

## **Design and Validation of a Cross-Platform Human-Machine Interface for Retrofitting Automation System in Industry 4.0 Environment**

**N. Nordin<sup>1,2\*</sup>, M.A.A Othman<sup>1</sup>, N. Abas<sup>1,2</sup>, M.M. Ghazaly<sup>1,2</sup> and M. Othman<sup>3</sup>**

<sup>1</sup> *Faculty of Electrical Technology and Engineering, Universiti Teknikal Malaysia Melaka*

<sup>2</sup> *Center for Robotics & Industrial Automation, Universiti Teknikal Malaysia Melaka*

<sup>3</sup> *Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn*

*\*Correspondence: N.Nordin*

### **Abstract**

This paper presents a practical framework for retrofitting legacy automation systems with ubiquitous Human-Machine Interfaces (HMIs) that align with Industry 4.0 standards. The core objective is to demonstrate how brownfield sites which are typically constrained by aging control hardware and isolated operator panels can be upgraded with secure, cross-platform access and modern visualization capabilities without a full infrastructure overhaul. Using an Omron NX1P2 PLC and NA5 HMI panel as the control backbone, a Node-RED dashboard is integrated via a TLS-secured MQTT broker to enable seamless remote monitoring and control across desktop, tablet, and mobile devices. The system architecture addresses common shortcomings of traditional HMIs, such as platform rigidity, single-point access, lack of alerting and log retention, and poor scalability. Although demonstrated on a garage-door prototype, this use case serves only as a benchmark scenario; the solution itself is generalizable to industrial environments requiring safe, responsive, and operator-friendly retrofits. A suite of usability and performance evaluations including task completion time and System Usability Scale (SUS) scores confirmed the system's low latency, strong cross-device consistency, and high user acceptance. The findings establish a transferable model for cost-effective modernization of brownfield control systems, bridging the gap between legacy assets and the connected ecosystems demanded by Industry 4.0.

**Keywords:** legacy system, brownfield automation, IR4.0, MQTT

## **Introduction**

The increasing push toward Industry 4.0 has highlighted a persistent challenge in modern manufacturing: how to bring aging, standalone control systems into the connected, intelligent industrial ecosystem without a full-scale replacement. Many factories and small industrial sites continue to rely on legacy Programmable Logic Controllers (PLCs) and fixed-location Human-Machine Interfaces (HMIs) that lack remote access, data logging, multi-device usability, and cyber-secure communication. These limitations restrict operational flexibility, slow down response times, and hinder the adoption of predictive maintenance or real-time monitoring. In short, the absence of a standardized, scalable, and secure HMI solution for older systems represents a critical barrier to digital transformation.

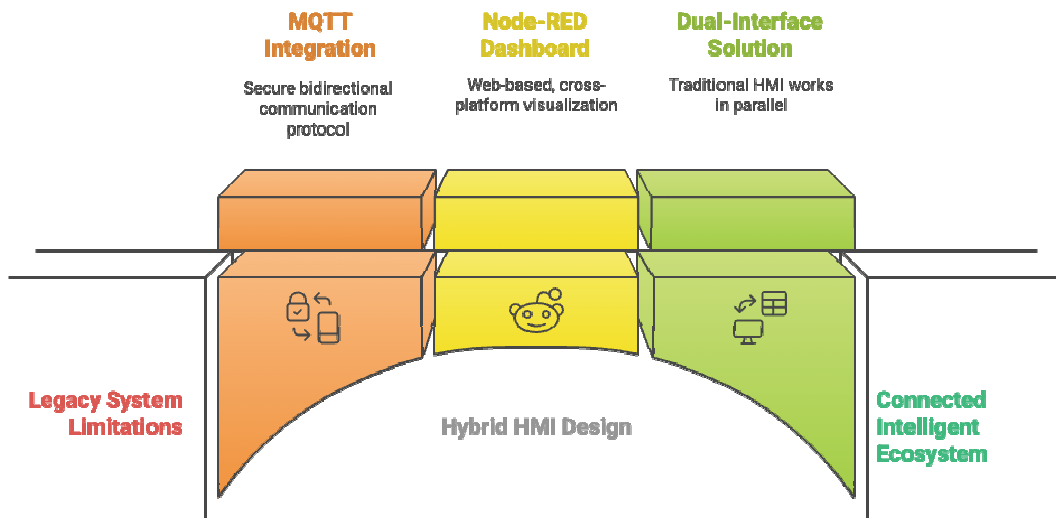
The integration of lightweight, interoperable technologies such as MQTT and Node-RED has emerged as a promising approach in modern industrial control systems. As the shift toward Industry 4.0 accelerates, the need for real-time data visibility, remote operability, and scalable interface design is becoming increasingly critical. Node-RED, a flow-based development tool with a visual programming interface, enables rapid prototyping and deployment of IoT applications, making it particularly attractive for industrial environments that require agility and low development overhead as depicted in Figure 1.

Numerous implementations have demonstrated the practicality of MQTT-Node-RED ecosystems across a range of industrial scenarios. For instance, Ferencz and Domokos illustrated a Combined Cycle Power Plant (CCPP) prototype that employed MQTT for efficient message delivery and Node-RED for dashboard-based visualization, offering insights into real-time plant performance [1]. Similarly, temperature monitoring systems developed using ESP8266 microcontrollers and Node-RED dashboards underscore the feasibility of lightweight remote sensing in industrial contexts [2], [3]. These solutions highlight the flexibility of Node-RED in bridging sensor networks, cloud interfaces, and operator terminals.

In supervisory control and data acquisition (SCADA) environments, Zare and Iqbal proposed a low-cost, scalable platform using ESP32, Raspberry Pi, MQTT, and Node-RED, enabling remote access to energy usage data, demonstrating the relevance of these technologies even in cost-sensitive setups [4]. Furthermore, recent applications in process automation, such as remote bottle-filling control using PLCs and Node-RED dashboards, affirm the versatility of this architecture in integrating heterogeneous systems and communication protocols [5].

Despite these advances, many industrial systems still lack a cohesive strategy for merging proprietary HMIs with web-based, cross-platform alternatives. Efforts to merge open-source visualization tools like Node-RED with commercial-grade HMIs and PLCs remain scarce in academic literature. There is a distinct gap in studies that attempt to create a dual-interface solution, where a traditional HMI (such as Omron NA5 or Siemens Comfort Panel) works in parallel with a web-based dashboard. Hence, the primary objective of this work is to design a hybrid HMI that provides consistent and secure multi-device access for monitoring and controlling existing machines. Secondary objectives include implementing TLS-secured MQTT messaging, configuring alarm and logging functions accessible from both the

industrial and web-based HMI, and conducting structured usability and latency testing to validate the retrofit.



**Figure 1:** Modernizing Legacy Industrial Control Systems

This study hypothesizes that such a system can achieve sub-second responsiveness, high usability scores, and functional parity across platforms—metrics that align with Industry 4.0 expectations and the Sustainable Development Goals (SDGs) related to resilient infrastructure (SDG 9) and responsible innovation (SDG 12).

The research design follows an experimental framework with hardware-in-the-loop validation. The legacy controller is programmed to work in tandem with an NA5 HMI for local operation and a Node-RED dashboard for remote access. MQTT over TLS is used for secure bidirectional communication. Tests simulate real-world scenarios with multiple users and device types to capture latency, task completion time, and usability across conditions. These measurements are analyzed using statistical tools including ANOVA and t-tests to determine the impact of platform and environment on system performance. **Literature Review**

### 2.1.1. Cross-platform and Ubiquitous HMI

Industry 4.0 encourages the development of human-machine interfaces (HMIs) that offer seamless cross-platform access, thereby improving operational flexibility across manufacturing environments. In response, Krupitzer and colleagues note that traditional HMIs tethering operators to stationary terminals undermine usability and impede productivity in situations where remote or mobile supervision is essential [6]. To address this shortcoming, Majrashi et al. introduce a Multiple User Interfaces framework [7], arguing that system designers can preserve a consistent look and feel by following clear, documented principles as users switch from smartphones to large wall-mounted screens. Meanwhile, concepts from ubiquitous computing and other recent Industry 4.0 studies demonstrate that multi-access interfaces can be

woven into established workflows, linking legacy machines directly with real-time cloud dashboards and advanced analytics [8].

This redesign is often most valuable during retrofits. Zawra's side-by-side evaluation of older automation stacks [9] explains how PLCs originally shipped decades ago can still secure meaningful connections with modern web-based dashboards if engineers follow a well-defined integration pathway. Guidelines such as the Material Design responsive spec summarize best practices for creating interfaces that stay responsive and intuitive across differing screen sizes and contexts [7]. Still, in the absence of written coding guidelines, open-source platforms such as Node-RED sometimes produce layout flows that undermine their advertised cross-platform consistency [10]. In practice, many projects neglect to synchronize live data streams and bypass structured usability tests, causing custom factory displays to operate in isolation from mobile or cloud dashboards.

As a result, many available toolkits do not offer retro-fit, industrial-grade HMIs that seamlessly marry devices such as the Omron NA5 panel with open-source IoT displays in a secure, user-first way. This work tackles that shortfall by linking a PLC-driven HMI touchscreen, a Node-RED web app, and MQTT messaging into a stable, multi-device monitoring and control system.

## *2.2. Usability and User Experience*

The usability and overall user experience (UX) of industrial human-machine interfaces (HMIs) determine how well operators can manage automated systems in harsh environments. Krupitzer et al. note that a subpar UX raises fatigue, mistakes, and sluggish response times, all of which undermine safety and slow production [6]. To avoid these failures, designers must create a clear visual hierarchy, easy navigation, and layouts that adjust to varying screen sizes. Majrashi's cross-platform usability framework reinforces this point by arguing that consistent patterns give the same functional hints whether the operator uses a fixed panel, a tablet, or a smartphone [7].

Controlled studies confirm that more intuitive, user-centered designs boost both task accuracy and operator morale on the factory floor [11]. Still, despite the many theories promoting strong usability, Industry 4.0 initiatives often skimp on basic testing routines and allow weaker interfaces to enter service. The System Usability Scale (SUS), a convenient and respected satisfaction index, is seldom paired with live evaluations of real-time industrial Internet-of-Things HMIs [12]. Naujoks et al. provide a helpful yardstick [10]: they combined expert reviews with user trials to measure car CMI performance across different levels of automation.

Our strategy follows recent recommendations—from Nikolova [13] and others on Industry 4.0 interfaces that urge designers to run systematic tests so that menus and controls stay clear and easy to repair, even in low-code setups like Node-RED. Still, there is almost no real-world evidence on how well people can use a system when commercial operator panels are linked with community-built IoT dashboards in the same workspace. To fill that gap, we weave formal usability checks, such as the System Usability Scale and standard timing-and-error records, into every test run. This data-driven review lets us confirm that the new HMI not only looks consistent across screens but also hits the usability benchmarks industry expects. By placing user

feedback at the center of the build process, our work links technical functionality with practical validation, addressing a key missing piece in many Industries 4.0 human-machine projects.

### 2.3. *Real-Time Communication Using MQTT and Node-RED*

Real-time signaling underpins the operation of next-generation factories that follow Industry 4.0 goals. The publish-subscribe messaging protocol MQTT has become the industry benchmark for linking IoT sensors, PLCs, and cloud dashboards, primarily because its small packet sizes ease network strain and accelerate deployment. Nevertheless, the protocol is not immune to attack, and its lightweight design amplifies existing security threats. Alsabbagh and colleagues illustrated the danger in their Pirates of MQTT paper [14], showing how an unauthorized client linked to an exposed broker can inject forged messages or listen in on control commands. In response, several research teams have proposed additive layers that work alongside standard MQTT to harden the protocol against such misuse. For example, Yan's Time-Specific Integrity Service (TSIS) timestamps each published record and checks the age of a message before acceptance, thereby raising confidence in the data that reaches distributed PLCs [15]. Chauhan and Gore go a step further with SMQTT, which coordinates light-weight clock synchronization so that timestamps match across IIoT devices and drift is kept negligible[16]. One more extension, usually branded PrioMQTT, introduces priority queues to the broker, allowing critical alerts to jump ahead of less urgent traffic even during heavy load [17]. Collectively, these enhancements reshape MQTT from a generic messaging standard into a more predictable and secure conduit suitable for industrial control applications.

At the network edge, Node-RED commonly partners with MQTT, serving as a nimble orchestrator that visualizes data flows and directs information between local programmable logic controllers PLCs and distant dashboard clients. Real-world study by Luis et al. [18] show that when an MQTT broker runs beside Node-RED, average end-to-end delays between production-line sensors and cloud interfaces can stay within 2 to 10 milliseconds, making the combination suitable for latency-sensitive tasks on the shop floor. Although those results highlight MQTT's potential for near-real-time industrial monitoring, many installations still overlook the security settings that Node-RED demands.

Weak protective measures such as unencrypted traffic or factory-default credentials leave networks exposed to rogue clients and trigger message-loss or replay errors. To close this gap, the design proposed here pairs a TLS-secured MQTT broker and client certificates with Node-RED flows that follow topic-hierarchy and error-handling best practices, thus safeguarding every real-time metric and alarm on the hybrid human-machine interface (HMI). In essence, the architecture preserves the agility and rapid prototyping that MQTT and Node-RED promise yet delivers that flexibility atop a firm foundation of encryption and authenticated access.

### 2.4. *Retrofitting and Legacy Upgrade*

Retrofitting legacy industrial systems has emerged as a pragmatic and cost-effective pathway toward achieving Industry 4.0 capabilities, particularly when compared to the substantial capital investment required for complete system overhauls. Zawra outlines a structured methodology for transitioning from traditional PLC-based control loops to smart-factory-compatible systems, while acknowledging that disparities in hardware capabilities and real-time data demands can introduce significant challenges during implementation [9]. A notable example of successful retrofitting is presented by Valaboju et al. [19], who extended the operational life of an aging assembly station by integrating a supervisory control layer and a real-time monitoring interface. This intervention effectively enhanced visibility and control without necessitating a complete system replacement.

Further reinforcing this direction, Yolov et al. demonstrated the feasibility of using the lightweight MQTT protocol to bridge legacy PLCs with cloud-based services [20]. Their prototype utilized an IoT gateway to establish secure remote communication, highlighting how incremental hardware enhancements can facilitate data-driven decision-making and remote supervision. Similarly, Malhão et al. introduced the Industrial IoT Smartbox [21], an edge device designed to extract sensor data from legacy machines, transmit it to the cloud, and provide operators with a basic on-site dashboard, thus significantly enhancing system transparency and data availability with minimal infrastructural changes.

Building on this foundation, Langås et al. proposed an inclusive digital twin architecture supported by edge computing, which enables the creation of live virtual models of existing equipment [22]. This framework keeps retrofitted assets synchronized with cloud-based analytics platforms, supporting predictive maintenance and immersive operator training through virtual reality, all without removing equipment from the production line. Complementing this, Zhang et al. showcased the use of embedded multicore edge controllers to offload computational tasks from traditional PLCs [23]. Their approach enables faster response times and supports advanced functionalities such as local data filtering and on-device preprocessing, thereby enhancing both performance and scalability.

Collectively, these studies illustrate a range of viable approaches to modernizing aging production infrastructure through strategic retrofitting. However, a recurring gap in many implementations is the lack of standardized, cross-platform human-machine interfaces (HMIs) and robust alarm systems; both critical for ensuring operator safety and operational efficiency. While establishing cloud connectivity for legacy PLCs is increasingly common, many retrofitting efforts rely on fragmented visualization solutions, such as mismatched display panels or ad hoc code, which undermine user experience and situational awareness.

To address this shortfall, our work presents a cohesive retrofit framework that integrates an industrial-grade Omron NA5 touch panel, a browser-accessible Node-RED dashboard, and a hardened MQTT communication layer. This configuration ensures a unified and resilient HMI environment that provides consistent real-time alerts and operational insights across all user devices. By aligning the legacy control network with modern visualization and alerting tools, the proposed solution not only enables connectivity to the Industrial Internet of Things (IIoT) but also elevates the

quality of human-machine interaction—a key dimension of successful digital transformation in manufacturing.

### 2.5. *Supporting Benchmark Studies*

A growing body of benchmark studies documents the deployment of Internet-of-Things and edge technologies in small and medium industrial settings, providing practical evidence rather than purely theoretical analysis. Chouragadey et al. describe a smart-home prototype that employs the lightweight MQTT protocol together with Node-RED to deliver near-real-time control and status displays on web and mobile clients [24]. Though set in a residential environment, their work confirms Node-RED can function as a cross-platform human-machine interface, a capability we adapt for monitoring and controlling an industrial garage door.

Opačin et al. build on that premise by outlining guidelines for configuring Node-RED flows so that memory and CPU usage remain within the limits of a typical industrial controller [25]. Their advice to reduce load on function nodes and to choose an appropriate Quality-of-Service level for MQTT echoes the resource-management principles discussed earlier in this article. Finally, Nenninger et al. present a case in which a programmable logic controller is linked to a cloud dashboard via publish-subscribe middleware like MQTT, allowing real-time performance monitoring from remote locations [26]. Their report adds further credibility to the hybrid model that couples local processing with distant visualization, a feature that reappears in the design proposed in the current work. Collectively, the reviewed studies demonstrate the practical viability of MQTT networking, Node-RED dashboards, and cloud-based integrations across a variety of real-world contexts, including laboratory setups, pilot-scale environments, and small production systems.

However, despite their promise, many of these implementations fall short of meeting the rigorous demands of industrial-grade deployments; particularly in areas such as cybersecurity, alarm management, and formal usability validation. As a result, these efforts often remain at the level of exploratory prototypes rather than fully trusted operational solutions. This study addresses these critical gaps by proposing a comprehensive, secure, and user-validated architecture that integrates an industrial HMI panel with a browser-based Node-RED interface over a hardened MQTT communication layer. The resulting system hypothesized to achieve sub-second response times, end-to-end encryption, and statistically validated usability across both local and remote interfaces. In doing so, this work may advance the discourse on Industry 4.0 retrofitting from basic connectivity toward a more robust paradigm of cyber-resilient and operator-centric design.

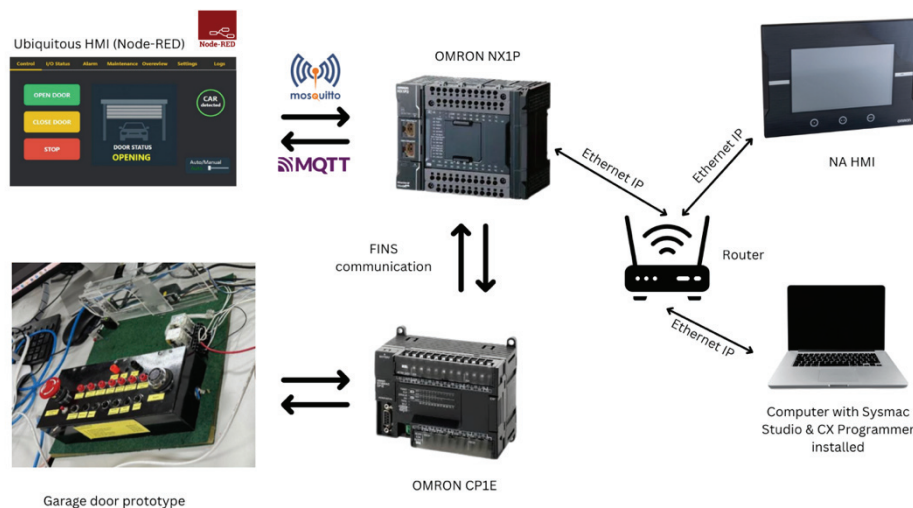
## 3. Methodology

This project uses an experimental, quantitative research design to test and validate a hybrid Human-Machine Interface (HMI) system developed for retrofitting conventional garage doors. The approach merges a working hardware prototype with ongoing real-time data capture, making every performance metric both measurable and repeatable. Controlled experiments place the garage-door system through a series

of repeatable operating scenarios. Metrics of interest include end-to-end response time, reliability of alarm notifications, integrity of secured MQTT traffic and consistency of usability scores across desktops, tablets and smartphones. The System Usability Scale (SUS) questionnaire provides a single numerical aggregate, allowing these subjective scores to be compared directly against widely accepted usability benchmarks.

### 3.1. Overall System Architecture

The proposed ubiquitous HMI system was designed to bridge legacy control architecture with Industry 4.0 capabilities using a layered integration of proprietary and open-source technologies as shown in Figure 2. The hardware configuration consisted of a legacy OMRON CP1E PLC as the core controller for a garage door automation testbed, interfacing directly with field components such as pushbuttons, limit switches, and relays. As a legacy model, the CP1E lacks native MQTT support; hence, to enable integration with modern visualization platforms, an OMRON NX1P2 PLC was introduced as an intermediary gateway.



**Figure 2:** Overall system architecture

The NX1P2 was selected for its ability to support multiple industrial protocols and to interface natively with both commercial and open-source systems. It communicated with the CP1E using the Factory Interface Network Service (FINS) protocol, a proprietary Omron standard. This configuration allowed for shared memory and data exchange across PLCs, enabling the NX1P2 to read and write control signals and sensor feedback from the CP1E in real-time. For instance, door open commands, sensor states, and alarm flags were mapped across shared memory areas using structured data addressing. The FINS protocol operates deterministically within Omron's architecture, requiring pre-defined node addressing and consistent I/O mapping to ensure synchronization and reliability.

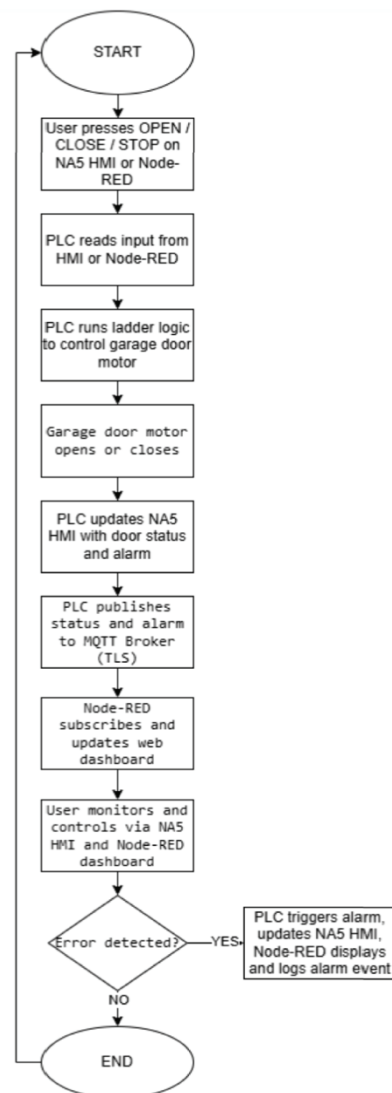
To enable ubiquitous access and enhance multi-platform operability, a secure MQTT communication layer was deployed between the NX1P2 and a Node-RED-based HMI dashboard. An on-premise Mosquitto MQTT broker, hosted on the same network via a wireless router, served as the publish-subscribe middleware. The NX1P2 was configured to publish process variables such as door status, system mode, and alarms to the broker on defined topics. Simultaneously, it subscribed to control topics to receive user commands issued from the Node-RED interface. TLS encryption was enabled on the broker to ensure secure data transmission and protect the system from unauthorized access or message injection. This MQTT layer provided low-latency, bi-directional communication that was compatible with mobile, tablet, and desktop-based clients.

In parallel, a conventional industrial-grade HMI (Omron NA5) was configured over Ethernet/IP for local operator control. Both the NA5 and the Node-RED dashboard accessed the same data model on the NX1P2, enabling dual-interface visualization and control. A laptop with Sysmac Studio and CX-Programmer installed was used for programming and debugging both PLCs over the same network. All devices such as PLC, HMI, broker, and PC were connected via a central router operating in static IP mode to maintain stable, closed-network communication. This hardware setup allowed for both backward compatibility with legacy equipment and forward compatibility with modern web-based control layers, thus providing a comprehensive and scalable retrofit model suitable for brownfield industrial automation. Figure 3 explains the system operation flowcharts.

### 3.2. Data Collection

The primary human participants in this study were six undergraduate students from a mechatronics engineering program, aged between 21 and 23, who had basic familiarity with HMIs but no prior exposure to ubiquitous dashboard systems. They were selected as representative of novice-level industrial operators in training. Participants were required to complete assigned tasks across different HMI platforms. No demographic restrictions were imposed, and participation was voluntary with informed consent obtained. The study received verbal approval aligned with faculty-level ethical guidance, as no sensitive or health-related data were involved.

Participants were recruited via convenience sampling from a senior-level control systems course. Five individuals who responded to the call and had no prior experience using Node-RED interfaces were selected. The testing environment was a controlled lab workstation mimicking an industrial setting. No incentives were provided. Data were collected over a single 2-hour session under observation by the researcher. The small sample size ( $n = 6$ ) was intended for preliminary usability and functionality validation rather than inferential generalization. Given the single-group, repeated-measure design across three platforms (NA5 HMI, Node-RED on desktop, Node-RED on mobile), the data were sufficient to detect user response patterns, identify interface challenges, and validate cross-platform consistency. No participants were excluded from the final analysis.



**Figure 3:** System operation flowchart

### 3.3. Outcome Measures

Usability was evaluated using the System Usability Scale (SUS) and task completion time. The SUS provided a subjective measure of system usability via a standardized 10-item Likert-type questionnaire. Each participant rated their experience with each platform independently. A multi-device test is performed using a PC, tablet, and NA5 HMI panel to ensure that layout, control elements, and alarm screens appear identically across platforms. Usability is scored with the System Usability Scale form for each device; ratings above the benchmark score of 70, as outlined by Naujoks et al. [10] and Drew et al. [12], are considered acceptable.

SUS contains ten statements, and subjects grade each one using a 5-point Likert scale anchored from 1, Strongly Disagree, to 5, Strongly Agree. The scoring rules

break out by item number, with odd-numbered questions Items 1, 3, 5, 7, and 9 giving credit equal to the given score minus 1 and even-numbered questions Items 2, 4, 6, 8, and 10 deducting the given score from 5. Totaling these adjusted values produces a raw score that the procedure then multiplies by 2.5, yielding the final SUS metric scaled out of 100 points. Task completion time was recorded for three pre-defined operations: opening/closing actuator, emergency stop reset, and alarm acknowledgement. Latency (round-trip MQTT message delay) was measured using timestamps logged on the broker and Node-RED client for real-time feedback assessment. No additional psychometric covariates were used in this preliminary phase.

This study adopted a within-subjects experimental design, wherein all participants interacted with three different HMI platforms in a randomized order to reduce sequence bias. The design allowed for direct comparison of usability, response time, and consistency across: (1) a commercial Omron NA5 HMI panel, (2) a Node-RED dashboard accessed via desktop browser, and (3) a Node-RED dashboard accessed via mobile phone. The prototype system simulated a garage door mechanism controlled via an Omron NX1P2 PLC, configured identically across all platforms. MQTT messages between the Node-RED dashboard and the PLC were routed through a TLS-secured Mosquitto broker hosted on a local server.

### 3.4. Experimental Setup and Procedures

Participants were briefed with a standardized instruction set and given a demonstration of the system workflow. They were then asked to perform three control tasks on each of the HMI platforms. The experimental system consisted of:

- a) Controller: Omron NX1P2 PLC programmed using Sysmac Studio
- b) Industrial HMI: Omron NA5 7-inch touchscreen panel
- c) Web-based Dashboard: Node-RED running on a laptop with a TLS-enabled Mosquitto MQTT broker
- d) Testbed Mechanism: Simulated garage door control circuit interfaced with input switches and motor actuators
- e) Network: Wi-Fi LAN secured with WPA2 encryption for all Node-RED interactions

The MQTT message payload was structured in JSON format and transmitted under three topic layers: */command*, */status*, and */log*. Message round-trip time was computed as:

$$RTT_{MQTT} = T_{received} - T_{sent}$$

where  $T_{sent}$  is the timestamp at which the command was published, and  $T_{received}$  is the timestamp when the acknowledgement was received. Participants completed the SUS questionnaire after interacting with each platform. Task completion time and MQTT latency were logged automatically. Observational notes were also taken to record qualitative feedback and errors.

## 4. Results

### 4.1. Multi-Device Usability

Usability testing included three device classes-PC browser, tablet display, and NA5 HMI panel. After each interaction, participants completed a standard System Usability Scale questionnaire. Given the small sample size ( $n = 6$ ), both the mean and median System Usability Scale (SUS) scores were calculated for each device category. While the mean offers a conventional estimate of central tendency, the median was also considered to mitigate the influence of any outliers or non-normal distribution, particularly in the tablet interface group. The System Usability Scale (SUS) yielded both an average and median exceeding 70 across all device categories, meeting the predetermined benchmark for accepted usability.

**Table 1.** SUS Score (Mean)

<b>User</b>	<b>HMI</b>	<b>PC</b>	<b>Tablet</b>
<b>1</b>	87.5	72.29	73.25
<b>2</b>	90	72.29	73.25
<b>3</b>	95	77.29	75.75
<b>4</b>	90	72.29	68.25
<b>5</b>	80	69.79	65.75
<b>6</b>	80	74.79	75.75
<b>Mean</b>	87.08	73.12	72.00
<b>Median</b>	88.75	72.29	73.25

A one-way ANOVA was conducted to determine whether usability scores differed significantly across three interface platforms: industrial HMI panel, PC-based interface, and tablet-based interface. Each group consisted of 6 participants (HMI:  $n = 6$ , PC:  $n = 6$ , Tablet:  $n = 6$ ). Data is presented as mean  $\pm$  standard deviation: HMI ( $87.08 \pm 6.00$ ), PC ( $60.83 \pm 2.58$ ), and Tablet ( $61.25 \pm 4.11$ ). The results revealed a statistically significant difference in usability scores among the interface groups,  $F(2, 15) = 68.30$ ,  $p < 0.001$ . This suggests that the type of interface platform had a significant impact on perceived usability.

**Table 2.** ANOVA result

<b>Source of variation</b>	<b>SS</b>	<b>dF</b>	<b>MS</b>	<b>F</b>	<b>P-value</b>	<b>F crit</b>
<b>Between groups</b>	2713.194	2	1356.597	68.3042	<0.0005	3.68232
<b>Within groups</b>	297.9167	15	19.86111			
<b>Total</b>	3011.111	17				

To identify where the differences lay, a post hoc Tukey HSD test was conducted. The results indicated that the HMI group differed significantly from both the PC group

(mean difference =  $-26.25$ , 95% CI [ $-32.93$ ,  $-19.57$ ],  $p < 0.001$ ) and the Tablet group (mean difference =  $-25.83$ , 95% CI [ $-32.52$ ,  $-19.15$ ],  $p < 0.001$ ). However, there was no statistically significant difference between the PC and Tablet groups (mean difference =  $0.42$ , 95% CI [ $-6.27$ ,  $7.10$ ],  $p = 0.986$ ).

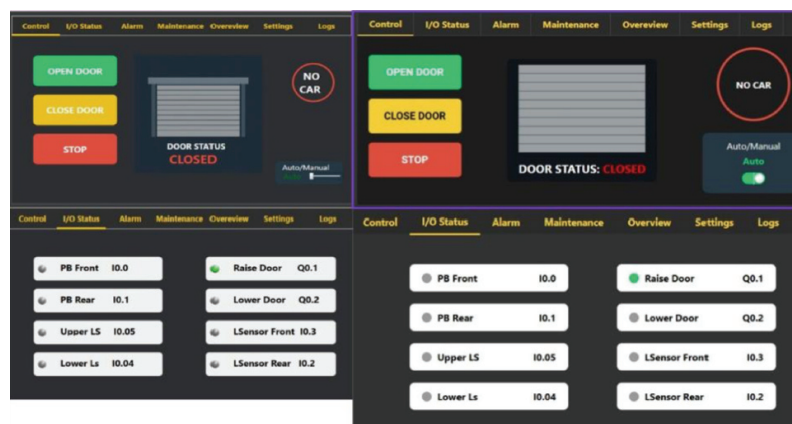
**Table 3.** Post hoc Tukey HSD result

<i>Group 1</i>	<i>Group 2</i>	<i>meandiff</i>	<i>p-adj</i>	<i>lower</i>	<i>upper</i>	<i>reject</i>
HMI	PC	-26.25	0.0	-	-	True
				32.9333	19.5667	
HMI	Tablet	-25.8333	0.0	-	-19.15	True
				32.5166		
PC	Tablet	0.4167	0.9857	-6.2666	7.1	False

While these findings strongly support the superior usability of the industrial HMI platform, caution should be exercised due to the small sample size ( $n = 6$  per group), which may limit generalizability and statistical power. Furthermore, no assessment of normality or homogeneity of variance (e.g., via Shapiro-Wilk or Levene's test) was conducted due to small samples and should be considered in future replications. Nevertheless, the results provide initial evidence that a dedicated industrial HMI yields significantly better user experience than general-purpose PC or tablet interfaces in this context.

#### 4.2. Visual fidelity

Visual fidelity is verified with an 8-item manual checklist form comparing status lamps and alarms between the physical NA5 HMI and the Node-RED dashboard view. The test applies standard HMI consistency practice as discussed by Zawra et al. [9] and Majrashi et al. [7]. Each participant graded visual fidelity on five-point scale where 1 is very poor match and 5 is a perfect match. Figure 4 and 5 shows the side-by-side comparison between the NA HMI and Node-RED version of the interface.



**Figure 4:** NA HMI (Left) vs. Node-RED HMI (main control panel & I/O Status)

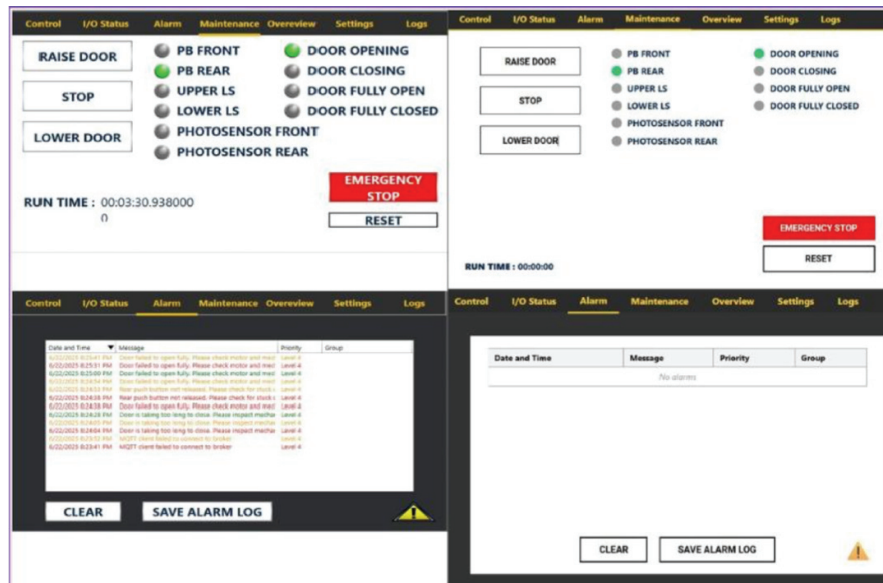


Figure 5: NA HMI (Left) vs. Node-RED HMI (maintenance & alarm)

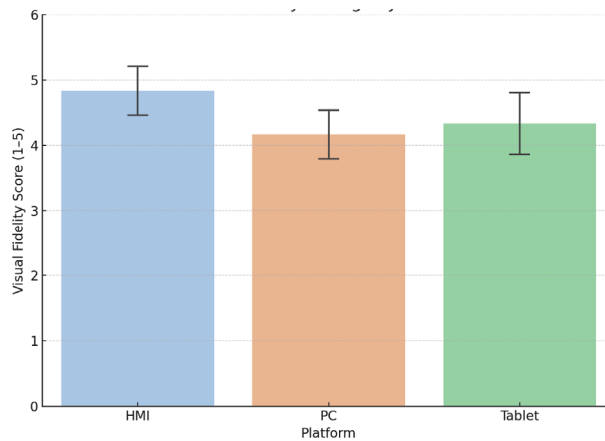


Figure 6: Visual fidelity ratings comparison

### 4.3. Task Completion (TCT)

Task time for six core operations and Open-Door Response Time is logged in CSV, using stopwatch and debug timestamps. The goal is to keep delay under 1 second for LAN, referencing real-time MQTT latency check guidelines [5][6]. The first test asks whether the combination of device class and network speed alters the time users need to finish a series of preset tasks with the system. Because TCT reflects how quickly users move through practical scenarios, it serves as a key gauge of overall efficiency.

#### a) NA HMI vs PC connected to a LAN network

The decision to evaluate task completion performance between the Omron NA HMI and PLC over a wired LAN connection reflects real-world industrial deployment

scenarios, where reliability and deterministic behavior are essential for safety-critical operations. Industrial HMIs are typically deployed in fixed locations and are expected to maintain uninterrupted communication with programmable logic controllers (PLCs) for real-time control, alarm monitoring, and operator interaction.

A wired Ethernet (LAN) connection ensures minimal latency, reduced packet loss, and immunity to wireless interference, which are critical in tightly synchronized automation environments. By simulating this industrially accurate configuration, the experiment captures latency and task responsiveness under stable network conditions, providing a robust baseline for comparing against more variable, consumer-grade wireless configurations.

A one-way ANOVA was conducted to examine whether communication latency differed significantly between two interface configurations: HMI connected via LAN (HMI\_LAN) and PC connected via LAN (PC\_LAN). Each group consisted of 36 latency measurements (HMI\_LAN:  $n = 36$ , PC\_LAN:  $n = 36$ ). Data is presented as mean  $\pm$  standard deviation: HMI\_LAN ( $8.34 \pm 3.06$ ) and PC\_LAN ( $12.92 \pm 7.72$ ).

**Table 4.** ANOVA result on task completion between NA HMI vs PC

Source of variation	SS	dF	MS	F	P-value	F crit
Between groups	377.2089	1	377.2089	10.9413	0.00149	3.97778
Within groups	2413.295	70	34.37564			
Total	2790.504	71				

The analysis revealed a statistically significant difference in mean latency between the two configurations,  $F(1, 70) = 10.94$ ,  $p = 0.0015$ , exceeding the critical F-value of 3.98. This suggests that interface type had a significant impact on LAN communication delay, with the HMI platform yielding substantially lower latency than the PC-based setup.

Although the difference is statistically significant, it is important to note the large variance observed in the PC\_LAN group ( $\sigma^2 = 59.57$ ), which may reflect instability in the system or sensitivity to external network factors. In contrast, the HMI\_LAN configuration showed more consistent performance with relatively low variance ( $\sigma^2 = 9.38$ ). These findings indicate that HMI-based implementations may provide more predictable and reliable communication performance in latency-sensitive industrial environments. Future studies are encouraged to replicate this analysis across different network conditions and with additional device types to ensure robustness and generalizability.

b) PC vs Tablet on Wi-Fi connection

Conversely, the Tablet and PC interfaces were tested using Wi-Fi to emulate flexible, mobile, and operator-level access scenarios that are increasingly common in Industry 4.0 and smart factory environments. As modern production floors integrate mobile

devices and browser-based interfaces into their human-machine interaction strategies, Wi-Fi-enabled control and monitoring are becoming prevalent for roles such as maintenance technicians, supervisors, and line-side operators. Testing under Wi-Fi conditions introduces realistic latency variability and signal integrity challenges that reflect actual usage environments for these devices. This setup allows the study to assess the practical performance limitations of wireless deployment, highlighting the trade-offs between mobility and responsiveness when interfacing with industrial systems through consumer devices.

A one-way ANOVA was conducted to examine whether task completion latency differed significantly between two wireless interface platforms: PC using Wi-Fi (PC\_WiFi) and Tablet using Wi-Fi (Tablet\_WiFi). Each group consisted of 36 measurements (PC\_WiFi:  $n = 36$ , Tablet\_WiFi:  $n = 36$ ). Data is presented as mean  $\pm$  standard deviation: PC\_WiFi ( $11.76 \pm 5.11$ ) and Tablet\_WiFi ( $13.33 \pm 6.56$ ).

**Table 5.** ANOVA result on task completion between Tablet vs PC

Source of variation	SS	dF	MS	F	P-value	F crit
Between groups	44.86237	1	44.25837	1.279464	0.261859	3.97778
Within groups	2421.393	70	34.59133			
Total	2465.651	71				

The analysis revealed no statistically significant difference in latency between the two platforms,  $F(1, 70) = 1.28$ ,  $p = 0.262$ . The mean difference did not exceed the critical F-value threshold of 3.98. This indicates that while the Tablet\_WiFi group showed a slightly higher average latency, the difference was not strong enough to infer a meaningful performance gap in terms of task completion time.

The Tablet\_WiFi group showed greater performance fluctuation, as indicated by a higher variance ( $\sigma^2 = 43.11$ ) compared to the PC\_WiFi group ( $\sigma^2 = 26.08$ ), pointing to less consistency in tablet-based interactions over a wireless network. Despite this, the absence of a statistically significant difference suggests that both platforms deliver comparable outcomes under typical Wi-Fi conditions. As such, either device could be considered suitable for general-purpose or non-time-critical industrial applications where the constraints of wired infrastructure are not a primary concern.

c) PC on LAN vs Wi-Fi

The comparison between PC on Wi-Fi and PC on LAN was designed to isolate the impact of network infrastructure on system responsiveness, using a constant hardware platform to eliminate device-related confounders. This experimental choice builds upon earlier tests where the industrial NA HMI was evaluated against the PC under LAN conditions, and the tablet was benchmarked against the PC over Wi-Fi. While those comparisons provided insight into how different interface devices perform

under similar network environments, the current test focuses specifically on how the same PC behaves under two contrasting network configurations: wired and wireless.

By holding the device constant, this experiment directly measures the latency trade-offs introduced by wireless connectivity which is an increasingly relevant consideration in the adoption of mobile or decentralized control solutions. The results from this setup help contextualize whether observed performance differences in earlier comparisons were predominantly due to the device type or network condition, thereby strengthening the overall validity of the experimental findings across all platforms.

A paired-samples t-test was conducted to evaluate whether there was a statistically significant difference in task completion latency when using a PC connected via LAN versus the same PC connected via Wi-Fi. The comparison included 36 matched observations collected under both network conditions. The mean latency for the PC using LAN was 12.92 ms ( $\pm 7.72$ ), while the same PC recorded a lower mean latency of 11.76 ms ( $\pm 5.11$ ) when operating over Wi-Fi.

The t-test revealed a statistically significant difference in mean latency between the two configurations,  $t(35) = 2.17$ ,  $p = 0.037$  (two-tailed). The result exceeds the critical t-value for a two-tailed test at the 0.05 level ( $t_{crit} = 2.03$ ), indicating that the network mode had a measurable effect on performance. Furthermore, the Pearson correlation between paired samples was high ( $r = 0.96$ ), suggesting strong within-subject consistency in how each system responded across both conditions.

While the Wi-Fi configuration yielded slightly faster average latency, this result should be interpreted with caution. The broader variance observed in the LAN setup ( $\sigma^2 = 59.57$  vs.  $\sigma^2 = 26.08$  for Wi-Fi) suggests possible fluctuations due to factors such as switch latency, Ethernet contention, or polling intervals on the wired network. Conversely, the tighter variance under Wi-Fi may reflect the benefits of modern wireless chipsets with adaptive rate control and buffer optimization. However, this does not imply that Wi-Fi is universally superior. In industrial environments, Wi-Fi performance can be affected by signal interference, congestion from multiple clients, or structural attenuation (e.g., metal enclosures or thick walls), which may not be present in the controlled lab setting of this study.

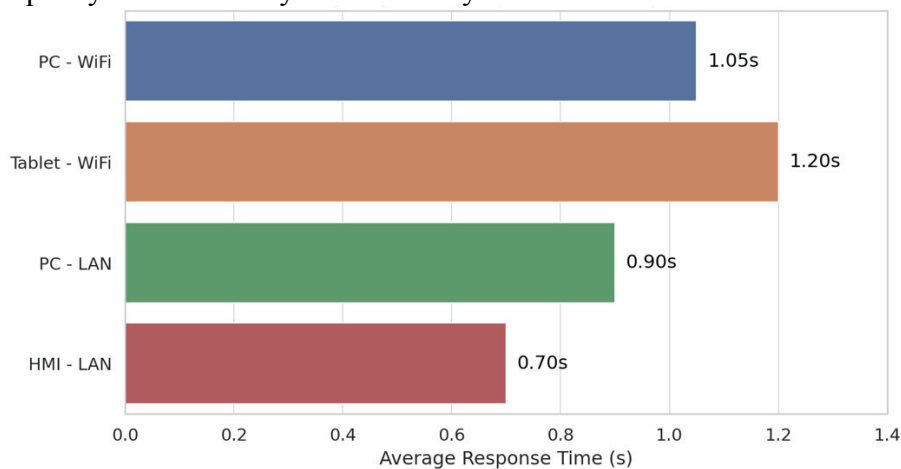
Moreover, network prioritization or Quality of Service (QoS) settings on the LAN infrastructure may have been misconfigured or absent during testing, inadvertently leading to higher jitter or queuing delay for the wired connection. Future studies should explore the influence of network topology, switch buffering, background traffic, and interference patterns to better understand these dynamics. Therefore, while the present data shows that Wi-Fi can match or even outperform LAN in specific contexts, long-term reliability, predictability under load, and security requirements should still guide deployment decisions in production-grade systems.

#### 4.4. Response Time (Open-Door Main Control)

Response time plays a vital role in industrial control systems, especially for time-sensitive actions such as opening a machine door. As the primary link between operators and PLCs, HMI platforms vary in performance depending on device type and network setup. This section compares average response times across four

configurations to assess how these factors impact system responsiveness during the execution of the "Open Door" control function.

Figure 7 illustrates the average system response time recorded across four device-network configurations during the execution of the "Open Door" main control function for five repetitions of the same control operation. The results clearly indicate that both network infrastructure and device class significantly influence control responsiveness. The **HMI-LAN** configuration yielded the lowest average response time (0.70 s), underscoring the advantages of using purpose-built industrial hardware over a stable wired connection. In contrast, the **Tablet-WiFi** configuration recorded the highest average response time (1.20 s), likely due to a combination of reduced processing capacity and variability introduced by wireless communication.



**Figure 7:** Average system response by device

The **PC-LAN** and **PC-WiFi** configurations demonstrated intermediate performance (0.90 s and 1.05 s, respectively), with the LAN-connected PC benefiting from lower latency compared to its wireless counterpart. This contrast highlights the performance penalty associated with wireless transmission, even when the device hardware remains constant.

These findings support the continued preference for dedicated, wired HMIs in applications where rapid and reliable system interaction is critical, such as real-time control operations in industrial settings. While tablet- and PC-based wireless interfaces offer flexibility and ease of deployment, their use should be restricted to monitoring tasks or non-critical operations where occasional latency is acceptable. Selection of interface platforms should therefore consider not only usability and accessibility, but also the deterministic response behavior required by the specific application context.

#### 4.5. Resource Usage Benchmark

To evaluate the computational efficiency of the Node-RED server under varying load conditions, both CPU and RAM usage were monitored across six device activity level; from idle to five concurrent devices. The setup is as shown in Figure 8. Resource data were collected via the Windows Task Manager and exported in CSV format for consistency. As shown in Table 4-5, the system remained stable throughout, with CPU

utilization peaking at only 0.2% and RAM usage increasing modestly from 89.2 MB to 120.3 MB as the number of active devices increased.

**Table 6.** Resource Usage for Node-RED dashboard

Active Devices	CPU Usage	RAM Usage (MB)
Idle	0%	89.2
1 device	0%	95.4
2 devices	0%	103
3 devices	0.1%	110.8
4 devices	0.1%	118.1
5 devices	0.2%	120.3

Notably, no performance degradation was observed during MQTT handshakes or dashboard interactions, even under maximum simulated load. This stable behavior aligns with prior findings that highlight Node-RED's suitability for lightweight industrial deployments due to its event-driven architecture and minimal overhead. The observed usage profile reflects well-documented performance characteristics of Node-RED in edge computing scenarios and confirms its appropriateness for multi-client control in production settings without the need for high-spec server infrastructure.



**Figure 8:** HMIs as accessed concurrently by 5 users

Taken together, the findings demonstrate that the integrated HMI architecture; comprising a hardened MQTT backbone and synchronized Node-RED dashboards; offers a robust and efficient alternative to traditional industrial interfaces. The NA HMI exhibited superior usability and task responsiveness under LAN conditions, affirming its suitability for latency-sensitive operations. While PC and tablet interfaces delivered comparable usability scores under Wi-Fi, their higher variability and longer task response times suggest limitations for real-time control. The PC's behavior across LAN and Wi-Fi further confirmed that network infrastructure significantly influences

system performance, even with identical hardware. Additionally, the Node-RED server showed negligible resource consumption under increasing concurrent access, confirming its scalability for small to mid-scale industrial deployments. These results collectively validate the proposed system's practical applicability and highlight key design considerations for balancing performance, mobility, and usability in modern Industry 4.0 environments.

### **Conclusion**

This study proposed and validated a hybrid HMI architecture that integrates an industrial-grade NA HMI with a browser-based Node-RED interface over a secured MQTT protocol, aiming to enhance usability, accessibility, and responsiveness in legacy industrial environments. The results confirmed the hypothesis that such integration could provide a scalable and cyber-resilient solution without compromising on real-time performance. The system delivered sub-second response times, statistically robust usability scores, and stable server behavior under concurrent load; demonstrating practical feasibility for deployment in small to mid-sized industrial setups.

Comparative evaluations between device types and network configurations revealed that wired interfaces (e.g., NA HMI, PC on LAN) consistently outperformed wireless counterparts in task completion and response variability. Nevertheless, the wireless Node-RED dashboard maintained acceptable performance, suggesting its utility for non-critical supervisory tasks in mobile contexts. These findings offer valuable insights for engineers balancing cost, mobility, and control performance in Industry 4.0 retrofitting projects.

Future work will focus on expanding the system's integration with alarm management standards such as ISA-18.2, enhancing cybersecurity through anomaly detection at the edge, and evaluating long-term reliability in larger-scale industrial deployments. Incorporating role-based access control, audit trails, and adaptive interface personalization may further strengthen the framework's readiness for critical operations in smart manufacturing environments.

### **Acknowledgement**

This research publication is funded by Universiti Teknikal Malaysia Melaka.

### **References**

- [1] K. Ferencz and J. Domokos, "Rapid Prototyping of IoT Applications for the Industry," in *2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, 2020, pp. 1–6. doi: 10.1109/AQTR49680.2020.9129934.
- [2] P. S. B. Macheso, T. D. Manda, A. G. Meela, J. S. Mlatho, G. T. Taulo, and J. C. Phiri, "Industrial Temperature Monitor Based on NodeMCU ESP8266, MQTT and Node-RED," in *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 2021, pp. 740–743. doi: 10.1109/ICAC3N53548.2021.9725469.

- [3] P. S. B. Macheso, T. D. Manda, A. G. Meela, J. S. Mlatho, G. T. Taulo, and B. M'mame, "Environmental Parameter Monitoring System Based on NodeMCU ESP8266, MQTT and Node-RED," in *2022 International Conference on Computer Communication and Informatics (ICCCI)*, 2022, pp. 1–4. doi: 10.1109/ICCCI54379.2022.9740787.
- [4] A. Zare and M. T. Iqbal, "Low-Cost ESP32, Raspberry Pi, Node-Red, and MQTT Protocol Based SCADA System," in *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, 2020, pp. 1–5. doi: 10.1109/IEMTRONICS51293.2020.9216412.
- [5] K. K. D. R. S. M V, K. S, G. R. M, and S. A. S Meashake, "IIOT based Remote Monitoring and Control of Bottle Filling Process Using PLC and Node-Red," in *2023 4th International Conference on Communication, Computing and Industry 6.0 (C216)*, 2023, pp. 1–5. doi: 10.1109/C21659362.2023.10430762.
- [6] C. Krupitzer et al., "A Survey on Human Machine Interaction in Industry 4.0," *ArXiv*, vol. abs/2002.01025, 2020, [Online]. Available: <https://api.semanticscholar.org/CorpusID:211020553>
- [7] K. Majrashi, M. Hamilton, A. L. Uitdenbogerd, and S. Al-Megren, "Cross-Platform Usability Model Evaluation," *Multimodal Technologies and Interaction*, vol. 4, no. 4, 2020, doi: 10.3390/mti4040080.
- [8] J. Mellado and F. Núñez, "Design of an IoT-PLC: A containerized programmable logical controller for the industry 4.0," *J Ind Inf Integr*, vol. 25, p. 100250, 2022, doi: <https://doi.org/10.1016/j.jii.2021.100250>.
- [9] L. M. Zawra, H. A. Mansour, and N. W. Messiha, "Migration of Legacy Industrial Automation Systems in the Context of Industry 4.0- A Comparative Study," in *2019 International Conference on Fourth Industrial Revolution (ICFIR)*, 2019, pp. 1–7. doi: 10.1109/ICFIR.2019.8894776.
- [10] F. Naujoks, S. Hergeth, K. Wiedemann, N. Schömig, Y. Forster, and A. Keinath, "Test procedure for evaluating the human-machine interface of vehicles with automated driving systems," *Traffic Inj Prev*, vol. 20, no. sup1, pp. S146–S151, Jun. 2019, doi: 10.1080/15389588.2019.1603374.
- [11] C. Rattanapoka, S. Chanthakit, A. Chimchai, and A. Sookkeaw, "An MQTT-based IoT Cloud Platform with Flow Design by Node-RED," in *2019 Research, Invention, and Innovation Congress (RI2C)*, 2019, pp. 1–6. doi: 10.1109/RI2C48728.2019.8999942.
- [12] M. R. Drew, B. Falcone, and W. L. Baccus, "What does the system usability scale (SUS) measure?," in *LNCS*, vol. 10918, 2018. doi: 10.1007/978-3-319-91797-9\_25.
- [13] Kalina Nikolova, "Framework and Practical Guidelines for Creating a Current State Analysis for (Re)Designing an HMI within the Industry 4.0 Era," University of Twente, Enschede, 2022.
- [14] W. Alsabbagh, S. Amogbonjaye, C. Kim, and P. Langendörfer, "Pirates of the MQTT: Raiding IIoT Systems with a Rogue Client," in *2024 8th Cyber Security in Networking Conference (CSNet)*, 2024, pp. 248–253. doi: 10.1109/CSNet64211.2024.10851733.
- [15] H. Yan, H. Hu, and Q. Ye, "Time-Specific Integrity Service in MQTT Protocol," in *2024 23rd ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2024, pp. 113–125. doi: 10.1109/IPSN61024.2024.00014.
- [16] S. B. Chauhan and R. N. Gore, "SMQTT: A Lightweight Clock Synchronization Algorithm for IoT Devices Using MQTT," in *2023 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS)*, 2023, pp. 1–6. doi: 10.1109/ISPCS59528.2023.10296995.

- [17] E. Patti *et al.*, “Event-driven user-centric middleware for energy-efficient buildings and public spaces,” *IEEE Syst J*, vol. 10, no. 3, pp. 1137–1146, 2016.
- [18] A. G. and W. T. G. and S. B. G. and J. S. A. Luis Chuquimarca Jiménez and Alba, “Development of Network System for Connection PLC to Cloud Platforms Using IIoT,” in *Advances in Digital Science*, T. Antipova, Ed., Cham: Springer International Publishing, 2021, pp. 433–443.
- [19] C. Valaboju, M. Amuda, S. C. NCh, S. R. Gopireddy, P. K. Hazaveh, and N. A. Rawashdeh, “A Supervisory Control System for a Mechatronics Assembly Station,” in *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2023, pp. 503–507. doi: 10.1109/ICCAE56788.2023.10111393.
- [20] S. Yolov, G. Hristov, and P. Zahariev, “Development and Integration of an Industrial Automation System based on the MQTT Protocol,” in *2025 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*, IEEE, Jan. 2025, pp. 79–84. doi: 10.1109/ECTIDAMTNCN64748.2025.10962086.
- [21] S. Malhao, R. Dionisio, and P. Torres, “Industrial IoT Smartbox for the Shop Floor,” in *2019 5th Experiment International Conference (exp.at'19)*, IEEE, Jun. 2019, pp. 258–259. doi: 10.1109/EXPAT.2019.8876562.
- [22] E. F. Langås, H. Zahra Bukhari, D. Hagen, M. H. Zafar, and F. Sanfilippo, “Inclusive Digital Twins with Edge Computing, Cloud Communication and Virtual Reality to Achieve Remote Human-Robot Interaction,” in *2024 12th International Conference on Control, Mechatronics and Automation (ICCMA)*, 2024, pp. 323–329. doi: 10.1109/ICCMA63715.2024.10843896.
- [23] X. Zhang, S. Gao, B. Yu, and W. Dai, “Applying Embedded Multi-Core Control Technologies for the Next Generation Industrial Edge Applications,” in *IECON 2024 - 50th Annual Conference of the IEEE Industrial Electronics Society*, IEEE, Nov. 2024, pp. 1–6. doi: 10.1109/IECON55916.2024.10905412.
- [24] P. Chouragadey, B. Rathore, and G. Khare, “Smart Home Automation with MQTT Using ESP32 and Node-RED,” in *2025 Fourth International Conference on Power, Control and Computing Technologies (ICPC2T)*, 2025, pp. 1–6. doi: 10.1109/ICPC2T63847.2025.10958730.
- [25] S. Opačin, L. Rizvanović, B. Leander, S. Mubeen, and A. Čaušević, “Developing and Evaluating MQTT Connectivity for an Industrial Controller,” in *2023 12th Mediterranean Conference on Embedded Computing (MECO)*, IEEE, Jun. 2023, pp. 1–5. doi: 10.1109/MECO58584.2023.10154921.
- [26] P. Nenninger, M. Gierl, and R. Kriesten, “A Contribution to Publish-Subscribe Based Communication in Industrial Applications,” in *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, IEEE, Oct. 2019, pp. 0424–0429. doi: 10.1109/IEMCON.2019.8936176.