Proceedings of the 19th Conference on Computer Science and Intelligence Systems (FedCSIS) pp. 655–660 DOI: 10.15439/2024F2423 ISSN 2300-5963 ACSIS, Vol. 39

Towards to an interface design for a building operations CPS

Filipe Moreira[1*], Rosana Alexandre[1], Rosa Silva[1], Joªo Oliveira[1], Manuel Alves[1], Joªo Pereira [2], Ana Colim[1], Nelson Rodrigues[1]

[1] DTx – Digital Transformation CoLAB, University of Minho, Portugal, e-mail {filipe.moreira, ana.colim}@dtx-colab.pt
[2] dst group – Rua de Pitancinhos - Palmeira 4700-727 Braga, Portugal, e-mail joaodaniel@innovpoint.com

***Abstract* — Cyber-physical systems (CPS) enable value creation in software development. However, they present several challenges that necessitate a structured approach to design and implementation. We present a concept on how to approach the design and development of a software service, based on multiple CPS. It delineates the specific characteristics of a CPS, including graphical user interface (GUI) components and architectural aspects, while recommending appropriate tools and design methodologies. This approach aligns with contemporary frameworks such as Industry 4.0 and Software as a Service (SaaS). The multidisciplinary nature of CPS leads to several development challenges, and we focus our work on using design thinking techniques, methodologies, and tools that are valuable solutions and key to the sustainability of new designs. These insights may help researchers and industrial practitioners to develop and commercialize service-based CPS.**

***Index Terms* — Cyber-Physical Systems; Industry 4.0; User Support; Graphical User Interface, Design Thinking, Building Operation.**

I. Introduction

The development and integration of industrial Cyber-Physical Systems (CPS), systems that integrate computation and physical processes [8], is a topic of utmost importance, given the significant challenges it presents. The severe consequences of system failures, which can impact safety, productivity, costs, and company reputation, underscore the critical nature of this field. Traditionally, components of CPS, such as physical assets, digital twins, industrial applications, and critical technological aspects like communication and cybersecurity, are developed independently. This traditional siloed approach poses a significant hurdle in harmonizing diverse development paradigms to deliver a seamless user experience while minimizing research and development costs.

Digital twins, unique and essential elements within CPS, are defined variably across literature but represent a digital replica of a physical asset [6]. They support various industrial applications through accessible data and services, operating on local servers or via cloud computing. The emphasis on user-centric design enhances value to industrial practitioners. The integration of digital twins requires a structured approach to mirror the connectivity of physical entities and ensure interoperability across different vendors and building operations stakeholders. The paper outlines the complexities of designing CPS by discussing integrating systems and subsystems into a unified framework that supports robust data management and system operation across different building operation phases, under the design perspective. This technical integration extends to aligning business models and ensuring system components deliver continuous value, adhering to traditional and emerging business practices. Moreover, this paper underscores the integration of human factors, recognizing the importance of seamless interaction between users and CPS. It involves developing adaptable interfaces and fostering safe and efficient human-machine collaboration, an essential aspect of CPS.

II. Methodology

This study is part of a project focused on developed by our design team within a project focused on the development of graphical user interfaces for software-as-a-service (SaaS) within a smart-building management operations context.

The design team’s main challenge was articulating the functional structure, of the several CPS, in a consistent, intuitive and pleasant user experience.

The project involves is a well-established and accomplished construction company in Northern Portugal with an extensive client portfolio. This project is part of a significant modular construction program that benefits from all the newest digital technologies a CPS can offer. Our approach consisted of using design thinking and systems thinking methodologies together to better address the complexity-related problems during CPS design and implementation [4].

To effectively execute this project, a multidisciplinary team established for this project involves distinct groups: industrial practitioners, a team of researchers from DTx’s Ergonomics and Engineering Psychology team, which includes two designers, the Edge and Embedded Computing group, and the DTx Data and Application Engineering group. Moreover, the structure of graphical interfaces should be agnostic enough to adapt to these diverse types of buildings, data collected, and

* Corresponding Author’s email: filipe.moreira@dtx-colab.pt

operations performed by users with different profiles and access permissions. The existence of a document with wireframe characteristics of the software provided by the project's leading company allowed the requirements-gathering work to progress more quickly. The leading company in the project will play a dual role, acting both as a vendor and as a client of the final SaaS that will be born from the developed prototypes. This unique position underscores the company's responsibilities and the need for a comprehensive understanding of the system's design and functionality.

From the beginning, it was clear that the terminology used required a phase of learning and studying the context. This required meetings to discuss and delve into the concepts reflected in this document. Clarifying any misunderstandings needed to be considered. These meetings were crucial in preventing the implementation of interfaces and back-end software that might need restructuring later.

The meetings also fostered a collaborative environment, allowing an understanding of the document's author, including his motivations, professional context, and business ideas. This insight improved the ability to reference a set of conditions, enhanced communication, and contributed to delivering a user-focused project.

After the first meetings to address the issues raised by the document with the project leader's wireframe, our work consisted of prioritizing some activities, which we will explore in more detail in this article. These activities included:

a) Creating a project glossary, where we could find a common language free from misunderstandings.
b) Mapping user stories of all potential users of the system, listing the accessible functionalities and activities of each one within the system.
c) Developing a low-fidelity prototype, where we organized the functional hierarchies into a structure of standard interfaces.
d) Crafting a service design blueprint, to uncover touchpoints that the user has with the service.
e) Articulating different users involved in the building maintenance ticketing process.
f) Creating a medium-fidelity prototype, with the fine-tuning of details and interaction mechanics per user, anticipating and paving the way for high-fidelity implementation and prototyping in Front-End.

Figure 1 illustrates the development stages of the main phases of the framework to support the design process. As mentioned previously, these five activities were carried out by the design team. Despite appearing separately, the process often occurred simultaneously and iteratively, as layers of information were added.

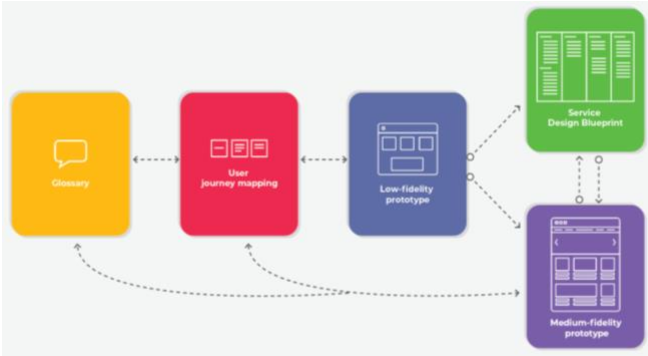

Fig 1: Representation of the main phases to support the design process

III. Results

From a human-centered design perspective, the design team conducted co-creation sessions with the stakeholders and the development team to understand users' needs and potential solutions [10].

A. Glossary

Given the complexities of the software and hardware platforms that make up this CPS, the initial focus of the interface design project was to initiate the development of a glossary that would allow discussing different views of the nomenclatures used, allowing for unification in the language used by the team multidisciplinary involved.

The creation of the glossary began with an in-depth reading of the initial wireframe, looking for inconsistencies in the terms used, discussing their meanings with the different stakeholders of the teams involved in the project, suggesting new nomenclatures, listing the different definitions and applications alphabetically, and considering their context and usage.

These discussions with the teams were carried out throughout meetings to approach the content of this document, simultaneously with the discussion of the organizational structure of the developing software and the identification of each of the potential users of the platform. In an effort to achieve gradual and sustained discovery, the goal was to identify a practical and optimal logic that will enable straightforward and clear use of its functionalities.

The determination of some of the nomenclatures, evident at the beginning for users with an elevated level of expertise, were gradually adapted whenever we found potential conflicts between interpretations, considering the different degrees of specialization and knowledge of the users who will use this SaaS. The effort allocated to the articulation between the different teams made it possible to gather consensus and compromise solutions to better create well-structured software adaptable to different user profiles.

This search for a common language, applicable both to the scope of the teams involved in the development of this CPS and to the scope of the different users who would operate as a team in its use, was fundamental to emphasize the social and human dynamics that should be an integral part of a CPS under construction [13].

B. *User Story Maps*

Another tool for the interface design was the application of User Story Maps. This technique allows for identifying what is being developed from the perspective of the end users and helps to understand the reasoning behind the solution. It provides an overview of the product/service while also allowing it to be visualized in a fragmented manner [11].

When building the user story map, the team identified the different contexts of use for each potential user of the CPS, the functionalities that each can and should access, and the access permissions at various levels of specification of each component. More specifically, this resulted in a visual representation of the users of the CPS, with a hierarchy of core activities to be performed in the system and subsequent tasks.

The original wireframe that served as the starting point for this exercise suffered from the bias of being built from the perspective of a user with Super Administrator privileges, which is why the need to analyze the role played by each of the potential users of the software became critical. **This analysis enabled** the team to structure more holistically the different types of interfaces necessary and applicable to each user profile.

At the top of the hierarchical structure of the system is the Super Administrator, who has access to all its functionalities: capable of editing and parameterizing the system, creating customer and user profiles stipulating roles and permissions, as well as creating, editing, and parameterizing all assets and equipment included in the system (Figure 2). All other users will have lower access levels to the system, depending on their role and the type of permissions assigned (Management Areas Interfaces, Maintenance Tasks Interface, or Final User/Client Interface).

The initial view of the software interface architecture, from the perspective of a Super Administrator, brought together all the accessible information of the recommended CPS, categorizing specificities and functionalities in a highly partial usage logic. Given the predictability small number of Super Administrator profiles, this view had limitation despite the breadth of processed, processable, and specifiable data.

Thus, a vision of a service-oriented architecture was required, making it more suitable for subsequent incorporations of new services and devices, commonly found in contexts of Smart Home environments. Such environments allow automation, emergency detection, and energy savings, among other features [12].

This Cyber-Physical Systems will have the functions of collecting and processing data to support the creation or maintenance of the building throughout its users. The structuring of data exchanges within the network of connected intelligent artifacts allows users to become its agents. Therefore, it is crucial to understand the central role of human beings in the complex organizational context of this case study. **Consequently**, the development of interfaces must respect the premise that the cyber-physical space must be designed and programmed by the social system presented in a work organization model.

Moreover, the human-machine relationship should create a potential symbiosis between the physical space (machines,

Fig. 2: Super Admin User story map

tools) and the social space (identities, mental references, decision-making, communication language, and organizational roles) through the effective design of its interfaces, with a clear objective of increasing productivity. Good interface design should allow increasing possibilities for intuitive interaction between humans and machines, facilitating operators' working conditions and improving the user experience in complex environments [3].

To this extent, using User Story Mapping makes the workflow or value chain visible, highlighting the relationships between different priority-level tasks, helping to confirm the integrity of the backlog, and providing a helpful context for prioritizing features for implementation. "Mapping your story helps you find holes in your thinking" [11, p. 11].

Additionally, user story maps facilitate productive, human-centered discussions about product creation, where it is possible to establish priorities regarding its activities and tasks. They provide a holistic view of the product, revealing logical and releasable slices of product increments that align with users' needs. Importantly, they also uncover potential impacts and areas of risk, allowing for early learning and decision-making. This early learning is a valuable tool for teams, enabling them to understand what works and what does not. Armed with this knowledge, teams can make informed decisions about where to focus first, maximizing usability, value, and feasibility in subsequent iterations. Furthermore, user story mapping can be used both as a form of documentation and communication between teams [9].

C. Low-fidelity Prototype and Service Design Blueprint

The following exercise was to create a low-fi prototype in Figma (https://www.figma.com/)where we sketched a set of screens used in the initial and exploratory phases of the project and a tool that helps in requirements gathering [2]. We established the various stages of interaction between each user and the software, starting from a login screen and trying to understand the natural sequences of the availability of different functionalities. During this exercise, since the IoT Hub (where all the data collected by the different equipment and sensors will flow) was still in the study phase, it was understood that it was essential to focus on the Super-Administrator's perspective.

The Super-Administrator, as the key figure in our system, articulates all the platform's capabilities and makes them accessible on the interfaces of this privileged user. The idea that proved to be most interesting advocated that all other users' interfaces should correspond to the same graphical structure, sharing common interaction elements, which would adopt more simplified profiles depending on the limited access permissions of each user.

Throughout the process, given the questions raised during the meetings to present the low-fi prototype to the project leader, numerous technical issues were triggered, leading to revisions of the original wireframe.

We understood at this stage that it would be beneficial, within the scope of analyzing the flow of information throughout the process of generating tickets for building maintenance operations, to use a tool such as Service Design Blueprint (Figure 3), allowing a reading of the flow of procedures of each actor in the ticketing process, organizing them along a timeline and identifying the points of contact between them and potential points of friction or gaps in development and planning at this stage.

Given the intangible nature of services and their complexity, discussing them can be challenging. To this

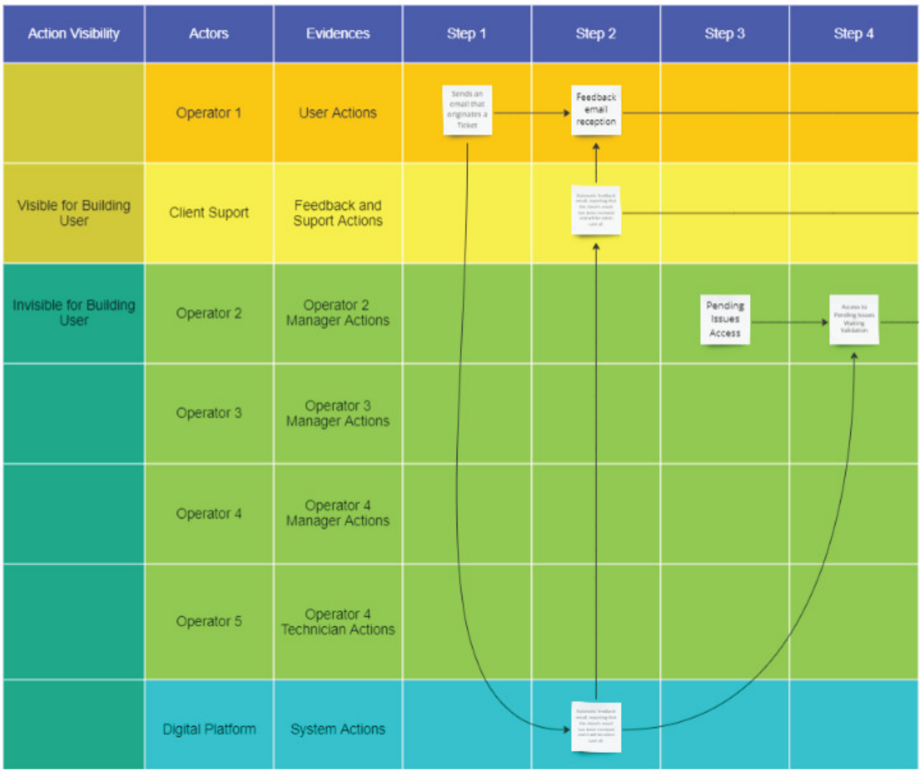

Fig. 3: Service design blueprint for the ticketing process

extent, blueprinting helps to create a visual representation of the service process that highlights the steps in the process, the touchpoints that occur and the physical evidence that exists from the point of view of the customer and the employees involved [5], [7]. Blueprinting helps an organization identify points of failure, improvement areas, innovation opportunities, and opportunities to increase profit. It allows for a standard visualization tool to identify and demonstrate how a service currently works or how to design a new service process. “Service blueprinting allows the capturing of dynamic processes in a visual manner. Relatively few methods allow for this type of dynamic, and at the same time visual, representation” [1, p. 92].

This tool demonstrated its usefulness by highlighting the need for structural changes to the interface design, introducing new users with their specificities regarding the software’s use. Identifying these changes in this initial phase of the project made it possible to prevent development and implementation work that would most likely prove redundant or even useless in later phases of the project.

These changes were introduced in the different iterations of the execution of this prototype, fueling the discussion and providing new insights into the expected functioning of the software, motivating a continuous search for solutions reflected in the tasks of back-end developers.

D. Mid-Fidelity Prototype

When we considered the structure defined in the low-fidelity prototype satisfactorily completed, articulating the different actors with their interfaces throughout the numerous procedures that the software would allow, including ticket management tasks and maintenance teams, creation of new user profiles, new assets, and identification of smart devices, we fine-tuned the interaction mechanics, namely: buttons, breadcrumbs, headers, footers, side menus, iconography, color schemes, font size, spacing grid and hierarchical articulation of the various graphic components.

The adjustments at this stage allow for a more in-depth assessment of what could become the final product from the end user’s perspective. Discussing this Figma prototype with the project leader, the questions that arise essentially concern preferences, personal tastes, suggestions for minor changes, identification of "nice-to-have" features, the refinement of aesthetic characteristics, and, as a rule, reduced impact on the final product’s structure.

The maturity level of the medium-fidelity prototype advances with each iteration, something that we were able to observe throughout the presentation meetings of the work carried out with the project leader, paving the way in a safe and sustained way for the implementation phase that will follow, translating this prototype into code, enabling us to better assess its integration with the Backend team’s work and the implementation of the necessary infrastructures for the final prototype’s proper functioning.

This project aims to create a prototype very close to a final product, making it sufficiently solid and configurable so that the necessary adaptations for the project leader’s placement on the market are aligned with the current state-of-the-art web applications that are easily accessible by any type of common digital device (tablet, smartphone, or laptop).

IV. Conclusion

The pervasive existence of CPSs in today’s everyday world is an unavoidable reality. Human beings’ dependence on complex systems that facilitate access to information from the digital and physical world and subsequent decision-making has grown in line with the certainties and security that rapid technological evolution has allowed.

It is essential to maintain sight of the design purpose of these kinds of systems, created by man with the tools he finds at his disposal, to use them as new instruments of decision, action, and reaction. Given its importance as the ultimate recipient of use, the human variable in cyber-physical systems demands to be placed first on the list of priorities. Applying human-centred design methodologies, using design thinking techniques and best practices, allows a deep understanding of the users and their motivations, preferences, constraints, and potential for involvement as an active element in the system.

The human variable must be meticulously considered whenever a cyber-physical system interfaces with humans. Given its natural tendency to increase entropy, the optimal design of interfaces becomes of utmost importance in any CPS project. This aspect cannot be overlooked, highlighting the urgency of its consideration in system development.

Acknowledgment

This article is a result of the Innovation Pact “R2UTechnologies | modular systems” (C644876810-00000019), by “R2UTechnologies” Consortium, co-financed by NextGeneration EU, through the Incentive System “Agendas para a Inovação Empresarial” (“Agendas for Business Innovation”), within the Recovery and Resilience Plan (PRR).

References

[1] Bitner, M., Ostrom, A., & Morgan, F. (2008) California Management Review Service Blueprinting: A Practical Technique for Service Innovation, p.92. https://doi.org/10.2307/41166446

[2] Dam, R. F. and Teo, Y. S. (2020, July 1). What Kind of Prototype Should You Create? Interaction Design Foundation - IxDF. https://www.interaction-design.org/literature/article/what-kind-of-prototype-should-you-create

[3] El-Haouzi, H. B., Valette, E., Krings, B. J., & Moniz, A. B. (2021). Social dimensions in cps & iot based automated production systems. Societies, 11(3). https://doi.org/10.3390/soc11030098

[4] Gürdür, D., & Törngren, M. (2018). Visual Analytics for Cyber-physical Systems Development: Blending Design Thinking and Systems Thinking. Presented at the 15th Annual NordDesign Conference (NordDesign 2018). Retrieved from https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-232358

[5] Interaction Design Foundation - IxDF. (2016, June 3). What are Service Blueprints?. Interaction Design Foundation - IxDF. https://www.interaction-design.org/literature/topics/service-blueprint

[6] Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020). Characterising the Digital Twin: A systematic literature review. CIRP Journal of Manufacturing Science and Technology, 29, 36–52. https://doi.org/10.1016/j.cirpj.2020.02.002

[7] Kalbach, J. (2016). Mapping experiences. O'Reilly Media.

[8] Lee, E. A. (2008, May). Cyber physical systems: Design challenges. In 2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC) (pp. 363-369). IEEE.

[9] Milicic, A., El Kadiri, S., Perdikakis, A., Ivanov, P., & Kiritsis, D. (2014). Toward the definition of domain concepts and knowledge through the application of the user story mapping method. International Journal of Product Lifecycle Management, 7(1), 3-16.

[10] Norman, D.A.: The Design of Everyday Things (Revised an). Basic Books (2013).

[11] Patton, J., & Economy, P. (2014). User story mapping: discover the whole story, build the right product. " O'Reilly Media, Inc."

[12] Romero, D., Hermosillo, G., Taherkordi, A., Nzekwa, R., Rouvoy, R., & Eliassen, F. (2013). The DigiHome service-oriented platform. Software - Practice and Experience, 43(10), 1205–1218. https://doi.org/10.1002/spe.1125

[13] Wang, F.-Y. (2010). The Emergence of Intelligent Enterprises: From CPS to CPSS. www.computer.org/intelligent.