

"THE HYPER EXPERT COLLABORATIVE AI ASSISTANT"

Deliverable 2.3

Low-fidelity simulation method workshop format

Version	1.0
Lead partner	imec-SMIT, Vrije Universiteit Brussel
Dissemination level	Public
Type of deliverable	Report
Contractual date	30/sept/2025
Actual delivery date	30/sept/2025

D2.3. Low-fidelity simulation method workshop format

Authors	**Organisations**
Ana POP STEFANIJA	imec-SMIT, VUB
Ine VAN ZEELAND	imec-SMIT, VUB
Rob HEYMAN	imec-SMIT, VUB

Revised by	
Princy ANDRIAMPANIRY	PDT
Benjamin VIQUERAT	PDT
Elise DURNERIN	CAT
Mehrdad ASADI	VUB

Disclaimer

This document reflects the authors' views only and neither the Agency nor the Commission are responsible for any use that may be made of the information contained

Table of Contents

1. Executive summary 5

2. Why we need simulation methods 7

2.1 Identifying shortcomings and limitations in AI systems development. 8

2.2 Mitigating shortcomings and limitations in AI systems development. 9

3. AI simulation methods for user involvement 13

3.1 Simulating AI applications for end-user testing 13

3.1.1 Scenarios and prototypes 14

3.1.2 Other aspects of early simulation testing 15

3.2 Levels of fidelity 15

3.2.1 Scoping a low-fidelity prototype 16

3.2.2 Validity of low-fidelity simulations 16

3.3 Types of low-fidelity simulations 17

3.3.1 Paper prototyping simulations 18

3.3.2 Wizard-of-Oz simulations 19

3.4 AI simulation protocols for user testing 23

3.4.1 Protocol for paper prototyping simulations 25

3.4.2 Protocol for Wizard-of-Oz simulations 28

4. Evaluating simulation methods 33

4.1 Why evaluate simulation environments? 33

4.2 Pre-testing and post-hoc evaluation 34

4.3 Categories of evaluation criteria 35

4.3.1 Evaluation criteria based on classical experiments 36

4.3.2 Evaluating participation 38

4.3.3 Optimising for what 41

4.4 Evaluation protocol 42

5. Discussion and recommendations 44

5.1 Ideal and less-than-ideal circumstances 44

5.2 Core principles for simulation testing 46

5.3 Evaluating and iteratively improving simulation testing 46

6. Conclusions 49

7. References 50

LIST OF TABLES

Table 1: Reporting guidelines for WoZ AI system simulations for end-user testing 23
Table 2: Preparatory steps (for both paper prototyping and Wizard of Oz protocols 24
Table 3: Quick guide for paper prototyping and Wizard-of-Oz protocols 33
Table 4: Core principles for simulation procedures 46

LIST OF FIGURES

Figure 1: Paper-prototyping simulation set-up 18
Figure 2: Wizard of Oz simulation set-up 21
Figure 3: The ladder of participation according to Arnstein (2019) 40
Figure 4: Simulation procedure and evaluation framework 46

1. Executive summary

This deliverable describes the work done in T2.3 "Low-fidelity simulation methods for AI systems". The aim of this task is to define, describe, and summarise the process of simulation development, testing, and evaluation. In doing so, it focuses first on the need for using simulation methods when developing AI systems. Then it provides a detailed overview of the types of simulation methods, based on a number of criteria, such as questions to be answered by the simulation and fidelity levels. The deliverable focuses on two types of approaches for simulating human-AI interaction: **paper prototyping** and **Wizard-of-Oz**. We also define evaluation protocols fit for this explorative, simulated test environment. Finally, we provide an extensive framework for simulation design and evaluation, which should serve as comprehensive guidance and a checklist for the entire simulation process and procedure (ideation, design, testing, and evaluation).

This deliverable provides:

- a discussion and elaboration of basic requirements for AI system simulation for user testing, detailing prerequisite steps for any kind of simulation testing, basic characteristics and core principles, as well as the reasoning behind user involvement in simulation testing.
- a comprehensive description of two approaches to AI system simulations for user testing, focusing specifically on low-fidelity simulation methods. We detail the characteristics of each of these approaches, elaborating on their uses, advantages, and shortcomings. We outline the different procedures and steps in the simulation procedure and setup, and requirements regarding the testing (scenarios, protocols, data collection, validity, and evaluation).
- a protocol and a framework for evaluating the simulation methods. We present a number of criteria that will allow us to assess the suitability of the methods, tools, procedures, participant selection and engagement, and optimization goals. Additionally, it will allow to evaluate the generalizability, representativeness, and validity of the simulation results. At a practical level, it should also serve as a ground to improve further iterations (of a same or higher fidelity).
- a comprehensive framework for simulation design and evaluation. Summarizing the evaluation protocol and the simulation protocol, we developed comprehensive and structured guidance for simulation procedures, from inception to final evaluation (preparation, testing, analysis, evaluation).

The descriptions, protocols, and frameworks developed in this deliverable will be useful for the activities undertaken throughout the PEER project, particularly the WP3 (D3.1, D3.2, D3.3) activities related to the design

and development of the MVPs, as well as for the testing activities in WP5 (D5.6 to D5.11, Reports on the evaluation processes). However, the value of this deliverable extends beyond the PEER project and its partners. The outputs presented can be used as standalone tools for anyone developing AI systems and various technologies in a participatory manner.

This deliverable is a result of an extensive literature review, the findings from our previous work within the PEER project, as well as our involvement in the meetings and activities leading to the D5.1 (Specification of Usage Scenarios and technical Validation Activities plan (1st version)[1] and D3.1 (MVP of the PEER collaborative AI assistant for lab evaluation)[2], D5.6 First Report on the Evaluation process[3], and the ongoing D3.2 (First iteration of the PEER collaborative AI assistant for use cases).

[1] Available at: https://zenodo.org/records/17201495

[2] Available at: https://zenodo.org/records/17206411

[3] At the moment of writing not yet available on Zenodo, but it should be available on the PEER project's Zenodo library

2. Why we need simulation methods

As we have already indicated in D2.1 Social and technical requirements[4] and D2.2 AI co-creation representations[5], one of the aims of the PEER project is to develop a technology-building process where the intended users are active participants in the design and development processes. To achieve this, we are incrementally building on various previous steps. In D2.1, we elaborated on the specific socio-technical requirements of the envisioned users for each use-case. This provided the foundation for the consequent technology building. In D2.2, we developed a co-creation method (AI cards[6]) that should translate parts of the AI system and create shared vocabulary between the envisioned users (non-experts) and the technology designers to further engage in the development of a particular AI system, and to advance the preferences and the needs of the users. The involvement of users in technology design and development is not a one-time event, but a process; this process should not be an afterthought of the development process, but an integral part of it. As such, it implies ongoing, continuous, and iterative, if needed, processes. One of those processes is the co-design and testing of an AI system through simulation methods (low to high fidelity). All these elements and processes (the AI cards, data sheets, and low-fidelity simulations) will allow end-users to consider and respond to AI system attributes before the system is (fully) developed.

The use of simulation methods, especially low-fidelity design representations of AI systems in the early stages of technology development, will allow designers and end-users to proactively explore the capabilities and limitations of AI in co-creation processes. However, as we already mentioned and will elaborate further below, this is a process and not a one-time event. It will require a few iterations, gradually increasing the level of fidelity (low to high) with each advancement in the technology design. Incrementally designing simulations, commencing from the simplest ones, such as **paper prototyping** or the **Wizard-of-Oz** approach, will enable the users to envision and, in a tacit manner, experience the potential AI system. When doing that at the early stages of development, users can provide critical input that will allow and enable to correct the system while the window of opportunities is not yet closed, meaning when the design phase is not finished. Once a design is completed, it is significantly more challenging, sometimes impossible, to introduce substantial (and infrastructural) changes in the AI system.

[4] This is a sensitive report, and it has not been published publicly

[5] Available at: https://zenodo.org/records/15130984

[6] Available at: https://zenodo.org/records/13861559

In that sense, to borrow the framing from Wang and Blok (2025), we need to approach co-design and co-participation as a process with three analytical stages: the input stage, the throughput, and the output stage. In the first, the input stage, we identify the values and goals that set up the AI design process (as we did for D2.1). The throughput stage, which encompasses the actual co-design process through the participation of various stakeholders (primarily intended end-users, but also technology developers and use-case owners), investigates how these values, needs, and requirements of the users, but also of the technology developers and the business (use-case) owners, are integrated into the technological design (D2.2 and this D2.3). The final output stage focuses on the outcomes of the AI system design process and includes an evaluation of the design and simulations (as was done in D5.6 First Report on the evaluation process).

The involvement of envisioned users in technology development should not be performative, however. It should also not use participation as a manipulation of legitimacy (Wang & Blok, 2025). If executed properly, it will bring several advantages and benefits both to those designing the technology and its intended users. We could summarize these as *identifying known shortcomings and limitations in AI systems development* and their *mitigation*.

2.1 Identifying shortcomings and limitations in AI systems development.

Early and iterative prototyping and simulations allow the capture and foregrounding of two important processes: the assumptions regarding the envisioned AI system, and the negotiations that take place as a result. When we talk about assumptions, we refer to the beliefs and expectations that various parties have regarding *what* the AI system should achieve, *how* it should achieve it, and what are the specifics of these processes. For example, what use-case owners want the system to achieve might not be what the intended users need from this system. Or how the intended users imagine the system to behave might not be what is technically possible at all. Similarly, what developers might plan to use as a technical tool might not be permitted by law. In that sense, designing simulations of the intended AI systems enables us to uncover these assumptions, which initiates the process of negotiating that follows. In technology development, all included parties (business owners, technology developers, and users) often need to adjust not just their expectations regarding the system, but also how that system will be technically implemented (*e.g.*, data, algorithms), the features it will offer (personalization, interactivity), the goals it will achieve (reduced risks and increased equity, accuracy, efficiency), and similar. Even more, sometimes the values of the different parties might clash, e.g., a focus on economic value and business interests might be misaligned with the social values of the end-users. These negotiations will take place in the design phase of the prototyping; they will be revisited again after simulation testing and will iteratively influence the consequent prototypes and simulations.

2.2 Mitigating shortcomings and limitations in AI systems development.

Including end-users and running simulation testing with them in the early and ongoing stages of AI systems development proves to be beneficial for the intended users themselves, for the technology developers, and for the system itself. It enables the mitigation of identified shortcomings and limitations, transforming them into benefits for all parties involved.

For the intended users, creating simulation prototypes and developing evaluation protocols enables them to play an active role in the design and development of the technology itself. More specifically, it provides them with opportunities to get familiar with the system, to have time and space to test the system, and to inspect the system. It allows for a feedback loop where the users test if the system (in its current form) responds to their needs and requirements, and they provide input accordingly. It also allows for their complex lived experiences and practices to be taken into consideration and tackled with(in) the proposed AI solutions.

For **technology developers**, the benefits it introduces can be summarized in the following way:

> Overcoming the user involvement dilemma. Oftentimes, technology development processes start with the assumption that the intended end-users lack the knowledge, qualifications, and expertise to understand the systems under development, and hence, they cannot understand what the AI system can do, and consequently, cannot modify or further adapt it. Thus, they may use it in entirely different ways than intended (Akrich, 1992). Therefore, although they are the ones who will use or be implicated in these systems, they are not consulted or engaged in the development process until it is too late in the process, and the technology design is closed. This can be mitigated by already planning co-creation sessions and input from end-users early in the ideation and design phase, where simplified versions and representations of the system will be translated into design material that end-users will understand. This requires more time and resources from the development team, but it will mitigate potential issues in the long term, especially technology acceptance and adoption.

> Overcoming well-identified limitations in technology design-thinking and development. These limitations originate in the difficulties of articulating in advance what the envisioned AI system can or cannot do, of not knowing how to express AI design ideas or to explain AI (intended) behaviours to users or to design open-ended interactions, and similar (Yang et al., 2020). Additionally, technology developers are often given a list of abstract values they need to integrate in the technology (e.g., privacy, transparency, explainability, *etc.*) (Wang & Beck, 2025) for which testing through simulations will enable a concrete translation into design elements, technological specifications, and AI models. This process consists of closely investigating how these abstract values are embodied in everyday technology use.

> Identifying and mitigating the points of friction between the system and the users, and between the visions of the designers and the needs of the users. Oftentimes, designers come up with their own ideas of what the actual problems of the intended users are that the systems should solve. However, this might be far from the lived experience and the daily practices of users (Akrich, 1992). This process consists of closely investigating how these abstract values are embodied in everyday technology use. These points of friction, if unidentified and left unaddressed, will impact the acceptance, adoption, and effectiveness of the final AI system.

> Uncover the negotiations taking place between the visions of the technology developers and the technical possibilities. Sometimes the ideas of technology designers are untranslatable in technology design, or there may be other constraints or limitations to enacting them in technology design. For example, there may not be appropriate models yet to achieve the system's goal; or the data for the models may not be possible to obtain (*e.g.*, due to regulatory limitations or requirements of the users); or the design possibilities may not correspond to the needs and requirements of the users, models, or business goals.

> The ability to intervene in and modify the technical design while the technology is still in the making. Early prototyping allows for testing of the technical affordances of the system and – if they are deemed inappropriate, insufficient, or incorrect — to intervene and modify them.

For the **AI system** (as a product):

> System built according to the needs and requirements of the actual users, not abstract representations. Often, technology designers have their own visions about the ideal or potential end-users of the system, including their characteristics, needs, and requirements. These visions can be both abstract and incorrect. When the development of the system is based on inputs from the actual intended users, and not on assumed and abstract representations of the system, we minimize the risk of fully developing a system based on faulty assumptions.

> Better acceptance and adoption. By engaging the intended end-users in the co-creation of the AI systems, we minimize the chances that the system will not respond to the requirements and the needs of the users, or that it will not take their expectations and lived experiences into consideration (Akrich, 1992).

> Lower financial costs. By adopting the approach of testing before investing, testing the AI system still in development through simulation methods will, in the long run, minimize financial costs (Grufberg & Holmquist, 2011). By allowing to test the system in low-fidelity environments (no algorithms needed, no data cleaning or structuring, or model development yet), assumptions can be corrected and decisions can be made that will show

what needs to be done in terms of data work and model-building, without very costly processes that ultimately might not lead to the intended results.

In summary, simulation methods, as part of the co-creation processes in the design and development phase of AI systems, help technology designers answer some of the crucial questions: do the chosen AI techniques make sense and help the users achieve their goals? Do the users trust this system as it is, or are interventions and modifications needed? Are users' needs and requirements accounted for, anticipated, and translated into technology design? If not, what additional requirements are needed? Do users feel empowered or constrained by the proposed AI solutions? Once these questions are answered, (technological) responses can be formed accordingly.

The typical life cycle of simulations and their evaluations can be summarized as a three-stage process (Salovaara et al., 2017). The initial stage is the stage of simulation initiation or *envisioning* what the particular technology (system) should be, what it should solve, improve, or similar, and how. This is a kind of a scenario for the future. The second stage is the *concretisation* of that future in the present in the form of simulations and prototypes. This is the step where concrete decisions are made, both regarding the system (which features of the system to simulate) and the particular settings of evaluating the simulations (participants, fidelity level, scenarios). To be able to gain valuable insights from this stage, it is a prerequisite to design the simulations in such a way that the workings of the system and the interactions with the respondents can be empirically observed and scrutinised (Salovaara et al., 2017). The final stage is *projection*, where the findings from the evaluation of the simulation are interpreted, and the evaluators are trying to make sense and "read" what the results in the present moment suggest about the envisioned (future) system. These three stages are closely related, and the validity of each stage is affected by the previous stage and dependent on the validity of the last stage in the consequent iterations.

However, we need to be careful not to overpromise or overestimate the role of simulations and prototyping in the success of the final AI product. As Salovaara et al. (2017) outline, when designing and developing a system, we are faced with a *present–future gap*, meaning the real use, context of use, and ways of using the system are partially "unknown future state of affairs" (2064). Consequently, we can never claim with utmost certainty that a particular prototype or a simulation will be adopted as intended or evaluated. However, not testing and evaluating simulations at different stages of system design and development risks designing and assessing "the wrong futures, contexts, or user groups, thereby leading to false conclusions and expensive design failures" (Salovaara et al., 2017: 2064). More so, involving the intended users enables a more user-centric approach to AI system development, and opportunities to detect a system's erratic behaviour at an early stage, prevent real-

world mishaps or even harm, and help overcome the blind spots of AI developers (Deng et al., 2023). While the future acceptance and adoption of an AI system can never be fully foreseen, and a simulation can never be fully evaluated, following a carefully designed simulation framework and evaluations should help close the present-future gap.

3. AI simulation methods for user involvement

A sub-objective of the PEER project is to develop design methods in which low-fidelity design representations make it possible for end-users to explore the capabilities and limitations of an AI system. This deliverable presents simulation methods to test human-AI collaborations, in particular methods based on the paper prototyping and on the Wizard-of-Oz approaches. These approaches involve producing low-fidelity prototypes to simulate AI systems for end-users before the systems are developed. To this end, scripts are created for how the AI system may behave, but all algorithmic responses are acted out by a human or a simple digital tool (*e.g.*, Figma), operating within the scripted boundaries. The goal of such testing is to ensure the development of behaviours of the final AI systems is aligned with end-users' needs and requirements.

User testing differs from heuristic evaluation, which focuses on the evaluation of a system's interface design (usually conducted by usability experts). User testing involves asking actual users to perform tasks by using a prototype of the system (Catani & Biers, 1998; Nielsen, 1990). When the aim is to align an AI system with end-user needs, simulating the performance of a system with representative end-users uncovers the most valuable information.

In designing the representations of the envisioned systems, all the parties involved in their design and development bring their own assumptions and values, and make a number of choices regarding the "future" (the goals) that these simulations need to create. These include, but are not limited to, the (choice of) recruitment of (specific) participants, the overall setting where the simulation study is conducted (*e.g.*, the physical, social, computing setting), the concrete scenarios and treatments to which the participants will respond, and the particular outcomes that will be tested and measured (Salovaara et al., 2017: 2065).

3.1 Simulating AI applications for end-user testing

Any user testing approach starts with the *why* and *what* questions that form the basis of its strategy (Bernsen et al., 1994; Camburn et al., 2017): Why are we testing the system, why at this stage of development, what are we aiming to achieve? As the previous chapter has made clear, user testing at various stages of the development process is crucial to identify and mitigate shortcomings and limitations in early stages of development, to lower development and adoption costs, and to introduce user-identified benefits for end-users. In the process of user testing, negotiations between different needs and limitations are uncovered and made available for explicit discussion.

From a creative development perspective, a major reason to simulate a system under development from the very beginning of the development process is to avoid “solution thinking”: defining a solution too early in the development process risks missing out on different technological approaches, narrowing down innovative thinking (Grufberg & Holmquist, 2011). Developers and testers are therefore advised not to let their assumptions influence (early) tests with users.

3.1.1 Scenarios and prototypes

End-users will rarely interact with developed systems in the ways developers expect and will attempt to de-script designs to serve their own purposes (Akrich, 1992). Testers must therefore prepare for different **scenarios** when drawing up scripts for simulated interactions (Bernsen et al., 1994). Scenarios are tasks that require the use of the AI system to be solved, but for which different paths to a solution are possible, leaving space for the idiosyncratic approaches of each user (Bernsen et al., 1994; Dahlbäck et al., 1993).

To prepare for different ways end-users may use a system and for different testing purposes, Nielsen (1990) suggests the following types of scenarios in user testing:

- **Design scenarios:** scenarios targeted at developing specifications through examples of user goals;
- **Experimental settings:** scenarios in which users are asked to perform tasks, to test usability;
- **Archetypal interactions:** descriptions of typical user behaviour;
- **Illustrative scenarios:** extreme examples of usability problems to test their severity;
- **Presentation scenarios:** visuals of interfaces or storyboards to gauge responses to the design;
- **Generic test suites:** descriptions of sets of tasks to test a class of applications of the system;
- **Documentation scenarios:** manuals, videos, or interactive tutorials showing the user how the system can be used for specific tasks (notably, created before development of the system).

When the focus of the user testing is on discovering explicit or implicit user requirements, all but the presentation scenarios are suitable. When the focus is merely on UX/user environment testing, presentations scenarios are most suitable, but documentation scenarios are advisable as well.

Presenting the scenarios happens in the shape of simulations with **prototypes** that make conceptual ideas concrete. Prototypes have been described as physical hypotheses in which form, function, interactivity, and other aspects embody "a theoretical proposition that can be subjected to test” (Wensveen & Matthews, 2014: 265). As physical manifestations of developers’ ideas and users’ requirements, prototypes stimulate reflection, not

only among users but also among developers (Lim et al., 2008). An additional objective of simulations may thus be exploration and active learning for developers (Camburn et al., 2017).

3.1.2 Other aspects of early simulation testing

Selecting the right participants for simulation testing is crucial. Even in the first experimental phase, participants should be selected to resemble the backgrounds and skills of the expected end-users, because characteristics such as level of education and domain-specific experience influence interactions with the AI system (Bernsen et al., 1994).

An element of end-user simulation testing that needs to be taken into account is the extent to which test users (simulation participants) can be exposed to harmful outputs of an unfinished system, such as offensive results (Wang et al., 2023). Apart from such safeguards as pre-testing and constraining outputs, adverse results in user tests can be mitigated by using low-fidelity prototypes that have a high level of human facilitator involvement (Wang et al., 2023).

3.2 Levels of fidelity

The resemblance of a prototype or simulation to the real-life conditions of use has been labelled a matter of **fidelity**. Generally speaking, the higher the fidelity of the prototype, the higher its cost (Lim et al., 2008). This cost should not only be understood as monetary cost, but also a cost for development and design; for instance, costs are high when end-users and developers are set on a narrowly specific development path because their design thinking has been influenced heavily by a too complete prototype.

This element of cost in creativity and analytic depth also translates back to monetary cost when part of the development process has to be retraced when it turns out the system does not conform to requirements in a later stage of development; “if everyone is involved from the beginning, possible issues can be highlighted and prevented before changes are more expensive” (Grufberg & Holmquist, 2011). Early prototyping is generally endorsed in both industry and research as critical for innovation (Camburn et al., 2017).

An important characteristic of an early-stage prototype is therefore incompleteness; the prototype is helpful exactly because of what is *not* yet included in it (Lim et al., 2008). Because of this element of incompleteness, early-stage prototypes can be used for discovering issues and requirements that are not evident in purely

theoretical and hypothetical descriptions of the system under development. What is not there is asking to be filled in. This underlines the benefits of keeping fidelity of the prototype low, especially in early stages of the development process (Lim et al., 2008; Sefelin et al., 2003).

3.2.1 Scoping a low-fidelity prototype

In practical terms, testers need to carefully consider what to include and what to exclude while designing a first prototype in an early development stage, to find the simplest form that leaves the most elements open for discussion, without affecting test users' understanding of the whole system (Camburn et al., 2017; Lim et al., 2008). At the other end of the fidelity scale lies pilot prototyping, which is a high-fidelity form of end-user testing that takes place when the system is almost finished and focuses the attention on fine-tuning the results of previous choices (Lim et al., 2008). In other words, levels of fidelity must be carefully adjusted to the questions that are open in the specific stage of development of the AI system, also known as **scoping** the prototype (Lim et al., 2008).

Aspects of a prototype to consider for inclusion and exclusion are elements of:

- appearance (*e.g.*, shapes, colours, textures),
- data (*e.g.*, types, visibility, privacy),
- functionality (think of the scenarios discussed in the previous section),
- interactivity (*e.g.*, feedback, input and output behaviour, operation),
- spatial structure (*e.g.*, how is the system connected to other systems).

Other dimensions can also be considered (Lim et al., 2008). It is important to note that while the simulation may be targeted at testing an aspect of functionality or data, there is no escaping the fact that the prototype will have an appearance and certain kinds of interactivity and structure, which testers will respond to, as well. These different dimensions are therefore inseparable and will all need to be carefully calibrated.

3.2.2 Validity of low-fidelity simulations

Apart from scoping, or finetuning what is left out of the prototype to evoke the right discussions among test users, questions of **validity** also play an important role in prototyping (Sauer & Sonderegger, 2009). Validity refers to the generalizability of the results of the test. In social science, this is often a matter of how representative and numerous the respondents in a study are, compared to the population for which the researcher wants to draw conclusions. There is some debate on the validity of low-fidelity prototyping, because

an approach that cuts costs too much and leaves too much out, can create uncertainty about real issues, endanger reliable analysis and raise false alarms or overlook problems (Cockton & Woolrych, 2002; Sauer & Sonderegger, 2009). As a consequence, there is a risk of drawing general conclusions that are actually a reflection of the limited simulation settings.

Therefore, cost-cutting should not be the main impulse for low-fidelity simulations. The validity of low-fidelity simulations can be improved if there are a sizable number and variety of end-user participants in testing the AI system (Cockton & Woolrych, 2002). Additional measures can be:

- improving ecological validity by conducting the simulation in a (near) realistic setting;
- conducting ample, interdisciplinary analysis of the results of simulations (Camburn et al., 2017; Cockton & Woolrych, 2002; Sauer & Sonderegger, 2009).

The next chapter (Chapter 4) will delve deeper into ways to evaluate and iteratively improve the validity of simulation methods.

3.3 Types of low-fidelity simulations

In the previous sections, the terms prototype and simulation have been used interchangeably, because authors differ in the extent of their interpretation of the term prototype, which sometimes includes the complete physical and digital testing environment (Lim et al., 2008) or sometimes refers more narrowly to an object of design (Wensveen & Matthews, 2014). From here onwards, we use the term simulation for the wider set-up of the test (for instance, is this experimental research, how realistic is the environment?), while the term prototype is reserved for the mock-up of the system in a slightly broader sense than merely a designed object, as it also includes paper and oral descriptions of systems.

Various kinds of low-fidelity simulation approaches are used in product and system design, such as functional decomposition, storyboarding, or method acting (Grufberg & Holmquist, 2011). A more recent type of low-fidelity prototyping is *digital wireframes*, which can also provide minimal interactivity. These prototypes focus on the representation of the system and a certain (not all of) its functionalities. They can be used for various goals, such as to illustrate a user experience, simulate (parts) of the system processes ("user journey"), possible interactions, and test different scenarios. *Figma*[7] is often used for this, and this was the case for the simulations

[7] *Figma* is a cloud-based collaborative web application for interface design and prototyping. It is used to produce both low-fidelity and high-fidelity prototypes. Available at https://www.figma.com/

of first MVPs (Minimum Viable Product) for some of the use cases within the PEER project (City of Amsterdam, Proditec, and Datacation). A more detailed overview of the particular digital wireframes and minimum interactive wireframes can be found in **D5.9** First report on the evaluation process[8] and **D5.3** Report for Integration and Technical Validation (1st Version)[9].

This deliverable focuses on two types of rapid prototyping that are commonly used in Human-Computer Interaction (HCI) design and research: paper prototyping simulations and Wizard-of-Oz simulations. Both types allow for feedback early in the development process. The latter in particular are recommended in the development of high-risk, cognitively demanding systems because they elicit much more complete information than other rapid prototyping techniques (Bernsen et al., 1994).

3.3.1 Paper prototyping simulations

The material for prototyping can be as simple as pen and paper (see Figure 1), the low fidelity of which does not affect the quality of users' experiences in simulations (Catani & Biers, 1998; Sauer & Sonderegger, 2009; Sefelin et al., 2003). If the aim of a simulation is to focus on the functionality of the system to be developed, a choice for the simulation can be to leave appearance untouched for the time being (Grufberg & Holmquist, 2011).

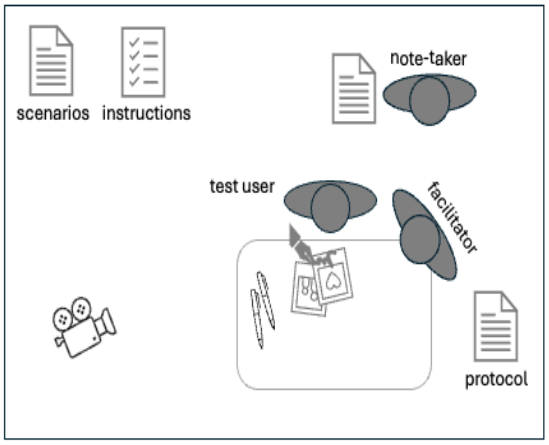


Figure 1: Paper-prototyping simulation set-up

Another reason to choose paper prototyping simulations can be that alternative tools do not support presenting the system elements that need to be tested, for example because too much about the system is still unclear to

[8] This report is defined as Sensitive and it is not publicly available. A public version of the same report will be later available in the Zenodo repository of the PEER project as D5.6 First report of the evaluation process.

[9] Available at https://zenodo.org/records/17201679 with restricted access.

be represented with higher fidelity (Sefelin et al., 2003). In other instances, software skills for computer-based simulations may be lacking among testers (Sefelin et al., 2003). A third reason to choose paper prototyping can be that a larger number of drawings (visual feedback) from end-users can support the system development (Sefelin et al., 2003). Moreover, a low-fidelity approach such as paper prototyping also means there is human control to prevent potentially harmful outputs for end-users (Wang et al., 2023).

Choosing a paper prototyping approach is first and foremost a result of the scoping of the simulation: when a tester decides to choose a paper simulation, there are specific reasons for this choice based on the open questions that need to be discussed about the system under development. In other words, choosing a paper appearance is the result of careful consideration. Nevertheless, the choice for paper prototyping is also often a consequence of time and budgetary pressures (Sauer & Sonderegger, 2009).

Paper prototypes can be of very low fidelity, in that every step in the test is drawn on the spot by a facilitator (Sefelin et al., 2003) or consist of AI cards, such as those presented in D2.2. Further elements need to be considered as well, such as the colours and elaborateness of the presentation of the system, interactivity and functionality (Lim et al., 2008).

The data resulting from paper prototype simulations are largely qualitative, also requiring qualitative approaches to data analysis. The data may consist of notes from facilitators or note-takers, drawings from facilitators or participants, video or audio recordings of participant behaviour and interaction patterns, surveys taken from participants, or other kinds of feedback data. These different types of data need to be coherently analysed, ideally by an interdisciplinary team of analysts.

3.3.2 Wizard-of-Oz simulations

The Wizard-of-Oz (WoZ) technique is a commonly used computer-based method in Human-Computer Interaction (HCI) studies to test user responses to “intelligent” automated systems (see Figure 2). The WoZ technique implies telling the users they are interacting with an automated system while in fact they are interacting with another person (Kelley, 1984; Riek, 2012). Compared to paper prototyping, WoZ simulations have the advantages of allowing for geographically dispersed participants; improving the understanding of the system’s functionality for users; and relatively easy upgrading to higher levels of fidelity (Camburn et al., 2017; Wang et al., 2023).

Preconditions for a WoZ simulation are that:

- it must be possible to simulate the automated system realistically, given human limitations;
- it must be possible to specify some of the system's future behaviour; and
- it must be possible to make the simulation convincing (Fraser & Gilbert, 1991).

Another aspect that must be taken into account is that WoZ simulations are rather costly when done well, which is why the costs must be weighed against the risks of not conducting this type of simulation test and having to re-build a prototype if it fails to meet requirements (Bernsen et al., 1994).

The WoZ technique may to some extent involve deception, as there is a human (the "wizard") interacting with the user, pretending to be the system (Kelley, 1984; Nasir et al., 2022). Ethical concerns are raised over deceiving participants in a WoZ simulation, whereas in some scientific communities, the right to know with whom a person is interacting was introduced (Fraser & Gilbert, 1991; Riek, 2012). This is why in some WoZ simulation studies, participants are told about the wizard beforehand (Kelley, 2018; Nasir et al., 2022), while others advocate not informing participants of the wizard to maintain a realistic simulation, but to remain vague, so as not to lie either (Bernsen et al., 1994).

Whether the human wizard is outed may not affect the interaction, and trust in wizards does not so much depend on whether they are revealed as humans, but more on the reliability of their advice (Nasir et al., 2022). For ethical reasons, the general recommendation for WoZ simulations is to explain the details of the set-up to participants at least afterwards (Kelley, 2018; Riek, 2012)

If the simulation does indeed include the deception that the AI system (not a person) is responding to the user, several aspects of the prototype must be designed in line with that deception. Computers and people differ in several important aspects (Fraser & Gilbert, 1991): "People are flexible, computers are rigid (or consistent); people are slow at typewriting, computer output is fast; computers never make small mistakes (*e.g.*, occasional spelling errors), people make them all the time" (Dahlbäck et al., 1993: 194).

Therefore, people pretending to be computer systems cannot take time to type out a reply to a user and cannot produce typos or other language mistakes. A wizard will have to use canned responses and templates, and ways to quickly retrieve information as needed in the interaction with the user (Bernsen et al., 1994; Dahlbäck et al., 1993). Multiple wizards may even be needed and/or assistants, with clear task allocations, to recreate the desired speed and accuracy of responses (Bernsen et al., 1994; Green et al., 2004).

Wizards (and assistants) must also be trained to respond as much as possible in the same way as the system under development would, and to control for influencing the simulation participants' responses (Bernsen et al.,

1994; Fraser & Gilbert, 1991; Riek, 2012). For example, a wizard should be careful not to respond in ways that surpass the system's capabilities, but reduced skills are difficult to simulate in real time (Bernsen et al., 1994). Wizard training will entail developing skills in handling the wizard interface of the prototype, practicing with pretend users, and testing the knowledge base for the system (Bernsen et al., 1994). The wizard training may already surface issues with the prototype (e.g., in the wizard or user interfaces) or wider simulation environment that require adjustments.

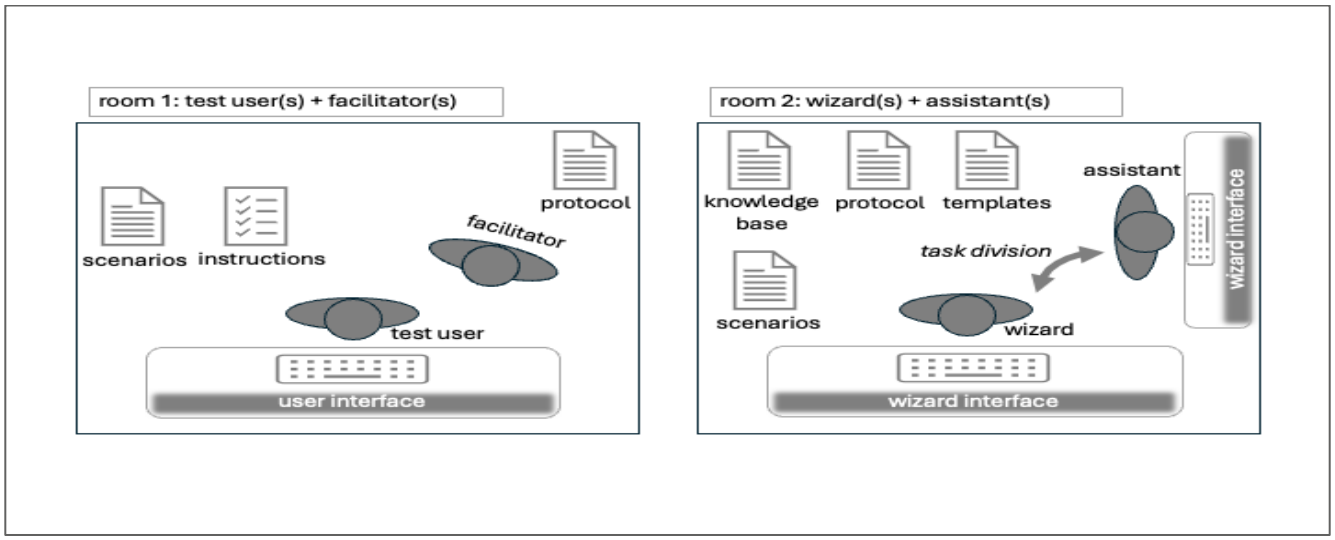


Figure 2: Wizard of Oz simulation set-up

Another element of preparation is that scenarios and instructions for simulation participants will have to be drawn up and tested for consistency, accuracy, and realism. As prototypes lack the data needed for a fully functional AI system, testers will have to be resourceful in obtaining sufficient data for the simulation, for instance, by profiling test users from their social media accounts to produce personalized responses (Browne, 2019).

Some studies report flaws in WoZ simulation approaches, as they may not sufficiently surface real-world issues of AI systems, especially when ecological validity (*i.e.* the realism of the simulation environment) is low (Wang et al., 2023; Yang et al., 2020). Consequently, creating a convincing WoZ simulation environment for multiple scenarios, training the wizard(s) and pre-empting a variety of user interactions can be costly and time-consuming (Bernsen et al., 1994; Dahlbäck et al., 1993; Fraser & Gilbert, 1991).

Not only the AI system under development, but also the simulation must be iteratively improved to overcome the paradoxical situation of attempting to realistically simulate a system that is being developed by simulating it (Kelley, 1984; Fraser & Gilbert, 1991; Riek, 2012). Iterative prototyping is especially recommended when the simulation strategy is aimed at refinement of a system and meeting challenging requirements (Camburn et al.,

2017). Each iteration of a simulation must be analysed, so the results of the analyses can be used to improve subsequent iterations of the simulated system (Bernsen et al., 1994). The WoZ technique therefore consists of three stages:

1. a **pre-experimental phase** in which wizards prepare and train, scenarios are developed, and a simulated knowledge base is created;
2. a **first experimental phase** in which there are few constraints on the wizard, to gather feedback on the simulation set-up;
3. subsequent **rounds of experimentation** with iterative improvements of the simulation and the envisioned system (Fraser & Gilbert, 1991).

The latter rounds of experimentation, when the simulation moves toward higher fidelity, can involve a "Wizard-*with*-Oz" set-up, using real technology in a simulated environment; or a "Wizard-*and*-Oz" set-up, using real technology in a real environment; or even an "Oz-of-Wizard" set-up, in which users are simulated but the system is real (Riek, 2012). Indeed, in recent AI development paradigms, iterative improvement of simulations (as well as systems) can continue well after market introduction (Wang et al., 2023).

Over several iterations, the time to produce a new iteration decreases, especially when the process involves evolutionary prototyping (Camburn et al., 2017). It is impossible to determine in advance how many iterations of the simulation will be sufficient, but an indication of saturation can be when no more substantial problems or new needs arise in simulations (Bernsen et al., 1994).

Similar to paper prototyping simulations, the data resulting from a WoZ simulation are largely qualitative, although there may be quantitative measurements as well (e.g. response times of test users and of the system itself). The analysis of the data resulting from a simulation should at least include:

- structural information (*e.g.*, users' problem-solving strategies),
- users' problems in interacting with the system, and
- developmental patterns over iterations of the simulations (Bernsen et al., 1994).

Riek (2012) suggests reporting guidelines for WoZ simulations to improve consistency and allow for academic publishing and knowledge sharing. Table 1 provides an overview of those guidelines adapted to simulations of AI systems.

Experimental component	Questions:
AI system	How many AI systems were used? What kind(s) of AI system(s)? What level(s) of autonomy? What are the AI system's capabilities? What hypotheses did the tester/researcher have?
User	How many users participated in total, and per simulation round? What were the user demographics, sampling procedure, etc.? What instructions were provided to the user? What behavioural hypothesis did the tester have about the user? Was the simulation convincing to the user? What expectations did the user have about the AI system, before and after simulation?
Wizard	How many wizards (and assistants) were used? What were the wizard demographics? Did the wizard know the behavioural hypotheses of the simulation? What were the wizard production variables (*e.g.*, response speed) and how were they controlled for? What were the wizard recognition variables (*e.g.*, is the wizard perfect?) and how were they controlled for? How did the experimenter control for wizard error (deliberate and accidental)? How much and what sort of training did wizards receive prior to starting the experiment?
General	Where did the simulation take place? What were the environmental constants and how were they controlled? What scenarios did the testers/researchers employ? Was this simulation part of an iterative design process? Does the report discuss the limitations of WoZ?

Table 1: Reporting guidelines for WoZ AI system simulations for end-user testing

3.4 AI simulation protocols for user testing

Summarizing the discussions of the previous sections, the protocols in this section contain specific recommendations for paper prototyping simulations and Wizard-of-Oz simulations of AI systems (see Table 2

below). Three steps precede the protocols, in the sense that the simulation approach depends on these basic considerations:

Step 1: Determine the purposes of end-user testing (*Why?*)

Answer such questions as: Why are we testing the system, why at this stage of development, what are we aiming to achieve? This is a matter of scoping the simulation; which questions specifically need to be answered by the simulation test?

Step 2: Describe the end-users envisioned for the AI system (*Who?*)

Be very specific when it comes to whom the system is for; this may be different groups, and they may have different purposes for the use of the AI system. Much depends on involving the right representatives of envisioned end-users in the simulations.

Step 3: Determine the necessary level of fidelity (*What?*)

Once test purposes and end-users have been clearly determined, the best level of fidelity for the prototype and simulation can be chosen. At that point, scoping of prototype and simulation is necessary: what needs to be included and excluded to achieve the test purposes, to which extent can the AI system be (realistically) simulated at the current stage of development, and what is already known about future specifications?

Based on the results from the three steps above and taking into account social science approaches to study protocol creation, the following protocols can be used for end-user simulations of AI systems under development.

Step	Key questions	Essential actions
1. Define purpose	Why test? Why at this stage? What goals?	List specific questions to answer • Scope simulation boundaries
2. Identify users	Who will use the system? Which different purposes do they have?	Define target user groups • Plan representative recruitment
3. Set fidelity	What to in- or exclude? Current capabilities? Which specs are known?	Choose appropriate fidelity level • Assess realistic simulation scope

Table 2: Preparatory steps (for both paper prototyping and Wizard of Oz protocols

3.4.1 Protocol for paper prototyping simulations

To reiterate, paper prototyping involves a very low level of fidelity in which the AI system and the interaction with it are described on paper, with a facilitator guiding the users through the test scenarios (see Table 3 for a summarized overview).

Preparation

1. Prepare scenarios

   Considering that end-users will have their own ideas about how the AI system can be useful for their purposes, different scenarios will support more realistic testing of the system. While the scope of the test (step 1 above) suggests particular scenarios (see section 3.1), it is wise to prepare for several other relevant scenarios as well. This can be especially useful if the envisioned end-users consist of different types of profiles (e.g., technical and less technical, different skills levels and experience, different cultural backgrounds, and similar).

2. Invite test users

   It may not always be easy to find sufficient numbers and diversity of participants for a user test. As soon as the envisioned end-users are clear (step 2 in Table 2), start inviting representative participants for the simulation. Invitations should include a brief description of the simulation and study setup. Include a consent request form for participants to consider beforehand.

3. Prepare materials for the scenarios

   Design and print the mock-ups or descriptions of the AI system (e.g., using AI cards, such as in D2.2[10]) and draw up the instructions for the users. Also prepare at least: introduction and wrap-up speeches; a detailed protocol for the specific simulation for the facilitator(s), note-takers, analysts and evaluators to consult; ample pens and paper for notes; a hand-out with contact details for participants to take home; name tags for facilitator(s) and participants; video or audio recording equipment; and a clock or other timekeeping device.

4. Prepare the facilitator(s)

   Since paper prototyping simulations intentionally and purposefully have low fidelity, facilitators have to be very careful not to suggest or imply system specifications that are not part of the prototype. Training and test simulations can help prevent influences from facilitators. Facilitators can have multiple roles

[10] The Human-AI collaborator cards, the template, and the facilitator's guide are available at https://zenodo.org/records/13861559

they may need to prepare for as well; for instance, they may need to take notes of interactions or user strategies, or they may need to draw consecutive stages in the interaction with the system. At the very least, facilitators should study the detailed protocol for the specific simulation.

5. Prepare the analysis

The framework for the analysis should be designed beforehand, especially the questions that the test seeks answers to (see step 1 before the protocol). This may involve specifying sensitizing concepts, i.e. concepts that analysists will have to pay special attention to when examining the results of the test. Testers and facilitators should also be prepared to take extensive notes on the details of the actual implementation of the simulation for later reporting.

Simulation procedure

1. Introduce the simulation

Timing for tests/experiments is always short, so introductions must be brief. In no more than a few minutes, the facilitator(s) introduce(s) her/him/themself and the other testers involved, the purposes of the simulation, and what will be done with the results. Brief the users on their tasks and clarify further instructions, such as the limitations of the interactions. Do not forget to request participants' consent for the processing of the simulation data.

2. Run the simulation

Facilitators guide users through the scenarios selected to achieve the purposes set for user testing, in accordance with the prepared detailed protocol for the specific simulation. Facilitators (or designated note-takers) take notes of user responses or script deviations that are not otherwise recorded.

3. Collect data

Paper prototyping simulations are particularly useful for collecting participant drawings or written feedback. Other types of data to be collected can be: notes from facilitators/note-takers, comments written on paper prototypes, video/audio recordings of participant behaviour or interaction patterns, or other kinds of feedback data. In some cases, researchers have also chosen to conduct small surveys after the test, to diminish potentially intrusive data collection during the test.

4. Wrap up the test

Whether or not a (short) survey is conducted among participants after the test, briefly and informally evaluating the simulation set-up (including the scenarios and the mock-up) with the participants can

provide more data on the realism of the simulation and feedback on the fidelity level, to verify ecological validity. Wrapping up also includes thanking the participants and providing contact details or other information for any further questions after the test.

5. Pre-analyse data

   Collect all data types, register them in an overview document that specifies data types (e.g., Excel sheet), and make sure they are stored in a secure location. Make careful notes of the procedure followed, from preparation stage to wrap-up of the test, for later reporting and to inform any subsequent iterations of the simulation. Also evaluate the representativeness of the actual participants, shortly after the test. Briefly go through all data, notes and evaluations to create and make notes of first impressions to prepare the analysis.

Analysis

1. Integrate the data

   While participant surveys can yield some quantitative data, most paper prototyping simulation data will be qualitative. Recordings may have to be transcribed; notes will have to be cleaned, structured and clarified. While it would be possible to do a thematic analysis purely on transcriptions or written texts, this would be a reductive approach to analysing participant responses if the other data types are not simultaneously and coherently included in the analysis. Going through the data chronologically, following the steps of simulation, does integrate the data in a meaningful way for the analysis.

2. Analyse the data in coherence

   Data in the shape of facilitator comments will not make much sense if not analysed in coherence with the data collected on the participants' actions at the same point in the simulation procedure. The same is true for most other data types (written feedback without the prototype prompt, for instance). It is therefore not advisable to analyse data types separately, but rather in coherence with simultaneously collected data.

3. Invite interdisciplinary analysis

   To improve the comprehensiveness and validity of the analysis, interdisciplinary analysis can support a wider range of perspectives on participant behaviour as well as on relevance of certain interactions or comments. Relevant disciplines to involve in the analysis, apart from system and algorithm developers, include UX designers, experts from the field of the envisioned system's application, and specialists in interpretative analysis (e.g., social scientists).

4. Evaluate the simulation

   Several of the collected data types specifically concern the simulation settings, including the notes from facilitators/note-takers on the procedure and the representativeness of the participants, participant responses in the wrap-up phase of the simulation, and possibly remarks made during the simulation. These data are used to evaluate the validity and reliability of the simulation (more on this in the next chapter), both for reporting purposes and to improve any subsequent iterations of the simulation.

5. Report on the simulation and the analysis

   Reporting on the validity and reliability of a simulation is important for internal and external purposes, such as keeping track of simulation improvements and test user responses, improving further iterations, and replicability of results. In addition, Table 1 provides an overview of elements of the simulation to report on for Wizard-of-Oz simulations, many of which are also relevant to paper-prototyping simulations, particularly those in the categories “User” and “General”.

3.4.2 Protocol for Wizard-of-Oz simulations

WoZ simulations can vary in level of fidelity, from only a slight step up from paper prototyping to very realistic, near-complete systems. Since this deliverable focuses on early-stage low-fidelity simulations, this protocol is most suitable for such WoZ approaches, but it can be easily adapted to higher-fidelity varieties (see Table 3 for a summarized overview).

Preparation

1. Prepare scenarios

   Different scenarios will support more realistic testing of the system. While the scope of the test suggests prioritizing certain scenarios (see section 3.1), it is wise to prepare for several other relevant scenarios as well, since users may use the prototype in idiosyncratic ways or for individual purposes. WoZ simulations are particularly useful for design and experimental scenarios, or to test either common functionalities or extreme examples of usability problems. Test the scenarios for consistency, accuracy, and realism.

2. Invite test users

   As soon as the envisioned end-users are clear, start inviting representative participants for the simulation. Invitations should include a brief description of the simulation and study set-up, with careful

consideration of the need for deception. Include a consent request form for participants to consider beforehand.

3. Prepare the prototype and the scenarios

   Prepare the instructions for the test users and a detailed protocol on the scenarios for the wizard(s) and any assistant(s). Design a realistic test-user interface for the computer-based simulation of the AI system. Prepare response texts and templates, and a knowledge base to retrieve information for the wizard(s) (and assistants) (see section 3.3.2). Also prepare introduction and wrap-up speeches for the facilitator; response registration; a hand-out with contact details for participants to take home; and name tags for participants.

4. Train wizards and assistants

   If there are multiple wizards and/or assistants, create clear task allocations. Train the wizard(s) (and assistants) to respond in the same way as the system under development would, and to control for influencing participants' responses. Other elements of wizard training entail developing skills in handling the wizard interface of the prototype, practicing with pretend users, and testing the prototype's knowledge base.

5. Prepare the analysis

   The framework for the analysis should be designed beforehand, focusing on the questions that the test seeks to answer. This may involve specifying sensitizing concepts, i.e. concepts that analysists will have to pay special attention to when examining the simulation data. As WoZ simulations are generally computer-based, many response types can be recorded in an automated manner, including response times of wizards and test users. The facilitator and wizard(s) should also be prepared to take notes on the details of the actual procedure of the simulation for later reporting (see Table 1).

Simulation procedure

1. Introduce the simulation

   Timing for tests/experiments is always short, so introductions must be brief. In no more than a few minutes, the facilitator introduces her/himself, the set-up and the purposes of the simulation, and what will be done with the results. Brief the test users on their tasks and clarify further instructions, such as the limitations of the interactions. Do not forget to request participants' consent for the processing of simulation data.

2. Run the simulation

The facilitator guides test users through the scenarios selected, in accordance with the prepared detailed protocol for the specific simulation. The wizard(s) and/or assistants respond as much as possible in the same way as the system under development would, using the prepared knowledge base, following the scripted scenarios, and staying within scripted limitations. The facilitator may take notes of script deviations that are not otherwise recorded.

3. Collect data

Most of the data resulting from a WoZ simulation will be qualitative, but there will also be quantitative data, such as response times of test users, wizards or the prototype – if it is already at any level of autonomy. Important data are user feedback on the usefulness of the prototype, structural information (e.g. users' problem-solving strategies) and users' problems in interacting with the prototype. Other types of data can be notes from the facilitator and possibly video/audio recordings. In some cases, researchers have also chosen to conduct small surveys after the simulation.

4. Wrap up the simulation

Whether or not a (short) survey is conducted after the simulation, briefly and informally evaluating the set-up (including the scenarios and the prototype and its interface) with the participants can provide more data on the realism of the simulation and feedback on the fidelity level, to verify ecological validity. If deception was involved, the wizard(s) can be outed during the wrap-up. Wrapping up also includes thanking the participants and providing contact details or other information for any further questions after the simulation.

5. Pre-analyse data

Collect all data types, register them in an overview document that specifies data types (e.g., Excel sheet), and make sure they are stored in a secure location. Make careful notes of the procedure followed, from preparation stage to wrap-up of the simulation, for later reporting and to inform any subsequent iterations of the simulation. Also evaluate the representativeness of the actual participants shortly after the test. Briefly go through all data, notes and evaluations to create and make notes of first impressions to prepare the analysis.

Analysis

1. Integrate the data

Most data will be qualitative (e.g., user interactions) but already recorded in a digital manner. Recordings may have to be transcribed; any facilitator notes will have to be cleaned, structured and clarified. While it would be possible to do a thematic analysis purely on transcriptions or written texts, this would be a reductive approach to analysing participant responses if the other data types are not simultaneously and coherently included in the analysis. Going through the data chronologically, following the steps of simulation, does integrate the data in a meaningful way for the analysis.

2. Analyse the data in coherence

Data in the shape of response times will not make much sense if not analysed in coherence with the data collected on the participants' actions at the same point in the simulation procedure. The same is true for most other data types. It is therefore not advisable to analyse data types separately, but rather in coherence with simultaneously collected data.

3. Invite interdisciplinary analysis

To improve the comprehensiveness and validity of the analysis, interdisciplinary analysis can support a wider range of perspectives on participant behaviour as well as on the relevance of certain interactions or comments. Relevant disciplines to involve in the analysis, apart from system and algorithm developers, include UX designers, experts from the field of the envisioned system's application, and specialists in interpretative analysis (e.g., social scientists).

4. Evaluate the simulation

Several of the collected data types specifically concern the simulation settings, including wizard actions, the notes from the facilitator on the procedure, testers' evaluation of the representativeness of the participants, participant responses in the wrap-up phase of the simulation, and possibly remarks made during the simulation. These data are used to evaluate the validity and reliability of the simulation (more on this in the next chapter), both for reporting purposes and to improve any subsequent iterations of the simulation.

5. Report on the simulation and the analysis

Reporting on the validity and reliability of a simulation is important for internal and external purposes, such as keeping track of simulation improvements and testing user responses, improving further iterations, and replicability of results. In addition, Table 1 provides an overview of elements of the simulation to report on for Wizard-of-Oz simulations.

Phase	Paper prototyping tasks	WoZ prototyping tasks	Required results
Preparation	Create and test scenarios • Create paper mock-ups • Train facilitator(s) • Recruit and invite test users • Document every step and prepare analysis	Create scenarios • Build computer interface • Create knowledge base • Train wizard(s) • Test scenarios and setup • Document every step and prepare analysis	Detailed scenarios and protocols • Simulation setup • Mock-ups / Interface • Trained team • Recruited participants • Note-taking / recording necessities
Simulation	Brief introduction • Guide users through scenarios • Take detailed notes • Gather drawings / feedback • Wrap up and evaluate with participants • Document all steps and prepare analysis	Brief introduction • Guide users through scenarios • Wizard responds realistically • Record interactions • Wrap up and evaluate with participants • Document all steps and prepare analysis	Recorded user interactions and transcripts • Response data • Behavioural observations • Technical logs • User evaluations • Potentially: user survey results • Documentation of proceedings
Analysis	Transcribe / organise notes • Integrate all data types and analyse coherently • Include interdisciplinary team • Evaluate simulation validity • Report results of simulation and evaluation	Process digital recordings • Integrate all data types and analyse coherently • Include interdisciplinary team • Evaluate wizard performance and simulation validity • Report results of simulation and evaluation	Integrated data collection • Findings report • Iteration recommendations • Validity and reliability assessment • Evaluation report (see Table 1)

Table 3: Quick guide for paper prototyping and Wizard-of-Oz protocols

4. Evaluating simulation methods

As highlighted in the previous chapter, evaluating simulation iterations is important for various reasons: to assess the extent to which the simulation characteristics have affected the outcomes, to assess generalizability and representativeness, and at a very practical level, to improve further iterations. It is also important to evaluate the complete methodology (beyond single iterations), first and foremost to assess whether it was indeed suitable for the questions that needed to be answered by the simulation. Another reason to evaluate the simulation procedure is to improve the procedure itself, based on insights from practice.

Several approaches to methodological evaluation are discussed in this chapter, which ends with practical recommendations for an evaluation protocol.

4.1 Why evaluate simulation environments?

Simulating and testing a system, as we elaborated so far, are crucial for its design and development, and for making sure that the final system design aligns with what is desirable (for the technology owners, and for the intended users), what is possible (technologically, legally), and what is required (for the intended users, legally). Since these processes of simulating and testing should occur iteratively, it is essential that we also evaluate the simulation environments we are using.

Although they are only a representation of the entire system, simulations, even low-fidelity ones, are still complex in nature. This complexity arises from the fact that when designing the simulation, we still need to take into consideration a number of factors and elements, such as user requirements, technical possibilities, goals of the system, and similar (see also 3.4.1 and 3.4.2). Hastily prepared simulation environments, or simulations that do not account for this complexity, risk producing poorly executed testing or incorrect results and assumptions, which can impact the subsequent system design and development. Therefore, it is a prerequisite that we also evaluate if the particular methods and tools work as intended, or if other methods/tools/features/respondents should be used and engaged.

This evaluation should take place in two phases (pre-testing and post-testing), and a number of elements should be evaluated according to several criteria (*e.g.*, reliability, validity, participation, optimization goals).

4.2 Pre-testing and post-hoc evaluation

End-user simulations can be evaluated at different stages of the procedure. For example, a very basic evaluation can consist of assessing the **face validity** of the procedure on paper, before the implementation: Does the procedure, at face value, appear to reflect the goals of the simulation, and to be robust and coherent? This can also be done by asking several outsiders to have a look at the procedure beforehand, for their quick evaluation.

For more in-depth evaluation, the protocols in section 3.4 presented pointers for evaluations to be carried out beforehand (**pre-test**) and afterwards (**post-hoc**). Important pre-tests are:

- Testing the scenarios for consistency, accuracy, and realism;
- For WoZ simulations: tests of the Wizard interface and knowledge base;
- A trial of the full simulation procedure with pretend users (*e.g.*, colleagues).

Most of the evaluation, however, will take place post-hoc, after the simulation. An important part of post-hoc evaluation is assessing the **validity and reliability** of the simulation test. Validity refers to the absence of systemic errors, such as peculiar results caused by structural errors in the procedure. In other words, the results can be considered valid if the simulation procedure was carried out without systemic errors. Specific aspects of validity, such as ecological validity (how realistic is the simulation set-up) and internal validity (which inferences can we draw from the findings), are discussed in section 4.3.1.

Validity also refers to the generalisability of the results: Can the findings in the simulation with these particular participants and in this particular set-up, be generalised to the larger population of envisioned end-users and use-cases? Section 4.3.2 will discuss the evaluation of the participants and the generalisability aspect of what is called **external validity** in more detail.

While validity refers to the absence of systemic errors, **reliability** refers to the absence of accidental or occasional errors. Three factors must be taken into account to assess reliability (Bryman, 2016):

- Stability: is the measurement stable over time?
- Internal reliability: is the scale or index of measurement consistent?
- Inter-rater reliability: are different analysts conducting the analysis consistently?

For AI system simulations with end-users, assessing internal reliability and inter-rater reliability may be very similar: do analysts agree on what they are observing in participant behaviour? (They may disagree in their interpretations of why participants behave as they do, but if they have observed different behaviour, there is a

problem with internal reliability). Nevertheless, internal reliability can also refer to consistent use of questionnaires or response time measurements.

Another form of post-hoc evaluation is to involve the participants in assessing the findings of the procedure. This technique is called **respondent validation,** and it can be helpful in providing nuance to conclusions or in confirming understandings of the analysts (Bryman, 2016). Respondent validation can be conducted at intermediary stages (e.g., after transcription of oral feedback, or after the initial analysis) or for the full report. The aim is to verify that the findings and impressions are consistent with the views of the participants. This technique may also show areas where the perspectives do not match up, which may provoke changes to the reported outcomes or compromises in the findings.

The simulation and the testing should be designed in a way that will enable the evaluation of all types of validity and reliability. However, if in certain instances this is not possible (e.g., respondent validation might not be feasible due to limited access to the respondents post-testing), certain trade-offs must be made. If that is the case, then this needs to be accounted for, and how trade-offs may impact the findings and their validity must be evaluated and taken into consideration for the interpretation of the findings.

Chapter 3 also emphasised the importance of **replicability** of the simulation, providing pointers for documenting preparations, and reporting about the procedure post-hoc. Clear and complete reporting allows for further evaluation and re-evaluation of a simulation (iteration) at later stages of the system development process.

4.3 Categories of evaluation criteria

In this section, we discuss a number of evaluation criteria, categorised in three main groups: criteria based on classical experiments, criteria related to end-user participation, and evaluation based on the optimisation goals. These criteria focus on the suitability of the chosen simulation method and the tools, the participant selection and the nature of their engagement, the simulation procedures and set-up, and the data and insights collected as a result. Assessing the simulation methods based on these criteria also enables an assessment of whether the findings of a simulation test are valid and therefore usable for subsequent iterations or the final design of a system.

4.3.1 Evaluation criteria based on classical experiments

Paper prototyping simulations and WoZ simulations often have characteristics of experiments, or they are presented as such. Classical experiments have a design that can be used as a standard to evaluate simulation tests which only have some experimental characteristics without being true experiments. This quality of classical experimental design is due to its robustness and the trustworthiness of causal findings (Bryman, 2016). True experiments have very strong **internal validity**, *i.e.* the relationship between the experimental stimulus (the prototype) and the dependent variables (the responses of the participants) can be established with a high level of confidence.

Classical experimental design: Randomised Controlled Trials (RCT)

Unfortunately, experimental design is very rigid and difficult to apply to social research settings such as AI system prototype simulations. In classical experiments, a situation is manipulated to establish the impact of that manipulation on the participants. Everything else in the situation stays the same. In other words, for a true experiment, every other factor than the stimulus is controlled – this is why such experiments are also called 'controlled trials'.

In controlled trials, no other factors than the stimulus influence the outcomes, or in the case of the simulations described in this deliverable: nothing but the prototype should influence the participant responses. Other factors that need to be excluded as rival causes for participant responses can vary:

- history (e.g., high-profile scandals related to the type of simulated system),
- external factors (e.g., participants may have slept badly, there is outside noise during the experiment),
- instrumentation (e.g., the formulation of a questionnaire may lead to biases),
- participants drop-out (e.g., participants may drop out over several iterations of simulations),
- participant maturation (people change; participants may change over several simulation iterations),
- participant selection (e.g., differences in experience with AI systems or with experimental settings),
- ambiguity over the direction of causal influence (*e.g.*, participant interaction also influences prototype outputs).

Classical experiments are often conducted in laboratory settings, to control all factors of the situation. However, laboratory settings are rarely the natural, everyday settings in which end-users deploy a system. Therefore, the laboratory setting itself can be a factor of influence on the findings of an experiment. This is a matter of ecological validity (see also section 3.2.2): "*This refers in particular to the physical and social aspects of the test environment (*e.g., *lab set-up, presence of observers)*" (Sauer & Sonderegger, 2009: 676). In addition, real user concerns are "subjective feelings experienced in everyday life", which may reflect larger concerns about the use

of AI in the workplace or its role in society (Wang & Blok, 2025: 5). After all, the introduction of an AI system in a workplace is an aspect of a larger transformation in society towards more use of AI systems that cannot be replicated in a small-scale, controlled laboratory setting.

Consequently, field experiments, *i.e.* tests in the actual environment where the system will be used, are considered to be more realistic and therefore have higher ecological validity (Bryman, 2016). They also allow assessment of the wider impacts on, or transformation of, the situation an AI system will be used in, including, for instance, shifts in the division of labour or in the balance of power (Wang & Blok, 2025). On the other hand, field experiments are more difficult to control, which may affect the internal validity (Bryman, 2016).

Another element of internal validity is that the selection of participants needs to be carried out with extreme rigour, and the participants need to be divided over experimental groups and a control group, each of which is confronted with a specific level of the stimulus (the prototype). This approach makes it possible to establish with a high level of certainty which differences in responses can be caused by which differences in the prototypes.

The division of participants in classical experiments is carried out via randomisation (hence the term 'Randomised Controlled Trial', RCT). Randomisation reduces participant selection-related systematic errors in the experimental set-up, improving the validity of the findings. In other words, randomisation diminishes biases that could result from the participant selection. As participants are randomly assigned to either an experimental group or a control group, a pre-test can establish whether the groups are sufficiently similar. For example, all groups can fill out a questionnaire to establish whether their responses to the hypothetical AI system are more or less the same.

Similarity of the groups on specific characteristics (e.g., duration of work experience, level of education) can be improved through matching. In precision matching, every participant in the experimental group has a 'twin' (regarding the relevant characteristics) in the control group, whereas in frequency matching, the relevant characteristics are equally distributed over both types of groups (*e.g.*, equal distribution of education levels and gender, but not all 'twins').

After the pre-test to establish experimental groups and control group are sufficiently similar, the experimental groups receive the stimulus. In AI system simulation tests, this would mean the experimental groups go through the simulation protocol, but the control group does not. After the stimulus, there is a post-test to establish how the stimulus has impacted the experimental groups. Since the control group will not have received the stimulus,

the differences in responses between the control group and the experimental groups can be attributed to the simulation.

Learning from RCT

It may be clear from the above explanation of the classical experimental design that several aspects of such a rigid design are not realistically applicable to simulation tests that aim to collect end-user input on purposes and functionalities in early stages of an AI system design. Hence, the classical experimental design is only useful for particular simulation test goals; for example, to confirm whether the system under development meets pre-established purposes or functionalities, whether it works as expected, or whether it violates users' sensibilities.

Furthermore, as was mentioned above, pure experiments are difficult to conduct because they require a strict approach to control for so-called **confounding variables**, or alternative influences on participants' responses.

Nevertheless, much can be learned from a comparison with true classical experiments:

- Most notably, the types of factors (confounding variables) listed above, which can cause erratic results in experiments, can be used in an evaluation of the findings of a simulation test. Have history, external factors, instrumentation, and so on, affected the results?
- The tension between ecological validity (realistic settings) and internal validity (controlled settings) can be explored, and choices in settings evaluated, depending on the importance allocated to different types of validity.
- The selection of participants and their relevant characteristics can be evaluated against the ideal of classical experiments.

The next section will delve deeper into the evaluation of the selection of simulation participants and their contributions to the AI system development process.

4.3.2 Evaluating participation

Who is invited to test a simulation is a very important factor in the validity of the simulation testing. However, there are a few steps before that decision is made. In an ideal scenario, the direct end-users, or those who will be impacted by the envisioned AI system, should be included in the testing (see section 3.1.2). This is not always an evident group of potential users/affected parties. For example, sometimes unexpected users emerge—delivery pickers and not just hypermarket individual customers. People who temporarily need to use a mobility-assistance device are emerging users, but not initially envisioned end-users. Sometimes the system has

unintended harmful consequences for a group of people not directly intended as users (*e.g.*, the implementation and adoption of an AI system may lead to the termination of certain job positions).

The selection of participants should closely follow the intended and affected users. While in the first stages of prototyping (especially if low-fidelity simulations are used), it may not seem crucial to include a sample of the users, efforts should be made to include them even in this early stage (see section 3.1.2). If this is not the case or not possible, it introduces the risk of incorrect assumptions in later stages, which will affect the subsequent development of the system. Technology designers or business owners should avoid skipping the inclusion of intended users, because, as we have pointed out earlier, this may lead to solution thinking. While technology designers often think that they, as experts, have complete knowledge and expertise of how a system behaves or should behave, this often clashes with the needs, and requirements of, and actual use by the end (or simulation) users.

The use of proxy users is possible; however, this should be an option that is considered only as a last resort (see also "Oz-of-Wizard" set-ups, section 3.3.2). If too far from the "real" intended users, these proxy users do not share the same needs, lived experiences, and exact contexts of use as those for whom the system is intended. For example, employees of the company developing the AI system, who are in no way the intended users, should not be involved in the simulation testing – this is certain to introduce biases and thus endanger internal validity, meaning the simulation findings cannot be considered valid.

If the envisioned AI system is to be used in a work setting and for work-related tasks, job-specific experience and insights will be lacking for proxy users, and shortcuts will constrain the proxies from correctly assessing the simulation. In other words, if test users are in higher or different positions than the ones who will use the systems, and they do not understand the type of work that is being tested, the results may not be valid, especially if the system, and not the interface, is being tested. In such situations, one needs to keep in mind that sometimes the workers' concerns involve subjective feelings, some of which may include the fear of losing their jobs, losing control, or "worry about being replaced by machines" (Wang & Blok, 2025: 4).

In that sense, technology developers (either directly involved in the development process or their colleagues) also should not be "testers" themselves, unless they are the intended users of the envisioned AI system. This will avoid the **fallacy of expertise**, where their expertise in systems is mistaken for their expertise in all things in general.

Attention should be paid to the potential slippery slope of turning participation into *de facto* nonparticipation, as elaborated by Arnstein (2019; 1969) (see Figure 3). If no real changes to the systems under development are

introduced after the engagement of the end-users in the simulations, participation may take the form of informing, consultation, or advice, but no real follow-through or embedding of the outcomes of the participation is taking place. If technology designers are in no position or do not plan to incorporate the needs and requirements of end-users after their participation, they should clearly acknowledge that beforehand.

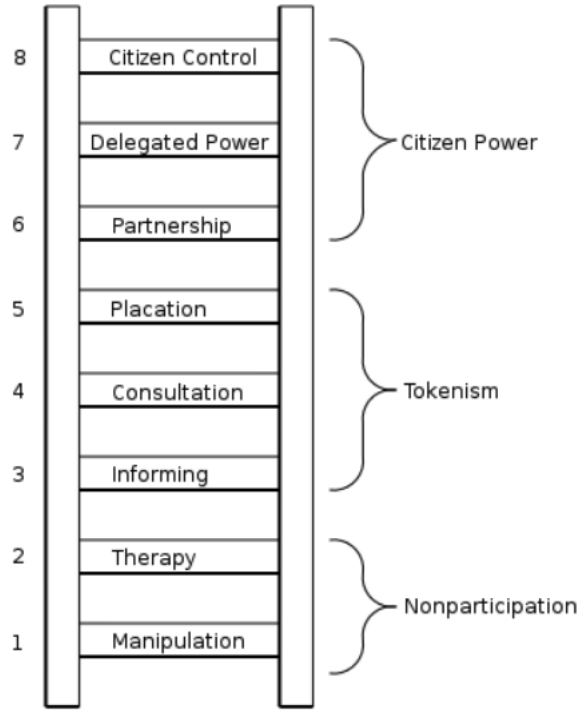


Figure 3: The ladder of participation according to Arnstein (2019)

How to evaluate participant selection

Participation should not be used as a manipulation of legitimacy (Wang & Block, 2025). This means that participation should not be seen merely as another box to be ticked. It should not be performative, and the simulation should, first, be built in a way that will enable true participation, and second, include protocols that lead to true consideration of the feedback and evaluation of the participants, and their inclusion in the envisioned system, otherwise it risks "offering symbolic rather than substantive involvement" (Gilman, 2022 in Wang & Blok, 2025: 5). Naturally, this does not imply that every need, requirement, or feedback of each individual will be taken into consideration. As we elaborated above, sometimes the state-of-the-art does not allow it, or there are legal and regulatory constraints, or trade-offs will have to be made. However, these negotiations should not, as Arnstein (2019) warns, become a tool for tokenism or participatory non-participation.

To evaluate the selection of participants, answer the following questions:

1. Who are the envisioned end-users of the system? (See step 2 in section 3.4.)
2. Have you selected the correct users?
3. Who have you invited to test the simulation?

4. Are these participants the intended end-users?
5. Have you reached a satisfactory number of participants to represent the number of intended end-users? (e.g., there will be a difference between the general public as envisioned end-users, compared with more specific end-users, like experts or certain professionals.)
6. Do you need a representative sample? If so, have you achieved that, and if not, why not? Do you plan to mitigate a lack of representativeness in subsequent iterations? If you do not need a representative sample, why not?
7. Have you used proxy users? If so, how are they similar to the intended end-users, and how do they differ?
8. How will you incorporate the feedback into the system or into the next simulation iterations?

4.3.3 Optimising for what

As mentioned at the start of this report, the goal of simulation testing is to ensure that the development of behaviours of the final AI system is aligned with end-users' needs and requirements. Notably, in addition to needs that end-users verbally express, there are also ethical, legal, and social requirements (Wang & Blok, 2025).

When designing and testing simulations, it is very important to keep in mind what we are optimizing for. As already discussed in Chapter 2, oftentimes what is desirable for the system developers and/or owners may not be desirable for intended end-users. Their ideas, needs, and values may clash, and this is something that needs to be recognized and accounted for. When evaluating a simulation, we need to keep in mind that the choices made by the simulation designers should be assessed from the perspective of “the presentation of the future” system as presented in the simulation (Salovaara et al., 2017: 2067). This presentation will affect the ability to draw valid conclusions and insights from the simulation. From this, several elements that need to be evaluated emerge:

1. What is being tested? Are we testing near-future prototypes where the focus is on iterations of the existing system, or are we testing far-future ones that involve speculation, unfamiliar features, and potentially novel technology (Salovaara et al., 2017)? This will impact the ability of the simulation participants to envision the system.

2. In that sense, which scenarios are we testing? The simulations and their evaluation protocols will differ depending on the questions that the procedure needs to answer. While the user interface or the simulation

representation is important and always part of the prototype/simulations, it plays a secondary role to testing capabilities and limitations of the envisioned system.

3. Does the understanding of the developers and owners regarding what should be tested overlap with what the end-users think? If not, how do these differ?

4. Do developers optimise for the same ends (*e.g.*, speed, efficiency, interactivity) as what the intended end-users would optimise for (*e.g.*, personalisation, serendipity, explainability)? If not, how will contrasts be resolved? What kinds of trade-offs will be made?

Additionally, the envisioned AI technologies and systems are increasingly subject to both general regulations (such as the AI Act, the GDPR, and the Data Governance Act) and industry-specific regulations. Having a good overview of these regulations even before designing and testing the simulations is crucial, since it will minimize the risk of testing something that is not compliant with laws or industry standards, or that will have to be removed from the system's features later in the process.

5. Are there elements and/or criteria that go against legal requirements (*e.g.*, excessive data collection in conflict with the GDPR's data minimisation principle or profiling employees as prohibited under the AI Act)?

6. Are there elements of the system that are in conflict with ethical or social/cultural expectations (*e.g.* impacting the dignity or autonomy of end-users or other affected parties)?

4.4 Evaluation protocol

The protocol we present below summarizes the elements and criteria we outlined in the chapters above. Its aim is to help guide the evaluation of the simulation, in terms of the suitability of the method (*e.g.*, low fidelity), the appropriateness of the tools (paper prototype, WoZ), the design of the scenario, the setting of the simulation testing, and the participant selection. This will allow us to test assumptions, identify procedures and processes that work and their potential shortcomings, and it will, in general, allow us to assess if the simulation was suitable for the questions that it poses and for the vision of the system it aims to achieve.

1. Assess your goals. Have you reached the goal(s) that you set for the simulation testing?

2. Assess your methods and tools. Were the methods you have chosen (*e.g.*, low fidelity simulation), the particular tools (*e.g.*, paper prototype), and the particular scenarios and settings appropriate for what you intended to find out?

3. Assess your procedures. How reliable are your procedures? Were your data collection and analysis consistent, was the simulation set-up realistic, how have the facilitators and wizards performed, have you selected the right participants? Have you engaged them in a way that will allow you to test your assumptions? Was the simulation procedure done in a way that allows for a productive engagement?

4. Assess your data. How have you collected your data? Have you set up the procedure in a way that allowed you to capture the data you need? Have you collected the right data? Can you use the data for analysis and for further iterations? Do you need more data or different types of data?

5. Assess your findings. Are your findings valid? How robust are your conclusions and can they be generalized to a larger population of end-users? What do your findings indicate? Do they provide the insights you need to move forward? Are there any surprising or unintended findings? How will you integrate the findings into the next iterations/final system design?

6. Reflect. What went well, what did not? What can be done differently? How should it be done differently? What do your participants think about the scenarios and how the simulation went?

7. Iterate. With the insights collected both from the simulation testing and the evaluation of the methods and procedures, design a new simulation procedure if needed (see section 5.3 for detailed guidance on iteratively improving simulation methods).

5. Discussion and recommendations

Summarizing the previous chapters, this chapter presents an integration of recommendations for simulations to test human-AI collaborations. The simulations discussed involve low-fidelity prototypes to simulate AI systems for end-users before the systems are fully developed. This serves the PEER project goal of ensuring the development of behaviours of the final AI systems is aligned with end-users' needs and requirements.

5.1 Ideal and less-than-ideal circumstances

The previous chapters mostly describe simulation approaches and evaluation techniques for ideal situations. Given all the time, resources and participants a tester may desire, the preparation, implementation and evaluation phases of a simulation will smoothly deliver the answers to the questions for which the simulation was intended. Unfortunately, circumstances are often less than ideal. In reality, time will often be too short, resources will be constrained, and representative participants may not be available, or only in limited numbers.

What is important in less-than-ideal circumstances, is to maintain the basic structure of the protocols, and to safeguard the core principles of simulation testing (see next section). Certain elements of the protocol require little time or no extra resources. For instance, given testers have succeeded in finding participants for the simulation, it will require little extra time or resources to conduct a wrap-up at the end of a simulation to ask participants to evaluate the scenarios and set-up of the simulation. Documentation of all steps may take some extra time but does not require extensive resources and is essential to answering the main purposes of the simulation.

A bare-bones protocol for a simulation test will therefore retain the three preliminary steps (1. define purposes, 2. identify users, 3. set fidelity), albeit within the realistic limitations of the circumstances - for example, the level of fidelity may not be a perfect fit for the simulation purposes, but testers will strive for as near a fit as can be achieved. The bare-bones protocol will also retain the overall structure of a preparation phase, an implementation phase, and an analysis phase, following the outlines in section 3.4.

At a very basic level, the preparation phase can under any circumstances address the following questions:

- Which scenarios address the purposes of the testers and the purposes of the end-users?
- Which participants can we invite (who are as near to representative as we can get)?
- How do we invite participants (including a description of the simulation set-up and a consent request)?

- Which materials do we need for the scenarios and the data collection?
- What should facilitators, wizards, assistants, and note-takers know and how can they be trained?
- How should we adapt the scenarios, prototype and instructions after a test run?
- How will we analyse the results and what do we need to prepare for that?

At a very basic level, the implementation phase can under any circumstances address the following questions:

- How do we briefly introduce the simulation and provide comprehensive instructions to the participants?
- Are we collecting as much feedback and other data as we can to answer the purposes of the simulation?
- What do the participants think of the simulation set-up, the scenarios and the instructions?
- Can the participants identify any biases or other confounding variables?
- How do we store the collected data in a structured and easy-to-analyse manner?
- Have we made ample notes of how the procedure went?
- How do we evaluate the simulation set-up, the procedure, the representativeness of the participants, and the level of fidelity directly after the procedure?

Lastly, at a very basic level, the analysis phase can under any circumstances address the following questions:

- How do we integrate all the collected data types and analyse them coherently and consistently?
- Even if involving multiple analysts from different disciplines is not possible, can different perspectives be taken on the findings, such as a design perspective, a behavioural interpretation perspective, and so on?
- How do we evaluate the reliability of the simulation procedure, is it consistent?
- How do we evaluate the ecological validity, *i.e.* the fidelity level and realism of the set-up?
- How do we evaluate the internal validity, *i.e.* the suggested relations between the stimulus from the prototype/scenarios and the participant responses?
- How do we evaluate the external validity, *i.e.* the generalizability of the results?
- How will we incorporate the participant feedback in next iterations or the final system?
- What should a next iteration or the final system improve?

As mentioned before, the participants themselves can to some extent be involved in the analysis phase, in the absence of sufficient analysts. In less-than-ideal circumstances, one needs to make the most of every opportunity.

5.2 Core principles for simulation testing

Throughout this document, we have emphasised that end-user involvement must be an integral part of the AI system development process, for reasons varying from improved system design (and lower costs) to acceptance and adoption of the system in practice, including the social acceptability of introducing the system in a societal context. We have also argued this inevitably involves negotiations between the different needs and perspectives of different stakeholders. Against that background, two approaches to simulations have been described that allow for such negotiations.

As for the procedural aspects of simulation testing and its evaluation, Table 4 presents the core principles behind the recommendations made in this deliverable.

Category	Essential elements
Preparation	Clear questions to be answered • Multiple scenarios • End-user definitions • Appropriate fidelity level • Correspondence between questions and methods
Team management	Trained facilitators / wizards / assistants • Note-takers • Technical support (WoZ)
Data analysis	Data types analysed in coherence • Multi perspective analysis • Evaluation of internal and external validity • Participant evaluation of simulation
Quality assurance	Pilot test protocol • Documented deviations • Secure data storage • Evaluation of reliability
Participation management	True involvement beyond tokenism • Representative participants • Ethical considerations • Evaluation of participant representativeness
Reporting	Documentation of all steps • Evaluation of simulation set-up • Recommendations for further iterations

Table 4: Core principles for simulation procedures

5.3 Evaluating and iteratively improving simulation testing

Considering everything elaborated, we summarized all the steps, protocols, and procedures in a Simulation procedure and evaluation framework (see Figure 4 below). This framework details the questions, assumptions, elements, and criteria that should guide the process of simulation evaluation. It strives towards comprehensiveness; however, it does not claim to achieve that fully. Some elements and criteria may not be applicable, and additional ones may be necessary. This will depend on the particular system and simulation in question.

The framework encompasses all the phases of the simulation process, from its ideation to the evaluation phase. Proper system development involves multiple iterations through various levels of fidelity (from low to high fidelity). The framework we present, along with the reporting based on it, should be used as input and guidance for each subsequent iteration (e.g., within the PEER project, as part of WP3 and WP5 activities). In that sense, the answers to the questions should be recorded and answered as detailed as possible. This will also serve as institutional memory later, either for the next iterations or as a learning material for new/similar simulations.

This framework is intended for use in both pre-testing and post-testing of the simulation. It should help guide the design of the simulations to assess the suitability of the simulation method, identify limitations and shortcomings, and develop mitigation strategies for subsequent simulation iterations.

Simulation procedure and evaluation framework[11]

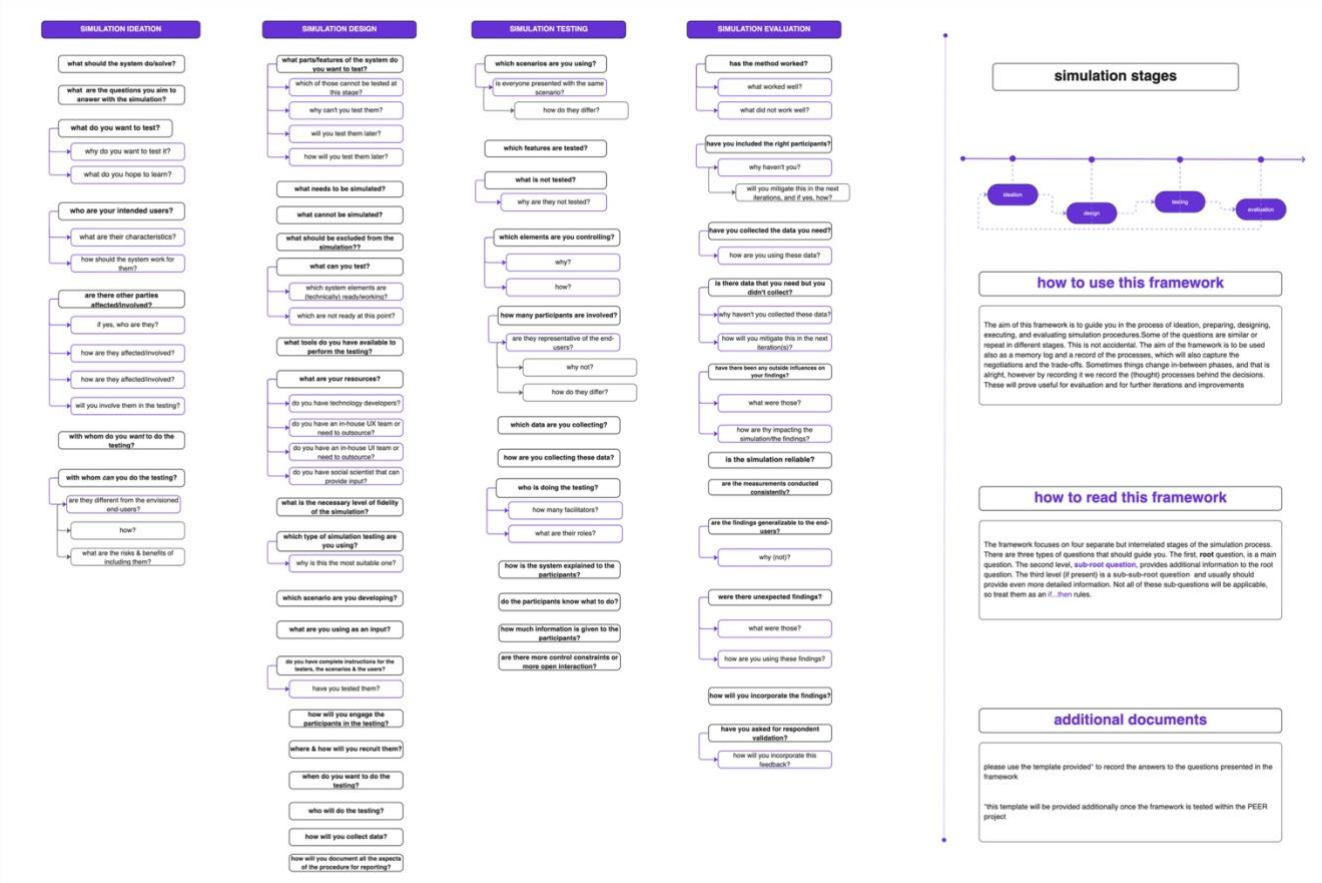


Figure 4: Simulation procedure and evaluation framework

[11] The high-res version is available in Miro, here.

This framework will be further tested within the PEER project, evaluated and modified accordingly, and presented in the D2.4 Guidelines to use AI co-creation and simulation methods.

6. Conclusions

This deliverable presents simulation methods to test human-AI collaborations, in particular, low-fidelity methods based on the paper prototyping and the Wizard-of-Oz approaches. These approaches involve producing prototypes to simulate AI systems for end-users before the systems are developed. The purposes and benefits of incorporating this type of simulation testing early in the process of technology development and engaging the potential end-users in co-creation processes are elaborated throughout the text. The goal of such testing is to ensure that the development of behaviours of the final AI systems is aligned with end-users' needs and requirements.

Testing an envisioned system in the early stages of its development is crucial because it enables us to minimize the *present-future gap*. There are many factors that influence the design, the acceptance, and adoption of a particular system/technology, and undertaking simulation testing and evaluation provides opportunities to account for them.

The goal of this deliverable is not only to present the benefits of simulation methods and testing, but also to make this process easily translatable into practice. To this end, we define, describe, and summarize the entire process of simulation testing, encompassing initiation, design, and development, testing, and evaluation stages. We provide guidance and describe protocols and elements that need to be followed and/or taken into consideration to ensure reliable and valid findings that could provide valuable insights to the questions asked and/or for future iterations. Finally, we provide an extensive framework for simulation design and evaluation, which should serve as comprehensive guidance and a checklist for the entire simulation process and procedure (ideation, design, testing, and evaluation). This framework will also be tested with its intended users within the PEER project, and based on feedback, modifications, and iterations, it will be presented in its final form in D2.4 Guidelines to use AI co-creation and simulation methods.

7. References

Akrich, M. (1992). The de-scription of technical objects. In *Shaping technology/building society, studies in socio technical change* (pp. 205–224). Cambridge: MIT press.

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. (2019). Guidelines for Human-AI Interaction. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 1–13. https://doi.org/10.1145/3290605.3300233

Bernsen, N. O., Dybkjær, H., & Dybkjær, L. (1994). Wizard of Oz prototyping: When and how? *Proc. CCI Working Papers Cognit. Sci./HCI*.

Browne, J. T. (2019). Wizard of Oz Prototyping for Machine Learning Experiences. *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, 1–6. https://doi.org/10.1145/3290607.3312877

Bryman, A. (2016). *Social research methods.* Oxford university press.

Camburn, B., Viswanathan, V., Linsey, J., Anderson, D., Jensen, D., Crawford, R., Otto, K., & Wood, K. (2017). Design prototyping methods: State of the art in strategies, techniques, and guidelines. *Design Science*, *3*, e13. https://doi.org/10.1017/dsj.2017.10

Catani, M. B., & Biers, D. W. (1998). Usability evaluation and prototype fidelity: Users and usability professionals. *Proceedings of the Human Factors and Ergonomics Society 42nd Annual Meeting*, *2*, 1331–1335.

Cockton, G., & Woolrych, A. (2002). Sale must end: Should discount methods be cleared off HCI's shelves? *Interactions*, *9*(5), 13–18. https://doi.org/10.1145/566981.566990

Dahlbäck, N., Jönsson, A., & Ahrenberg, L. (1993). Wizard of Oz Studies—Why and How. *ACM Intelligent User Interfaces '93*, 193–200.

Deng, W. H., Lam, M. S., Cabrera, Á. A., Metaxa, D., Eslami, M., & Holstein, K. (2023). Supporting User Engagement in Testing, Auditing, and Contesting AI. *Computer Supported Cooperative Work and Social Computing*, 556–559. https://doi.org/10.1145/3584931.3611279

Fraser, N. M., & Gilbert, G. N. (1991). Simulating Speech Systems. *Computer Speech and Language*, *5*, 81–99.

Green, A., Huttenrauch, H., & Eklundh, K. S. (2004). Applying the Wizard-of-Oz framework to cooperative service discovery and configuration. *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No.04TH8759)*, 575–580. https://doi.org/10.1109/ROMAN.2004.1374824

Grufberg, K., & Holmquist, L. E. (2011). Magical Bits: Designing through experiencing the future end product. *Procedings of the Second Conference on Creativity and Innovation in Design*, 127–130. https://doi.org/10.1145/2079216.2079233

Kelley, J. F. (1984). An iterative design methodology for user-friendly natural language office information applications. *ACM Transactions on Office Information Systems*, 2(1), 2641.

Kelley, J. F. (2018). Wizard of Oz (WoZ)—A Yellow Brick Journey. *Journal of Usability Studies*, *13*(3), 119–124.

Lim, Y.-K., Stolterman, E., & Tenenberg, J. (2008). The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Transactions on Computer-Human Interaction*, *15*(2), 1–27. https://doi.org/10.1145/1375761.1375762

Nasir, J., Oppliger, P., Bruno, B., & Dillenbourg, P. (2022). Questioning Wizard of Oz: Effects of Revealing the Wizard behind the Robot. *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 1385–1392. https://doi.org/10.1109/RO-MAN53752.2022.9900718

Nielsen, J. (1990). Paper versus Computer Implementations as Mockup Scenarios for Heuristic Evaluation. *Proceedings of the IFIP Tc13 Third Interational Conference on Human-Computer Interaction*, 315–320.

Riek, L. (2012). Wizard of Oz Studies in HRI: A Systematic Review and New Reporting Guidelines. *Journal of Human-Robot Interaction*, 119–136. https://doi.org/10.5898/JHRI.1.1.Riek

Salovaara, A., Oulasvirta, A., & Jacucci, G. (2017). Evaluation of Prototypes and the Problem of Possible Futures. *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2064–2077. https://doi.org/10.1145/3025453.3025658

Sauer, J., & Sonderegger, A. (2009). The influence of prototype fidelity and aesthetics of design in usability tests: Effects on user behaviour, subjective evaluation and emotion. *Applied Ergonomics*, *40*(4), 670–677. https://doi.org/10.1016/j.apergo.2008.06.006

Sefelin, R., Tscheligi, M., & Giller, V. (2003). Paper Prototyping—What is it good for? A Comparison of Paper- and Computer-based Low-fidelity Prototyping. *NEW HORIZONS*.

Wang, Q., Madaio, M., Kane, S., Kapania, S., Terry, M., & Wilcox, L. (2023). Designing Responsible AI: Adaptations of UX Practice to Meet Responsible AI Challenges. *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 1–16. https://doi.org/10.1145/3544548.3581278

Wang, H., & Blok, V. (2025). Why putting artificial intelligence ethics into practice is not enough: Towards a multi-level framework. *Big Data & Society*, *12*(2), 20539517251340620. https://doi.org/10.1177/20539517251340620

Wensveen, S., & Matthews, B. (2014). Prototypes and prototyping in design research. In P. A. Rodgers & J. Yee (Eds.), *The Routledge Companion to Design Research* (1st ed., pp. 262–276). Routledge. https://doi.org/10.4324/9781315758466-25

Yang, Q., Steinfeld, A., Rosé, C., & Zimmerman, J. (2020). Re-examining Whether, Why, and How Human-AI Interaction Is Uniquely Difficult to Design. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–13. https://doi.org/10.1145/3313831.3376301

PEER will focus on how to systematically put the user at the centre of the entire AI design, development, deployment, and evaluation pipeline, allowing for truly mixed human-AI initiatives on complex sequential decision-making problems. The central idea is to enable a two-way communication flow with enhanced feedback loops between users and AI, leading to improved human-AI collaboration, mutual learning and reasoning, and thus increased user trust and acceptance. As an interdisciplinary project between social sciences and artificial intelligence, PEER will facilitate novel ways of engagement by end-users with AI in the design phase; will create novel AI planning methods for sequential settings which support bidirectional conversation and collaboration between users and AI; will develop an AI acceptance index for the evaluation of AI systems from a human-centric perspective; and will conduct an integration and evaluation of these novel approaches in several real-world use cases.





This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101120406.