Leveraging Transdisciplinary Engineering in a Changing and Connected World
P. Koomsap et al. (Eds.)

doi:10.3233/ATDE230629

Exploring the Adoption of UX-Driven Approaches to Design Industrial PLC User Interfaces

Margherita PERUZZINI[a,1], Lorenzo VALENTINI[a,1], Alessio TUTARINI[a,1], Pietro BILANCIA[b] and Roberto RAFFAELI[b]
[a] *Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, Modena, Italy*
[b] *Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Reggio Emilia, Italy*

Abstract. Modern automated production systems (e.g., automatic machines, assembly lines, robotic cells) are typically governed by dedicated industrial controllers, such as Programmable Logic Controllers (PLCs), which supervise and coordinate the process by exchanging I/O data, sequencing tasks or triggering actions with the involved automation modules. Different solutions have been developed to offer an intuitive Human-Machine Interface (HMI) programming to the user, based on PLC HMI editors, according to vendor-specific programming languages. However, in the current industrial practice, user interfaces (UIs) are usually generated by software specialists and far from adopting any user-centered approach. As a result, the generated UIs are poorly usable and hard to understand for end users (e.g., operators), diverging from Industry 5.0 ideas that put humans at the center of the modern factory design. In this context, the present paper aims at exploring how the adoption of User eXperience (UX) driven approaches can benefit the design of industrial PLC UIs, reflecting on advantages and limits, and transdisciplinary perspectives. A case study utilizing Beckhoff TwinCAT as PLC environment and Adobe XD as UX design tool is examined and discussed.

Keywords. User experience, User-centered design, Human-machine interface, User interface, Programmable Logic Controllers.

Introduction

While Industry 4.0 focuses mostly on connectivity and data analytics, the novel Industry 5.0 paradigm pays attention to the transition towards a human-centric, sustainable, and resilient industry, shifting from smart manufacturing to an effective human-machine co-working [1]. This implies to move from a technology-driven approach to a human-driven design strategy, where the human-centricity concept should be adopted to design every factory component, starting from the UIs: indeed, UIs are responsible to dialogue with people, throughout the whole manufacturing value chain, and must enable the workers' up-skilling and re-skilling when needed [2].

Automated systems represent a good example of industrial equipment where human-centered design approaches are scarcely implemented yet [1]. As they are conceived to work automatically, designers and engineers usually think that they don't need to interact with people. Conversely, they require intuitive and efficient UIs to enable rapid

[1] Corresponding Author, Mail: margherita.peruzzini@unimore.it.

interaction with the machines and the entire process. An optimized interaction could lead to numerous benefits: improving process efficiency and quality, shortening the learning process, enhancing people motivation and satisfaction, improving process safety and knowledge management. All these aspects become essential for guaranteeing overall factory productivity. The HMI is a UI that specifically allows humans to interact with machines [3] and it can be seen as a crucial part of any industrial control system. From a technological viewpoint, it represents the middle layer between the factory operations and devices (i.e., sensors, actuators, and PLCs) and the management layer (i.e., manufacturing execution systems and enterprise resource planning), according to the automation pyramid shown in [4]. The most common type of industrial HMI is represented by a control panel through which a manufacturing machine is partly or fully operated by a human worker, providing the necessary information about the controlled process or occurring events. Five interaction modalities (i.e., gaze, voice, gesture, tactile, and haptic interactions) can be identified to interact with industrial HMIs, depending on the different tasks and type of communication required [5]. Therefore, the design of the HMI directly influences its usability which in turn affects the overall machine efficiency, the manufacturing operations, and the interdependent relations between human, technological and organizational factors within the factory [6]. Even though new models to include digital assets within the factory have been recently proposed (e.g., Reference Architectural model for Industry 4.0 - RAMI4.0) [7], there is still a lack of practical methods to enable an effective interoperability between the different automation levels, for instance to link the PLC level with the HMI level.

This research moved from the lack of study and specific guidelines about the design of human-centric HMI in the industrial context, specifically for PLC. This paper proposes to adopt a transdisciplinary view, based on system thinking and collaborative practices among different competencies, to promote the adoption of a UX-driven approach to help engineers in designing industrial PLC HMIs. Currently, HMIs are usually designed without a solid knowledge of target users and without looking at UX design principles, so that industrial HMIs are usually affected by problems of comprehension, confusion in handling the large number of adjustable variables as request to control modern machines, especially for the less skilled operators [8]. Although recent papers have already defined the urge for a specific design strategy to ensure the PLC HMIs usability and relevance [9], there are no practical guidelines so far. The goal of the paper is to explore the combined use of UX design tools and commercial HMI programming tools and to guide engineers and decision-makers in adopting a human-centric view to design highly usable and comprehensible interfaces. A UX-driven approach for PLC HMI design is defined according to participatory practices and typical user-centered design phases. A use case consisting of a testing machine for industrial servodrives, whose logic is entirely controlled with a Beckhoff PLC, is considered to conduct the research, and validate the design approach. Instead of a typical industrial HMI design process, mainly managed by software specialists, a UX expert was involved in the design team and collaborated with software programmers to carry out a preliminary analysis of target users and to properly conceive the HMI visual design and information architecture by directly addressing the users' needs. The remaining of the paper is structured as follows: Section 1 outlines the existing design flow for PLC HMIs adopted in industry and discusses its related challenges. Section 2 describes the proposed UX-approach, whereas Section 3 reports its direct implementation on an industrial use case. Subsequently, Section 4 presents a set of generally applicable guidelines for designing PLC HMIs, whereas Section 5 summarizes the study's findings and innovations.

1. PLC HMI design in industry

A PLC is a dedicated industrial computer with a reconfigurable input/output system and a fully programmable logic module which is used to automate a wide range of industrial machines and processes. The PLC software can be modified to suit the specific requirements of the controlled device(s), making it highly versatile and adaptable to different industrial settings. A PLC-based automation system allows the operator to interact through HMI to control the real-world processes in real time; the user can see the information on a graphical interface and interact with items (e.g., buttons, menus) to make the control decision and provide the necessary inputs to the PLC logic that, in turn, manipulates the process variables at lower levels. The following sections examine the flow currently employed in industry for designing PLC HMIs and recall its primary issues. The typical PLC HMI design process is carried out by software or automation specialists, with an informatics or electronics engineering background, without proper consideration of the human-machine interaction to realize. It starts with the analysis of the HMI functions and the definition and positioning of the main UI elements, according to a reference wireframe. Frequently, the design is not completely original, but consists of adding functions of pre-exiting interfaces. Sometimes, not so commonly, external UI designers are involved in proposing some graphical interface concepts, without a direct relation with the implementation side. Therefore, the implementation phase is characterized by a graphic and functional downgrade of the proposed concept, due to the excessive effort at the programming level. In some cases, the implementation has very tight deadlines between the machine's realization and the launch on the market, so that there is no time to properly design the human interaction. In addition, the large number of machine variants requiring specific HMI designs and their relatively short lifespan in the modern market inevitably reduces the available time [10]. To speed up the process, pre-established templates are repeatedly used and adapted to the new HMI prototypes, according to practical consideration about the layout and the programming effort, without taking into consideration the operators' experience. Additionally, the limited knowledge of UX-driven approaches in industry further hinders the consideration of user needs during the design process. As a result, HMIs are difficult to understand by operators and show important limitations in terms of usability.

Based on experience, industrial PLC HMI designs often fail due to various issues. Commonly encountered user-related problems include: fragmentation (e.g., users have to constantly switch between screens to complete a single procedure), confusion (e.g., too much data and information are displayed without a clear structure), inconsistent controls (e.g., similar controllable objects react differently depending on their location or dissimilar controls all related to the same task), irrelevant operational steps (e.g., additional steps unrelated to the procedure but required by the internal functions of the control system), slow response (e.g., system feedback is delayed despite improving computer performance), overload (e.g., users receive many system messages and alarms that they cannot deal with). The best way to solve these problems is to adopt a transdisciplinary view: according to system thinking, we should consider the users as a part of the factory and include their requirements into the overall system design, combining technical and socio-related aspects. More specifically, experts in UX should be included into the HMI design team, similarly to different fields (e.g., e-commerce interfaces), to define UX-driven requirements and support the software specialists into the following HMI implementation.

2. UX-driven approach

UX design places the users' experience, including behaviors, emotions, preferences, perceptions, physical and psychological responses, at the core of the design process [11]. By comprehending the needs, wants, and actions of users, UX design approach informs and directs product design, and aims at facilitating the creation of user-friendly, intuitive products that deliver a positive experience for the user. The research proposes to extend the typical UX design cycle for HMI design into six phases, each of them consisting of specific HMI design techniques, as presented in Figure 1. Each phase is detailed in the following paragraphs.

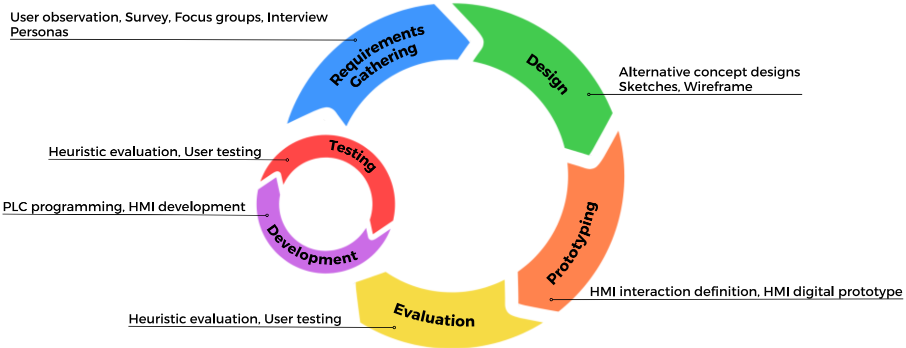

Figure 1. UX-driven cycle for HMI design.

1. **Requirements Gathering:** after the creation of a multidisciplinary team, made up of people with different background, this phase aims at collecting valuable insights into the users' needs, desires, and actions, by employing a range of discovery techniques, such as user observation, surveys, focus groups, and interviews. Once data is collected, they can be synthesized using personas to define the target user profile and pinpoint design opportunities. This phase allows to gain a comprehensive understanding of user needs and behaviors, to define a reliable list of user requirements for the HMI.
2. **Design:** this phase involves creating a range of HMI design solutions to meet the user requirements. Design starts from concept designs, which can be presented in different formats such as sketches and wireframes, providing a preliminary representation of the final product appearance and functionality.
3. **Prototyping:** this phase involves creating a digital prototype of the HMI representing the human-machine interaction flow. According to the design stage and the realism of the functions replicated, prototypes can be low-fidelity or high fidelity. The purpose of prototyping is to test and evaluate the design solution created during the design phase, identify any usability or functionality issues, and refine the design. The multidisciplinary team is necessary in this phase to evaluate the prototype. The aim of this stage is to develop a prototype-ready design for the subsequent phase.
4. **Evaluation:** the evaluation phase is fundamental to ensure that the design meets the user requirements and uses the prototypes as produced in the previous phase to organize different evaluations, involving experts (e.g., heuristic evaluation) or users (e.g., usability testing). The output includes a detailed evaluation report and a list of recommendations for improvements.

5. **Development:** the successful HMI design prototype, from the previous stages, is developed and implemented in the PLC. The implementation process includes the PLC programming and the implementation of the HMI based on the prototype, linking the PLC variables with the interface elements and interactions.
6. **Testing:** it consists of a final usability testing conducted on the developed HMI on PLC, involving users in the real context of interaction. From this phase, further optimization recommendations could be defined. The cycle repeats until the UX target is reached.

3. Industrial use case

The design approach outlined in Section 2 has been validated on industrial equipment, specifically a test-rig for robotic servodrives (for detailed descriptions refer to [12], [13]). At the initial stage, the rig lacks a proper HMI, which means that it can only be operated within the PLC programming environment by accessing and editing variables. Two specific reasons led to the selection of this use case for the present research, 1) the complexity of the system, and 2) the commonly used PLC environment. The system consists of two servodrives (controlled in position/velocity/torque depending on the requested operation mode), a set of sensors to be constantly monitored (i.e., encoders, thermocouples, torquemeters, etc.) and a large library of programmed tasks (based on the experiment to be run). The conceptual system architecture is presented in Figure 2. This requires a meticulous UI design and integration of multiple functions that allow for in-depth exploration of the UX-driven approach. The system uses Beckhoff TwinCAT as automation software for the primary PLC governing the rig functioning. Such package is widely used in industry and integrates dedicated tools for HMI development. Next sections report the tools for HMI design and prototyping and discuss the development and implementation phases.

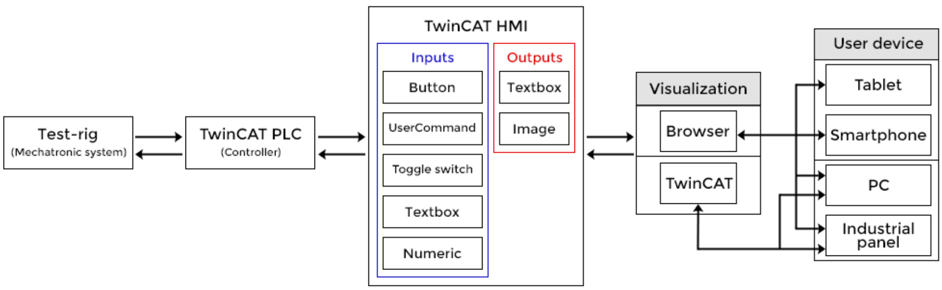

Figure 2. Conceptual system architecture composed of the mechatronic system, the controller, the HMI created through TwinCAT and displayed on a device through a visualization bridge.

3.1 Software tools

Two commercial software tools have been considered, one related to UX design (i.e., Adobe XD) and one for the HMI development on the PLC-based system, available from the technology provider (i.e., TwinCAT HMI by Beckhoff). Adobe XD is well-known for graphic UI designers to prototype digital HMIs. In the study, it has been used to design, prototype and evaluate the HMI as digital applications, able to create the UI behaviors and simulate user interactions and animations. TwinCAT HMI allows customized UIs and dashboards for industrial automation systems to be created. It has been used in the study for HMI development into the PLC environment.

3.2 Interface design process

The HMI use case was designed according to the UX-driven cycle as shown in Figure 1. Firstly, a team of four people with different backgrounds was created (i.e., a UX expert, an HMI designer, an automation expert, a software programmer). For the definition of the user requirements, two users were observed while interacting with the machine and subsequently interviewed to elicit their needs (i.e., intuitive viewing of the main parameters, display of the operating status, activation of a set of main functions, execution of a set of experimental procedures with a manual control on specific parameters) and preferences. Needs were defined by combining verbal and non-verbal language as well as the interaction modalities collected during the user observations. As a result, two personas were defined (i.e., Machine coordinator, PLC programmer). At the end of this phase, a prioritized list of user requirements was obtained (Table 1).

Table 1. List of personas' requirements obtained.

Machine coordinator	PLC programmer
1. Intuitive access to all standard functions.	1. Clear and intuitive visualization of data during operations.
2. Conducting experiments, possibility to save obtained results, and a checklist of required preliminary operations.	2. Preservation of the interactive flow structure.
3. Displaying the status of operations.	3. Visualization of the Test RIG configuration before startup.

After that, different solutions were designed, starting from the homepage. Preliminary prototypes based on paper-based sketches were discussed by the design team and showed to the sample users to collect their impressions. The feedback suggested the solution to be chosen and some improvements, i.e.: i) simplifying the display of information; ii) repositioning certain functions; iii) displaying the machine's status; iv) inserting an easily accessible and always visible stop button and v) displaying different test-rig physical configurations. Then, the improved solution was digitally designed and prototyped using Adobe XD as shown in Figure 3.

Figure 3. HMI homepage as designed – optimized solution.

Such design contains the machine front view to obtain a more direct and intuitive information display. The upper bar informs the user about the machine status, while the lower bar contains the tabs of the different pages. The designed solution consists of three pages: i) Home, which provides a synthetic overview of the machine and its fundamental parameters to control; ii) Manual, where the user manually sets the machine parameters; and iii) Experiments, where the user can choose to run specific experimental procedures from a predefined list. Due to the complexity of the machine and the great number of operation modes, it has been necessary to add various UI elements to allow the user to properly control and monitor the machine: from switch buttons, to scrolls, status bars, buttons, alarms, drop-down menus, popups, color variations of labels, interactive loading bars. Such a solution was evaluated by experts and users using the Adobe XD interactive prototype on a tablet alongside the machine, to replicate the real context of use. Subsequently, usability tests were carried out with two users to identify problems and to define consequent corrections.

Once the prototype had been tested and optimized, it was possible to proceed with the development phase. Two approaches can potentially be followed at this point: 1) Exporting the interface from Adobe XD and importing into TwinCAT HMI using HMTL and CSS, 2) Creating the interface in TwinCAT HMI, using the provided libraries. The first approach, the most efficient and promising at first sight, turned out to be very complex due to a lot of incompatibility issues arising during the import process, which led to errors in the interface layout and content transferring. Unfortunately, such issues cannot be totally solved by manually modifying the HTML code. Moreover, once the interface structure is imported, appearance and animations would need to be re-programmed using CSS and JavaScript, respectively. On the other hand, the second approach seems to be very time-consuming, but it was found the most convenient from a practical point of view. Indeed, the creation of the interface as a new item in TwinCAT HMI was relatively easy thanks to the features and libraries provided by the software, although the initial UX design had to be slightly revised, and some specific interactions had to be realized by additional coding. The second approach allowed to achieve the goal, thanks to the following workaround in TwinCAT HMI:

a) TwinCAT HMI does not natively support the designed animation on the navigation button located in the bottom bar (i.e., the button remains highlighted in a different color compared to the other buttons, indicating which page the user is currently on). To obtain this effect, a color variation of the three buttons was associated to the click of the mouse through programming;
b) TwinCAT HMI does not allow to position the UI elements by imposing their distance from other elements, but only from the page borders. To address this limitation, all element positions in Adobe XD were aligned with the same reference points used in TwinCAT HMI.
c) TwinCAT HMI recommends the use of *Regions* (i.e., elements allowing the display of a specific *Content*) for creating multiple pages, which can be activated/deactivated according to a specific input. However, every time a *Region* is changed, its predefined value is restored, and an animation occurs that brings variables from the predefined state to the state defined by the connected PLC variable. To overcome these problems, the PLC variables have been connected to the HMI variables so that they remain in the same state as defined by the PLC variable, and the *Regions* were managed in a different way (i.e., instead of activating them through an input variable, we managed each page as open but not

visible time by time). This guaranteed that the HMI variables were updated with the state of the PLC variables.

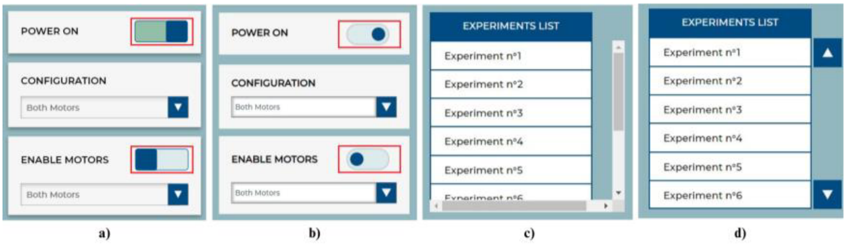

Figure 4. *Toggle switch*: a) Designed on Adobe XD, b) Developed on TwinCAT HMI. *Scroll bar*: c) Designed on Adobe XD, d) Developed on TwinCAT HMI.

d) TwinCAT HMI *Toggle Switch* does not allow any graphic customization, limiting the style of the interface. In this case, the design was not significantly affected. Similarly, the *Scroll Bar* is not customizable. For the same reason, buttons were used to allow the user to scroll a menu step-by-step (Figure 4).

e) TwinCAT HMI *Combobox* should be used to create drop-down menus, although it does not allow advanced graphic customization. For this reason, the *UserCommand* tool can be used which allows to create a command from scratch to develop a drop-down menu graphically in line with the one designed in Adobe XD.

f) Blocks for data input (i.e., the TwinCAT HMI *Numeric* and *Textbox* features) have a default inner shadow that cannot be changed). This could be problematic as different shading communicates different information to the user.

The final homepage developed in TwinCAT HMI, with the above-mentioned workarounds, is shown in Figure 5. A Direct comparison of Figure 3 and Figure 5 shows that the designed homepage and the developed homepage appear to be the same except for minor details (specified in Figure 5). The same result was obtained for the other pages of the HMI. Then, this final project was tested with users according to structured usability testing.

Figure 5. HMI homepage on TwinCAT HMI – as developed.

4. Design guidelines for PLC HMIs

Based on the experience gained from the use case, we have defined a set of guidelines for the design of PLC HMIs, as a list of best practices for those who like to approach the design of industrial interfaces in a user-centered way.

1. Create a multidisciplinary team, including experts in UX and human factors: creation of a multifaceted team is mandatory to generate a usable and intuitive HMI, able to include the different points of view about a same problem. This is not common in the industrial practice, but it is essential;
2. Be careful on the user research stage: the requirements gathering phase has to include careful user research, where users are observed in their daily working conditions and data about the human-machine interaction is collected by survey or interviews. This is necessary to elicit a robust list of user requirements, based on the real users' needs;
3. Structure the information architecture from your conceptual HMI design: any HMI design has to start with a robust UI information architecture before proceeding with the development, depending on the tools chosen. This allows to overcome some minor issues that can occur in the development stage (i.e., adequate positioning and correct management of some features to view information as expected);
4. Try to use commercial software for the HMI development: the commercial tool used for the HMI development (i.e., TwinCAT HMI) has some limits in the graphic design options, but they can be easily handled to create a harmonious graphic style and avoid mismatches between the designed UI and the developed UI;
5. Think outside the box: during the HMI development thinking about alternative solutions can help to overcome some issues, by redefining non-standard elements and combining several standard elements. Non-standard animations and graphical effects (i.e., not available by the HMI development software such as TwinCAT HMI) can be reproduced using fictitious variables to be manipulated both through coding and tools made available by the software itself;
6. Test your solutions with users: even if in a very preliminary stage, users can be involved to evaluate the HMI prototype, also using paper sketches, to understand if the design is going in the right direction.

Thanks to these guidelines, the developed HMI can achieve an adequate balance between the overall UX, the design and development efforts, and the related cost. An alternative solution, not described in this paper, consists of adopting a middleware (i.e., software that acts as an intermediate layer to enable communication and data exchange, between the UI design environment and the PLC environment). Examples of commercial solutions include AVEVA PI Vision and AWS IoT SiteWise. Despite their advantages in terms of flexibility and data visualization/management, these platforms offer a limited level of customization in their graphical library for UI creation. Then, incorporating a commercial middleware entails other challenges. In particular, the integration with existing systems is required as well as additional software skills for personnel and costs for purchasing its license. Overall, this leads to higher expenses and extended implementation periods in contrast to the use of TwinCAT HMI, which is seamlessly integrated into the Beckhoff suite and can be readily used by PLC programmers.

5. Conclusions

The aim of this study is to encourage engineers and decision-makers to adopt a human-centric approach when defining the key aspects of an industrial HMI. A UX-driven approach is proposed to ensure the development of highly usable and understandable PLC HMIs that meet the user needs, integrating human, technological, and organizational factors within the factory according to a transdisciplinary view. The UX design practices were adopted to a use case referring to the development of an HMI to control an industrial equipment, i.e. a novel test-rig for servomotors. A set of guidelines were defined to help software developers to integrate UX design practices into the HMI development using commercial controllers. Furthermore, the use case demonstrated that modern commercial PLC tools offer useful features to completely build a UI, without compromising the proposed design.

References

[1] M. Breque, L. De Nul, A. Petridis, and European Commission. Directorate-General for Research and Innovation., *Industry 5.0: towards a sustainable, human-centric and resilient European industry*, 2021, https://data.europa.eu/doi/10.2777/308407, accessed July, 5 2023.

[2] X. Xu, Y. Lu, B. Vogel-Heuser, and L. Wang, Industry 4.0 and Industry 5.0—Inception, conception and perception, *J Manuf Syst*, 2021, Vol. 61, pp. 530–535, doi: https://doi.org/10.1016/j.jmsy.2021.10.006.

[3] P. Papcun, E. Kajati, and J. Koziorek, Human Machine Interface in Concept of Industry 4.0, *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA), IEEE*, Aug. 2018, pp. 289–296. Doi: 10.1109/DISA.2018.8490603.

[4] ANSI/ISA-95.00.05-2018, Enterprise-Control System Integration - Part 5: Business-to-Manufacturing Transactions.

[5] N. Kumar and S. C. Lee, Human-machine interface in smart factory: A systematic literature review, *Technol Forecast Soc Change*, 2022, Vol. 174, 121284.

[6] M. Oliveira, E. Arica, M. Pinzone, P. Fantini, and M. Taisch, Human-centered manufacturing challenges affecting European industry 4.0 enabling technologies, *HCI International 2019–Late Breaking Papers: 21st HCI International Conference, HCII 2019*, Orlando, FL, USA, July 26–31, 2019, Proceedings 21, Springer, 2019, pp. 507–517.

[7] T. A. Abdel-Aty, E. Negri, and S. Galparoli, Asset Administration Shell in Manufacturing: Applications and Relationship with Digital Twin, *IFAC-papersonline*, 2022, vol. 55, no. 10, pp. 2533–2538.

[8] B. A. Kadir and O. Broberg, Human-centered design of work systems in the transition to industry 4.0, *Appl Ergon*, 2021, Vol. 92, p. 103334.

[9] V. M. Koshti and S. M. Joshi, DESIGN OF HUMAN MACHINE INTERFACE FOR PLC BASED AUTOMATION SYSTEM, *IFAC Proceedings Volumes*, 2007, Vol. 40, no. 18, pp. 343–346, doi: 10.3182/20070927-4-RO-3905.00057.

[10] B.-I. Ionescu, *HUMAN-MACHINE INTERACTION IN INDUSTRY 4.0 AND BEYOND*. https://revija-ventil.si/en/human-machine-interaction-in-industry-4-0-and-beyond/, accessed July, 5 2023.

[11] N.N., ISO 9241-210:2010 'Ergonomics of human-system interaction - Human-centred design for interactive systems', ISO, 2010.

[12] M. Belloni, P. Bilancia, R. Raffaeli, M. Peruzzini, and M. Pellicciari, Design of a Test Rig for Tuning and Optimization of High Dynamics Servo-Mechanisms Employed in Manufacturing Automation, *Procedia Manuf*, 2021, Vol. 55, pp. 48–55, doi: 10.1016/j.promfg.2021.10.008.

[13] P. Bilancia, L. Monari, R. Raffaeli, M. Peruzzini, and M. Pellicciari, Accurate transmission performance evaluation of servo-mechanisms for robots, *Robot Comput Integr Manuf*, 2022, Vol. 78, 102400, doi: 10.1016/j.rcim.2022.102400.