

Supporting Usability Engineering in Small Software Development Organizations

Nis Bornoe and Jan Stage

Aalborg University
Aalborg, Denmark
[nis, jans]@cs.aau.dk

Abstract. Despite an interest and use of different usability engineering methods small software development organizations find it challenging to implement usability engineering into the software development process. We present the results from a study of usability engineering in practice. Through a series of semi-structured interviews, we want to get an understanding of how usability is implemented into the organizations and how it's practiced in reality. We found that the developers found it problematic to combine agile software development methods with classic usability engineering methods. A lack of solid usability engineering expertise and not least experience seems to be a central obstacle to a successful implementation of usability engineering into current software development practices. They are requesting methods and procedures that fit better with their current practices and strategies to implement usability engineering into the organizations.

Keywords: Usability engineering, software development

1 Introduction

Usability Engineering continues to be an essential part of software development, but it's not straightforward to combine usability engineering with software development. Usability is an ambitious concept that goes by many names with no clear, tangible definition [18]. Usability evaluations need to be conducted with a clear objective. Otherwise it's a waste of resources and can even become counterproductive [7].

Through a survey, Bak et al. looked into the obstacles for deploying usability evaluations in software development organizations. They found that the most significant obstacles were missing knowledge and competencies about conducting usability evaluations, the resources required for successful evaluations, and problems with developers neglecting usability perspectives [1]. Several studies have pointed out that small development organizations rarely hire usability specialists or external consultants due to the costs involved [3]. Other studies have found that usability practitioners rarely follow a systematical approach when conducting evaluations [17] and usability evaluations are analyzed informal and according to light-weight and home-brewed approaches [6]. As a result, the quality of usability evaluations is often questionable [15].

Proceedings of the 36th Information Systems Research Conference in Scandinavia (IRIS 36), August 11-14, 2013, Gran, Norway.

This work is licensed under a Creative Commons Attribution 4.0 International License

In this paper, we present a study of usability engineering in small software development organizations. We found that usability engineering is a concern and to some extent a priority, but primarily a lack of competencies and a theoretically founded approach stop these organizations from taking full advantage of the benefits of successful usability engineering. Hiring a dedicated usability engineer was not feasible for several reasons, mainly the size of the organizations did not allow for such an investment. As has been documented in the past usability engineering is not straightforward [10]. This especially holds for mobile usability engineering [13]. Identifying and categorize usability problems requires skills and expertise [20] so it's not surprising that small organizations find usability engineering challenging.

We also found that the use of agile software development methods was a common obstacle as the organizations found it difficult to combine classic usability engineering methods with agile development methods, for example, Scrum. New development patterns and requirements, especially development models based on agile software development, cannot easily integrate with conventional usability evaluation methods and practices. Instead, new practices and not least new ways of planning usability evaluation and user experience design are required [22] [5].

Explicate the need of usability evaluation to the customer was also a challenge. Were usability engineering mainly, if not only, has been focusing on the given users of a system, it has been suggested that understanding the business goals of the system also can be beneficial [11]. A more explicit focus on business goals might make it easier to involve the customer and improve the product.

We believe this research can be of interest to both the scientific community and practitioners, especially small software development organizations. The scientific community has requested more research about how usability evaluations are conducted and analyzed in practice [6] and research has suggested that practitioners can benefit from more systematically and formal methods for both conducting and analyzing usability evaluations [17]. With this study, we wish to take a step towards the development of usability engineering methods and practices that can support small software development organizations. This paper serves as an introduction to get a better holistic view of usability engineering in practice. Where other papers have focused on how usability was done in general, this paper specifically looks into small software organizations operations within the agile development umbrella.

With the study presented it's the intention to get a broad understanding that can be used as a foundation for more focused studies.

2 Background

Several studies have recent years looked into usability engineering from different views including training software developers in usability evaluations [3], how usability evaluations are conducted and analyzed by practitioners [6, 15, 17] and new ways of integrating usability engineering into an agile development process and environment [19]. Recent research has looked into different approaches towards easier implantation of usability engineering in the software development process. This includes studying the

effects of training software developers in conducting user-based usability evaluations [4, 20, 21]. The overall conclusion was that software developers being trained in usability evaluation were able to catch a significant amount of usability problems in comparison to the problems identified by usability specialists. The studies indicated that missing practical experience was the main obstacle for locating more usability problems. Further one of the studies found that the novice usability evaluators delivered incomplete and hard to understand descriptions of the identified usability problems [20].

A few studies have looked into how usability evaluations are conducted in practice, both when it comes to conducting an evaluation and how the data gained was analyzed. Through an explorative study, Nørgaard & Hornbæk observed how usability practitioners conducted thinking aloud usability tests [17]. They found that a systematic analysis of usability problems rarely took place and that evaluators not always got a common understanding of essential observations from the test sessions. Further, they report that sometimes usability tests were mainly used to confirm known or suspected usability problems. The questions asked by the evaluators did not focus on how to get a better understanding of the usability problems. Rather the questions were aimed towards predicting possible problems. They also found that common challenges for the evaluators included conducting evaluations on incomplete prototypes and limited knowledge about the system they evaluated. Nørgaard & Hornbæk suggests that new tools and methods fitting better into reality are developed.

Through a survey study [6] wanted to investigate how practitioners analyzed usability data. Two third of the respondents reported from usability testing and one-third reported from usability inspection. When it came to identifying usability problems the main strategies were observed problems and responses from the users. The main resource for the analysis was “My professional experience” and expertise when conducting heuristic evaluations. When it came to documenting usability problems, descriptions were common and followed mainly a homebrewed format and explained in plain prose.

The context and circumstances in which usability evaluations were conducted are far from static making the usability evaluators evolve analysis practices and adapt tools and methods per a given situation. Følstad et al. conclude that tools must fit the analysis context, which is “fast-paced.” Another point is that usability methods are not seen as indivisible wholes as the study shows that people work with usability methods and approaches as components. How usability evaluations are done, in reality, does not cohere with theory. The methods are considered components that can be combined as needed. Both Følstad et al. and Nørgaard & Hornbæk concludes that the analysis process is challenging and only limited literature about how to conduct an analysis exists.

Some research has focused on ways to support novice evaluators. Skov & Stage has presented a one-page tool called “Usability problem identification tool” [20]. As mentioned identifying and categorize usability problems requires skills and expertise. This tool is intended to support and stimulate the analytic skills of novice usability evaluators conducting website development in an agile environment without resources and skills needed for more extensive and expertise driven usability engineering. In an evaluation,

non-experts were all together able to identify 72% of the usability problems identified by experts. On the positive side, the non-experts identified almost all critical problems.

Research has also evaluated methods to support practitioners. For instance, Kjeldskov and colleagues have introduced what they call “Instant Data Analysis” (IDA) [12]. This approach is designed to conduct an agile analysis of think-aloud usability evaluation sessions. The overall procedure is to conduct a one-hour brainstorming and analysis session in which the data logger and test monitor discuss and identify critical usability problems. The result is a list of usability problems divided into different categories based on severity. Afterward, the screenshots and notes from the sessions can be used to further document and explain the identified usability problems. With this method, the authors want to introduce a quick yet effective method for usability analysis. The claim is that IDA only requires about 10% of the time needed for conducting a full-blown classic video analysis while still catching essential and critical usability problems. The downside is that a classic video analysis will catch more problems and can provide more detailed information about what causes a given problem.

When it comes to documenting and reporting Vilbergsdóttir et al. argue that highly detailed and very structured usability reports are successful [23]. On the other hand, Sy argues that within an agile development frame the reporting should be done in a light manner in the form of meetings and presentations [22].

Combining agile development methods with different user experience approaches and usability engineering has recent year received attention from the scientific community [19]. Here two case studies have been selected. Both Sy and Budwig et al. have through case studies described the implementation of the interaction design process into an agile development framework [5, 22]. These case studies provide an overview of how agile software development was mixed with user experience design and usability engineering. Both case studies outline the overall strategy they followed and the experiences they gained through the process. In both case studies usability was not considered a standalone discipline. Rather they both considered usability, user experience design, and user-centered design loosely under the same umbrella with no strict lines in between.

A defining feature of agile software development methods is that the development process is centered around iterative development cycles and that there is a focus on the creativity of the people as opposed to more strict processes. [8]. Here both Sy and Budwig et al. found the main approach to be strategies that could be used to implement interaction design into these iterative cycles.

Agile development processes are highly feedback driven those completing a frontend design does not make sense. The focus was kept on a few designs at a given time, and usability testing was only conducted in relation to the design parts in the works at the moment. This is by Sy coined “just-in-time design.” Conducting usability testing in this kind of environment was found to be a challenge because only chunks of the design would be completed at a time. A related challenge is that the workflow of a design was impossible to test. The chosen strategy was to conduct quick usability tests with people in easy reach such as internal users. As testing with external users is time-consuming and highly resource-demanding tests with external users was only con-

ducted on relative large design chunks. During the final stages of the development actual workflows could be evaluated. This sort of ongoing usability evaluation conducted within strict development cycles is only successful if the evaluators can maximize the information gained from these evaluations.

Based on the case studies it's concluded that by conducting usability testing within an agile development framework it has been easier to act upon issues and making changes to the product, but it is challenging to have the interaction design team and the development teams running somewhat in parallel. Designing and coding are two quite different processes and in an agile environment, time is always limited.

Also, while doing design in chunks, it's essential to keep a holistic view, which can be tough in an agile environment. These two case studies clear show that despite clear benefits such as the options to make changes to a design during the development process and faster and less resource demanding usability testing, expertise and focus on method is essential. Both studies report how missing knowledge and experience about integrating interaction design into an agile process caused huge problems.

3 Method

During the spring of 2013, we conducted five semi-structured interviews with representatives from different Danish software development organizations. The number of employees ranged between 4–34 employees making it fair to classify them as small businesses. The overall topic of the conversations was “the interplay between usability engineering and software development.” Each interview was divided into two main categories. The first category of questions was related to the type of products being developed and how the development process was organized. The participants were asked to answer based on actual experiences from recent projects. The second category was about usability engineering. Here the questions were centered around how usability was defined and what purpose usability played during the software development phase. Based on recent experiences the participants were asked how usability evaluations were conducted in practice. By having the participants talk about recent experiences we hoped to get more concordant stories and a broader understanding of the development process that could reveal more about how usability, in reality, is being conducted. Especially since usability evaluations and considering usability far from always is an explicit state in the development process. The goal with interviews was to get a somewhat holistic overview of how software development took place while keeping a focus on the interplay between usability engineering and software development. The analysis of the data was done according to observer impression. The data was loosely coded. This coding is partly reflected in how the findings are organized.

The participants had several different backgrounds including academic degrees in the areas of computer science and user experience design some had more autodidact backgrounds, and one oversaw quality assurance. Despite that, the participants worked in small software development organizations that by first glance seemed quite similar it was clear that there were as many differences as similarities. This was both reflected

in the development process, type of products they had specialized in, and the organization of the companies.

4 Findings

All the organizations we interviewed followed some kind of agile development model, formal or informal. Even that several similarities could be located all the organizations had unique routines. This was found to be due to the size of the organizations and because they specialized in different types of products.

The customer is the main priority, and the business goals are essential. A characteristic of the companies is that the developers, in general, have no formal or very limited training in usability evaluation. A common denominator is that the companies had not hired dedicated user experience specialists, but in one case used freelancers for some of the frontend development. A large majority of the development projects are completed within weeks to a few months. Especially this factor is central as such fast-paced environments do not leave much time for traditional usability evaluations. Further resources are limited, and it's a balance between what is requested by a customer, deadlines, demands, and budget. In one company, they had received training in conducting usability evaluations, but this had not been maintained, and the gained expertise was somewhat lost.

The diversity of platforms makes it even more challenging as solutions now must be accessible on several different platforms, especially mobile platforms.

4.1 Roles

The roles of the developers varied between the organizations. Some developers will take on several different roles. A single developer would mainly complete smaller projects. This depends on the type of projects accepted by the different companies. For example, developing a website for a customer only takes a few people, and they will code, design, and evaluate. Where larger projects take up more people, the division between the people involved is stricter. For the small organizations, it's too resource demanding filling out all roles required in a classic development cycle such as the Waterfall model. This was reported as a problem when it came to following a formal agile development approach such as Scrum.

4.2 Usability and user experience

Usability was used a vague term and often mixed with user experience (UX) that also would be used in a vague manner, but it was clear that the motivation for usability is there. Usability was to some extent considered as part of the overall design strategy as opposed to a single process task, but when talking explicitly about usability, the comments would typically be about classic usability evaluation. That being said, usability is considered in relation to other design technics such as wireframing, mockups, and prototyping. This is somewhat opposite to some of the studies mentioned earlier that

looked into evaluation and mainly focusing on the evaluation and analysis of traditional usability evaluation. An umbrella of different methods is used. Again, this would be both formal and informal.

What can be called spontaneous usability testing was quite common. This form of usability testing would take place ad-hoc and conducted very informal. For example, by asking a colleague to try out a feature or get feedback on a certain design.

4.3 Explicate benefits of usability engineering

As the customer has the final saying, especially for none off-the-shelf products, it can be hard to persuade the customer into having usability evaluations conducted. Especially explicating the benefits can be somewhat difficult to document and explain. For example, providing measurements of the value of conducting a series of usability testing is tough. Pointing out the exact benefits for a given product was found to be challenging. Usability was often seen as a feature of a product both by the organizations and the customers. Some customers would consider usability testing as “nice to have” rather than considering it as part of the overall development process. Some organizations would also offer usability evaluations as an option. Because they would not have the expertise themselves, they would need to get an external consultant and pass the bill on to the customer. As mentioned above some usability engineering is part of the development process, but running final tests with users on a more or less fully functional system and afterward making adjustments was the big challenge.

4.4 Following good practices

As mentioned, usability was part of an overall design context. Especially developers with multiple roles would mix design and development into one process. Here it's also relevant to consider that these developers would work more or less independently. Here usability would rely on expertise, experience, and gained knowledge. Typically, by following good practices and reuse standards. For example, a participant explained how he had built up best practices over time and was using an abstract model for designing websites so he would always follow some general designs when creating a website. This included always adding certain elements and placing them in certain orders.

4.5 Reporting usability problems

Reporting of usability problems was done in a wide variety, with the common denominator that it was all lightweight. Detailed reports were by some seen as “heavy” material that would not be read by anyone and not fit in the fast pace of the development environments. In some organizations reporting was not done at all. Rather the developer would rely on quick ad-hoc evaluations and act directly upon the feedback. In other organizations, this would be done through short descriptions, for example, in the “to-do” log for the programmers or by short notes and presentations.

5 Discussion

5.1 Usability engineering in practice

There is no one solution or direction for usability engineering if one wants to fully take advantage of usability engineering [16]. As advocated by Woolrych and colleagues [24], the focus on usability should be on: "...resources combined within specific approaches..." rather than usability methods as hole units. The context and circumstances in which usability evaluations are conducted are far from static making the usability evaluators evolve analysis practices and adapt tools and methods according to a given situation [6]. Følstad et al. mention an idea about using usability methods as components. Sy (2007) talks about "juggle" different designs and evaluations during usability evaluation sessions. The results of this study support these ideas. What is of concern is the potential of increased complexity. These organizations are currently lacking the needed expertise. "Juggling" with different usability resources might even make usability evaluations more unstructured and disadvantageous.

Where Følstad et al. and Nørgaard et al. investigated practitioners without looking into the context they operated in, we specifically considered small software development organizations operating in an agile development community. When investigating usability in practice, we believe especially the development framework is essential to take into consideration.

It's suggested that more research is needed about what usability analysis is, but we also found that in addition, practitioners are highly requesting hands-on methods for conducting an analysis, and tools and methods that fit better into their current practices. Rather than methods, this could be guides or approaches as seen contextual design such as the book "Rapid Contextual Design" [9].

The case studies presented by Sy and Budwig et al. are both conducted at large organizations. Directly exporting their experiences and suggestions to small software development organizations is not a feasible solution. The size of the development teams (sometimes just consisting of a single person) and that the developers can be assigned multiple roles, for example, both being the designer and developer inquiry different approaches. The organizations simply do not have the resources for full-blown agile developing and design teams. Further the projects are often of a size that cannot justify large development and design teams. Rather a selection of different components could be part of the solution.

5.2 Secure the investment

Ordering a piece of software is an investment and usability engineering can be considered one of several approaches to secure the investment. If the software is unusable by the desired target group, it's not a safe and good investment. Here usability engineering should be considered part of quality assurance and the target group should be an explicit and essential constant during the process. Again, usability is not only a figurehead that simply should be considered a last step or add-on.

Defining a target group from the beginning can turn out to be very helpful usability wise. This is one view of usability engineering that can be used to define and explicate usability engineering in software development projects. Here the organizations need to consider usability engineering from different views and as more than evaluations. This especially applies to the quality assurance of a project and can help keeping a project on track by continuously considering a predefined target group. Potentially a better understanding of what usability is will make it clearer to the customer what usability engineering is and why it's something that can be worth investing in.

5.3 The complexity of usability engineering

In Brooks' famous essay "Silver Bullet: Essence and Accidents of Software Engineering" [2] he talks about "essential complexity". That is the complexity, which is a result of the problems to be solved. Conducting usability evaluations in an agile development environment does not mean that the degree of complexity changes. Rather the outcome should have more value. Lightweight usability methods and tools do not imply cutting corners, and should not be interpreted this way, the complexity of usability engineering remains. Rather it's about having the right usability engineering tools and methods for the given development model, in this case agile software development models. Here the so-called lightweight methods seem to fit better into the agile development concept in comparison to more classic usability evaluations. Rather they could be more useful and simply provide more insights resulting in better products. The focus needs to be on the outcome and better integration with the overall development framework. It's about improving the quality and usefulness of usability engineering. A promising approach is explicit integration of UX in the development cycles [5, 22], but to be able to do this specific methods needs to be in place. This introduction study shows that for small software development organizations methods supporting an ad-hoc approach mixed with lightweight approaches not requiring a complex usability lab infrastructure could highly improve the incentive to conduct more structured and higher qualitative usability evaluations.

6 Conclusion

Especially the variety of platforms and the popularity of agile development methods have changed the landscape of usability evaluation. To some extent, usability evaluation should be rethought when it comes to productive and useful usability evaluations in an agile development context. This is both backed up by the case studies presented earlier and through the results of this study. Using quick methods while explicit focusing on maximizing the output of usability evaluations should be the aim. An approach could be somewhat strict guidelines and efficient methods. The word "guideline" is used to reflect flexibility, as we also found that different organizations operate differently, the nature of the projects are different, and even that they all organizations followed some sort of agile development approach, they still have unique routines.

The resources, both when it comes to hiring dedicated designers and usability evaluators and expertise are limited. The organizations highly lack experience and expertise. It's also quite clear that simply hiring designers is not necessarily the one and only path to walk. Integration of usability engineering into the development process is not a straightforward task, yet this could be one approach to integrating more formal and useful usability engineering into small development organizations. Flexible tools and approaches seem to be essential. It's suggested that approaches focusing on the fast-paced nature of these organizations are used for development of new usability approaches or modifications to existing ones.

When studying usability in practice context, product types being developed, the development method and the size of the organization needs to be considered.

7 Future work

Where the majority of past research has looked into single usability test sessions, typically some form of think aloud usability evaluation, we want to focus the research on a holistic view of the integration of usability engineering into the software development process. As mentioned only very few studies mention anything about the development model used, a parameter we believe is very essential, so we will specifically consider usability engineering in an agile development environment. Based on the study we believe that small software development organizations not will benefit much from only getting introduced to, for example, lightweight usability methods such as "instant data analysis." Instead, they should learn how to integrate usability engineering directly into agile software development process and be able to organize and use different usability methods so they can become capable of adjusting the entire development process from project to project. As has been pointed out by several studies, not two software development projects are identical [6, 24] and they propose that research in usability engineering can benefit from case-studies.

The next planned step is to carry out a study in cooperation with an industry partner to gain more insights into the organizational context and what can be done to improve the usability aspects of the software development process. When it comes to a research method and approach an engaged research lens is interesting because the philosophy of engaged research is both to advance scientific and practical knowledge. Under this umbrella action research (AR) is one proposed approach [14]. AR looks interesting as an overall research model because this model explicit is aimed at understanding an organizational context with the intent to intervene into existing practices. An essential feature of AR is that it's explicitly constructed to engage collaboration between the academic world and practitioners. This research path could result in development and extension of lightweight methods, a better understanding of how usability problems are solved when identified, and how to better train practitioners in usability analysis. With the common use of ad-hoc evaluations, better methods for ad-hoc evaluation could be useful for practitioners.

References

1. Bak, J.O., Nguyen, K., Risgaard, P., Stage, J.: Obstacles to usability evaluation in practice: a survey of software development organizations. Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges. pp. 23–32 ACM, New York, NY, USA (2008).
2. Brooks, F.P.: No silver bullet: Essence and accidents of software engineering. *IEEE Comput.* 20, 4, 10–19 (1987).
3. Bruun, A.: Training software developers in usability engineering: a literature review. Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries. pp. 82–91 ACM, New York, NY, USA (2010).
4. Bruun, A., Stage, J.: Training software development practitioners in usability testing: an assessment acceptance and prioritization. Proceedings of the 24th Australian Computer-Human Interaction Conference. pp. 52–60 ACM, New York, NY, USA (2012).
5. Budwig, M., Jeong, S., Kelkar, K.: When user experience met agile: a case study. CHI '09 Extended Abstracts on Human Factors in Computing Systems. pp. 3075–3084 ACM, New York, NY, USA (2009).
6. Følstad, A., Law, E., Hornbæk, K.: Analysis in practical usability evaluation: a survey study. Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems. pp. 2127–2136 ACM, New York, NY, USA (2012).
7. Greenberg, S., Buxton, B.: Usability evaluation considered harmful (some of the time). Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 111–120 ACM, New York, NY, USA (2008).
8. Highsmith, J., Cockburn, A.: Agile software development: the business of innovation. *Computer* (Long Beach, Calif). 34, 9, 120–127 (2001).
9. Holtzblatt, K., Wendell, J.B., Wood, S.: Rapid contextual design: a how-to guide to key techniques for user-centered design. Morgan Kaufmann (2004).
10. Hornbæk, K.: Current practice in measuring usability: Challenges to usability studies and research. *Int. J. Hum. Comput. Stud.* 64, 2, 79–102 (2006).
11. Hornbæk, K., Frøkjær, E.: Making use of business goals in usability evaluation: an experiment with novice evaluators. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 903–912 ACM, New York, NY, USA (2008).
12. Kjeldskov, J., Skov, M.B., Stage, J.: Instant data analysis: conducting usability evaluations in a day. Proceedings of the third Nordic conference on Human-computer interaction. pp. 233–240 ACM, New York, NY, USA (2004).
13. Kjeldskov, J., Stage, J.: New techniques for usability evaluation of mobile systems. *Int. J. Hum. Comput. Stud.* 60, 5–6, 599–620 (2004).
14. Mathiassen, L., Nielsen, P.A.: Engaged scholarship in IS research. *Scand. J. Inf. Syst.* 20, 2, 3–20 (2008).
15. Molich, R.: The Quest for Quality: Usability Testing Assessment. Proceedings of the 11th Danish Human-Computer Interaction Research Symposium. pp. 56–59 (2011).
16. Nielsen, J.: Usability engineering. Morgan Kaufmann (1993).
17. Nørgaard, M., Hornbæk, K.: What do usability evaluators do in practice?: an explorative study of think-aloud testing. Proceedings of the 6th conference on Designing Interactive systems. pp. 209–218 ACM, New York, NY, USA (2006).
18. Seffah, A., Metzker, E.: The obstacles and myths of usability and software engineering. *Commun. ACM.* 47, 12, 71–76 (2004).
19. da Silva, T., Martin, A., Maurer, F., Silveira, M.: User-Centered Design and Agile Methods: A Systematic Review. Agile Conference (AGILE), 2011. pp. 77–86 (2011).

20. Skov, M.B., Stage, J.: Supporting Web Developers in Evaluating Usability and Identifying Usability Problems. In: Spiliotopoulos, T., Papadopoulou, P., Martakos, D., and Kouroupetroglou, G. (eds.) *Integrating Usability Engineering for Designing the Web Experience: Methodologies and Principles*. IGI Global (2010).
21. Skov, M.B., Stage, J.: Training software developers and designers to conduct usability evaluations. *Behav. Inf. Technol.* 31, 4, 425–435 (2012).
22. Sy, D.: Adapting usability investigations for agile user-centered design. *J. usability Stud.* 2, 3, 112–132 (2007).
23. Vilbergsdóttir, S.G., Hvannberg, E.T., Law, E.L.-C.: Classification of usability problems (CUP) scheme: augmentation and exploitation. *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*. pp. 281–290 ACM, New York, NY, USA (2006).
24. Woolrych, A., Hornbæk, K., Frøkjær, E., Cockton, G.: Ingredients and Meals Rather Than Recipes: A Proposal for Research That Does Not Treat Usability Evaluation Methods as Indivisible Wholes. *Int. J. Hum. Comput. Interact.* 27, 10, 940–970 (2011).