The scheme likely employs SVD to decompose the host media, allowing efficient embedding of the watermark. Theoretical discussions may cover embedding/extraction processes, robustness to attacks, and imperceptibility. Experimental results likely quantify effectiveness using metrics like PSNR or visual quality assessments.

Lu and Chou propose a denoising method utilizing the directional weighted-median filter for salt-and-pepper noise [21]. Modifications likely include adaptive window size, directional filtering, and post-processing. Theoretical discussions cover algorithmic details, parameter selection, and computational aspects. Experimental results demonstrate effectiveness through metrics like PSNR, SSIM, or visual comparisons.

The paper [22] introduces a stochastic denoising method adapting patch size based on local image characteristics for improved performance. Adaptive patch size dynamically adjusts denoising, enhancing spatial structure and texture detail capture. The method's stochastic nature suggests utilization of probabilistic models or random processes. Experimental results likely showcase effectiveness through metrics like PSNR, SSIM, or visual comparisons, advancing image denoising quality and robustness.

Impulse noise, or salt-and-pepper noise, and Gaussian noise are common in images, prompting the need for denoising filters [23]. The survey likely explores traditional methods like median, Gaussian, and adaptive filtering, as well as advanced techniques such as machine learning and deep learning-based approaches. The paper discusses theoretical foundations, algorithmic details, and practical considerations of each filter, including strengths, limitations, and optimal usage scenarios. Experimental results, including metrics like PSNR and SSIM, demonstrate the effectiveness and performance of the surveyed denoising filters.

PRNU, inherent in digital cameras, acts as a unique noise pattern for image authentication and forgery detection [24]. The paper likely introduces a method using binary similarity measures of PRNU to differentiate natural images from computer-generated ones. Techniques such as cross-correlation or machine learning classifiers trained on PRNU patterns may be involved. Experimental results likely demonstrate effectiveness in image origin identification, contributing to multimedia forensics.

Watermarking embeds imperceptible information into digital media for ownership assertion. The SVD-based scheme preserves perceptual quality while embedding watermarks [25]. Mohammad et al. likely enhance this scheme's robustness against attacks, potentially modifying embedding processes and incorporating error correction coding. Experimental results demonstrate effectiveness, quantified through metrics like BER and robustness against attacks, contributing to multimedia security by improving ownership protection.

Orchard, Ebrahimi, and Wong propose an NLM denoising acceleration method using SVD [26]. By decomposing image patches, computational complexity is reduced, enhancing efficiency. The paper likely discusses theoretical foundations, SVD integration details, and practical considerations. Experimental results are expected to demonstrate improved efficiency and denoising effectiveness, contributing to image processing.

Rohit, Abdu, and George propose a robust face hallucination technique employing adaptive learning [27]. This method likely adjusts parameters dynamically based on input or training data characteristics. The paper discusses theoretical foundations, algorithmic details, and high-frequency detail learning from training data to enhance low-resolution facial images. Experimental results likely include metrics like PSNR, SSIM, or visual comparisons, showcasing effectiveness in high-resolution facial reconstruction.

ShouDong, Yong, et al. propose a method using Gaussian super-pixels for image segmentation [28]. Gaussian super-pixels represent compact and perceptually meaningful image regions, followed by efficient segmentation using graph cuts. The paper likely discusses theoretical foundations, including Gaussian super-pixel generation and graph cuts optimization. Experimental results are expected, evaluating segmentation accuracy and computational efficiency, contributing to image processing with improved segmentation methods.

The paper by Song-Tao, Zhen-Xing, and Ning proposes a method using saliency maps and sub window search for target segmentation in infrared images [29]. They leverage saliency maps to highlight regions of interest and employ an efficient sub window search technique for accurate target localization. The fused saliency map integrates multiple saliency detection methods, while sub window search optimizes segmentation results iteratively. The paper likely discusses theoretical foundations, algorithmic details, and experimental results demonstrating effectiveness in segmenting targets in infrared images.

The paper by Thanh, Hai, Prasath, et al. introduces a two-stage filtering approach for high-density salt and pepper noise removal [30]. The method likely involves identifying and isolating noisy pixels before reconstructing them based on surrounding information. Theoretical foundations and algorithmic details of each filtering stage are discussed, along with practical considerations such as parameter tuning and computational complexity analysis. Experimental results likely demonstrate effectiveness through metrics like PSNR, SSIM, or visual comparisons, contributing to improved denoising quality in image processing.

Umam and Yunus propose using the Quaternion Wavelet Transform for image denoising, which extends wavelet analysis to quaternion-valued signals, suitable for complex data like colour images [31]. The QWT potentially offers a more comprehensive representation of image features, including spatial structures and directional information. The paper likely discusses the theoretical foundation of QWT, denoising algorithm employing QWT coefficients, and experimental results evaluating denoising effectiveness via metrics like PSNR or SSIM. Overall, the paper contributes to image denoising by introducing a novel approach based on QWT, promising improvements over traditional scalar wavelet methods.

Wang, Istepanian, and Yong propose using the Stationary Wavelet Transform (SWT) for denoising microarray images, aiming to preserve important features [32]. The SWT maintains data points in both input and output domains, suitable for stationary signals like images. The paper likely discusses SWT's theoretical foundations, its application to microarray image denoising, and the denoising algorithm based on SWT decomposition. Experimental results, evaluating denoising effectiveness through metrics like SNR and MSE, contribute to enhancing microarray image analysis.

Yahya, Tan, Su, et al. propose adaptive enhancements to BM3D denoising [33], integrating collaborative and Wiener filtering with adaptive techniques. These methods dynamically adjust filter parameters based on local image characteristics like noise variance and texture complexity. The paper discusses theoretical foundations, algorithmic details, and computational complexity analysis. Experimental results demonstrate effectiveness via metrics like PSNR, SSIM, contributing to improved denoising quality and robustness in various noise conditions.

Yang et al. propose employing NSST for image representation and TWSVM for denoising [34]. NSST decomposes images into shear let coefficients capturing spatial and frequency data, while TWSVM learns mappings between noisy and clean patches. The paper likely discusses NSST and TWSVM's theoretical foundations, algorithmic integration, and model training. Experimental results may demonstrate denoising effectiveness using metrics like PSNR, SSIM, or visual comparisons, contributing to image processing with improved denoising quality.

The paper [35] introduces a novel image denoising method using sparse gradients and a coupled system. It exploits sparsity in the gradient domain to effectively remove noise while preserving image details. The approach likely involves representing the image as a coupled system and discussing theoretical foundations and algorithmic details. Experimental results are provided, evaluating denoising performance using metrics like PSNR and SSIM, contributing to improved image processing techniques.

The paper introduces a cascaded neural network architecture for seamless handling of multiple image restoration tasks [36]. Each network specializes in tasks like inpainting, deblurring, or denoising, working sequentially. It discusses deep learning theories, training procedures, and optimization strategies for restoration. Experimental results demonstrate effectiveness using metrics like PSNR, SSIM, and visual comparisons, contributing to versatile image restoration.

PROPOSED SYSTEM

3.1 DESCRIPTION OF THE PROPOSED SYSTEM

Milk processing factories rely on SCADA (Supervisory Control and Data Acquisition) systems to automate and control critical processes such as pasteurization, temperature regulation, and recipe management. However, as these systems become increasingly interconnected and digitized, they are exposed to various cybersecurity threats that can compromise the integrity, availability, and confidentiality of industrial processes. Fig. 3.1. shows the overall structure of the proposed system.

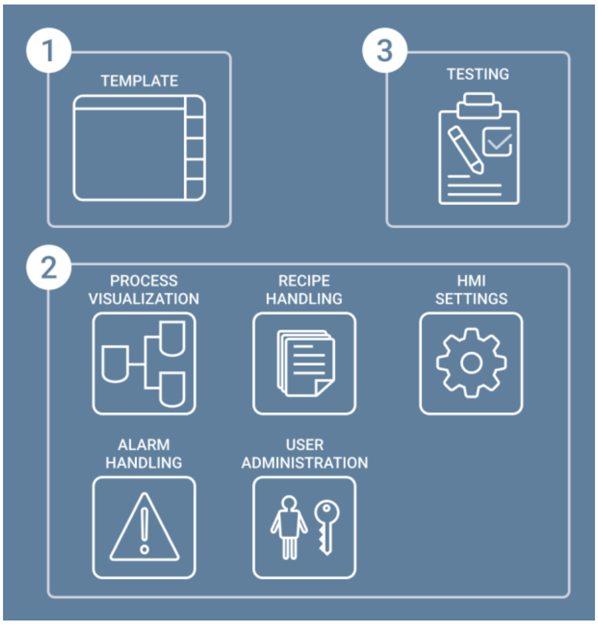


Fig. 3.1. Schematic diagram of the proposed system

Fig. 3.2 shows the template of the proposed system. Despite the availability of advanced security features, many SCADA systems in milk factories lack robust protection mechanisms that can mitigate these risks. Siemens TIA Portal, an industry-standard engineering platform, provides security tools that can enhance user management, protect recipe data, and ensure real-time anomaly detection through improved alarm systems. This project addresses these vulnerabilities by integrating advanced security protocols into the SCADA system, ensuring a secure and resilient operating environment for the milk factory.

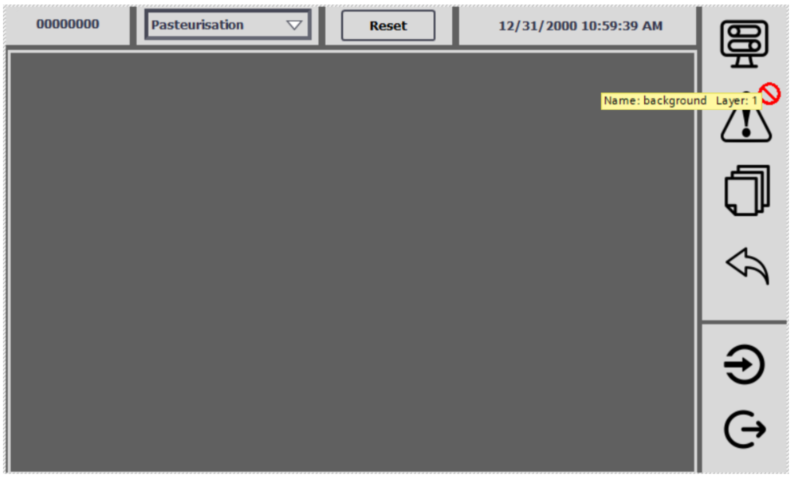


Fig. 3.2. Schematic diagram of Template

3.1.1 Preprocessing

The increasing reliance on automation and digital control systems in the dairy industry has revolutionized the efficiency and precision of milk processing operations. However, as these systems become more interconnected and complex, they also become more vulnerable to cyber threats and security breaches. Ensuring the security of SCADA (Supervisory Control and Data Acquisition) systems used in milk factories is no longer optional but a necessity to protect critical processes, sensitive data, and operational integrity.

This project, titled "Enhancing the Security of SCADA System for Milk Factory", focuses on strengthening the security framework of a SCADA system by addressing key vulnerabilities in user administration, recipe management, and alarm systems using Siemens TIA Portal. Through the implementation of advanced security measures such as role-based access control (RBAC), encrypted recipe storage, and real-time anomaly detection, this project aims to establish a secure and resilient operational environment.

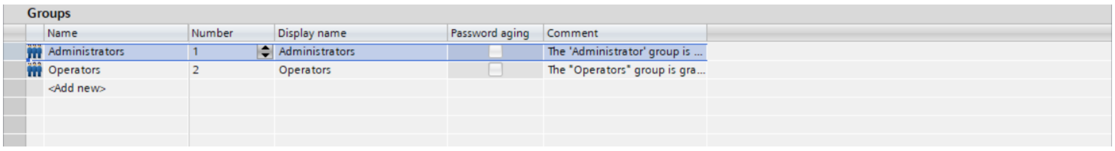
The journey of this project involved understanding the intricacies of SCADA systems in milk processing, identifying potential security risks, and applying appropriate solutions to mitigate these risks effectively. Siemens TIA Portal, with its comprehensive automation and security features, was instrumental in implementing these enhancements, ensuring a seamless integration of security protocols into the existing SCADA infrastructure.

This report provides a detailed overview of the project's objectives, methodologies, and outcomes, highlighting how the proposed security improvements contribute to safeguarding the milk factory's operations. It is hoped that the insights gained through this project will serve as a valuable reference for industries seeking to enhance the security of their SCADA systems and maintain operational excellence.

3.1.2 Creating the User Administration

Creating user administration for the HMI involves in different steps like user groups, groups, authorizations.

Step 1: Creating user groups is an essential step in enhancing the security of a SCADA system by implementing role-based access control (RBAC). In a milk factory, user groups can be configured based on different roles, such as operators, supervisors, and administrators, each with specific access privileges. This ensures that only authorized personnel can modify critical system parameters, access sensitive data, or make configuration changes. By assigning permissions at the group level, the system reduces the risk of human error and unauthorized access. Using Siemens TIA Portal, user groups can be easily created and managed, ensuring a secure and organized approach to system access.

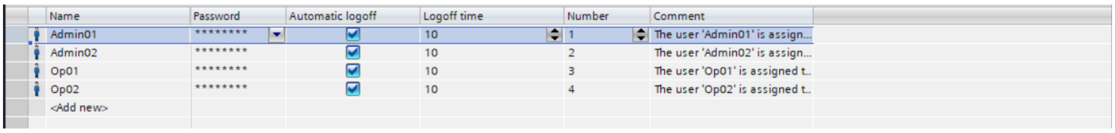


Groups

Name	Number	Display name	Password aging	Comment
Administrators	1	Administrators	□	The 'Administrator' group is ...
Operators	2	Operators	□	The "Operators" group is gra...
<Add new>				

Fig. 3.3. Creating user groups

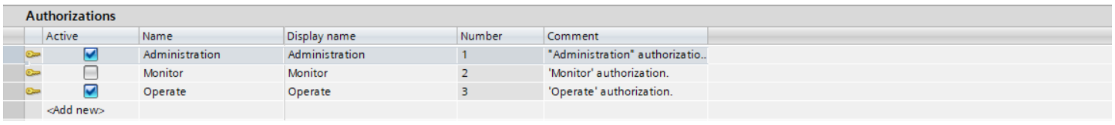
Step 2: Creating user groups in a SCADA system helps control access by assigning specific permissions to different roles, such as operators, supervisors, and administrators. This approach ensures that only authorized users can access or modify critical system settings. Using Siemens TIA Portal, user groups can be efficiently configured to enhance security and streamline access management.



Name	Password	Automatic logoff	Logoff time	Number	Comment
Admin01	********	☑	10	1	The user 'Admin01' is assign...
Admin02	********	☑	10	2	The user 'Admin02' is assign...
Op01	********	☑	10	3	The user 'Op01' is assigned t..
Op02	********	☑	10	4	The user 'Op02' is assigned t..
<Add new>					

Fig. 3.4. Creating users

Step 3: Authorizations in a SCADA system define the level of access and control granted to different user groups based on their roles. By assigning appropriate authorizations, critical operations such as modifying recipes, managing alarms, and configuring system settings can be restricted to authorized personnel only. Siemens TIA Portal allows for precise authorization management, ensuring enhanced security and operational control.



Authorizations

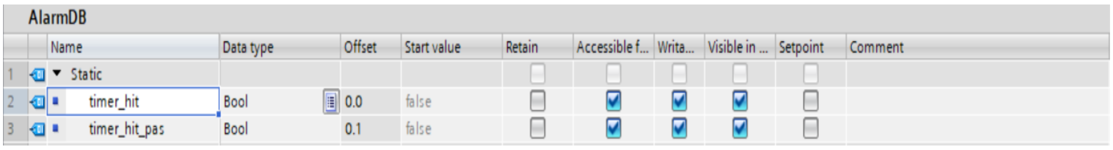
Active	Name	Display name	Number	Comment
☑	Administration	Administration	1	"Administration" authorizatio..
□	Monitor	Monitor	2	'Monitor' authorization.
☑	Operate	Operate	3	'Operate' authorization.
<Add new>				

Fig. 3.5. Assigning authorizations

3.1.3 Creating Alarm

Step 1: Creating a database (DB) for the alarm system in a SCADA environment is essential for efficiently storing, managing, and retrieving alarm-related data. The alarm database captures critical information such as alarm types, timestamps, priority levels, acknowledgment status, and associated process parameters. This data ensures that all alarm events are logged for future analysis, enabling operators to identify recurring issues and optimize system performance.

To ensure optimal performance, the database size and memory allocation should be carefully planned. Depending on the alarm frequency and retention period, storage requirements can vary, with high-frequency alarm systems requiring more storage capacity. Implementing data archiving and automatic cleanup mechanisms helps manage memory efficiently, preventing database overload and ensuring long-term system reliability. Siemens TIA Portal allows seamless integration of the alarm system with a secure and optimized database, ensuring smooth data management and retrieval.



AlarmDB

	Name	Data type	Offset	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	▼ Static				□	□	□	□	□	
2	timer_hit	Bool	0.0	false	□	☑	☑	☑	□	
3	timer_hit_pas	Bool	0.1	false	□	☑	☑	☑	□	

Fig. 3.6. Database Block of alarm

Step 2: Discrete alarms, also known as digital or binary alarms, are triggered when a predefined condition or state changes between two distinct values, typically **ON/OFF**, TRUE/FALSE, or **1/0**. These alarms play a crucial role in monitoring critical processes in a milk factory by alerting operators when abnormal conditions occur, such as temperature deviations, valve malfunctions, or equipment failures.

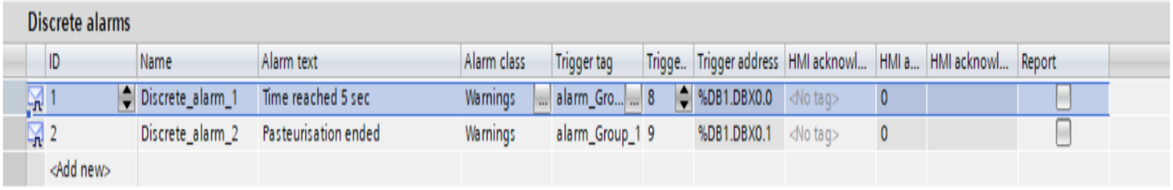
Steps for Creating Discrete Alarms:

1. Define Alarm Conditions: Identify the conditions that will trigger the alarm, such as a sensor exceeding a threshold or a motor switching to an unexpected state.
2. Configure Alarm Tags: Create digital input tags in Siemens TIA Portal to monitor specific states of devices or process variables.
3. Set Alarm Parameters: Define alarm properties, including priority level (low, medium, high), acknowledgment requirements, and reset conditions.
4. Assign Alarm to HMI/SCADA Interface: Link the alarm tags to Human-Machine Interface (HMI) displays for real-time visualization and notification.
5. Test and Validate Alarms: Simulate different conditions to verify that the alarms trigger correctly and notify operators as expected.

Details of Discrete Alarm Configuration:

- Alarm ID: Unique identifier for each discrete alarm.
- Trigger Condition: Binary condition that activates the alarm.
- Priority Level: Determines the urgency of the alarm.
- Alarm Action**:** Specifies actions, such as activating an audible alert or sending notifications.
- Acknowledgment Requirement**:** Indicates whether operator acknowledgment is required to reset the alarm.

Using Siemens TIA Portal, discrete alarms can be easily configured, visualized, and managed to ensure timely detection of critical events, enhancing the safety and efficiency of the milk factory’s operations.



Discrete alarms

ID	Name	Alarm text	Alarm class	Trigger tag	Trigge..	Trigger address	HMI acknowl...	HMI a...	HMI acknowl...	Report
1	Discrete_alarm_1	Time reached 5 sec	Warnings	alarm_Gro...	8	%DB1.DBX0.0	<No tag>	0		□
2	Discrete_alarm_2	Pasteurisation ended	Warnings	alarm_Group_1	9	%DB1.DBX0.1	<No tag>	0		□
<Add new>										

Fig. 3.7. Database Block of Discrete alarm

Discrete alarms offer high reliability and minimal false positives when properly configured, ensuring that only genuine events trigger notifications. Additionally, alarm data is logged for historical analysis, which helps in identifying recurring issues and improving process efficiency. By leveraging discrete alarms effectively, milk factories can enhance system security, minimize downtime, and ensure product quality while maintaining compliance with industry standards.

Step 3: Alarm classes are used to categorize alarms based on their priority, severity, and type, allowing operators to quickly assess and respond to critical system events. In a milk factory's SCADA system, defining alarm classes helps prioritize alarms and ensures that the most urgent issues are addressed promptly. Siemens TIA Portal provides the capability to configure and manage multiple alarm classes, ensuring better alarm management and improved operational efficiency.

Common Alarm Classes and Their Details:

1. Critical Alarms (Emergency/High Priority)
   - Description: Indicates severe conditions that require immediate attention to prevent equipment failure, safety hazards, or production downtime.
   - Examples: Equipment malfunction, high-temperature deviations, and power failures.
   - Action Required: Immediate response and acknowledgment by operators.
2. Warning Alarms (Medium Priority)
   - Description: Signals abnormal process conditions that may not immediately impact production but could escalate if not addressed.
   - Examples: Pressure deviations, abnormal valve positions, or temperature approaching critical limits.

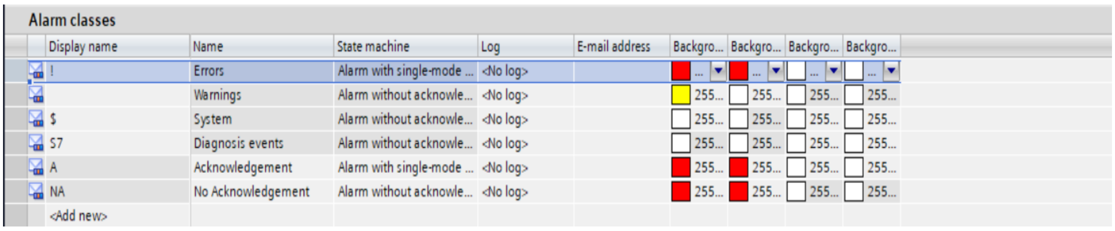
   - Action Required: Timely intervention to prevent potential system failure.

3. Information Alarms (Low Priority)
   - Description: Provides system information or status updates that may not require immediate action but should be monitored.
   - Examples: System start/stop notifications, device status changes, or operator actions.
   - Action Required: Acknowledgment for logging purposes, but no immediate corrective action.
4. Maintenance Alarms
   - Description: Notifies operators or maintenance teams about preventive maintenance schedules or system diagnostics.
   - Examples: Lubrication reminders, filter replacement, or scheduled inspections.
   - Action Required: Scheduled maintenance to ensure system reliability.
5. Process Alarms
   - Description: Monitors deviations in process parameters that may affect product quality or system efficiency.
   - Examples: Incorrect recipe parameters, unexpected flow rates, or temperature inconsistencies.
   - Action Required: Operator intervention or parameter adjustments.

Configuring Alarm Classes in Siemens TIA Portal:

- Define specific alarm classes in the alarm configuration panel.
- Assign alarm tags to corresponding classes based on priority and event type.
- Set appropriate notification and acknowledgment rules for each class.

By properly defining alarm classes, operators can focus on the most critical issues first, ensuring smooth and safe operation of the milk factory’s SCADA system.



Alarm classes

Display name	Name	State machine	Log	E-mail address	Backgro...	Backgro...	Backgro...	Backgro...
!	Errors	Alarm with single-mode ...	<No log>		🟥 ... ▼	🟥 ... ▼	□ ... ▼	□ ... ▼
	Warnings	Alarm without acknowle...	<No log>		🟨 255...	□ 255...	□ 255...	□ 255...
$	System	Alarm without acknowle...	<No log>		□ 255...	□ 255...	□ 255...	□ 255...
S7	Diagnosis events	Alarm without acknowle...	<No log>		□ 255...	□ 255...	□ 255...	□ 255...
A	Acknowledgement	Alarm with single-mode ...	<No log>		🟥 255...	🟥 255...	□ 255...	□ 255...
NA	No Acknowledgement	Alarm without acknowle...	<No log>		🟥 255...	🟥 255...	□ 255...	□ 255...
<Add new>								

Fig. 3.8. Database Block of Alarm Classes

Siemens TIA Portal enables seamless configuration of these alarm classes, allowing operators to assign appropriate priority levels and define acknowledgment and reset conditions. By organizing alarms into well-defined classes, milk factories can streamline their alarm management processes, minimize downtime, and ensure efficient operations.

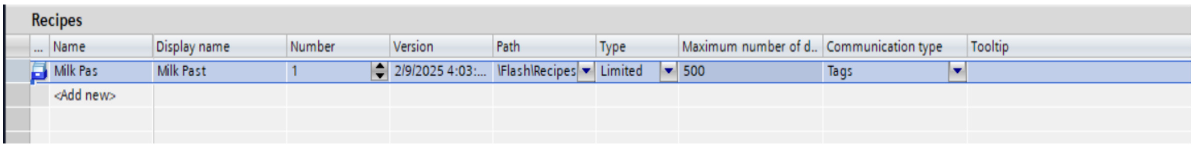
3.1.4 Creating Recipie

Step 1: Recipe

The recipe for Milk Pasteurization (Milk Past) in the Siemens TIA Portal is configured to handle process parameters essential for maintaining consistent product quality. Recipes in a SCADA system allow operators to store, retrieve, and apply a set of predefined values for process variables, ensuring that different batches follow the same operational standards. In this configuration, the Limited type indicates that the number of datasets stored in the system is capped at 500, ensuring that the memory allocated for recipes is managed efficiently. Each dataset contains a unique set of parameters such as temperature, pressure, and timing values required for the pasteurization process.

The Flash Recipes path suggests that the recipe is stored in non-volatile flash memory, making the data resistant to power loss and ensuring that recipe settings are retained after system reboots. Using Tags as the communication type means that the recipe values are dynamically mapped to PLC tags, enabling real-time updates of process parameters during operation. Additionally, recipe management in TIA Portal allows for easy versioning and traceability, making it possible to track changes over time and revert to previous versions if necessary. Since no tooltip or additional comments have been configured, operators would need to rely on documentation or internal notes to understand the specifics of this recipe.

In industrial automation, a recipe is a predefined set of parameters or values used to control and configure a process, ensuring consistency and repeatability across different production batches. In the case of the Milk Pasteurization (Milk Past**)** recipe configured in **Siemens TIA Portal**, the recipe manages critical process variables such as temperature, pressure, flow rate, and processing time. Recipes in TIA Portal allow operators to quickly switch between different product configurations without manually adjusting process parameters, reducing downtime and minimizing errors. The Limited type of recipe indicates that the maximum number of datasets or variations that can be stored for this particular recipe is restricted to 500, ensuring that system resources are optimally utilized. Each dataset within the recipe can contain a different combination of parameters, allowing the same recipe to be adapted for various operating conditions. The recipe is stored in the Flash Recipes directory, which suggests that the data is saved in non-volatile flash memory, ensuring that the recipe parameters are retained even after system reboots or power failures.



Recipes

...	Name	Display name	Number	Version	Path	Type	Maximum number of d..	Communication type	Tooltip
	Milk Pas	Milk Past	1	2/9/2025 4:03:...	\Flash\Recipes	Limited	500	Tags	
	<Add new>								

Fig. 3.9. Database Block of Recipe

Benefits of Recipe Configuration in TIA Portal:

- **Consistency and Quality Control:** Ensures that each batch follows the same process parameters, maintaining product quality.
- **Reduced Setup Time:** Switching between different recipes is quick and seamless, minimizing production downtime.
- **Improved Traceability:** Version control and timestamping help monitor changes and maintain a history of modifications.
- **Flexibility and Adaptability:** Allows multiple datasets to be stored under a single recipe, accommodating different operational scenarios.

Step 2: Data Records

The image shows the Data Records section configured for the recipe in Siemens TIA Portal. Data records allow multiple sets of parameters to be stored under a single recipe, providing flexibility to handle different operating scenarios. In this configuration, two data records have been defined:

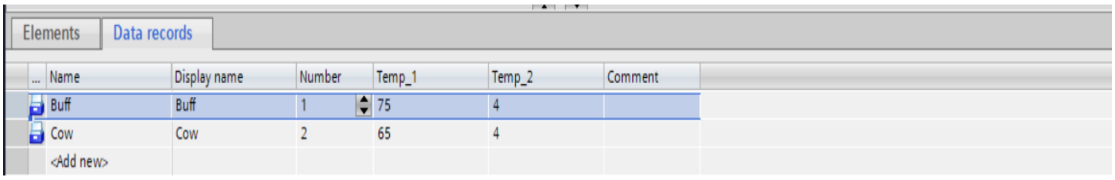
1. **Buff (Buffalo Milk Pasteurization):**
   - Number: 1
   - Temp_1**:** 75°C
   - Temp_2**:** 4°C
   - Comment: No additional comments provided
2. **Cow (Cow Milk Pasteurization):**
   - Number: 2
   - Temp_1: 65°C

- Temp_2: 4°C
- Comment: No additional comments provided

Each data record corresponds to a different type of milk, with defined temperature settings for pasteurization. Temp_1 appears to represent the heating or pasteurization temperature, while Temp_2 is likely the cooling or storage temperature. For buffalo milk, the pasteurization temperature is set at 75°C, while for cow milk, it is set at 65°C, with both types being cooled to 4°C after processing.

Benefits of Using Data Records:

- Flexibility and Efficiency**:** Operators can easily switch between different datasets without reconfiguring process parameters manually, reducing setup time and increasing production efficiency.
- Consistency and Accuracy: Data records ensure that the correct values are consistently applied for each product type, minimizing errors and maintaining quality standards.
- Scalability: Additional data records can be added to accommodate new product variations, enhancing the adaptability of the system.
- Traceability: Version control and timestamping allow operators to track changes made to data records, ensuring traceability and compliance with quality standards.



...	Name	Display name	Number	Temp_1	Temp_2	Comment
	Buff	Buff	1	75	4	
	Cow	Cow	2	65	4	
	<Add new>					

Fig. 3.10. Database Block of Data Records

Step 3: Elements

the **Elements** section of the recipe configuration in **Siemens TIA Portal**, where individual parameters or variables are defined and linked to the associated PLC tags. In this configuration, two elements have been created:

1. **Temp_1:**
   - **Display Name:** Temp_1
   - **Tag:** Milk Pas_temp_1
   - **Data Type:** Integer (Int)
   - **Data Length:** 2 bytes (16-bit integer)
   - **Default Value:** 0
   - **Minimum Value:** -32,768
   - **Maximum Value:** 32,767
   - **Decimal Places:** 0
2. **Temp_2:**
   - **Display Name:** Temp_2
   - **Tag:** Milk Pas_temp_2
   - **Data Type:** Integer (Int)
   - **Data Length:** 2 bytes (16-bit integer)
   - **Default Value:** 0
   - **Minimum Value:** -32,768
   - **Maximum Value:** 32,767
   - **Decimal Places:** 0

These elements define the process parameters required for the pasteurization process and are linked to the respective PLC tags **Milk Pas_temp_1** and **Milk Pas_temp_2**. The **data type** for both elements is **Int**, which uses 2 bytes (16-bit) to store numerical values. The **default value** is set to **0**, and the allowable range of values for these parameters spans from **-32,768 to 32,767**, which is the typical range for a signed 16-bit integer.

Temp_1 and **Temp_2** likely represent key temperature parameters in the pasteurization process, with **Temp_1** being the heating temperature and **Temp_2** being the cooling or storage temperature. Since the **decimal places** are set to **0**, these values are treated as whole numbers without fractional precision. The absence of tooltips suggests that no additional descriptions have been added for operator reference.

By defining these elements, the SCADA system ensures that process parameters are accurately communicated with the PLC, enabling real-time monitoring and control. This setup enhances process consistency and facilitates seamless automation by dynamically applying the correct values when different data records are selected.

Benefits of Using Elements in TIA Portal

1. **Parameter Standardization and Accuracy**

   Elements provide a standardized way to define and manage process variables such as temperatures, pressures, and other key parameters. By linking these elements to PLC tags, the system ensures that accurate values are consistently applied during the operation, minimizing the risk of errors and enhancing process reliability.

2. **Seamless Communication with PLC Tags**

   Elements are directly mapped to PLC tags, enabling real-time communication between the SCADA system and the PLC. This allows for dynamic updates of process variables, ensuring that any changes in the data records or recipe parameters are reflected immediately in the control system.

3. **Flexibility and Easy Adaptation**

   Using elements allows operators to easily modify, add, or delete process parameters without altering the overall system structure. This flexibility makes it simple to adapt the system to new requirements, product variations, or process changes without significant downtime.

4. **Efficient Recipe and Data Record Management**

   Elements enable efficient handling of multiple data records within a recipe. Each data record can reference different values for the same element, allowing for quick switching between different process configurations. This functionality reduces setup time and enhances production efficiency.

5. **Range Control and Data Validation**

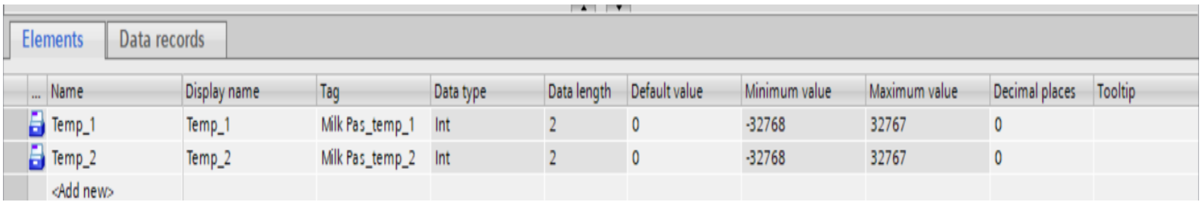
   By defining minimum and maximum values for each element, the system ensures that only valid parameter values are used during the process. This built-in validation helps prevent out-of-range errors, enhancing system safety and preventing damage to equipment or product quality issues.

6. **Scalability and System Expansion**

   Elements provide a scalable structure that allows new process parameters to be added easily. As the system expands or new recipes are introduced, additional elements can be created and mapped to PLC tags, enabling smooth scalability without major reconfiguration.

7. **Improved Troubleshooting and Maintenance**

   Clear definition and linkage of elements with tags facilitate easier troubleshooting and maintenance. When an error or deviation occurs, operators can quickly identify which parameter is causing the issue, reducing system downtime and ensuring smooth operations.



...	Name	Display name	Tag	Data type	Data length	Default value	Minimum value	Maximum value	Decimal places	Tooltip
	Temp_1	Temp_1	Milk Pas_temp_1	Int	2	0	-32768	32767	0	
	Temp_2	Temp_2	Milk Pas_temp_2	Int	2	0	-32768	32767	0	
	<Add new>									

Fig. 3.11. Database Block of Elements.

3.1.5 : Creating Screens

Step 1 : Alarm

The alarm screen shown in the figure is part of an HMI (Human-Machine Interface) used to monitor and manage alarms in an industrial process. The interface is typically seen in SCADA (Supervisory Control and Data Acquisition) systems, often configured in Siemens TIA Portal or WinCC. Below are the detailed insights:

1. Alarm Table Overview

The main section of the screen contains a table that displays information related to alarms. The table consists of several columns that provide details about each alarm event:

- No.: Sequential number assigned to each alarm or event.
- Time: Time of occurrence of the alarm.
- Date: The date on which the alarm was generated.
- Status: Indicates whether the alarm is active, acknowledged, or cleared.
- Text: A brief description or message explaining the alarm condition.
- Acknowledge Group: Used for grouping alarms to enable operators to acknowledge them collectively.

2. Process/Recipe Selection Dropdown

- The dropdown menu located at the top left corner, labeled “Pasteurization,” indicates the selected process or recipe.
- Operators can choose between different processes or recipes to monitor alarms related to specific operations.

3. Reset Button

- Positioned next to the dropdown, the Reset button is used to clear or reset alarm conditions after the necessary corrective actions have been taken.
- Clicking this button typically removes alarms that have been acknowledged and are no longer active.

4. System Date and Time Display

- The upper right corner displays the system’s current date and time in the format MM/DD/YYYY HH:MM:SS AM/PM.
- This timestamp is crucial for monitoring and comparing alarm occurrences and acknowledgments.

5. Alarm Control Icons (Right Panel)

On the right side of the screen, there is a vertical row of icons that provide control and management options for alarms. These icons include:

- Alarm List Icon: Opens or displays active alarms.
- Disable/Block Alarm Icon: Temporarily disables alarm generation.
- Alarm Acknowledge Icon: Used to acknowledge active alarms. Acknowledging informs the system that the operator is aware of the condition.
- Event/History Icon: Displays historical alarms or events for reference and analysis.
- Undo/Redo Icon: Allows operators to navigate back and forth between alarm states or views.
- Refresh/Update Icon: Updates the alarm list to show the most recent alarm conditions.

6. Bottom Left Control Buttons

- There is a button at the bottom left that switches between different views:
    - Summary View: Displays a condensed list of alarms for a quick overview.
    - Detailed View: Provides detailed information about each alarm, including the exact condition and potential corrective actions.

7. Acknowledge and Alarm Log Buttons

- In the bottom right corner, there are buttons for acknowledging alarms and accessing the alarm log for reviewing historical events.
- Acknowledge Button: Confirms that the operator has recognized the alarm and is addressing the issue.
- Log Button: Opens the event log or historical data for analysis and troubleshooting.

8. Importance of the Alarm Screen

The alarm screen is essential for ensuring the safety, reliability, and efficiency of the system by providing real-time information about abnormal process conditions, helping operators take corrective actions, and maintaining a historical record of alarm events for analysis and compliance.

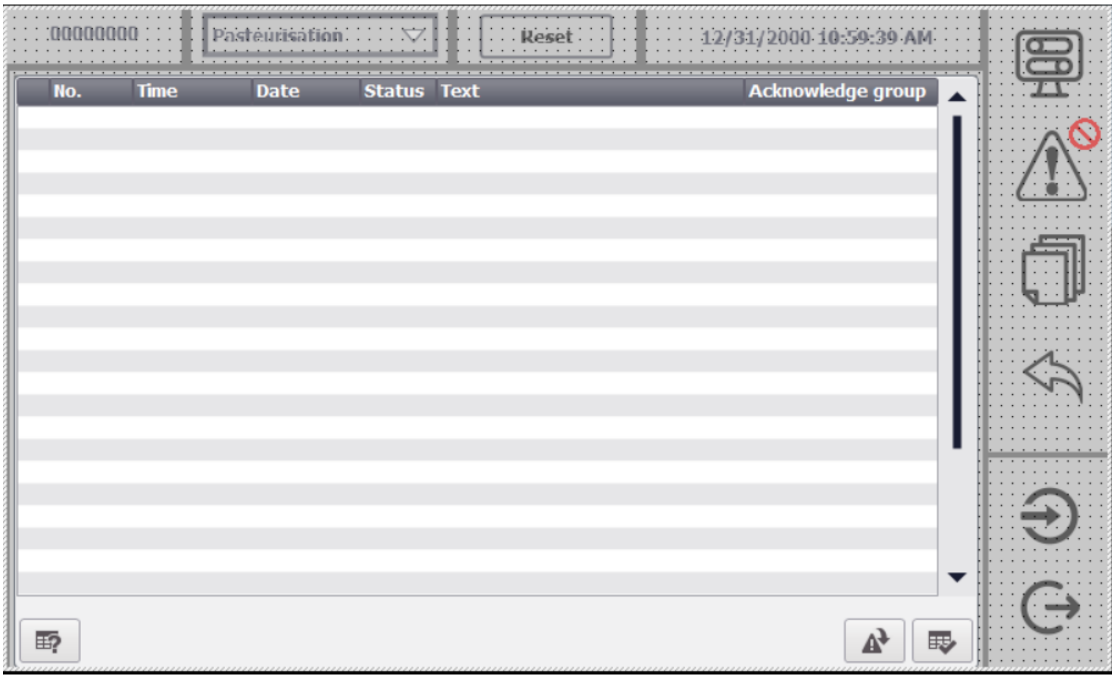


Fig. 3.12. Database Block of Alarm Screen.

Step 2 : Pasteurization

1. Process Overview

- Objective:
  - The pasteurization process involves heating the liquid to a specific temperature for a defined period and then cooling it rapidly.
  - This process kills harmful pathogens without compromising the nutritional value of the product.

2. Main Components in the Screen

Input Tank (Left Side):

- The tank on the left stores the raw liquid (e.g., milk) that needs to be pasteurized.
- A flow meter is connected to the output line of the tank to monitor the liquid flow rate in litres per hour (LPH).
- Flow Display: The value displayed (currently set to "000") indicates the current flow rate.

- Valve: A red valve regulates the flow of liquid into the pasteurization system, ensuring consistent feed.

Flow Measurement Sensor:

- Positioned after the tank, it measures the flow rate of the liquid entering the system.
- The reading is shown below as Flow (LPH) in litres per hour.

Heat Exchanger / Pressure Pipes (Centre):

- The heat exchanger or pressure pipes facilitate the heat exchange process required for pasteurization.
- The liquid is heated to a specific temperature and maintained at that temperature for a defined period to ensure effective pasteurization.
- Pressure Measurement:
    - A pressure sensor monitors the system's internal pressure, ensuring that it remains within safe operating limits.
    - The pressure value is shown as Pressure (psi) in pounds per square inch (currently "00").

Output Tank (Right Side):

- This tank stores the pasteurized liquid after it passes through the heat exchanger.
- Level Measurement:
    - The level of pasteurized liquid is displayed as Level (cm) to indicate the quantity collected in the tank.
    - Currently, the level is set to "00".
- Outlet Valve:

    - The red valve near the output tank controls the flow of the pasteurized liquid and ensures regulated discharge.

3. Control and Monitoring

System Control Options:

- Start/Stop/Reset:
    - The system can be started, stopped, or reset using the control buttons at the top.
- Process Monitoring:
    - Real-time monitoring of flow, pressure, and levels helps ensure smooth operation.
- Alarm and Warning System:
    - Alarm icons on the right may indicate potential system faults, high pressure, or flow irregularities.

Automation Integration:

- The pasteurization process is likely integrated into an automated system that adjusts the temperature, flow, and pressure dynamically.
- Automation ensures consistency in pasteurization and maintains quality standards.

4. Key Parameters to Monitor

- Flow Rate (LPH):
    - Ensures that the required amount of liquid is entering the pasteurization system.
- Pressure (psi):
    - Prevents overpressure scenarios and ensures the pasteurization process is conducted under optimal conditions.

- Level (cm):
    - Monitors the level of pasteurized liquid in the output tank to prevent overflow or insufficient collection.

5. Benefits of Using This System

- Enhanced Safety: Kills harmful bacteria and pathogens effectively.
- Consistent Quality: Maintains product quality by controlling flow, temperature.
- Real-time Monitoring: Provides accurate real-time data for better control and decision-making.

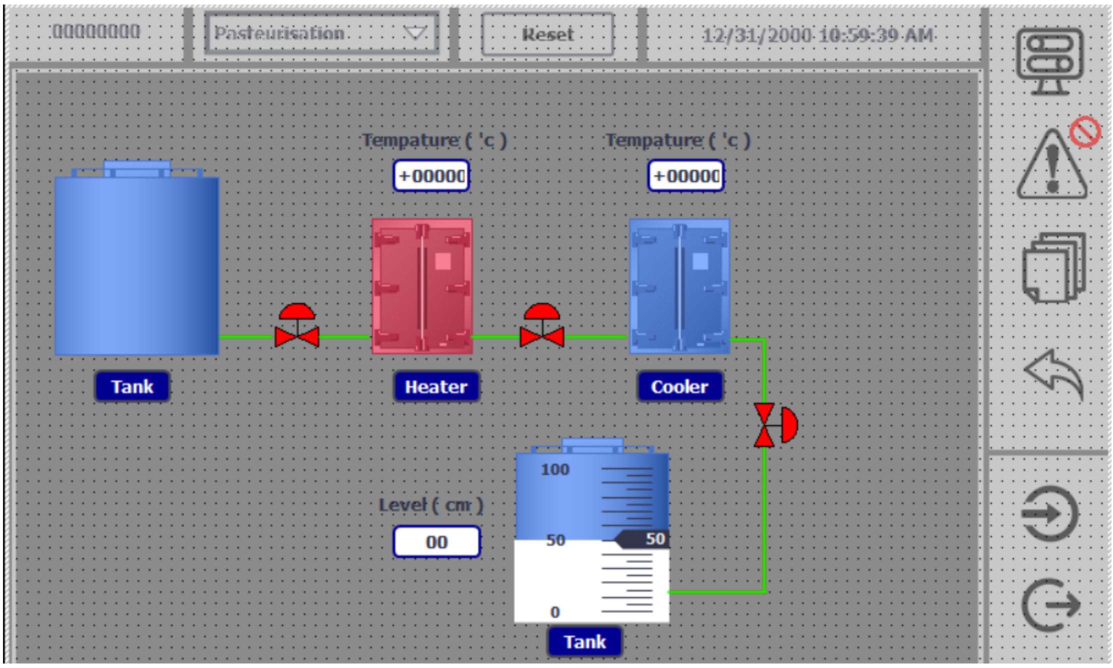


Fig. 3.14. Database Block of Pasteurization Screen.

Step 3 : Homogenization

1. Overview

The pasteurization screen provides a graphical representation of the pasteurization process, where raw liquid (such as milk) is heated to a specific temperature to kill harmful bacteria while maintaining product quality. This process ensures that the liquid is safe for consumption and prolongs its shelf life. The screen allows operators to monitor and control critical process parameters like flow rate, pressure, and liquid levels in real-time, ensuring that the system operates efficiently and safely.

2. Main Components

- Input Tank:

  The input tank stores the raw liquid before it enters the pasteurization system. The liquid flows from this tank into the system through a regulated pipeline.

- Flow Meter:

  Positioned after the input tank, the flow meter measures the volume of liquid passing through the system in Liters Per Hour (LPH). The display currently shows a value of 000, indicating no flow at the moment. Monitoring flow is crucial to maintain the correct processing speed.

- Controlvalve:

  This valve, located after the flow meter, controls the flow of liquid entering the heat exchanger. It ensures that the liquid enters at the desired rate to maintain optimal pasteurization conditions.

- HeatExchanger/PressurePipes:

  The heat exchanger (labled as Pressure Pipes in the diagram) is responsible for heating the liquid. It maintains consistent heating to kill bacteria effectively without compromising product quality. The pressure inside the heat exchanger is monitored to prevent overpressure situations.

- PressureSensor:

  The pressure sensor monitors the pressure inside the system, measured in psi. Maintaining the correct pressure ensures the system's safety and prevents equipment failure. The current reading on the display shows 00 psi, indicating no active pressure.

- OutputTank:

  After pasteurization, the liquid is collected in the output tank. This tank stores the processed liquid before it is transferred for further use or packaging.

- LevelIndicator:
  The level indicator shows the amount of liquid present in the output tank in centimetres (cm). Proper monitoring of the liquid level prevents overflow or underfilling. The current level reading shows 00 cm.

- OutletControlValve:
  Positioned before the output tank, this valve controls the flow of pasteurized liquid into the tank. It ensures that the processed liquid flows at an optimal rate.

3. Key Parameters to Monitor

The pasteurization process requires constant monitoring of three key parameters:

- Flow(LPH):
  The flow rate ensures the correct volume of liquid is being processed. Any deviation may affect the quality of the pasteurization process.

- Pressure(psi):
  Proper pressure management is essential for maintaining safe operating conditions. If the pressure goes beyond the safe limit, the system could face malfunctions or potential safety hazards.

- Liquid(cm):
  Monitoring the liquid level in the output tank helps avoid overflows and ensures that the pasteurized product is stored properly.

4. Purpose and Benefits

The pasteurization screen ensures that the process runs efficiently while minimizing the risk of errors. It provides improved safety by eliminating harmful bacteria, which enhances the quality and shelf life of the product. The system also offers real-time monitoring, allowing operators to make timely adjustments to avoid costly errors. Additionally, automation increases efficiency by reducing manual work and ensuring consistent product quality.

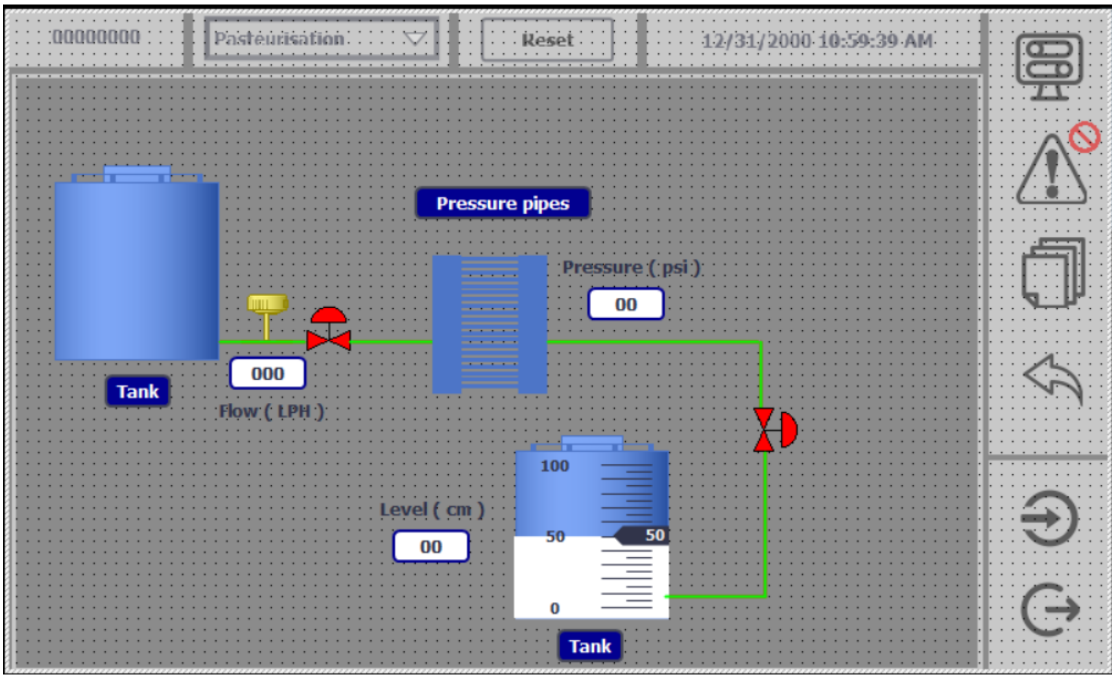


Fig. 3.15. Database Block of Homogenization Screen.

Step 4 : Centrifuge

1. Centrifuge Position and Role

- The centrifuge is located at the centre of the process flow between the Tank and the separated Cream and Milk containers.
- It is responsible for separating raw milk from the tank into two different products:
    - Cream – The higher fat content part of the milk.
    - Milk – The remaining low-fat or skimmed milk.

2. Functionality of the Centrifuge

- Separation Process:
    - The centrifuge operates based on the principle of centrifugal force.
    - When the milk enters the centrifuge, it spins at high speed, forcing the heavier portion (milk) to move outward and the lighter portion (cream) to collect at the centre.
    - This separation occurs due to the difference in densities between cream and milk.
- Valve Control:
    - A valve is placed between the tank and the centrifuge to control the flow of milk into the centrifuge.

    - The valve can be opened or closed based on process requirements or triggered by alarms or safety conditions.

3. Input and Output Connections

- Input from the Tank:
    - The milk flows from the tank through a green pipeline into the centrifuge.
    - Flow regulation is likely managed by the valve depicted before the centrifuge.
- Output to Storage Tanks:
    - The separated cream and milk exit the centrifuge through two different pipelines:
        - Cream Outlet: Directed towards the cream storage container.
        - Milk Outlet: Directed towards the milk storage container.

4. Milk and Cream Level Monitoring

- Milk Level Indicator:
    - Located next to the milk container, it shows the milk level in centimeters.
    - Ensures the separated milk quantity is monitored and maintained at the desired level.
- Cream Level Indicator:
    - Positioned next to the cream container, it displays the cream level in centimeters.
    - Helps in managing cream quantity and avoiding overflow.

5. Data Monitoring and Control

- Milk and Cream Level Values:
    - The numeric display below the containers shows real-time levels of milk and cream in their respective containers.
    - These values can be used for monitoring and controlling the output from the centrifuge.
- Process Automation:The centrifuge is likely part of an automated system where values from the tank, cream, and milk levels are monitored through sensors and managed through the HMI.

6. Benefits of Using the Centrifuge

- Efficient Separation: Ensures quick and efficient separation of milk and cream.
- Consistent Quality: Provides uniform cream and skimmed milk quality.

- Automated Control: Reduces manual effort by automating the separation and flow control.
- Reduced Wastage: Prevents product loss due to precise monitoring and regulation.

7. Possible Alarms or Warnings

- High/Low Milk or Cream Levels: Alarms may trigger if the milk or cream levels exceed or drop below a defined threshold.
- Blockage or Valve Malfunction: Indicates possible issues in flow or valve operation.
- Centrifuge Speed Faults: Monitors speed variations that may affect separation efficiency.

The centrifuge plays a critical role in ensuring smooth and efficient separation, contributing to the overall performance of the pasteurization or dairy processing system.

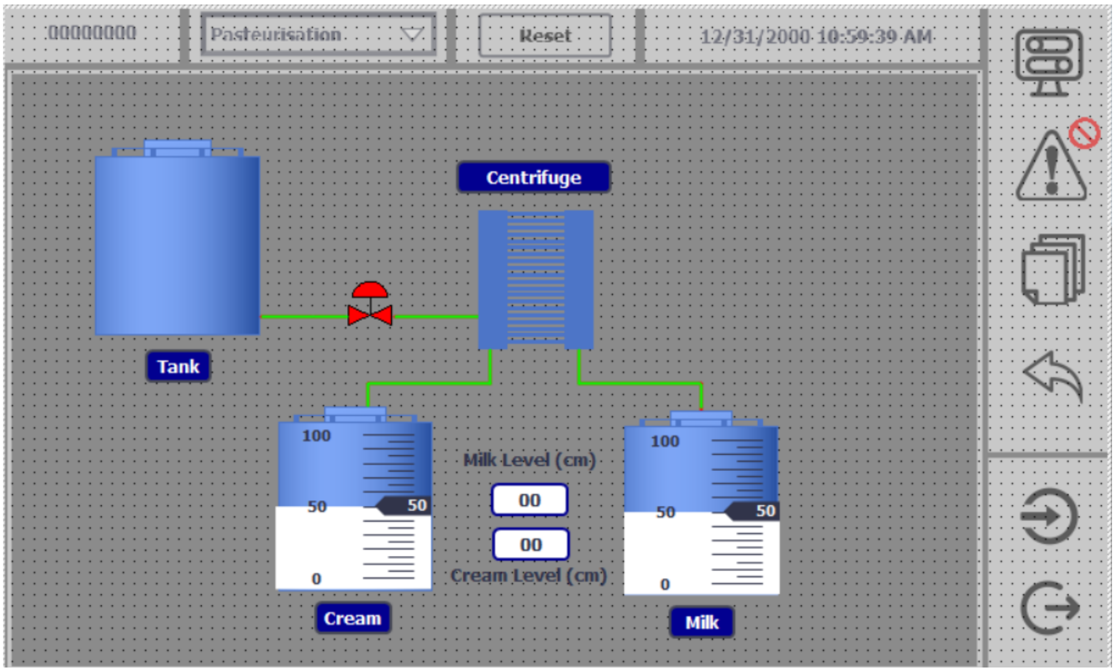


Fig. 3.13. Database Block of Centrifuge Screen.

4.1.1 Creating Tags

Step 1: PLC Tags

The Default Tag Table shown in the figure represents a list of PLC tags used in a control system. Tags in PLC (Programmable Logic Controller) systems are used to define and identify different variables or signals that are monitored or controlled by the system. Each tag has a name, data type, address, and associated attributes that determine how the tag interacts with the system. Below is an overview of the key details about the tags shown in the table:

- Tag Names and Functions

Each tag is given a specific name to represent the corresponding process or device it controls or monitors. For instance:

  - Filtering_ stop, Homogenization_ stop, Centrifuge_ stop: These tags are Boolean (Bool) variables that control the stop functionality for respective processes such as filtering, homogenization, and centrifuge operations.
  - Pasteurization stop and cooling_ stop: Tags that are also Boolean and responsible for stopping the pasteurization and cooling processes.
  - HMI_ start: This tag initiates the start process via the Human-Machine Interface (HMI).
  - Clock_ Byte and Clock Frequencies: These tags include a variety of Boolean tags that generate timing pulses with different frequencies, such as Clock_10Hz, Clock_5Hz, Clock_2.5Hz, Clock_1Hz, etc., which are useful for time-based operations.
  - up_ cv and water_ pipe_ steps: These tags are Double Integers (DInt), used for counting or storing large values.

- Data Types

The data types used in this tag table include:

  - Bool: Boolean tags that can hold only two states—TRUE (1) or FALSE (0), commonly used for start/stop or on/off operations.
  - Word: A 16-bit data type that can store integer values or status information.
  - DInt (Double Integer): A 32-bit signed integer used to store larger numerical values.

    - Byte: An 8-bit data type that stores a small set of binary information.
    - Time: Represents time-related data, useful for time calculations or delay functions.
    - Int: A 16-bit integer data type for smaller numerical values.
- Addressing Format

Each tag is mapped to a specific memory address that defines where the data is stored or accessed. Addressing formats include:

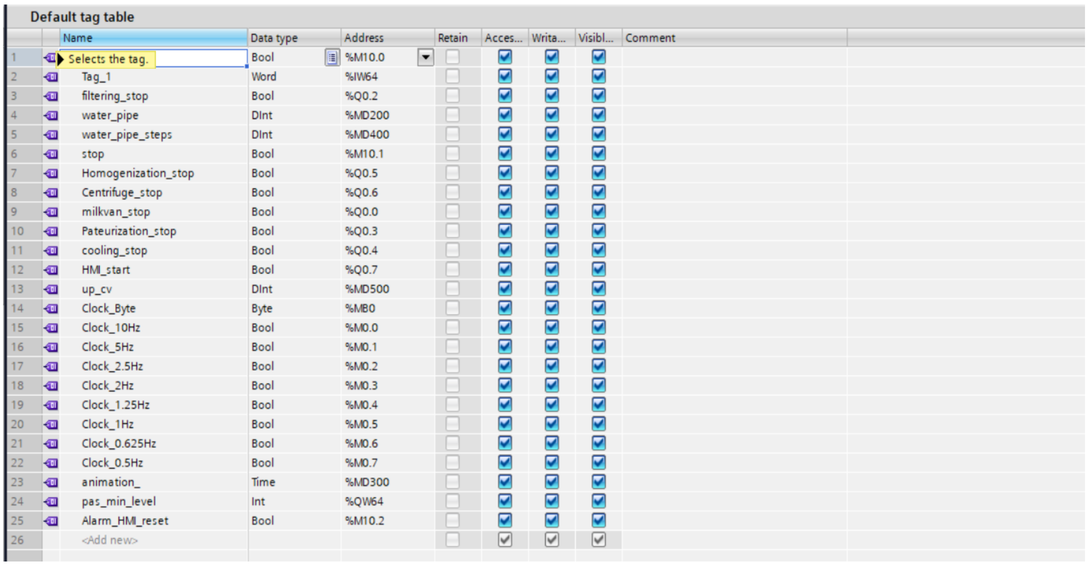
    - %M: Internal memory bits or words used for storing intermediate values or system states.
    - %Q: Output memory bits or words that control external devices or actuators.
    - %MD: Double-word memory addresses used to store larger numerical values.

For example:

- %M10.0 corresponds to the tag Selects the tag., which is a Boolean variable.
- %MD500 is assigned to up_ cv, which stores a Double Integer value.
- %QW64 is used for pas_ min_ level, indicating an integer value from the output area.
- Retain, Access, Write, and Visibility
    - Retain: Indicates whether the tag value is retained after a power cycle. None of the tags in the figure have this option enabled.
    - Access, Write, and Visibility: These options determine whether the tag can be accessed, modified, or made visible for HMI operations. In this case, all tags are enabled for writing and visibility, ensuring complete access for process control and visualization.
- Purpose and Usage in the Control System

These tags play a critical role in automating and controlling different processes in an industrial system. They enable seamless communication between the PLC and external devices such as motors, valves, and sensors. Monitoring tag values ensures that the process operates within safe limits, and any deviation can be quickly corrected.

The tags are also essential for generating time-based pulses, storing data, and controlling equipment with start/stop operations. Moreover, they provide real-time feedback for HMI visualization, allowing operators to oversee the process.



Default tag table

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	Selects the tag.	Bool	%M10.0	□	☑	☑	☑	
2	Tag_1	Word	%IW64	□	☑	☑	☑	
3	filtering_stop	Bool	%Q0.2	□	☑	☑	☑	
4	water_pipe	DInt	%MD200	□	☑	☑	☑	
5	water_pipe_steps	DInt	%MD400	□	☑	☑	☑	
6	stop	Bool	%M10.1	□	☑	☑	☑	
7	Homogenization_stop	Bool	%Q0.5	□	☑	☑	☑	
8	Centrifuge_stop	Bool	%Q0.6	□	☑	☑	☑	
9	milkvan_stop	Bool	%Q0.0	□	☑	☑	☑	
10	Pateurization_stop	Bool	%Q0.3	□	☑	☑	☑	
11	cooling_stop	Bool	%Q0.4	□	☑	☑	☑	
12	HMI_start	Bool	%Q0.7	□	☑	☑	☑	
13	up_cv	DInt	%MD500	□	☑	☑	☑	
14	Clock_Byte	Byte	%MB0	□	☑	☑	☑	
15	Clock_10Hz	Bool	%M0.0	□	☑	☑	☑	
16	Clock_5Hz	Bool	%M0.1	□	☑	☑	☑	
17	Clock_2.5Hz	Bool	%M0.2	□	☑	☑	☑	
18	Clock_2Hz	Bool	%M0.3	□	☑	☑	☑	
19	Clock_1.25Hz	Bool	%M0.4	□	☑	☑	☑	
20	Clock_1Hz	Bool	%M0.5	□	☑	☑	☑	
21	Clock_0.625Hz	Bool	%M0.6	□	☑	☑	☑	
22	Clock_0.5Hz	Bool	%M0.7	□	☑	☑	☑	
23	animation_	Time	%MD300	□	☑	☑	☑	
24	pas_min_level	Int	%QW64	□	☑	☑	☑	
25	Alarm_HMI_reset	Bool	%M10.2	□	☑	☑	☑	
26	<Add new>			□	☑	☑	☑	

Fig. 4.1. Database Block of PLC Tags.

Step 2: HMI Tags

The attached figure shows a list of HMI (Human-Machine Interface) tags that are used to facilitate interaction between the operator and the PLC system. These tags store and transfer data between the HMI and the underlying control system. Each tag is defined with a name, data type, connection type, and PLC name (though PLC names are not specified in this case). Below is a detailed explanation of each parameter and the associated tags:

- Tag Names and Their Purpose

Each tag in the list serves a specific function within the HMI and control system:

  - _iScreen_number: An integer tag that stores the current screen number, allowing the HMI to identify which screen is currently displayed.
  - _iUser_group_number: Another integer tag that identifies the user group currently accessing the HMI. This tag helps in controlling access based on the user’s group or role.
  - _sUser_name: A Wide String (WString) tag that holds the name of the logged-in user. This tag is useful for maintaining audit logs and tracking user activity.
  - Milk Pas_temp_1: An integer tag that records the temperature from the first milk pasteurization sensor. This tag is used for monitoring and ensuring the pasteurization process maintains the required temperature.

  - Milk Pas_temp_2: Similar to the first sensor, this tag captures data from the second milk pasteurization sensor, allowing for redundancy and cross-checking of temperature values.

- Data Types

Each tag is assigned a data type that determines the kind of information it can hold:

  - Int (Integer): Tags like _iScreen_number, _iUser_group_number, Milk Pas_temp_1, and Milk Pas_temp_2 use the Integer data type to store numerical values.
  - WString (Wide String): _sUser_name uses the WString data type, allowing it to store alphanumeric characters (text) such as user names.

- Connection Type

All tags in this figure use the <Internal tag> connection type. This means the data is stored and processed locally within the HMI and is not directly linked to any external PLC memory or address. Internal tags are primarily used to manage information that affects HMI visualization and user interaction without involving the PLC.

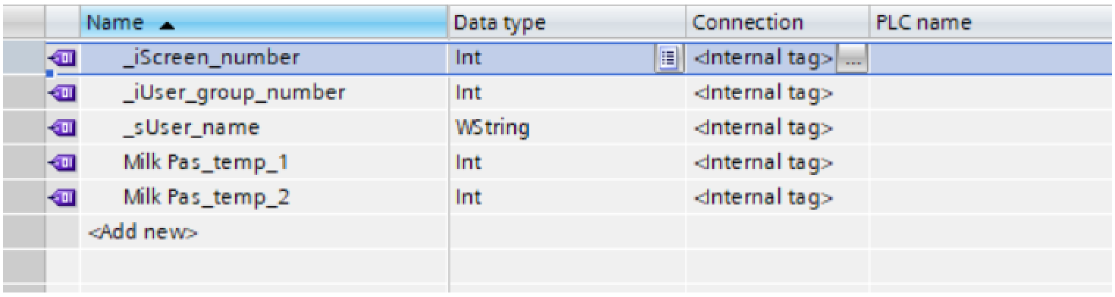
- PLC Name

The PLC name column is left empty, indicating that these tags are not mapped to any external PLC memory or devices. Since these are internal tags, they reside within the HMI itself and are used for HMI-specific operations such as managing screen transitions, tracking user sessions, and storing intermediate process values.

- Application and Importance

These HMI tags play a critical role in improving system control and monitoring. Tags such as _iScreen_number and _iUser_group_number ensure smooth navigation and enforce role-based access to different screens. _sUser_name provides user identification and helps in auditing actions taken by operators. The Milk Pas_temp_1 and Milk Pas_temp_2 tags monitor process-critical parameters, ensuring that the pasteurization process meets the required standards.

In summary, these tags collectively enhance the functionality and security of the HMI system, while also contributing to process efficiency and regulatory compliance.



Name	Data type	Connection	PLC name
_iScreen_number	Int	<Internal tag>	
_iUser_group_number	Int	<Internal tag>	
_sUser_name	WString	<Internal tag>	
Milk Pas_temp_1	Int	<Internal tag>	
Milk Pas_temp_2	Int	<Internal tag>	
<Add new>			

Fig. 4.2. Database Block of HMI Tags.

4.1.2 Designing Ladder Diagram

Animation Ladder Diagram

The attached ladder diagram consists of two main rungs that contribute to the overall process, likely for an alarm or monitoring system associated with tank level and animation. Below is a detailed breakdown of each component and its functionality:

- Rung 1: Timer for Animation Process
  - %M10.0 "start" – This is a memory bit that initiates the process. When this bit is set to TRUE (activated), it starts the sequence.
  - TON (On-Delay Timer) %DB2 "animation" –
  - This timer is triggered by the start bit %M10.0.
  - PT (Preset Time): T#100s – The preset time for the timer is set to 100 seconds, which means that after the input is enabled, the timer will count 100 seconds before activating its output.
  - Q (Timer Output): When the timer completes its countdown, the Q bit turns TRUE.
  - ET (Elapsed Time): This shows the elapsed time while the timer is running.
  - %MD300 "animation_" – The output associated with the timer stores or reflects the animation process or transition. Once the timer is completed, this memory location updates to indicate that the animation has been successfully completed.
- Rung 2: Counter for Tank Level Monitoring and Alarm
  - %M0.0 "Clock_10Hz" – This is a pulse clock that runs at 10 Hz frequency, providing a periodic pulse signal. It acts as a trigger to increment the counter.
  - CTU (Count Up Counter) %DB3 "tank level" –

  - CU (Count Up Input): Connected to the 10 Hz clock pulse %M0.0. Every time this pulse turns ON, the counter increments by 1.
  - R (Reset): A constant false condition prevents the counter from being reset automatically.
  - PV (Preset Value): The preset value is set to 90, indicating that the counter is set to trigger an event or alarm when the count reaches 90.
  - CV (Current Value): The current count value, stored in %MD500 "up_cv", increments with each pulse.
  - %MD500 "up_cv" – This holds the current value of the counter and increases with each pulse from %M0.0.
  - When the CV (Current Value) reaches the PV (Preset Value) of 90, an action such as an alarm or stop condition can be triggered.
- Summary of Process Flow
  - StartCondition:

    When %M10.0 is set to TRUE, the timer %DB2 is activated, and it begins counting for 100 seconds. Upon completion, the output %MD300 is updated to reflect the animation process.
  - TankLevelMonitoring:

    Simultaneously, %M0.0 generates 10 Hz pulses that increment the counter %DB3. The counter increases with each pulse and updates the value in %MD500. Once the count reaches 90, a predetermined action such as triggering an alarm can be initiated.
  - FinalAction:

    The final condition based on the completion of the animation and the counter reaching 90 will determine if an alarm or alert is raised to notify the operator of the situation.
- Potential Alarm Triggering
  - Alarm Trigger: When the tank level counter %DB3 reaches the preset value of 90, the system could trigger an alarm indicating a high tank level condition.
  - Visual/Animation Feedback: The timer's completion %MD300 could potentially be used to initiate a visual indication or animation on the HMI to notify the operator.

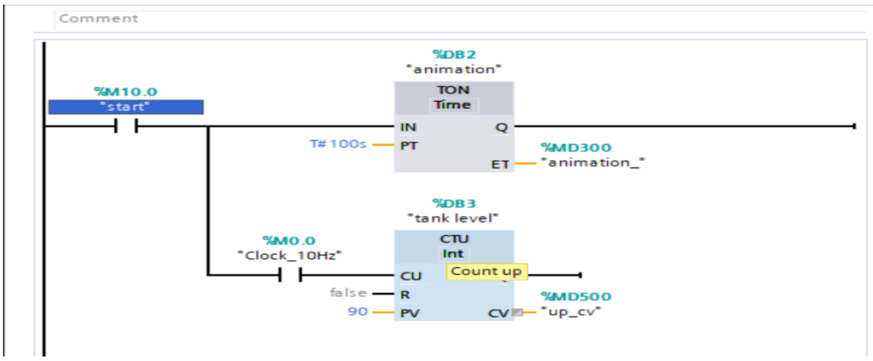


Fig. 4.3. Ladder Diagram for Animation.

Alarm Ladder Diagram

The provided ladder diagram consists of two rungs that handle different alarm conditions based on the elapsed time of an animation and other monitored events. Below is a detailed explanation of the process:

- Rung 1: Alarm Trigger Based on Animation Time (55 Seconds Timer)
  - %MD300 "animation_" == T#55s
  - This condition checks whether the value of %MD300 (which holds the animation or process time) has reached 55 seconds.
  - Once this condition is met, it activates the output.
- %DB1.DBX0.0 "AlarmDB".timer_hit
  - When the timer condition is satisfied, this bit %DB1.DBX0.0 is set to TRUE.
  - This bit acts as a flag to indicate that the 55-second time threshold has been reached and that the corresponding alarm condition has been triggered.
- %M10.2 "Alarm_HMI_reset"
  - This memory bit is used as a reset condition.
  - If the reset condition %M10.2 is activated, it can reset the alarm state or stop further actions related to the alarm.
- Rung 2: Alarm Trigger for Pasteurization Time (30 Seconds Timer)
- %MD300 "animation_" == T#30s
  - This condition checks whether the animation time %MD300 has reached 30 seconds.

    - Similar to the first rung, this condition triggers an event when the time reaches 30 seconds.
- %DB1.DBX0.1 "AlarmDB".timer_hit_pas
    - When the 30-second condition is satisfied, this bit %DB1.DBX0.1 is set to TRUE.
    - This bit indicates that the pasteurization process or associated task has reached its 30-second time threshold, potentially triggering another alarm or indication.
- %M10.2 "Alarm_HMI_reset"
    - As in the previous rung, this bit is used as a reset condition.
    - Activating %M10.2 resets the state of the alarms and stops further triggering.
- Summary of Process Flow
- Animation Time Check for Alarm 1:
    - When %MD300 reaches 55 seconds, the condition becomes TRUE, setting %DB1.DBX0.0 to signal an alarm.
    - The reset condition %M10.2 can be used to reset or acknowledge this alarm.
- Animation Time Check for Alarm 2:
    - When %MD300 reaches 30 seconds, the condition is met, setting %DB1.DBX0.1 to indicate another alarm.
    - %M10.2 also acts as a reset condition for this alarm.
- Alarm Management and Reset Logic
- Alarm Trigger:
    - Two alarms are triggered at different time thresholds (55 seconds and 30 seconds).
    - These alarms are linked to bits %DB1.DBX0.0 and %DB1.DBX0.1, which can be monitored in the HMI or SCADA system.
- Alarm Reset Condition:
    - The %M10.2 bit serves as a reset condition to reset or clear the alarms.
    - This allows for manual or automatic acknowledgment of the alarms after addressing the issue.

This ladder logic ensures that multiple time-based alarms can be triggered and managed efficiently, with reset functionality for operator intervention or automatic reset scenarios.

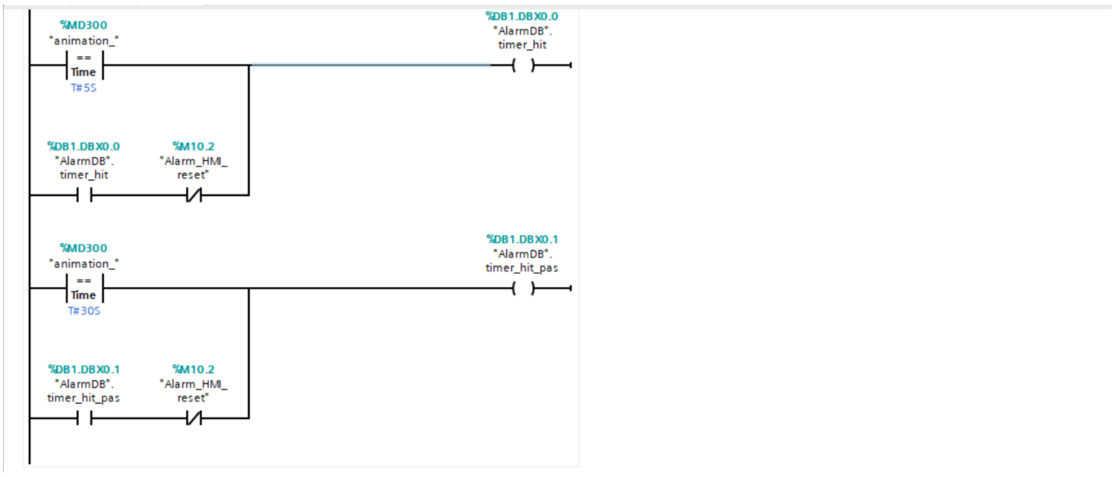


Fig. 4.4. Ladder Diagram for Alarm.

4.1.3 Creating Graphic Process Image

Username and Password:

The attached image shows an HMI (Human-Machine Interface) screen from a TIA Portal project, displaying a Login Dialog Box where a user is prompted to enter credentials. However, the image does not provide any visible information about the actual username or password.

- Login Interface Overview:
    - User: This is a text input field where the operator or technician must enter their assigned username.
    - Password: This is a password-protected field (masked with asterisks or dots) where the corresponding password is entered.
    - OK Button: Confirms the entered credentials and attempts to log in.
    - Cancel Button: Cancels the login attempt and closes the dialog.
- Where to Find User Credentials in TIA Portal:

To find or manage usernames and passwords in TIA Portal:

1. Open TIA Portal Project.
2. Navigate to:
    - HMI Configuration > Runtime Settings > Security > User Administration.
3. Here, you can view or modify:

    - Usernames
    - Passwords
    - User roles and access levels.
- Important Notes:
    - The credentials are stored securely in the project and can only be accessed with appropriate privileges.
    - Passwords are encrypted and cannot be retrieved directly.
    - Only users with administrative access can modify or reset user accounts.

The image shows an HMI (Human-Machine Interface) screen from a TIA Portal project, where a login popup is displayed, prompting the user to enter a username and password for authentication. This login functionality is typically used to restrict access to certain areas or functions within the HMI, ensuring that only authorized personnel can perform critical operations such as resetting alarms, modifying setpoints, or accessing sensitive data. The login dialog box consists of two input fields — one for the username and another for the password. After entering the required information, the user can click the "OK" button to validate the credentials or choose "Cancel" to abort the login attempt.

In TIA Portal, user credentials and roles are configured under the User Administration section, which can be accessed by navigating to:

- HMI Configuration > Runtime Settings > Security > User Administration.

This section allows the project developer to create different user profiles, assign roles, and configure permissions based on the operational requirements. Each user can be assigned specific privileges, such as access to certain screens, acknowledgment of alarms, or modification of process values. Passwords are securely stored in the project and encrypted to prevent unauthorized access. During runtime, when a user successfully logs in, their assigned role determines which functions they can perform within the HMI.

If a user forgets their password or needs a role change, only an administrator with the required access level can make these modifications. Additionally, TIA Portal offers features such as automatic logout after a period of inactivity to enhance security further. Proper management of user credentials ensures that safety and operational integrity are maintained throughout the system.

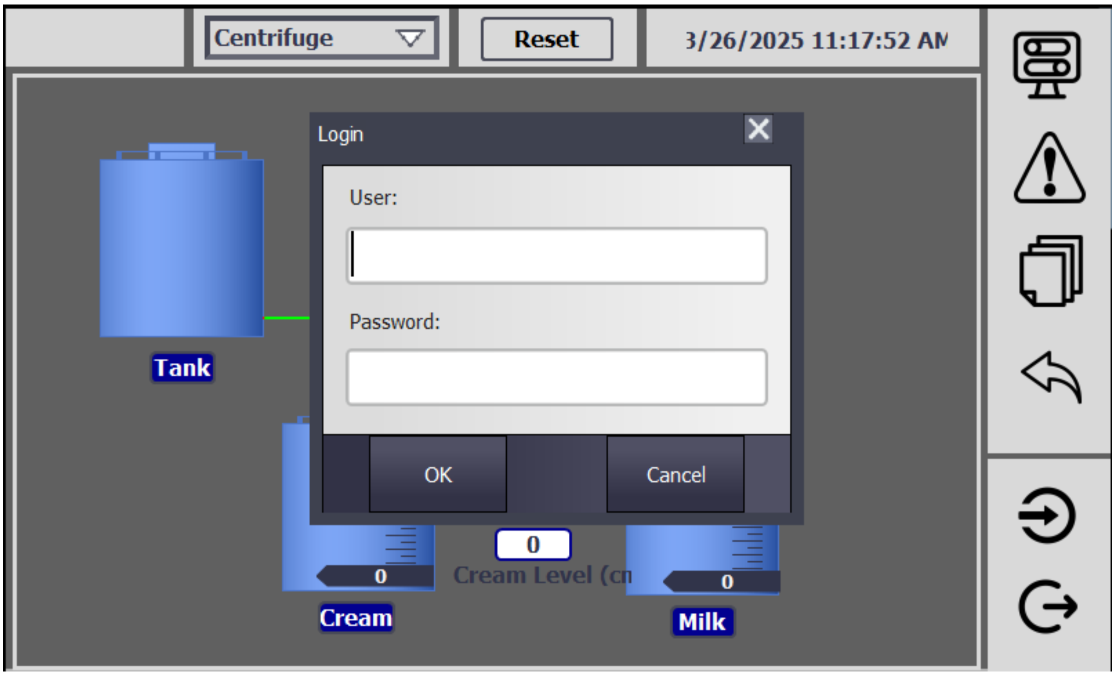


Fig. 4.5. Graphic Process Image (Username and Password).

Admin and Operator

The HMI screen shown in the image is from SIMATIC WinCC Runtime Advanced in TIA Portal. It displays a process control interface with a logged-in user, indicated as "Admin01" at the top left. User roles such as Admin and Operator in this type of HMI system have different levels of access and control, ensuring that certain critical tasks are only performed by authorized personnel.

- Admin Role (Admin01)
    - Level of Access: Full Access
- Permissions:
    - Modify System Parameters: Admins can modify and adjust setpoints, process variables, and system limits.
    - Acknowledge and Reset Alarms: They can acknowledge alarms and reset the system, as shown by the available "Reset" button.
    - User Management: Admins can add, modify, or delete user accounts and assign appropriate access levels.
    - Access to Configuration Screens: They can access and modify critical system configurations, including recipe management, historical data, and security settings.
    - HMI and PLC Control: Admins have the ability to make changes that affect the control logic and behavior of the PLC.

    - Alarm Handling: They can acknowledge and reset alarms or modify alarm settings.
    - Maintenance Mode: Admins can put the system in maintenance mode for system updates, hardware changes, or safety checks.
- Operator Role
    - Level of Access: Limited/Restricted Access.
- Permissions:
    - Basic Operation: Operators can view the current process and monitor key performance indicators (KPIs), such as tank levels, temperature, and system status.
    - Acknowledge Alarms: Operators can acknowledge alarms but may not be able to reset them unless explicitly allowed.
    - Start/Stop Control: Operators can start and stop predefined operations or processes but cannot modify system settings or control parameters.
    - Limited Access to Configuration Screens: Operators may only view certain configuration settings without permission to modify them.
    - No Access to User Management: They cannot add, modify, or delete user accounts.
- Typical Scenario in the Image:
- Admin01 is logged in and has the ability to:
    - Reset alarms or the system using the Reset button.
    - Switch between different process screens such as "Pasteurisation" via the drop-down.
    - Modify system settings or access deeper configuration options, which an operator would not be able to perform.

In real-time applications, role-based access control (RBAC) ensures that only users with the appropriate permissions can alter critical aspects of the system, preventing accidental changes and maintaining the safety and efficiency of industrial processes.

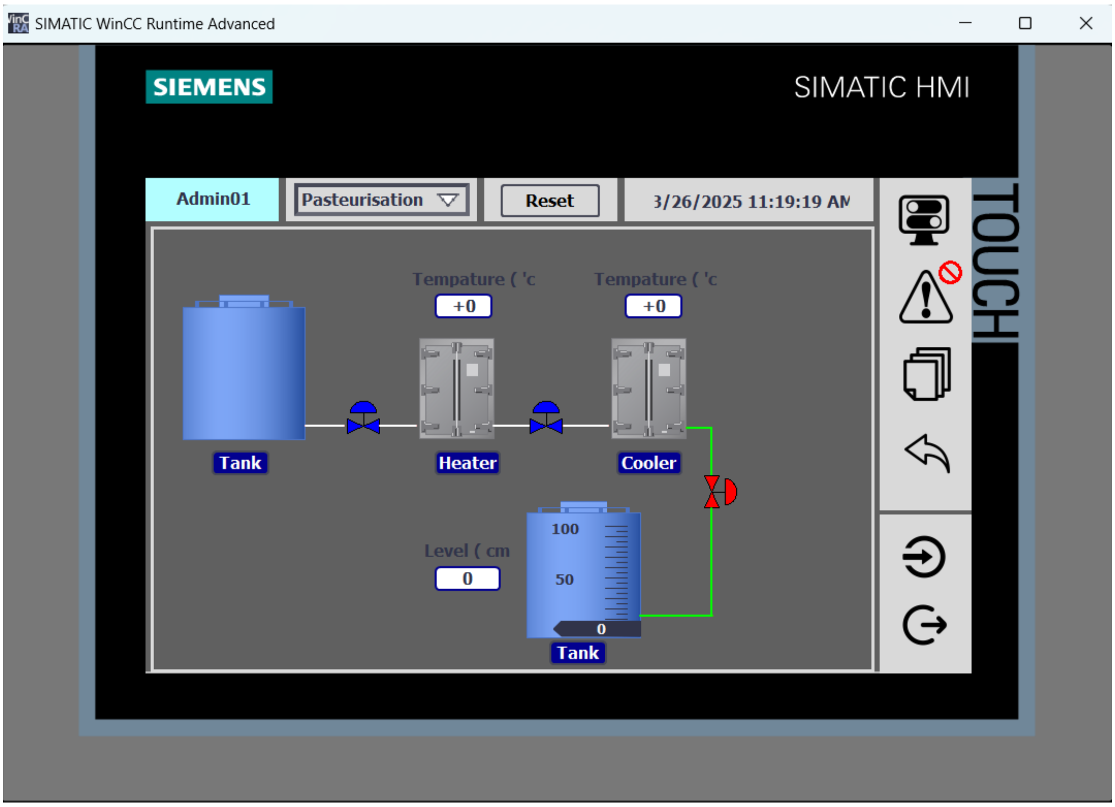


Fig. 4.6. Screenshot Of Admin LOGIN.

In industrial automation environments using SIMATIC WinCC Runtime Advanced, user roles such as Admin and Operator are critical for maintaining security, operational efficiency, and system integrity. These roles define the scope of actions a user can perform within the Human-Machine Interface (HMI), ensuring that only authorized personnel can make modifications or execute sensitive commands.

The Admin Role is typically reserved for senior-level engineers, system integrators, or maintenance supervisors who require full access to the system. Admins can modify system configurations, change process setpoints, and manage user accounts, giving them complete control over the HMI and PLC (Programmable Logic Controller). This includes modifying alarm settings, changing recipes, and performing software updates or system diagnostics. Admins also have the privilege to reset alarms and process faults, ensuring that critical issues are addressed promptly.

On the other hand, the Operator Role is designed for personnel responsible for the daily operation and monitoring of the system. Operators typically have limited access to system parameters, ensuring that they can only interact with the interface to start or stop processes, acknowledge alarms, and monitor real-time data. They do not have the authority to modify core system configurations, which protects the process from unintended errors or unauthorized changes. In the current screen, "Admin01" is logged in, which indicates that the user has administrative privileges, enabling them to perform actions like resetting alarms and switching between different process states.

By implementing role-based access control (RBAC) in industrial automation systems, organizations can ensure that their operations are both secure and efficient, reducing.

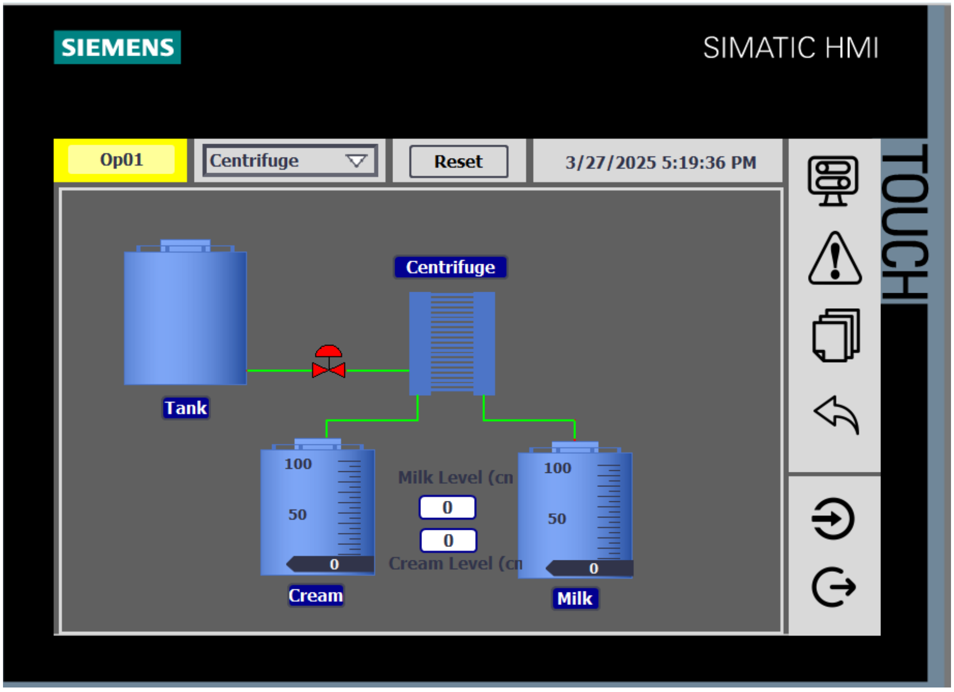


Fig. 4.7. Screenshot Of Operator LOGIN.

5.

Conclusion:

Securing a SCADA (Supervisory Control and Data Acquisition) system in a milk factory is essential for ensuring product quality, operational continuity, and protecting sensitive data. As milk processing involves multiple critical stages—such as pasteurization, cooling, and storage—securing the SCADA system mitigates risks associated with unauthorized access, cyber threats, and process malfunctions. Implementing multi-level user authentication (such as Admin and Operator roles), encrypted communication protocols, and robust firewall settings significantly strengthens system security. Regular security audits, timely software updates, and real-time monitoring of system anomalies further enhance protection against potential cyber-attacks. Additionally, integrating role-based access control (RBAC) ensures that only authorized personnel can modify sensitive parameters, reducing the risk of human errors or malicious actions.

By adopting a layered security approach that includes physical security, network protection, and user access management, milk factories can maintain high standards of safety, quality, and regulatory compliance. Ultimately, enhancing SCADA system security ensures operational efficiency, product consistency, and customer satisfaction, while minimizing downtime and safeguarding the factory's reputation in an increasingly digital landscape.

References

[1] Stouffer, K., Falco, J., & Scarfone, K. (2015). Guide to Industrial Control Systems (ICS) Security. National Institute of Standards and Technology (NIST). NIST Special Publication 800-82, Revision 2. Available at: https://www.nist.gov

[2] Kriaa, S., Pietre-Cambacedes, L., Bouissou, M., & Halgand, Y**.** (2015). A survey of approaches combining safety and security for industrial control systems. *Reliability Engineering & System Safety,* 139, 156-178. https://doi.org/10.1016/j.ress.2015.02.008

[3] Igure, V. M., Laughter, S. A., & Williams, R. D. (2006). Security issues in SCADA networks. *Computers & Security,* 25(7), 498-506. https://doi.org/10.1016/j.cose.2006.03.001

[4] International Society of Automation (ISA). (2018). *ISA/IEC 62443 Series: Security for Industrial Automation and Control Systems.* Available at: https://www.isa.org/standards-and-publications/isa-iec-62443

[5] Zhu, B., Joseph, A., & Sastry, S. (2011). A taxonomy of cyber attacks on SCADA systems. *Proceedings of the 2011 IEEE International Conference on Internet of Things and Cyber, Physical and Social Computing,* 380-388. https://ieeexplore.ieee.org

[6] Yan, Y., Qian, Y., Sharif, H., & Tipper, D. (2012). A survey on cyber security for smart grid communication. *IEEE Communications Surveys & Tutorials,* 14(4), 998-1010. https://doi.org/10.1109/SURV.2012.010912.00035

[7] Siemens AG. (2020). *Security Guidelines for Industrial Automation Systems.* Available at: https://new.siemens.com

[8] Weber, R. H. (2010). Internet of Things – New security and privacy challenges. *Computer Law & Security Review,* 26(1), 23-30. https://doi.org/10.1016/j.clsr.2009.11.008

[9] Sadeghi, A. R., Wachsmann, C., & Waidner, M. (2015). Security and privacy challenges in industrial Internet of Things. *Proceedings of the 52nd Annual Design Automation Conference (DAC),* 1-6. https://doi.org/10.1145/2744769.2747942

[10] Galloway, B. & Hancke, G. P. (2013). Introduction to industrial control networks. *IEEE Communications Surveys & Tutorials,* 15(2), 860-880. https://doi.org/10.1109/SURV.2012.071812.00124