Combining user centered design and system engineering to the design of a generic AI-based assistant

Kahina AMOKRANE-FERKA
IRT SystemX
91120 PALAISEAU, France
+33 (0)1 69 08 06 17
Kahina.amokrane-ferka@irt-systemx.fr

Virgil ROUSSEAUX
IRT SystemX
91120 Palaiseau, France
virgil.rousseaux@irt-systemx.fr

Matthieu DUSSARTRE
RTE
92800 La Défense, France
matthieu.dussartre@rte-france.com

Moustafa ZOUINAR
Orange
92320 Châtillon, France
Moustafa.zouinar@orange.fr

Nicolas RENOIR
SNCF
93418 Saint-Denis, France
Nicolas.renoir@sncf.fr

Abstract.

Human users' needs must be considered at the beginning of system design. However, classical systems engineering approaches consider the needs of several stakeholders (clients, authorities, etc.) but those of end users are often less considered. Hence, neglecting or oversighting such needs will lead to unacceptance and non-adoption of systems by end users. This paper introduces an original approach that combines System Engineering (SE) and User-Centered Design (UCD) approaches to address the needs of end users from the early stages and throughout the design process of a generic system. This approach is applied to the design of a bidirectional AI-based assistant, incorporating principles of human-machine teaming. It aims at assisting operators in real time network supervision and piloting activities.

Introduction

With the emergence of AI based systems, like virtual assistants, we are witnessing new forms of human-machine relationship that reach human-machine cooperation. However, in this line of thinking, AI is intended to perform part or the whole human task or mission. Nevertheless, in critical or complex sociotechnical systems, problems come when unexpected situations occur, and AI may not work any longer, because the system's performance is out of its validity context and the human may feel out of the loop. Thus, effective decision-making should keep humans at the center and the human must be an integral part of the system with a primary role in determining overall system performance and degree of mission success. To design systems with this perspective, end users' needs must be considered at the beginning of the design process and, in the case of the combination of use domains, the solution must be as generic as possible for time and deadline savings as well as future reuse.

Different approaches exist to design AI based systems, especially systems that could “collaborate” with the users. While each of these approaches offers specific advantages, none has yet presented a comprehensive methodology for addressing the overarching challenge of designing a generic system based on users’ needs. On the one hand UCD is an iterative design process in which designers focus on the users and their needs in each phase of the design process. In UCD, design teams involve users throughout the design process via a variety of research and design techniques, to create highly usable and accessible products for them [ISO13407]. On the other hand, system architecture is the fundamental and unifying system structure, defined in terms of system elements, interfaces, processes, constraints, and behaviors [INCOSE SAWG]. An architecture framework defines a common approach for the development, presentation, and integration of architecture descriptions. The application of a framework, based on a common set of accepted views, enables stakeholders to contribute more effectively to building interoperable systems and to manage the associated complexities [Handley 2019] [Krob 2017]. Until the last decade, on the one hand, humans were studied in ergonomics/human factors, a discipline that seeks to improve the usability of systems according to human capabilities and limitations. On the other hand, system design was studied in systems engineering [Boy 2014] [Millot 2012]. Then, a convergence of these two domains occurred through the field of Human System Integration (HSI). The main objective of HSI is to provide methods and tools that support the systems engineering community by ensuring that humans are considered in a logical and efficient way all along the system design process [Madni 2011] [Boy 2017]. Up to now, HSI has focused on methodology and tools [Watson 2017], but the questions related to the approach have still been little discussed.

In this paper, we introduce an original design approach that we developed as part of the CAB (Cockpit Assistant bidirectional)[1] project. This research project focuses on developing an AI-based assistant, from first design principles to its implementation. This assistant aims to support the operator’s decision-making process. One distinguishing aspect of the system is that it can learn from the human operator and vice versa, when necessary, thus achieving a two-way interaction, called “bidirectional interaction”. This operator can be a pilot of a business aircraft or a supervisor of various complex systems (telecommunication network or IT system, high voltage electrical network, railway network).

The major challenge of CAB project lies in the creation of a generic platform, offering a modular architecture that can be reused at low cost in various contexts, while responding in a specific way to the various end users’ needs and contexts.

At the heart of these different professional domains, the CAB project is confronted with the need to consider the specific constraints of each field of activity in a synergistic way. This synergistic approach has made it possible to transcend professional boundaries, highlighting on the one hand, the common essence of critical tasks such as diagnosis, anticipation, warning and, on the other hand, the need for effective support. In this way, the CAB project embodies a fruitful collaboration in which the diversity of professions is transformed into a shared strength, drawing on similarities to build an innovative and generic solution.

Combining user centered design to system architecture

The user-oriented design must anticipate future developments in the businesses for which the system is intended. Designers must be able to consider potential changes in professional practices, regulatory requirements and technological advances. The constant evolution of AI based systems raises profound questions about the very nature of the professions that use them. The impact of these

[1] https://www.irt-systemx.fr/projets/cab/

technologies on the world of work is palpable, redefining traditional roles and prompting critical reflection on the limits inherent in these emerging systems.

One major issue is the redefinition of the daily tasks of professionals working with these systems, particularly to capitalize on the respective strengths of humans and machines.

How AI may help operators (aircraft pilots, power grid supervisors, railway network supervisors or telecoms supervisors) in their daily tasks? How can we design AI based systems that leverage the respective strengths of human and machine? How can we design a system that covers different work contexts?

To address these issues, on the one hand, it is necessary to establish a generic system architecture to meet the requirements of versatility. On the other hand, a UCD approach must be followed, to satisfy the specific requirements of end users. To reconcile these two essential characteristics of the CAB assistant, we have introduced an approach consisting of merging the system architecture and UCD approaches within iterative cycles. This approach allowed users' needs to be encompassed before designing the CAB assistant. In the rest of this article, the five stages of this approach will be described.

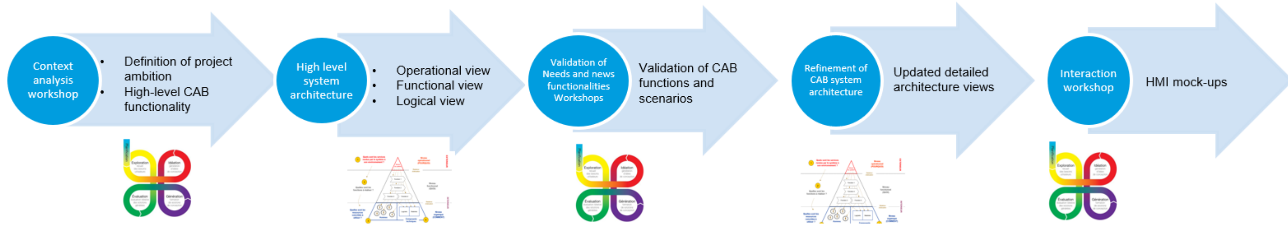

Figure 1. steps of our approach

Context of use analysis workshop

During the first stage of the design process, the goal was to understand the context of use relative to each use case (aircraft piloting, power grid supervision, railway network supervision and telecom/IT monitoring). Holding four separate workshops, each corresponding to a specific use case, was a crucial stage in our approach.

The main objective of these workshops which were attended by operators or experts having knowledge of the activities concerned was to determine the potentially relevant functionalities of the CAB assistant, while identifying the common needs shared by the various use cases. To this end, each industrial partner undertook an analysis of the current or the future contexts of use of the system to be designed, focusing on the tasks performed by the operators, their tools, the difficulties encountered in their activities, and the procedures they use. This analysis was based on observations, analysis of existing documents, in-depth interviews with operators or experts, and in-situ observations. The analysis of the activities and contexts of use provided a detailed vision of the operational scenarios, laying the foundations for the definition of the project's ambitions.

Working together, the participants to the workshops identified common needs that transcended the various professions involved, such as the visualization of certain data, contextual information, task sharing, the bidirectional concept, etc. This feedback enabled the operational requirements to be fine-tuned, ensuring that the specifics of each use case were considered. At the same time, the workshops allowed the identification of high-level CAB functionalities, providing a basis for the development of solutions that respond in a targeted way to specific operational needs.

High level system architecture

The results obtained at the end of the context of use analysis phase formed the basis for the development of the high-level system architecture, by following the CESAMES framework, [Krob 2017], which is a product of NASA and INCOSE systems engineering frameworks [Krob 2014].

This stage began with a harmonization process aiming at unifying the various use case's needs. The first challenge was to cope with the diversity of the specific terminology used by the project partners. For example, unexpected situations requiring operator intervention are named "constraint" by RTE, "incident" by SNCF, "failure" by Dassault Aviation and "incident" by Orange. After the harmonization process, these concepts were referred to the generic term "event". Similarly, the notion of 'solution' (to problems that operators should fix), is another example of vocabulary standardization as it replaces the word "parade" for RTE, "procedure" or "Unwinding" for Dassault Aviation, and "procedure" or "plan re-planning" for SNCF. Other specific concepts have also been identified and integrated into our conceptual framework. At the end of this stage, we drew up a specific taxonomy to the project, adopted in all the use cases. This taxonomy constitutes a common language, offering a shared semantic basis for describing work activities related to each use case.

The objective was then to build the three views of the CAB assistant architecture. This phase marked the transition from abstract concepts derived from the analysis of contexts of use to a systemic architecture, conveying a tangible and functional representation of the CAB assistant with its three views as described below.

Operational view. The assistant shares its environment with an operator, business tools and, in our case, a simulator to reproduce the real environment. This assistant will be in continuous interaction with an operator and provided with real-time data by the simulator.

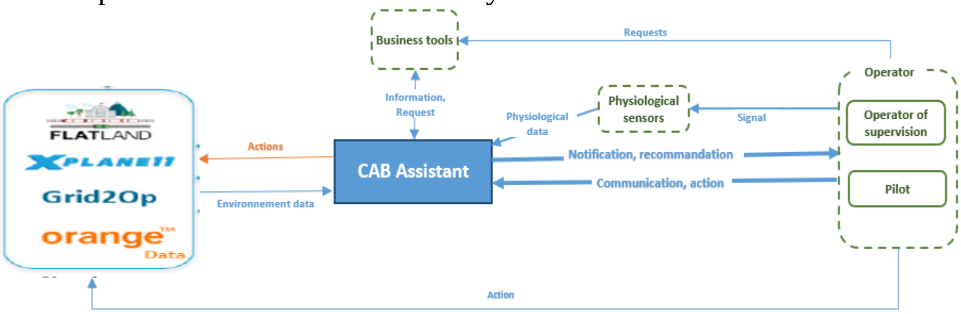

Figure 2. Environment diagram of the CAB assistant

Operational scenarios. The operational scenarios allow us to identify the sequence of existing interactions, organized in time, between the CAB assistant, the operator, and the simulator. The interaction describes the actions of the operator and those of the CAB assistant. It also describes the events provided by simulators. These scenarios allow to challenge both the CAB assistant functions, and the interaction between the CAB assistant and the operator. Twelve scenarios have been defined, such as event management, event anticipation, multi-events management, tasks organization. Concretely, these scenarios consist in describing, in a generic way, how and when CAB assistant can help an operator.

Up to now, we focused on the scenario "event management" only. First, this scenario is common to all use cases (manage an overload of a power grid for RTE, manage a railway disruption due to a traveler's discomfort for SNCF, manage an engine failure for Dassault Aviation, and diagnosing application failures in the information system for Orange). Second, it covers several identified issues such as when and how to communicate the event to the operator. Who initiates the interaction? How can CAB help the operator? Can it take over certain tasks?

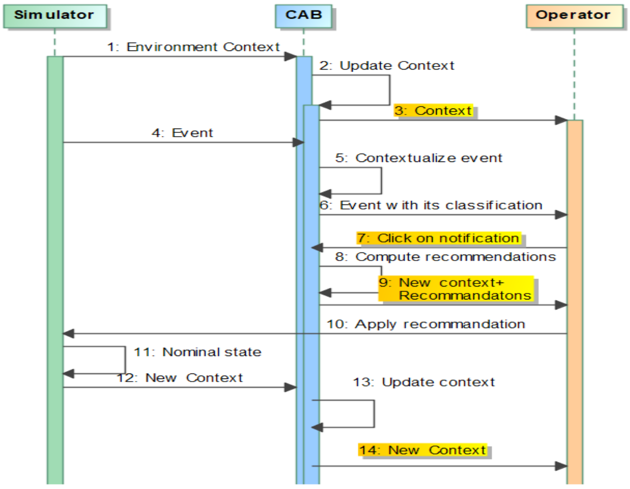

Figure 3. Example of operational scenario: managing an event

Functional view. After defining how the assistant may be useful in the context of the four industrial use cases, we now specify what an assistant should accomplish when interacting with an operator. To this end, we have based on the human AI guidelines, defined in [Amershi, 2019], to define and implement the functionalities of this assistant.

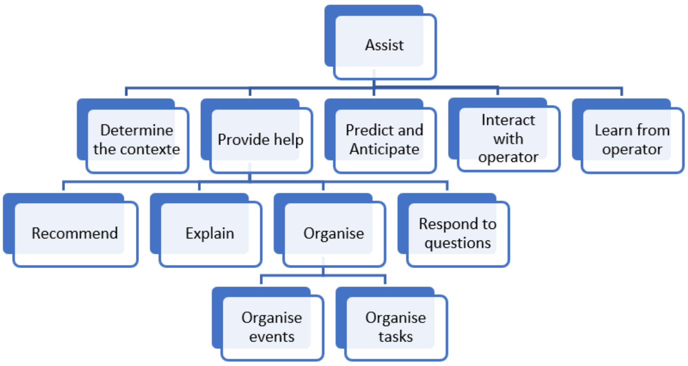

Figure 4: Functional decomposition of CAB

We identified four principals' functions. *Determine the context* is a function that discerns the external context and the operator behavior by collecting and analyzing all the data received from the simulator. *2) Interact with the operator* handles any exchange between the operator and the CAB assistant. *3)Provide help to the operator* defines how CAB answers the operator's needs (recommendations, explanations, event classification, etc.) after a solicitation from the operator.

Logical view. CAB is composed of ten modules. *Communication module* handles all communications with the operator, regardless of the interaction modality (for example, vocal, textual, etc.). *Context module* in charge of collecting and analyzing all data related to external or internal contexts. *Parameter module* is responsible for collecting all information related to the configuration and settings of a session/mission. *History module* saves all actions (operator's or assistant's), events, data, etc., related to one or more sessions. *Recommendation module* in charge of helping operators to manage a problematic event by giving several "solutions". This module is based on machine learning technologies. *Knowledge module* responds to operator's request based on ontology techniques. *Capitalization module* determines and saves new knowledge. *Learning module* allows the further training of the recommendation module algorithms. *Cognitive workload estimation module* that allow to estimate operator cognitive workload by considering physiological signals, basing in data fusion. *Diagnosis module* in charge to evaluate the gravity on the events.

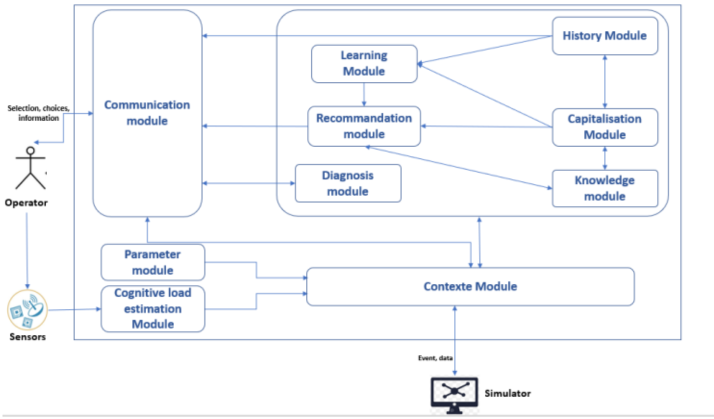

Figure 5: Logical architecture view

Validating Needs and Gathering New Functional Requirements Workshops

The validation of a system's functionality with end users is a crucial step in UCD. In this perspective, the operational scenarios illustrating CAB's functions were confronted to end-users. Specifically, the aim was to validate the relevance and realism of the CAB scenarios from an operational point of view, as well as the suitability of the CAB system for the activities of operators for each use case.

For this purpose, the generic operational scenario (managing an event) was instantiated on all use cases. For example, for RTE, the starting event is a potential overload on a power line. The CAB assistant offers its assistance to the operator by recommending solutions, based on following the subsequent steps presented in Figure 6. Subsequently, these scenarios were presented to operators during the workshop [Zouinar 2023]. Operators validated some of the concepts and corrected others, with particular emphasis on interrelated functions.

Anticipation management Scenario
The operator is at the beginning of the 8 o'clock shift.) He has a list of instructions to process
Overload anticipation on line 1.2, CAB assistant Notify the operator: An orange "overload anticipation" notification
Operator sees the notification
operator clicks on the notification
The CAB assistant updates the information in the HMI (context, timeline), CAB Assistant recommends three solutions (P1, P2, P3)
The operator consults the P2 solution
The operator applies the P2 solution

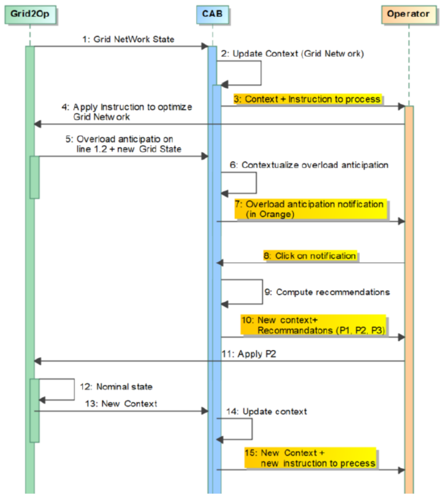

Figure 6; Example of operational scenario instantiation (RTE use case)

Refinement of the System Architecture

The identification and validation of common functionalities, expressed in a transversal way by all end-users, were crucial steps. By focusing on shared needs, the refinement process evolved towards the integration of generic functionalities that respond in a unified way to the requirements of the various use cases. At this stage, most of the functions presented to operators have been validated. Some functions have been refined. For example, the function *determine the context* has been recomposed into two sub-functions, *determine the temporal* context and *determine the geographical* context, the function *recommend* is recomposed into two sub-functions, *recommend a procedure*, *recommend a solution*.

Thus, the refinement of the system architecture was not limited to a simple technical iteration but was the fruit of a collaborative and reactive approach, shaped by constant exchanges between stakeholders and, above all, by careful consideration of end-users' feedbacks.

Interaction workshop

An initial work was conducted into pooling operators' needs in terms of interaction with the assistant. A wireframe mock-up consisting of four main zones was produced, suggesting an initial generic structure for the user interface, with the aim of initiating end users' creativity . In addition, interaction scenarios were created in service blueprint format to highlight interaction touchpoints, i.e. the moments when a user interacts with CAB assistant in the context of their activity. The wireframe mock-up and these scenarios were presented to end-users in four workshops (one for each use case) to validate the general structure of the user interface and populate it with content. The information and interaction requirements expressed by operators during these workshops were brought together and transcribed in a generic interface framework form that adapts to the context of each use case [Zouinar 2023]. In terms of interaction with AI, we have implemented the following principles: it is at the operator's initiative that AI proposes recommendations or information on the context, the operator has the choice of applying one of the recommendations or proposing a solution.

Refinement of the Functional Architecture: Interact with the Operator

Following the interaction workshop, the function dedicated to the interaction of the system with the operator was clarified and enriched with end-users' feedbacks. Common interaction sub-functions were identified and validated. These include consulting an event, updating the context and the timeline, responding to operator solicitation, displaying recommendations, allowing the choose of recommendation, allowing recommendations comparison, applying a recommendation.

Workshops for Validating HMI Mockups

The aim of this last series of workshops was to validate a first version of the CAB assistant prototype with end users. In preparation step, *medium fidelity prototypes* (HMI wireframes enhanced with interaction functionalities and content) were produced with Figma and instantiated for each use case. Each of the HMI zones was populated with data specific to each use case [Zouinar 2023].
These workshops had the same format as the interaction workshops, with the same participants and was held face-to-face and recorded.

Refinement of the logical Architecture: Operator-CAB Interface

After the previous workshops, validated concepts and functionalities were converted into graphical assets that could be implemented by the development team [Zouinar 2023].

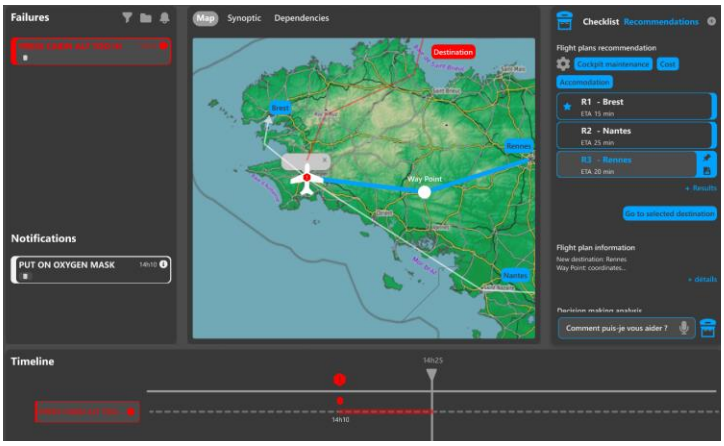

Figure 7. Example of CAB HMI

Conclusion and Future Directions

The iterative and collaborative approach adopted in the design process for the CAB assistant has paved the way for a methodology that helped us to address the challenging goal of designing a generic AI-based system, which can be adapted and potentially reused, even outside the initial project framework. The cycles of workshops with end users played a crucial role in the design process. They helped us to define and validate in a continuous way the users' needs, the system's functionality and the user interface. It also played an important role in refining the system's architecture. In other words, the iterative approach has enabled the CAB's functionality to be constantly adjusted in response to end-users' feedbacks, ensuring that it continues to meet operational needs. The next step of the project is to evaluate with end users the final prototype of the system which is currently under development The gains resulting from this approach are significant, both in terms of time and the harmonious integration of diffrent use cases. This approach and its adaptability are a major asset, not only for future developments of the CAB project, but also for other similar projects seeking to better integrate UCD and system engineering in their design process.

Nowadays, we have applied this approach on four different fields. As this method is independent on a particular domain of application, we have already begun to apply it to other domains, such as cyber security. This will allow us to evaluate its adaptivity and improve it if necessary.

Acknowledgments

This work has been supported by the French government under the "France 2030" program, as part of the SystemX Technological Research Institute within the CAB project[1]

References

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., ... & Horvitz, E. (2019, May). Guidelines for human-AI interaction. In *Proceedings of the 2019 chi conference on human factors in computing systems* (pp. 1-13).

Boy, G. A. (Ed.). (2017). *The handbook of human-machine interaction: a human-centered design approach*. CRC Press.

Boy, G. A., & Narkevicius, J. M. (2014). Unifying human centered design and systems engineering for human systems integration. In *Complex Systems Design & Management: Proceedings of the Fourth International Conference on Complex Systems Design & Management CSD&M 2013* (pp. 151-162). Cham: Springer International Publishing.

Handley, H. A. (2019). *The Human Viewpoint for System Architectures*. Springer International Publishing.

INCOSE SAWG. (1996). INCOSE Systems Architecture Working Group at INCOSE '96 in Boston, MA on July 8,

ISO13407. (1999). Human-Centred Design Processes for Interactive Systems, ISO/IEC 13407: 1999 (E).

Krob, D. (2014). Éléments de systémique. Architecture des systèmes. *Complexité-Simplexité. Collège de France*.

Krob, D. (2017). Cesam: Cesames systems architecting method-a pocket guide.

Madni, A. M. (2011). Integrating humans with and within complex systems. *CrossTalk*, *5*.

Millot, P., & Boy, G. A. (2012). Human-machine cooperation: a solution for life-critical Systems?. *Work*, *41*(Supplement 1), 4552-4559.

Watson, M. E., Rusnock, C. F., Colombi, J. M., & Miller, M. E. (2017). Human-centered design using system modeling language. *Journal of Cognitive Engineering and Decision Making*, *11*(3), 252-269..

Zouinar, M., Amokrane-Ferka, K., Rousseaux, V., Vallet, F. (2023). Conception d'un assistant bidirectionnel générique a base d'IA pour la supervision de réseaux et le pilotage d'aéronefs. In Proceedings of ERGO-AI conference,

Biography

Kahina Amokrane-Ferka.is a senior research engineer in Human Machine Interaction and is working as a system architect at IRT SystemX, France. Master's research degree from Evry Val d'Essonne University, France. She received a PhD degree in Information Technologies and Systems from UTC of Compiègne, then worked as an academic researcher for over four years. Her research interests include knowledge management, human factors, system engineering, Human System Integration, Human AI-collaboration, Human Autonomy Teaming

Virgil Rousseaux. is a junior research-engineer specialized in Human Machine Interaction. After a master's degree in mechanical engineering, industrial design and ergonomics at the Université de Technologie de Compiègne, he specialized in cognitive ergonomics and user centered design through an exchange program at McMaster University (Canada). In the last three years, he has been working at IRT SystemX where he has been focusing on studying complex human machine interactions, such as interactions with AI-systems.

Matthieu Dussartre. is researcher in RTE's R&D for 20 years, he has accumulated recognized expertise in the electrical system and decision support tools. Currently involved in collaborative projects on AI and augmented engineering initiatives, his skills cover, in electrical grid operation processes (designing innovative solutions, managing input data, producing relevant results, and creating intuitive interfaces for operators). With a strong track of successful projects, he has played a key role in advancing these topics at RTE.

Moustafa Zouinar. is researcher in the Human and Social Sciences Department (SENSE) at Orange and Associate Professor of Ergonomics at the CNAM. He is interested in Human-machine interaction, user experience, and the design and the analysis of the uses of digital technologies in a variety of contexts. He is currently working on the design, the usage and the social implications of AI systems in work settings.

Nicolas Renoir. is a cognitive engineering graduate from ENSC Bordeaux, began his career as an operator, learning to drive metros and trams, then later transitioned to automated environments. He has been working in railway R&D for seven years. Nicolas is dedicated to finding solutions that ensure optimal operator-machine collaboration, aiming to enhance the safety and performance of railway operations