1 Introduction to Evaluation

Interactive systems have come to play an essential role in the lives of individuals, in shaping how communities develop, and in our economies. One of the key ideas in HCI is to take an interest in the people that use such systems (which Part **??** called being human-centered). Therefore, we should assess that the interactive systems we develop work as intended and that adverse effects do not occur. We must ensure that the systems are practical, usable, accessible, and much more to deliver the value to the people that its designers imagined. In HCI, such assessments are called *evaluations*. This part describes how to conduct evaluations of interactive systems.

Evaluations have taught us much about which systems work and in which way they do not work. For instance, Consolvo et al. [7] developed a mobile system to encourage physical activity and combat a mainly sedentary lifestyle. They evaluated how 12 people used the system over three weeks as part of their everyday life. This evaluation showed us which types of physical activity that participants performed and which activities that the system could and could not infer. As another example, Amershi et al. [1] collected 18 guidelines for successful human-AI collaboration. They show how practitioners can use the guidelines to evaluate systems that rely on AI. For instance, a guideline called "Make clear why the system did what it did" helped practitioners identify many cases where systems violated the rule. Finally, the side box shows a classic evaluation in HCI, which used experiments and quantitative measures of usability to improve a system called Superbook over several iterations. This evaluation helped improve Superbook from being a reasonable manual to be better than paper manuals.

In general, *evaluation* refers to the attribution of value, for instance, stating whether something is 'good' or 'bad', or if it 'fails' or is 'acceptable'. To arrive at such a judgment for an interactive system, one must not only collect data on some *evaluation criteria*, but also have a justified way to conclude if the criteria are met. Evaluation in this sense is different from the common-sense understanding of the word. A subjective opinion is not enough. The goal is an appraisal that one can trust, base actions on, and that others can replicate and scrutinize.

It is worth being clear on *why* we evaluate systems. These reasons have made evaluation part of most models of human-centered systems development. They include the following.

- It is impossible to make systems correct in the first attempt. Gould and Lewis [10] pointed out that everybody builds a prototype. Some people evaluate it one or more times, and some people merely deliver it to the customer or the marketplace.

- Evaluation pays off. The literature on the return-on-investment of evaluations is unequivocal: Whenever money is spent on an evaluation, they are returned many times [5].

- Evaluations typically involve and engage users. Involving stakeholders in the design and evaluation of interactive systems leads to better uptake of the systems and helps keep "continuous focus on users and their tasks" [10].
- Early evaluation helps prevent bad ideas from being constructed and introduced to people. This, again, save resources. For comparison, such assessments are frequently done well with respect to the technical workings of an interactive system: computer scientists and engineers check and test that the systems store data reliably, do not crash, and give accurate results.
- A lack of evaluation may reflect negatively back on us and our organizations. The Association of Computing Machinery's code of ethics states that a computing professional should "strive to achieve high quality in both the processes and products of professional work"[1].

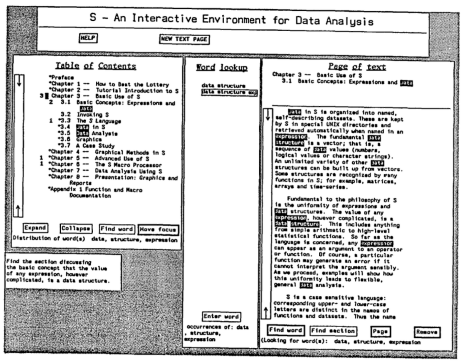
Evaluation is related to other parts of the book, yet distinct. User research (Part **??** is about obtaining concrete insights about *particular* users, their activities, their needs and wants, and their context of use. These insights are not evaluative and are typically collected before anything is designed. In contrast, evaluation is about assessing how well interactive systems work (or representations of interactive systems); it is done after at least some representation of an interactive system has been completed. In other words, evaluation is about how good an interactive system is, whereas user research is about informing what a good system might be. Somewhat confusingly, some empirical method may be used for either purpose. An interview (see Section **??**), for instance, may be used to understand users' activities and to evaluate a system; A think-aloud study (see Chapter **??**) may be used both to learn about a users' work and to identify usability problems. What differs is the intention of the researcher applying the method.

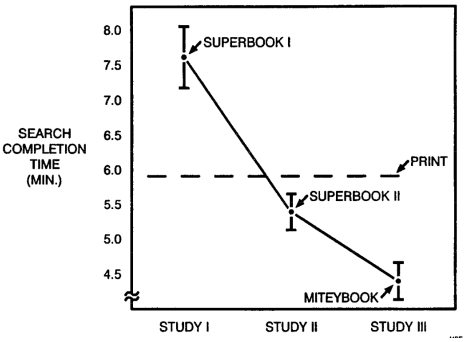
Next, we first discuss in more depth the goals of evaluation and then turn to the yardsticks that may be used as the standard of evaluation. Based on this, we give an overview of evaluation methods and discuss their key quality characteristics.

[1] https://www.acm.org/code-of-ethics

Can computers beat paper in support of reading?

Throughout the history of HCI, researchers have tried to understand why paper, as a material, is so good for reading and how to create interactive systems that could perform better than paper. The work on Superbook [9] is an outstanding example of the iterative evaluation of a system, compared to the performance of paper. Superbook, a version of which is shown below on the left, is a hypertext browsing system giving access to content about statistics.





The evaluations of SuperBook consisted of three rounds of iterative software development and evaluation. The evaluations were done as comparisons of searching in SuperBook and printed manuals offering the same information across tasks that required finding answers to a question and an essay writing task.

The evaluations show three important results. First, as shown in the figure above, the first version of SuperBook performs worse on question answering than the printed manual. People's use of paper is flexible, and here even some of the brightest minds of HCI could not make a system that in one good performed better than the paper baseline. Second, over iterations, careful observations of interaction patterns and usability problems help improve the system. Evaluation works! Third, the work shows that evaluations can use challenging tasks. In one test, participants wrote essays using either SuperBook or the printed manual. An essay could, for instance, be about comparing three different functions of the statistics system described. Among other things, an expert in statistics assessed the essays. This example shows that an evaluation can tap even complex problem solving.

1.1 Goals of Evaluation

Evaluations may be undertaken for a variety of reasons. In general, those reasons shape what is done in an evaluation, how it should be carried out, and how it is interpreted and reported. Therefore, being clear upfront about the goals of an evaluation helps to take the appropriate choices and to plan the evaluation in the best possible manner. Start with why!

One primary reason for doing evaluation is to *improve an interactive system.* This is a frequent activity in the practical development of interactive systems. Most software companies have people engaged in conducting human-centered evaluations of their software.

Attribute	Measuring concept	Measuring method	Worst case	Planned level	Best case	Now level
Initial use	Conferencing task	No. of successful interactions in 30 min	1–2	3–4	8–10	?
Infrequent use	Task after 1-2 weeks disuse	% of errors	Equal to product Z	50% better	0 errors	?
Preference over product Z	Questionnaire score	Ratio of scores	Same as Z		None prefer Z	?

Table 1.1: Examples of quantitative goals of a summative usability evaluation; adapted from [24], p. 799

These evaluations may identify features of the system that, unexpectedly, do not work well for a particular group of users, a particular task, or in a particular use situation. We may then change those features in a future version of the system. For this reason, evaluations for this purpose are called *formative* because they help shape a system. Formative evaluation is essential in the iterative development of system because it gives input to what to fix in the next iterations. As an example of improving a system with evaluation, the icons in the early computer Xerox Star were extensively evaluated [4]. Among other things, the evaluation investigated whether users could precisely associate a name with an icon and to which extent they found an icon easy or difficult to "pick out of a crowd". This evaluation helped pick the set of icons used for the Star.

The other primary reason for evaluations is to discover *how well an interactive system performs with respect to some given objective*. The goal here is not to inform design, but to ensure that the system fulfils the objective. Such an objective may be part of a requirement specification, be included in a contract, be used to select a system for procurement, or to establish a superior design. This use of evaluation is often called summative. As an example, Whiteside et al. [24] proposed to establish quantitative objectives for different aspects of an interactive system. Table 1.1 shows three examples of such objectives for a conference system. The specific methods used to investigate these objectives may be of any kind.

A related but important goal of evaluation is to *identify system features that work well*. This is unrelated to the two main goals of evaluation but is important because it may help to confirm that our expectations of a particular design has indeed been fulfilled. Moreover, pointing out the positive features of a design has been pointed out to be a good way of making the stakeholders of evaluation, for instance, developers of interactive systems, appreciate the evaluation. Thus, usability reports often list positive findings and meeting non-functional objectives (like those of Table 1.1) may also be an important finding. Another secondary goal of evaluations is to teach us about things not strictly about whether or not a particular standard is met. Even though it is not the primary purpose, evaluations may teach us about users and their tasks. For instance, a user in

a think-aloud study may react to a task by saying "we do not usually do it that way" which is important information. Therefore, we are essentially getting information from evaluation that is usually obtained through user research.

Finally, evaluations are widely used in research in HCI to give evidence for particular theories, such as those discussed in the part on understanding people (see Part **??**). For instance, we may derive predictions from a theory and evaluate them to other theories. Some of the methods used to do such evaluations are described in this part (e.g., experiments, think-aloud studies), although the goal behind their use is to understand people.

1.2 Yardsticks of Evaluation

Evaluations strive to *attribute value* to an interactive system. Such attribution requires some way of figuring out if a system is valuable. However, to say that something is 'good' in the case of interaction is tricky. What is 'good'? If a system is very responsive, does it mean it is usable? It depends. If a user is skilled at using a poorly designed system, does it mean the system is usable for that person? Again, it depends. A defining characteristic of evaluation in HCI is that neither people nor technology alone can determine evaluation. The focus is on interaction.

The basic question in evaluation, then, is what our understanding of good is, or the yardstick against which we evaluate systems. Such yardsticks are typically derived from our view of what a good interaction is (see Part **??**) or from the objectives of user interfaces (see Part **??**). For instance, an interactive system is only useful if it is useable. Thus, we may use a particular task as a yardstick and assess a system as good if a large percentage of users (say, 80%) can complete it.

Table 1.2 shows some examples of what yardstick interactive systems may be evaluated against. Some of these are absolute. For instance, we may assess an interactive system on user satisfaction and use a questionnaire for which we have standard. Some ways of measuring user satisfaction provides reference values (e.g., the System Usability Scale gives a score between 0 and 100; typically, systems score 70 [2]). We may also set a goal for a system that it should be learnable in a certain duration. Absolute yardsticks may be applied to a single system. Other yardsticks are relative. That is, an interactive system can only be considered valuable in comparison to another system, be it a competitor, an earlier version, or an alternative user interface. This is often accomplished with experiments.

To use a yardstick for evaluation, it needs to be *operationalized*. That means that general constructs (such as finding "no usability problems") need to be turned into procedures and measures that allow us to compare a concrete system to a standard. For example, *usability* is often operationalized as a usability test where selected tasks are given to invited participants to complete, while their task completion time, errors, and satisfaction are measured.

A key consideration in operationalization is how to measure such constructs. For example, 'error' can be measured in many ways, such as inaccurate presses, misconceptions,

Yardstick	Definition	Example method
No usability problems	The system does not make the user pause, confused, or give up a task	Think aloud study
Don't make the user think	The system should not "make me think" unnecessarily [15]	Think aloud study
Comply with guidelines	The system should comply with known characteristics of good systems as captured in guidelines.	Heuristic evaluation
Meets usability goals	The system should meet specified, quantitative goals for non-functional requirements	Summative usability test
Compares favorably to X	The system should be better than system X on some measure of usability, accessibility, or similar	Experiment
Reduce cost of maintaining system	Reduce calls to support center	Deployment study

Table 1.2: Examples of yardsticks against which to evaluate interactive systems.

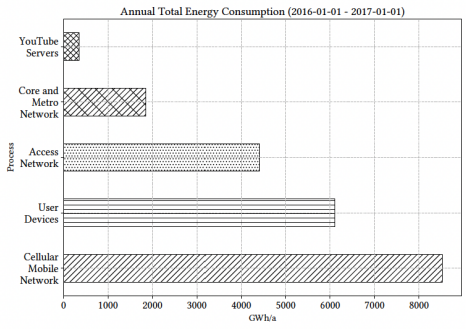
or entering faulty states in the system. Finally, one needs to make *conclusions* based on the obtained data. If a user fails three times in completing a task out of ten, can the system be considered 'usable'? Because the constructs are about good interaction, many of them have been covered in earlier parts of the book. For instance, they include usability, accessibility, autonomy, awareness, memory load, and many more.

Note that describing the interaction with a system does not constitute an evaluation. We may describe which commands people use to interact with a system, which posture they interact in, or what type of content they engage with. None of these, however, gives us information about whether the interaction is good or bad. For instance, even something as straightforward as time requires a standard: low time usage may be interpreted as a lack of engagement, or it may be interpreted as high efficiency [12]. What valuation you make depends on your evaluation standard. Descriptions of interaction may be useful in themselves. They may also, with a yard stick of what is good or bad, be used for evaluation. Such a standard may be an optimal set of commands to compare against, a biomechanically efficient pose, or the content of positive valence. Understanding patterns of interaction and system use is widely practiced in HCI research.

Evaluating Sustainability? Yardsticks for evaluation may be complicated to articulate and operationalize. This is because evaluation may concern any aspect of when the interaction is good. To illustrate this complexity, let us consider how to evaluate whether a system is sustainable.

Sustainability has emerged as a topic in HCI since around 2007 [6]. Naturally, sustainability can act as a yard stick against which to evaluate a system. One such evaluation was done by Preist et al. [21], who were interested in how digital service provides might accurately assess the greenhouse gas emissions associated with particular services. In particular, they were interested in quantifying how much greenhouse gas emissions result from a year of use of YouTube.

The evaluation showed that YouTube use in 2016 created green house gas emissions similar to those of Frankfurt or Providence.



Moreover, people often use YouTube for audio only. In the case where half of YouTube streaming of music is audio only, it is possible to directly save 6% of the total greenhouse gas emissions. Moreover, the evaluation shows that the infrastructure for streaming content, in particular user's devices and the mobile network, contributes the most to emissions. According to Preist et al. [21], those network costs should form part of evaluating sustainability.

1.3 Evaluation Methods

Evaluations are done with established, systematic procedures for carrying them out. Such procedures are called *evaluation methods*. Evaluation methods contain at least some yardstick of evaluation, a process for conducting the evaluation, tools for supporting the evaluation, and a standardized way of reporting the evaluation. The reason for using these systematic methods is that considerable focus has gone into establishing good ways of evaluating systems that are valid, reliable, and useful as a way of giving input to the design of interactive systems.

Table 1.3 shows an overview of some key evaluation methods that will be covered in detail in subsequent chapters. These methods differ in a few key ways. Often *analytical* and *empirical* methods are distinguished. In analytical evaluation methods, an evaluator

compares an interface to guidelines, principles, or theories for good interaction. The assessment of the interface does not involve users. In empirical methods, users interact with the interactive system and that is used as the basis for evaluation.

Another distinction is between evaluation methods is whether they are done in the *laboratory* versus evaluations done in the *field*. Laboratory evaluations are done 'in vitro', away from the users' usual use context. They may therefore easily be done on interactive systems that are not finished. Field evaluations are done during actual use of a system. They emphasize realism [17]. Neither of these two approaches is superior to the other, each of them simply has different benefits and limitations. Please refer to Part **??** for a recap of realism, generalizability, and precision, and for McGrath's arguments about the fallibility of all methods.

Evaluation methods also differ with respect to what *representations of systems* they may be used on. Thus, early representations of systems, including use cases, scenarios, and storyboards may be evaluated. Paper prototypes, hi-fidelity prototypes, and interactive systems that have been in use for years may also be evaluated by some techniques. Generally, though, evaluation methods exist for all types of system representation.

It is worth noting that the methods discussed in Part **??** may also form part of evaluations of interactive systems. Thus, all forms of interviews discussed in Chapter **??** may form part of the evaluations. The defining feature of evaluations merely is that they assess the value of systems relative to some yardstick. In themselves, interviews do not help do that—we need some way to turn the content of interviews into a valuation of the interactive system.

1.3.1 Tailoring Evaluation Methods

Evaluation methods are not generic across all people, activities, technologies, and contexts. They may be tailored to particular instances of those, just like we learned that understanding people may draw on specific information about a group of people or a certain domain. For instance, Druin [8] discussed the strengths and challenges of using children to test interactive systems. For instance, she mentioned that children may offer suggestions in tests that are surprising to adults. They also need to be handled in different ways from adults during the testing. The literature also contain adaptations to elderly users or users with disabilities [23].

Similarly, for specific technologies, we might also draw on evaluation approaches that are tailored to those technologies. For instance, researchers have developed guidelines that fit a range of technologies, including artificial intelligence [1], displays that are distributed in the environment [16], or groupware application [20].

Evaluation methods may also be tailored to particular activities. The evaluation of games, for instance, has seen the development of particular heuristics [22] and methods [18], The evaluation of mobile computing has seen extensive use of treadmills as a way of simulating the demands of walking [3].

In all cases, the evaluation methods are improved by being made more specific and tailored towards specific users, activities, contexts, or technologies. Unfortunately, little is known about how these tailored evaluation methods perform relative to the generic ones.

Method	Definition	Pros	Cons
Heuristic evaluation	An analytic evaluation method in which evaluators go through an interface using a list of features of good user interfaces	Inexpensive	False positives
Think aloud testing	Users verbalize what they think about while they solve tasks with an interactive system; the thinking aloud is analyzed to find usability problems.	Inexpensive, convincing	Short-term
Usability test	An evaluation of the usability of an interactive system with representative users doing representative tasks	Direct assessment of usability	Misses the broader context of use
Experiment	An experimental comparison of the usability of at least two user interfaces	High precision; clean comparisons	Limited realism
Deployment study	Measure evaluation criteria after deployment with real user	Problems are real	Expensive

Table 1.3: Central evaluation methods and their pros and cons.

1.3.2 How to Select an Evaluation Method?

At this stage, the reader might ask: Which of all these methods should we use in a given project? Because of the diversity in interactive systems, user goals, and use contexts, there is no silver bullet of method choice. Professional evaluators master a toolbox of methods and tailors them in a case-specific manner. Part of their considerations revolve around the goals of the evaluation relative to the pros and cons of each method. For instance, analytic methods work best for relatively simple designs and assume that experienced evaluators are available. Due to their high false negative rate, they should not be relied on for complex or safety-critical systems.

1.4 Validity, Reliability, and Impact

As with methods for user research (see Part **??**), evaluation methods raise several fundamental questions about the quality of their application. Some of these are similar to those discussed in the earlier part, but some are different.

Validity of an evaluation is about whether the comparison of an interactive system compared to a standard reflect the real state of the system. For instance, usability problems predicted by an evaluation method should be real problems for real users doing real tasks; otherwise, the evaluation is invalid.

In general, the results on the validity of different validation methods are mixed. A couple of findings stands out, however. First, numerous studies have shown that analytic evaluation can find a high proportion of problems that cannot be found in think-aloud studies. Second, even usability problems found in think aloud tests might not be serious if the users that run into them in a test find a workaround that they can apply every time they subsequently face the problem.

Reliability of evaluations refer to whether the findings of an evaluation would be changed with another set of evaluators or if ran again. If that is the case, the trustworthiness of findings is reduced, and it is unclear if action needs to be taken on the problems, as they might disappear if the evaluation was run again. Reliability has been the topic of much work on evaluation methods [e.g., 19, 13]. The CUE studies, for instance, have numerous times compared the performance of different evaluators or teams of evaluators on the same evaluations find markedly different usability problems[2].

Impact is about ensuring that the evaluation results can be used for their purpose, in particular regarding formative evaluation. Results with impact will help change systems for the better by being convincing and offering concrete input or ideas on how to solve problems. John and Marks [14] compared what they call the persuasive power of usability evaluation methods; persuasion here refers to whether a developer actually makes a change to the interactive system after seeing the results of the evaluation.

One of the advantages of usability tests, for instance, is that they help convince developers that problems are significant. By seeing video of users struggling often help convince developers that they must fix usability issues within a system.

[2]See more at `https://www.dialogdesign.dk/cue-studies/`

1.5 Is Evaluation Needed?

The assumption behind this part of the book is that evaluation, in particular empirical evaluation, is indispensable in HCI research and practice. Not everybody agrees with this assumption.

One example is the paper by Greenberg and Buxton [11], entitled "Usability Evaluation Considered Harmful (Some of the Time)". They argued that often usability evaluation are done without thinking, merely because it is usually required or, as we argued earlier, that evaluation is indispensable in HCI. However, such thoughtless evaluations may harm scientific and practical progress. For instance, evaluation does not help idea generation and may squash it. Evaluation does not help anticipate how people will adopt and adapt technology. And if done to a vision of future interfaces, We agree with the premise of the paper: finding the right method requires thinking, and evaluation is not always a suitable approach. However, the subtitle "(some of the time)" means that evaluation is nevertheless often needed and important.

Another counter-argument to the need for evaluations is the view that interactive systems are objects of design or art. Their value is derived from the artist's or designer's vision and intuition; it is therefore unnecessary to evaluate them. We believe that this view only holds if one lets loose of human-centeredness. To claim that a design process has been human-centered places a burden of responsibility to show evidence that the outcomes achievable with a system can be positive.

In sum, evaluation is about deciding on the goal of the evaluation and the appropriate yardsticks against which to appraise the system. Such yardsticks vary greatly but may include benchmarking against other systems, avoiding usability problems, or increasing subjective satisfaction. Evaluation is an essential for a human-centered development process and, additionally, indispensable for research. The reason is that interactive systems are complex, and assuming that they work as intended by their design is naive. Thus, we need to evaluate the systems.

Summary

- Evaluation is necessary because systems are never perfect and because of the complexity of people, their activities, and physical, social, and organizational context.

- Evaluation methods have different strengths and weaknesses; they may be tailored to specific technologies and user groups

- Validity, realiability, and impact are key concerns for evaluation methods.

Bibliography

[1] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen, J. Teevan, R. Kikin-Gil, and E. Horvitz. Guidelines for human-ai interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–13, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359702. doi: 10.1145/3290605.3300233. URL `https://doi.org/10.1145/3290605.3300233`.

[2] A. Bangor, P. T. Kortum, and J. T. Miller. An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, 24(6):574–594, 2008.

[3] J. Bergstrom-Lehtovirta, A. Oulasvirta, and S. Brewster. The effects of walking speed on target acquisition on a touchscreen interface. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, pages 143–146, 2011.

[4] W. L. Bewley, T. L. Roberts, D. Schroit, and W. L. Verplank. Human factors testing in the design of xerox's 8010 "star" office workstation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '83, page 72–77, New York, NY, USA, 1983. Association for Computing Machinery. ISBN 0897911210. doi: 10.1145/800045.801584. URL `https://doi.org/10.1145/800045.801584`.

[5] R. G. Bias and D. J. Mayhew. *Cost-justifying usability: An update for the Internet age*. Elsevier, 2005.

[6] E. Blevis. Sustainable interaction design: invention & disposal, renewal & reuse. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 503–512, 2007.

[7] S. Consolvo, D. W. McDonald, T. Toscos, M. Y. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, I. Smith, and J. A. Landay. Activity sensing in the wild: A field trial of ubifit garden. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, page 1797–1806, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605580111. doi: 10.1145/1357054.1357335. URL `https://doi.org/10.1145/1357054.1357335`.

[8] A. Druin. The role of children in the design of new technology. *Behaviour and information technology*, 21(1):1–25, 2002.

[9] D. E. Egan, J. R. Remde, L. M. Gomez, T. K. Landauer, J. Eberhardt, and C. C. Lochbaum. Formative design evaluation of superbook. *ACM Trans. Inf.*

Syst., 7(1):30–57, Jan. 1989. ISSN 1046-8188. doi: 10.1145/64789.64790. URL `http://doi.acm.org/10.1145/64789.64790`.

[10] J. D. Gould and C. Lewis. Designing for usability: key principles and what designers think. *Communications of the ACM*, 28(3):300–311, 1985.

[11] S. Greenberg and B. Buxton. Usability evaluation considered harmful (some of the time). In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 111–120, 2008.

[12] K. Hornbæk. Current practice in measuring usability: Challenges to usability studies and research. *International journal of human-computer studies*, 64(2):79–102, 2006.

[13] N. E. Jacobsen, M. Hertzum, and B. E. John. The evaluator effect in usability tests. In *CHI 98 conference summary on Human factors in computing systems*, pages 255–256. Citeseer, 1998.

[14] B. E. John and S. J. Marks. Tracking the effectiveness of usability evaluation methods. *Behaviour & Information Technology*, 16(4-5):188–202, 1997.

[15] S. Krug. Dont make me think: a common sense approach to usability testing. *New Riders, Berkeley*, 2005.

[16] J. Mankoff, A. K. Dey, G. Hsieh, J. Kientz, S. Lederer, and M. Ames. Heuristic evaluation of ambient displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, page 169–176, New York, NY, USA, 2003. Association for Computing Machinery. ISBN 1581136307. doi: 10.1145/642611.642642. URL `https://doi.org/10.1145/642611.642642`.

[17] J. E. McGrath. Methodology matters: Doing research in the behavioral and social sciences. In *Readings in Human–Computer Interaction*, pages 152–169. Elsevier, 1995.

[18] M. C. Medlock, D. Wixon, M. Terrano, R. Romero, and B. Fulton. Using the rite method to improve products: A definition and a case study. *Usability Professionals Association*, 51:1963813932–1562338474, 2002.

[19] R. Molich. Are usability evaluations reproducible? *Interactions*, 25(6):82–85, 2018.

[20] D. Pinelle and C. Gutwin. Groupware walkthrough: adding context to groupware usability evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 455–462, 2002.

[21] C. Preist, D. Schien, and P. Shabajee. Evaluating sustainable interaction design of digital services: The case of youtube. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–12, 2019.

[22] G. F. Tondello, D. L. Kappen, E. D. Mekler, M. Ganaba, and L. E. Nacke. Heuristic evaluation for gameful design. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, CHI PLAY Companion '16, page 315–323, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450344586. doi: 10.1145/2968120.2987729. URL `https://doi.org/10.1145/2968120.2987729`.

[23] T. Van der Geest. Conducting usability studies with users who are elderly or have disabilities. *Technical Communication*, 53(1):23–31, 2006.

[24] J. Whiteside, J. Bennett, and K. Holtzblatt. Usability engineering: Our experience and evolution. In *Handbook of human-computer interaction*, pages 791–817. Elsevier, 1988.