INTERNATIONAL JOURNAL OF HUMAN–COMPUTER INTERACTION, *21*(2), 173–196


# *The Impact of Usability Reports and User Test Observations on Developers' Understanding of Usability Data: An Exploratory Study*

**Rune Th. Høegh**
Department of Computer Science, Aalborg University

**Christian M. Nielsen**
NNIT A/S, Denmark

**Michael Overgaard**
KMD A/S, Denmark

**Michael B. Pedersen**
ETI A/S, Denmark

**Jan Stage**
Department of Computer Science, Aalborg University

A usability evaluation provides a strong and rich basis for understanding and improving the design of user interaction with a software system. Exploiting this evaluation requires feedback that significantly impacts the developers' understanding of usability data about the interaction design of the system. This article presents results from an exploratory study of 2 ways of providing feedback from a usability evaluation: observation of user tests and reading usability reports. A case study and a field experiment were used to explore how observation and usability reports impact developers' understanding of usability data. The results indicate that observation of user tests facilitated a rich understanding of usability problems and created empathy with the users and their work. The usability report had a strong impact on the developers' understanding of specific usability problems and supported a systematic approach to deal effectively with problems.

We are grateful to the four developers from B-Data and Net-Mill who contributed to our two empirical studies. We also thank the anonymous reviewers for providing constructive comments and ideas. The project in which the research behind this article was carried out is partly financed by the Danish Research Councils (Grant 2106–04–0022).

Correspondence should be sent to Rune Thaarup Høegh, Department of Computer Science, Aalborg University, Fredrik Bajers Vej 7, DK–9220 Aalborg East, Denmark. E-mail: runethh@cs.aau.dk

## 1. INTRODUCTION

Implementing software systems in organizations can be a difficult task, and many systems fail to fulfil their goal or have serious limitations (Frøkjær & Korsbæk, 1992). Such limitations include lack of adequate support to the core tasks of the user and unsuitable designs of user interaction and interfaces (Landauer, 1996). Usability evaluations are conducted to overcome such problems. Usability is related to a software system's ability to help specified users achieve specified goals in a particular environment in an effective, efficient, and satisfying way (International Organization for Standardization, 1998). A software system's usability is closely related to the design of the user interaction component of the system. A main purpose of a usability evaluation is to assess the quality of a user interaction design and establish a basis for improving it (Rubin, 1994). This goal is accomplished by identifying specific parts of a system that do not properly support the users in carrying out their work. Thus usability evaluations and the related activities can help designers make better decisions and thereby allow them to do their jobs more effectively (Radle & Young, 2001)

The usability evaluation is the whole process of judging the usability of a specific system. A usability evaluation starts with planning; it involves some systematic assessment of the system in question, and it produces some sort of result, typically a usability report. The systematic assessment is the heart of the evaluation. It can be conducted in different ways. A common approach is to run a series of user tests, where each user test involves a prospective user of the system who applies the test to solve certain tasks (Rubin, 1994).

The interplay between user interaction design and usability evaluation activities can be illustrated as in Figure 1. The design process produces descriptions and implementations of various parts of the complete user interaction design. Some of these design products are applied in a usability evaluation, and the results of this evaluation are subsequently fed back into the design process. The purpose of the feedback is to provide a solid basis for improving the quality of the interaction design; then, the circle may start over again.

Many different design products can be subjected to usability evaluation. The most obvious product is the system itself. Other relevant products are design specifications, functional or paper prototypes, and other artifacts from the development process that represent certain qualities of the system. Usability evaluations conducted early in the design phase of the product life cycle, using very early design

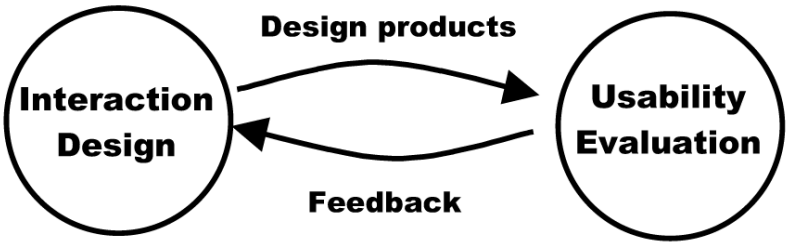


**FIGURE 1** The interplay between interaction design and usability evaluation.

sketches, such as paper prototypes or a similar low-fidelity method of exploring a potential interface, are denoted exploratory tests (Rubin, 1994). The objectives of an exploratory test are to explore the potential of preliminary design concepts and to examine whether the design is based on faulty assumptions or misunderstandings about the needs of the users, which may be difficult and expensive to remedy later in the development process.

Another type of test, which is typically used on a less complete but still operational prototype of the system, is the assessment test (Rubin, 1994). The main goal of an assessment test is to ensure that the assumptions for a system remain relevant and that specific and detailed design choices are appropriate. Furthermore, the assessment test tends to focus more on the level of functionality and usability issues, for example, whether the system supports the users in completing their tasks and whether the system satisfies all user needs. A usability evaluation based on the final system has been denoted as a validation test (Rubin, 1994) and is conducted to ensure that all product design goals have been met. Validation tests aim to evaluate the actual functionality and performance of the system and often present the first opportunity to evaluate all of the elements in the entire system together, although the elements might already have been evaluated individually.

Feedback is the information that is fed back from a usability evaluation to the design process. No matter which evaluation is conducted, feedback to the designers of the evaluated system is crucial to support improvements to the system as a part of an iterative development process. The feedback from the usability evaluations may take a variety of forms. By far the most typical is a written report that presents a number of usability problems. Other forms have also been explored, such as meetings with designers, edited videos, observation, and redesign proposals.

In this article we present the results from an exploratory study of how two different kinds of feedback impact the developers' understanding of the usability of the system that is being evaluated. The study includes two empirical studies of the impact of observation of user tests and reading of usability reports. In section 2, we provide a survey of previous research on the interplay between usability evaluation and interaction design and on different means for providing feedback. In section 3, we present the first of our empirical studies, a case study of the impact of developers observing user tests. The second empirical study is presented in section 4. We conducted a field experiment, where we inquired into the impact of a traditional usability report and its different elements. Section 5 discusses the additional observations that emerged through the two empirical studies. Finally, section 6 provides the conclusion.

## 2. RELATED WORK

The interplay in Figure 1 involves two roles: designer and evaluator. The literature on usability engineering includes a significant body of research that deals with the relation between designers and the evaluators on an organizational level. We have identified three ways of structuring the relation between designers and evaluators: (a) The evaluators are integrated in the development teams and conduct evaluations as part of the work in the team; (b) the evaluators form a separate organiza-

tional unit within the development organization, and they conduct evaluations as a service to development teams; and (c) the evaluators are employed by a different organization, and evaluations are outsourced from the development organization to this organization.

The first way—the integration approach—focuses on the organizational and interpersonal aspects of usability evaluation in a software development organization. Mayhew (1999a, 1999b) suggested that usability engineers are best adopted and introduced into existing development groups. Efforts have been undertaken to simplify the integration problem by training designers to conduct usability evaluations. Other researchers attend to how all levels of an organization can be directed toward usability (Ehrlich, Beth, & Pernice, 1994; Radle & Young, 2001). When the organizational setup is based on usability specialists as part of the development team, there is little need for formalized forms of feedback, because evaluators bring results directly into the development process (Bærentsen & Slavensky, 1999). The results of the evaluation just need to be documented for later reference.

The second way—the separate unit approach—has been discussed by many authors. The idea is to let different people conduct development and evaluation to get a more objective evaluation but to conduct both activities in the same organization so evaluators can benefit from domain knowledge. Rohn (1994) portrayed a usability engineering group inside SunSoft, which provides support and performs usability evaluations across the organization. Several authors have described the use of specialized usability groups or departments employing usability professionals (Blatt, Jacobsen, & Miller, 1994; Fowler, Stuart, Lo, & Tate, 1994; Lund, 1994; Muller & Czerwinski, 1999; Palmiter, Lynch, Lewis, & Stempski, 1994; Salzman & Rivers, 1994; Zirkler & Ballman, 1994). With this organizational structure, there is a manifest need for some form of formalized feedback.

The third way—the outsourcing approach—is the consequent form of separation between development and evaluation. So far, it has had little attention. One strongly separated alternative to the integration and separate unit approaches is third-party vendors providing services to other companies (Dolan & Dumas, 1999). There are also documented examples of projects in which the evaluation has been outsourced (Murphy, Howard, Kjeldskov, & Goschnick, 2004). This approach requires even more formalized feedback compared to the second approach.

With any of these three approaches, some form of feedback is needed. The literature on strengths and weaknesses of different forms of feedback is lacking. A reason for this is that most research on design and evaluation of specific systems take place within the integration approach (the first way previously cited). In a review of 58 articles that present usability evaluations of mobile systems, all 58 showed the designers and the evaluators to be the same individuals (M. C. Nielsen, Overgaard, Pedersen, & Stenild, 2004). Thus in research experiments it is often the designers themselves who perform the usability evaluation. This approach has both advantages and disadvantages concerning the outcome of a usability evaluation. The advantage is that the evaluators are familiar with the application domain and the functionality and design of the system (Hartson, Shivakumar, & Pérez-Quiñones, 2004). The disadvantage is that the lack of independence between designer and evaluator might result in a less objective evaluation, because the designer risks being biased toward the system (Bachrach & Newcomer, 2002).

When the separate unit or the outsourcing approach is employed, there is a need for formalized feedback. In that case, most of the literature seems to take for granted that this feedback must be a written report. Dumas and Redish (1993), Rubin (1994), and Molich (2000) suggested usability reports as the mean for communicating results of a usability evaluation. A study has shown that test reports are very common and standardized documents (Borgholm & Madsen, 1999). Muller and Czerwinski (1999) also described the use of reports within Microsoft to share usability engineers' findings and recommendations by making them available on the company intranet. Cockton (2004) extended this with the idea of a value-centered approach in which the interplay between design and evaluation extends from a simple feedback loop to a chain of mediations where the focus is on the intended value of the system.

One study of usability reports suggests four basic guidelines for providing feedback from evaluation to design: emphasize the positive, express your annoyance tactfully, avoid usability jargon, and be as specific as you can (Dumas & Redish, 1993). A few authors have presented specific advice on the structure and content of a usability report (e.g., Perfetti, 2003; Redish et al., 2002), and an ANSI standard (American National Standards Institute, 2001; Scholtz & Morse, 2002) specifies the elements that should be reported from a summative usability evaluation. The advice on feedback suggests that a description of the goal of the evaluation is included and that the feedback is only limited to a manageable number of problems, as a too-long list of problem can be overwhelming to the recipients of the feedback. The feedback should also include an executive summary listing the most important findings, possibly along with a video that illustrates the problems. Furthermore, it is suggested that identified problems are classified according to their severity along with the frequency with which different numbers of participants encountered the problem. Others, such as Sy (1994), have argued that the feedback should clearly state the implications of not changing the product and that the report also should include graphical illustrations to aid in quick retrieval of information when the report is aimed toward the management.

The widespread use of usability reports as the key mechanism for providing feedback does not imply that there is a generally agreed-upon report format. In one comparison of usability evaluation methods, the usability reports varied in length from 5 to 36 pages, and there were many differences in content. The problem lists were also presented very differently (Molich, Ede, Kaasgaard, & Karykin, 2004).

Different authors have discussed the extent of the relevancy of positive evaluations in the report. Both Perfetti (2003) and Redish et al. (2002) supported the idea of including positive findings in usability reports. Developers have argued that positive findings are always nice, but they cannot really use them for improving the system. Hence they do not find any reason for spending a lot of time on finding positive aspects. Including redesign proposals in the feedback from usability evaluations, however, have received a more positive response from developers. Frøkjær and Hornbæk (2005) studied how developers received such proposals. They found that redesign proposals were useful tools for developers, because the proposals would help them understand the usability problems, and the proposals were useful as inspiration to finding alternative solutions on how to address a usability problem.

There is less research on other forms of feedback situation than the traditional usability report. Sy (1994) suggested a brief meeting with the development team as a part of the feedback. The purpose of the briefing is to give a walk-through of the findings and help the developers to prioritize the issues to deal with. Sy specifically suggested an action list as the product of the discussion with the attendees.

The overall concepts defined in XXXXXXXXXX and the three approaches to structuring the relation between designers and evaluators that were previously discussed reflect our fundamental understanding of usability work. Here, design and evaluation are considered two separate activities in a system development process. This understanding is basically influenced by the work of J. Nielsen (1993) and Rubin (1994). It should be emphasized that we have made this choice well aware that other approaches are based on a fundamentally different understanding. Norman (1998) emphasized that design and evaluation should be integrated into the design process. This is also the focus of Moran's (2002) notion of adaptive design.

## 3. STUDY 1: A CASE STUDY COMPARING OBSERVATION AND USABILITY REPORT

In this section we present a case study of how direct observation of a series of user tests impacted two developers' understanding of usability problems in the system they had developed. Next, we describe the method of the case study and the results we obtained.

### 3.1. Method

The purpose of this empirical study was to make a qualitative assessment of observation as a means for providing feedback from a usability evaluation to the development team. We carried out a study of the impact of observation of user tests on software developers and compared it to the impact of a traditional usability report.

The study was made in the natural settings in the field, and we gathered information from two developers from the same development organization. These are key characteristics of a case study (Benbasat, Goldstein, & Mead, 1987; Wynekoop & Conger, 1990).

***Participants.*** A research group consisting of five usability experts conducted the usability evaluation. Two of the authors of this article were members of that group. The group had experience from more than 30 usability evaluations conducted for software development organizations and customers. The usability evaluation referred to in this article was made for a customer who intended to buy the software system under evaluation.

The usability evaluation was based on five user tests with five different users. The system was intended for the health care area, and the five users were nurses or medical doctors.

A project manager and a senior developer from the development team observed the five user tests.

***Materials.*** The evaluated system is an interactive online facility for booking doctors and operation units for operations in a hospital. The system enables access to and reporting of data about commissioned operations for patients. The system is normally accessed from a personal computer connected to the hospital network.

The results of the usability evaluation were documented in a usability report with a list of contents as in Table 1. This structure is based on Rubin's (1994) guidelines for structure and contents of a usability report. The usability report included a description of the evaluation method and the setting, demographic data about the test subjects, a list of problems sorted by severity, detailed descriptions of the problems, log files with transcripts of the individual usability tests, and user scores in a TLX-test. The list of problems sorted by severity is the most essential information in the usability report. It details the identified usability issues, and the problems are described in detail. In addition, the log files describe the situations where the user encountered the different usability issue. The TLX-test is a multidimensional rating procedure to measuring users' work effort based on a weighted average of ratings on six subscales (Hart & Staveland, 1988). The 52-page report included transcripts and screen dumps of the graphical user interface. Fifty-one usability issues were found and categorized using a scaling from Molich (1998). The problems were rated as 19 critical problems, 15 serious problems and 17 cosmetic problems.

***Procedure.*** The usability evaluation was formative, as the purpose was to provide the software company with a basis for improving the system. The evaluation was based on five user tests that were conducted by using the think-aloud protocol (Rubin, 1994). The user's interaction and utterances were recorded on video for later analysis. Five test participants completed a series of seven tasks. The session for each user lasted about 45 min.

Developers were sitting in an adjacent other room with live video screen display and audio from the test room (see Figures 2 and 3).

After the user tests were completed, a detailed video analysis was conducted in order to identify and categorize usability problems. The report was sent to the software company approximately 1 month after the evaluation.

**Table 1: The Structure of the Usability Reports**

| *Usability report structure* | | |
|---|---|---|
| **1. Summary** | **3. Expenditure of time** | **5. Conclusion** |
| **2. Method** | a) Expenditure of time on tasks | **Appendix** |
| a) Purpose | b) Discussion of expenditure of time | a) Tasks |
| b) Procedure | **4. Usability problems** | b) Introduction and questionnaires |
| c) Test participants | a) Problem list | c) Log files |
| d) Test procedure | b) Detailed description of problems | d) Screen dumps |
| e) Location & equipment | | |
| f) Identification & categorization of problems | | |

*Note.* Bold numbers denote chapters; letters denote sections.

**FIGURE 2** Setup in the usability laboratory: Test participant and test monitor.

***Data collection.*** After every second user test, we had a short discussion with the two developers. During the discussion, we noted their expressions about the users, the system, and the usability problems they had observed.

Three months after the evaluation was completed, we conducted individual semistructured interviews (Kvale, 1997) with the two persons who observed the user tests. In a semistructured interview, the interviewer uses a set of prepared questions but is free to follow and investigate interesting topics during the interview. The interviews focused on how the observation of the user test had impacted the further development of the software and how the usability report was used in that process. The interviews were tape recorded with the developers' permission and lasted about 40 min. The tape recordings of the interviews were fully transcribed to facilitate detailed qualitative analysis. The result was a 20-page document for each interview.

***Data analysis.*** The interviews were interpreted and analyzed using grounded theory (Strauss & Corbin, 1998). This was done by two of the authors of

**FIGURE 3** Members of the development team, observing by video from an adjacent observation room.

this article through a four-step process. First, they separately analyzed the transcripts of the interviews using open coding. The codes were used to identify a range of categories and properties in the data. The analyzer defined the codes, and each code referred to one or more quotations in the document. After this step was completed, both authors had a set of open codes from the interview transcripts. Second, the two sets of codes were merged, and a final set of codes was agreed on. Third, the two authors separately conducted a metalevel analysis to create families of codes, here called *categories*. As the categories emerged, they were compared to other incidents in the transcripts to broaden or narrow the category. Fourth, the two lists of categories were joined, and a final list of categories was agreed on.

The merged list included 56 codes. For each code there were references to between 1 and 15 quotations. From these codes, we defined nine categories, each including between 1 and 16 codes. The validity of this empirical study relies on the grounded theory approach. Grounded theory is a methodology for collecting and analyzing empirical data. The purpose of the method is to support the building of a theory from a set of data, typically a large text. The theory takes the form of a conceptual framework that reflects the understanding that the researcher has developed from the text. An overview of grounded theory as a research approach in human–computer interaction is provided by Pace (2004).

### 3.2. Results

The nine categories covered the issues presented in Table 2. Although there are only 56 codes, the number of codes in all categories adds up to 60, because 4 codes were related to two different categories.

The first four categories cover demographic issues, the fifth is an example of usability problems that the interviewees mentioned as examples, and the ninth represents new ideas to feedback. These categories are not discussed in this article, as we focus on the observation and the usability report. Thus we only discuss Categories 6, 7, and 8.

***Observing user tests.*** This category includes 16 codes with 47 quotations. The codes with the largest number of quotations are experience evaluation (8

**Table 2: Categories Identified in the Data**

1. Demography of the user tests (2/6)
2. Demography of the interview persons (4/18)
3. Relation to customers and users (6/24)
4. Test and usability evaluation in the company (10/45)
5. Examples of usability problems (1/6)
6. Observing user tests (16/47)
7. Impact of the usability report (9/39)
8. The developers' reactions to problems (6/16)
9. Ideas to new forms of feedback (6/40)

*Note.* The parenthetical numbers indicate the number of codes included in each category and the number of quotations for all codes of the category.

quotations), disseminating experience (9 quotations), and effect on developers (5 quotations).

The developers expressed that having observed the evaluation had several positive effects on the development process. One of the first things they noticed was that they quickly got a feeling of the most severe problems with the software.

> Well, I think that, once we got started, we realised pretty fast where the biggest problems were. It was actually already after the first few users that it was pretty clear to us where the biggest problems were. And I think that once we got to user number 3 and 4 then … it was the same problems we saw. So perhaps it became a bit repetitive for us.

The development team was able to address those severe problems the day after the user tests, as the developers had already seen them and experienced how the problems influenced the usability of the software. They did not feel the need to wait for the report on the usability evaluation, because some of the problems were obvious. The two observers made their own prioritized list of problems to address straight after the user tests. The list of problems that they had noted included eight of the problems later categorized as "critical" by the usability evaluation team.

The observers focused on the most severe problems. The less critical problems were filtered out and not remembered. The project manager explained that they could not recognize some of the problems when they got the usability report.

It is the main issues you remember when you get home. It is not until you get this report that … It is like an "ahh yes" experience, and then you can use what is in the report. … Well some of them, and I will say that some of them I cannot even remember where they were, or why they were there.

This illustrates the added value of the systematic analysis for usability problems that is documented in the usability report. With the report, the less critical problems are not forgotten. It is, of course, a question if all problems should be reported or if too long a list will exhaust or confuse the developers. In this particular study, the development team decided to focus on the critical problems, and the remaining problems were used as examples of problems to avoid in the future.

***Impact of the usability report.*** This category includes nine codes with 39 quotations. The codes with the largest number of quotations are use of the report (13 quotations), impact of problem list (7 quotations), and time between user tests and report (6 quotations).

The project manager who observed the evaluation said that the report was used very little. The development team had focused on the problems they had seen during the observation of the usability evaluation, and there had been no effort to systematically address the problems described in the usability report. A reason for that was that the development process had very limited resources. The issues described in the usability report were viewed more as "problems to avoid in the future" than something that needed systematic fixing in the current project. The project manager even said that the report was only interesting as an argument to management to get more resources for the project: "Also simply because this piece of paper, it only works for management. The report is an argument to our management—only."

There is, of course, a question of whether the project manager fully understood the purpose of the report, but her statement clearly describes how she views the report. It is not because she does not recognize that there are usability issues in the software, but she does not agree that the report is the best way to communicate the problems.

The developers also emphasized that the delay of 1 month from the user tests to the receipt of the report was too long. They needed more immediate feedback into the development process.

***The developers' reactions to problems.*** This category includes six codes with 16 quotations. The code with most quotations is user training, and other codes cover accepting or rejecting problems and impact in terms of either changes to the current system or transfer to the development of other systems.

The interviews revealed that there is much more information in a usability evaluation than is possible to include in a traditional usability report. The information is related to seeing and hearing the test participants interact with the system in real time in a live observation. The experience of observing the test participant gives a stronger impression of the systems weaknesses and how the prospective users struggle to overcome them.

> It may be that you have written a lot about how they sit and fumble with they keys or how they really have problems figuring out what they should do. But as a developer, to see the woman that really works with the software, and to really understand that she is completely lost. It may be that you write that the user does not see the navigation options, or looses her orientation . . . . It is just not the same, as when the developer is seeing it herself. She has no idea about what to do.

This is an example of a reaction from one of the observers. Her point is that observing the user gave her a much stronger impression of the problems than she got from reading about it. She develops a stronger empathy because she can see the users struggle. This empathy supports her in understanding the problem with the software. The other observer expressed it this way:

> I would say that I was probably more influenced that day, because I saw how severe problems they had. Rather than if I get it on an email, where I would normally think "Oh well, it can not be that bad."

Observing the evaluation is obviously a stronger experience than reading about it. One of the observers added that it was harder not to recognize the existence of the problems. Another benefit of observing the usability problems with the software was that the observers felt that they could more easily explain the problems to the other developers on their team. They had a better feeling of the problems, and they could show the developers the exact situation in which the test participants had experienced the usability problems.

The observers also discussed whether the cause of the usability problems was the system or the amount of training that the users had received. One of them thought these sources were about equally important:

> I think the reason is both the problems of the system and the amount of training. The system has problems, and we have seen that clearly in the test, no doubt about that. But there are also situations where we think the user lacks training.

In our conversations with the developers throughout the test, it was also clear that they started being very defensive. After the first two users had worked with the system, we talked with the developers. They did not express it explicitly, but they clearly conveyed that they were surprised we had been able to find two such incompetent users. After the next two users, they were very quiet, and after the fifth user they told us directly that they wanted to collaborate with us to solve these problems. Thus they gradually became convinced of the problems. The other developer expressed it this way: "I dare to say … without putting it into percentages, I dare to say the major problem is the design of the system." Observing the user tests made the developers much more empathetic to the prospective users of the system, and this empathy was still in place at the time we conducted the interviews.

## 4. STUDY 2: A FIELD EXPERIMENT WITH USABILITY REPORTS

This section describes our second study, a field experiment in which we conducted a qualitative assessment of the impact of usability reports and their individual elements on the developers of a system.

### 4.1. Method

The purpose of this study was to examine how usability reports can impact the developers' opinions about the major strengths and weaknesses of a system. The basis for the study was two usability reports that were made through a usability evaluation in a different but related study. It has been emphasized that usability reports are often very extensive, take a long time to produce, and involve a heavy workload for the author (Borgholm & Madsen, 1999). Therefore, it is paramount that the feedback designers receive from such reports is useful. Otherwise, producing the report would be a waste of resources. Based in this, our aim was furthermore to identify the report elements that the developers found most useful and relevant.

The study was made in the development organization, which is the natural context for working with design issues. The experiment involved systematic manipulation of one variable, which was the reading of usability reports, and measurement of another variable, which was the developers' opinion about the system. Thereby, it can be characterized as a field experiment (Benbasat et al., 1987; Wynekoop & Conger, 1990).

***Participants.*** This field experiment was made in collaboration with a software company developing a mobile system. Two developers from the company were involved in the experiment. They were responsible for the design of the user interface of the mobile system. They described themselves as experienced interface designers based on their educational background and their previous work.

Three of the authors of this article acted together with a fourth person as evaluators of the mobile system. They were split into two teams, and each team made an evaluation and a report. All four were trained usability evaluators with experience from previous usability evaluations. The users that participated in the usability evaluation were 14 tradesmen who had received basic training in using the mobile system.

***Materials.*** The system that was the focus of the field experiment is used by tradesmen and people with similar occupations for registering the consumption of time, materials, mileage, and equipment and for providing online access to the inventory while working in the field. The system runs on a regular mobile phone with a barcode scanner attached. Most of the registrations and interactions with the system are based on barcodes that are taken from a small booklet.

The evaluation was recorded using a mobile camera and recording unit. Both the test monitor and the test participants wore a wireless microphone that transmitted a signal to the recording unit. One member of the group operated the mobile camera and recording unit.

The usability reports from the evaluations are named R1 and R2. The two reports were made by two different teams of evaluators and thereby emphasized somewhat different problems with the system. The structure of the two reports is similar to the one described in the Materials section of Study 1 and Table 1. The enumerations in that table are used as a reference next, where we present our results. The lengths of the reports were also comparable to the one described in the Study 1 Materials section. Table 3 shows the number of usability problems documented, described, and rated according to severity in each of the reports. R1 reported 15 critical, 16 severe, and 17 cosmetic usability problems, and R2 reported 14 critical, 14 severe, and 6 cosmetic usability problems to the developers. Note that the purpose here is not to compare R1 and R2.

***Procedure.*** The two teams conducted their evaluations separately. In each evaluation, the same team of two evaluators carried out the entire process of con-

**Table 3: Number of Usability Problems Found in the Two Evaluations Categorized According to Severity**

| | *R1* | *R2* |
|---|---|---|
| Critical | 14 | 15 |
| Severe | 14 | 16 |
| Cosmetic | 6 | 17 |
| Total | 34 | 48 |

ducting the evaluation, analyzing the data, and writing the usability report. Both teams employed a common severity based on the categories proposed by Molich (2000).

The field experiment involved the five steps shown in Table 4. The table also shows how the two developers were involved in each step. In Step 1, they were instructed about the procedure of the investigation, and then they wrote down their initial understanding of usability and usability evaluation. In addition, they were asked to write down their expectations to the usability reports. In Step 2, they were interviewed about their initial opinion on strengths and weaknesses in the system. In Step 3, one of them received R1 and the other R2. After reading the report, they were asked to describe and explain five strengths and five weaknesses of the system. In Step 4, each developer received and read the other report and was interviewed again about his conception of strengths and weaknesses. In both Steps 3 and 4, the developers also ranked the strengths and weaknesses in a list. The developers were worked separately in Steps 2, 3, and 4. In Step 5, the two developers met and discussed their lists of strengths and weaknesses. They were asked to work out a common list. First the final list from each developer was written on a white board without ratings. Then the developers were asked to discuss and finally agree on a rating for all of the items in the two lists. The rating was important by itself, and it also forced the developers to discuss and reflect on each item.

***Data collection.*** The developers were interviewed when they made their lists with strengths and weaknesses. Our approach was a semistructured interview (Kvale, 1997).

**Table 4: Overview of the Structure of the Field Experiment**

| *Step* | *Developer A* | *Developer B* |
|---|---|---|
| 1 | Outline the process for the developers. | |
| 2 | Semistructured interview on initial opinions on strengths and weaknesses. | Semistructured interview on initial opinions on strengths and weaknesses. |
| 3 | Receive and read R1.<br>Semistructured interview on strengths and weaknesses.<br>Interview conducted by one of the writers of R1. | Receive and read R2.<br>Semistructured interview on strengths and weaknesses.<br>Interview conducted by one of the writers R2. |
| 4 | Receive and read R2.<br>Semi-structured interview on strengths and weaknesses.<br>The developer is also asked to comment on the usefulness of the reports and its individual elements.<br>Interview conducted by one of the writers R2. | Receive and read R1.<br>Semi-structured interview on strengths and weaknesses.<br>The developer is also asked to comment on the usefulness of the reports and its individual elements.<br>Interview conducted by one of the writers R1. |
| 5 | Group discussion where the developers are presented with each other's list of strengths and weaknesses.<br>The two developers develop a common list. | |

***Data analysis.*** To analyze the interviews, we transcribed the interview using opinion condensation as described by Kvale (1997). This was done 2 days after the interviews. Through this kind of transcription, opinions expressed by the interviewees are transformed into shorter and more precise formulations. The intention of the condensation is to be as precise as possible, which means that we maintain the keywords that the interviewee uses. Longer pieces of speech are condensed into a single or a few sentences. The advantage of opinion condensation is that is can help present a relatively large amount of empirical data in an easy-to-read fashion while both preserving and clarifying important issues. Opinion condensation can never be considered equal to traditional transcription of the interview, which has significantly higher level of detail and involves less processing of the original source.

### 4.2. Results

In this section, we present the results from the field experiment. This includes the developers' attitude to usability, the impact of the usability reports on their opinion about the system, and their judgment about the relevance of different parts of the usability reports.

***The concept of usability.*** In the first step, before reading any usability report, both developers were asked to express how they understood the term *usability*. Both of them were able to formulate this in specific terms. Developer A found that *intuitive* was the word that described it best but also mentioned *easy* and *straightforward* to use without "having to read several manuals." Developer B defined *usability* as the specific screens in the system, where the design of the screens should target the user and the information presented should be relevant. In addition, the user interface should be easily understood and look nice. The developers stated that usability is and always has been important in their daily work but that time issues prevent them from analyzing and considering different ideas.

***Opinions about strengths and weaknesses.*** Developer A initially had some difficulties in naming as many as five strengths and weaknesses of the system (see Table 5). He was also somewhat reluctant in prioritizing the items in his lists. His strengths reflected the arguments that the system was sold upon, whereas the weaknesses reflected technical issues encountered in the development process.

After reading the first report, Developer A did not change his list of strengths noticeably; the items were merely rephrased. Contrary to this, his list of weaknesses was completely altered. He adopted many of the issues described in the usability report, and he was now able to expand the list to five items. These items were also more reflected issues that concerned the interaction with the system. Furthermore, social implications caused by the use of the system became evident to the developer.

The second usability report did not profoundly influence his belief about the strengths of the system. It made him rearrange two subjects and add the use of

daily technology as an advantage. Regarding weaknesses, reading the second report only made him rearrange the rankings in the list and expand and rephrase the descriptions of two items.

Unlike his colleague, Developer B was from the beginning able to list five items of strengths and four weaknesses. Most of his initial items were maintained throughout the entire process, and his list was only slightly altered and rearranged (see Table 6). After reading the first report, he only elaborated on the description of the highest ranking strengths and replaced the second highest item. On the other hand, his list of weaknesses was expanded with a problem of understanding the possibilities of interaction as a new item that ranked highest. The remaining items were only rephrased and rearranged. Reading the second usability report did not influence Developer B enough for him to make noticeable changes. He made alterations only to descriptions in the list of strengths and the ranking of the two last items.

The common list, shown in Table 7, was generated in a collaboration between the two developers. This gave rise to debate between the two, where especially the ranking process initiated heated discussion. The developers discussed each top item on their lists until an agreement was reached. It is interesting that all the strengths in the final list can be traced back to the developers' initial lists, either in one or both lists. Some items have been rephrased but emphasize the same advan-

**Table 5: The Lists Generated by Developer A**

| | *Developer A* | |
|---|---|---|
| *List* | *Strengths* | *Weaknesses* |
| 1 Before reading reports | 1. Online: The system can provide relevant real-time information. | 1. GPRS: Limited coverage. |
| | 2. Barcode scanners: All interaction begins with the user scanning. | 2. Barcodes are used to interact with the system instead of the mobile phone. |
| | 3. No software on the mobile phone. | 3. Online: Problem when no connection is available. |
| 2 After first report | 1. Online/No software on mobile phone. | 1. No manual or documentation. |
| | 2. The use of barcode technology. | 2. Error messages. |
| | 3. Customizable. | 3. Handling of logical errors. |
| | | 4. Input of data through the mobile phone is problematic in relation to target user group. |
| | | 5. Human resistance toward the system. |
| 3 After second report | 1. Online/No software on mobile phone. | 1. Human resistance toward the system. Employees feel that they are under surveillance. |
| | 2. Customizable. | 2. No manual or documentation. |
| | 3. The use of barcode technology. | 3. Many barcodes needed to navigate the system. |
| | 4. Hardware: Mobile phone. Everybody knows it. | 4. Browser technology/phone restrictions: Input of data through the mobile phone is problematic in relation to target user group. |
| | | 5. Error messages and handling of logical errors. |

*Note.* GPRS = XXXXXXXXXXXXXXXXX.

**Table 6: The Lists Generated by Developer B**

| | *Developer B* | |
|---|---|---|
| *List* | *Advantages* | *Disadvantage* |
| 1 Before reading reports | 1. Hardware: Mobile phone.<br>2. Few scans necessary.<br>3. Customizable.<br>4. Online—real time.<br>5. Simple solution with limited interaction. | 1. Screen size<br>2. Problems with GPRS. Often slow.<br>3. No manual or documentation.<br>4. Only works on some types of mobile phones. |
| 2 After first report | 1. Hardware: Mobile phone. Everybody is familiar with the technology.<br>2. Displays only necessary information.<br>3. Customizable.<br>4. Online all the time.<br>5. The system is simple and uniform. | 1. The text describing each of the barcodes.<br>2. More user education in needed.<br>3. System reply time.<br>4. Screen size. Difficult to maintain an overview.<br>5. System is interpreted differently on different phones. |
| 3 After second report | 1. Hardware: Mobile phone. Everybody knows it.<br>2. The system is simple and uniform.<br>3. Customizable.<br>4. Displays only necessary information.<br>5. Online all the time. | 1. The text describing each of the barcodes.<br>2. More user education in needed.<br>3. System reply time.<br>4. Screen size. Difficult to maintain an overview.<br>5. System is interpreted differently on different phones. |

*Note:* GPRS = XXXXXXXXXXXXXXXXXXXX.

**Table 7: The Common List Made in Collaboration Between the Two Developers**

| | *Joint List - Developer A & B* | |
|---|---|---|
| | *Strengths* | *Weaknesses* |
| List 4: After Group Interview | 1. Online–real time.<br>2. Customizable.<br>3. Rely on commonly known technology: mobile phone.<br>4. Simple and small barcode scanner.<br>5. The system is simple and uniform. | 1. Online: Problems with GPRS.<br>2. No manual or documentation.<br>3. Human resistance toward the system. Employees feel that they are under surveillance.<br>4. More user education in needed.<br>5. Error messages and handling of logical errors. |

*Note.* GPRS = XXXXXXXXXXXXXXX.

tage. The story is different when it comes to the weaknesses. Here, the two top items also originates from both developers' initial lists, but the third item is A's final top weakness. The last two items were derived from Developer B's final list and are items that were added to his list in the course of the field experiment. Hence it is clear that the usability reports have influenced the developers' opinion about the system's weaknesses.

***Usefulness of the usability reports.*** Both developers used the same approach when reading the two usability reports. Basically, the reports were read from the beginning to the end. Occasionally the Appendixes were used to see the design of the tasks (see Table 1). The log files (5d + 5e) were not read in their entirety but were used to examine details concerning a problem, if developers were uncertain why a problem had occurred. When asked, Developer B stated, "I used the log files to gain further insight into what happened."

Both Developer A and B mentioned that the overview of the usability problems (3c) and the elaborating descriptions (3d) were important for the future work on the system: "I really like the problem list and it is something I can use concretely in my work." The log files (5d) were good, because "they describe what they (the test participants) did. It provided a better feel of what they did, why they could not figure it out, and what they did next." This shows that log files are useful for providing further insight when trying to understand some of the problems in detail.

Log files can provide almost firsthand insight into the specific actions the user performed. Although they cannot be used directly to resolve the problems, the developers also find them important in understanding the conditions under which the tests have been conducted (2). This was mentioned by both developers as being very important in respect to how they rate the validity of the evaluation. On the contrary, Developer B mentioned that "the other assessments and similar elements are quite amusing to read, but they are not very useful," referring to the summary (1) and the conclusion (4).

The developers found the NASA-TLX (3a) method interesting, but they experienced some problems in interpreting the tables with the results of this test. Developer B found that R2 lacked a transcription of the debriefing conducted at the end of each test. This was important, as "it would provide me with a better insight into the participants' attitude towards the system."

During the final interview (Step 5), the developers brought up the issue of using video recordings. In relation to some of the problems encountered in " … the first few minutes, when the user for the first time was presented with the system on the mobile phone," it would have been beneficial if the video material had been available. This would have given him a chance to see the test participants' first reactions.

***Social and organizational aspects.*** A project resource such as time is an overall topic throughout the interviews. On several occasions the two developers used this as an excuse for some of the usability problems that were identified. In the beginning of the experiment, before having seen any of the reports (Step 1), Developer A said, "We know that many of the things are there—many things that we would really like to correct if we had the time." Numerous times both developers mentioned that designing the user interface is an important and necessary part of their job, but they cannot spend much time on analyzing and considering different ideas. They are simply too busy. Developer A expressed that this should be taken into account when evaluating the usability of a system.

As developers, they often find themselves thinking in "states" and "actions" of the system, but according to both developers, the reports can help them to gain fur-

ther insight into how the users think, when they use the system, thereby turning their attention more toward the interaction with the system.

***Evaluation conditions and procedure.*** One of the issues frequently referred to during the interviews was that the users were very inexperienced, and if they were more experienced, the result of the evaluation would have been different. This may be correct, but it does not imply that usability problems found by relatively inexperienced users do not exist. We see this more as a defensive reaction toward a perhaps overwhelming number of usability problems seen from the developers' point of view. This is supported by the developers acknowledging that many of the problems were relevant and should be fixed. When Developer A was asked about his general opinion on the evaluations, he replied that "many of the things mentioned have applicability in our further work" and added that he "can relate to the findings and use them positively."

Another point of critique presented by the developers was that the tasks were not realistic and that this might have affected the outcome of the tests. According to Molich (personal communication, May 18, 2004), this is a typical objection raised by developers. Still, Developer A mentioned, "I am impressed with how many strange errors the users manage to provoke, which we have never thought of ourselves."

## 5. DISCUSSION ACROSS STUDIES

The two sections above have presented the results of the two empirical studies. During the studies and the analysis, topics related to both studies emerged. The key topics are discussed in this section.

### 5.1. Developer Motivation

In both studies, the developers emphasized motivation as a key topic. On one hand, they expressed that usability is a very important issue for a development organization. On the other hand, knowledge about usability problems had only very limited impact on their practice. In addition, there is typically no systematic processing of the usability report. The reason may be that the developers have no motivation to focus on usability.

Observation of users was different, because the developers gained a strong impression of the way "real" users worked with the system:

> When you suddenly stand out there and face them in the real world, and face their lack of competence. … This is about the developers' attitude and understanding. Therefore, it is important to motivate developers by letting them experience "the real world."

Observation of a user test gives this firsthand experience of the real world. Thus this developer believes that observation of system use by real users will increase the motivation of the developers to improve the usability of the system.

### 5.2. Enhancing the Impact of the Usability Report

During the two studies, the participants articulated several ideas that could improve the feedback from the evaluations. A major point with the usability report was the problem list, which was considered to be very important. It was emphasized that the problem list could be improved by giving the reason why a problem occurred. For example, the feedback could say that there is too much information on a page or that a button is placed without a clear reference to its context. The developers further expressed a desire to discuss such explanations in a meeting between developers and evaluators.

> It would be a very good for some of the most important issues, or the most critical issues, to talk them over with the test monitor, or some of the others that had been involved in the user tests. To have them explain why something is a usability problem.

This result shows that the developers do not get all the information they need from the usability report in the traditional form. The developers also requested redesign proposals to key usability problems. At a first glance there seems to be some issues to consider for usability specialists before giving redesign proposals. For example, the usability evaluators do not know the entire system design; they do not know if there are any technical issues or any special customer demands. Moreover, acquiring this information would make the task of generating redesign proposals highly resource demanding. Yet there are promising research results that challenging these concerns. Frøkjær and Hornbæk (2004) reported from a series of interviews where practitioners criticized the form of traditional usability reports, and they concluded that developers were more interested in constructive proposals for redesign than mere descriptions of problems, even if the redesign proposals were only based on sketches. The developers in our study themselves suggested that the best way to working out redesign proposals might be to let the development team and the usability evaluators meet and discuss the key usability problems.

Radle and Young (2001) recognized the importance of interpersonal skills when addressing usability through spoken language in relation to development teams, and Sy (1994) presented additional advisement on how communication of the evaluation results can be improved apart from a usability report. If possible, a meeting should be held to go through the findings with the appropriate people. In such a meeting, designers and evaluators can work out ideas for solutions in collaboration. This way the proposals can be assessed before being implemented and the proposals would still be based on knowledge of the system design.

### 5.3. Combining Multiple Media

Observation of the user tests had a strong impact on the observers who participated in our case study. It increased their empathy for the users and their understanding of the usability problems. Thus, observation seems to be an appealing way of providing feedback from usability evaluation to interaction design. Yet in our case study with direct observation, we identified problems that need to be re-

solved before observation would be more generally applicable as a feedback technique. The first fundamental problem is the time it takes to observe the user tests. For large development teams, it is simply not practical that all members observe all user tests or watch a full video recording, because it consumes too much time. The simple solution is to involve only part of the development team in the observation. This will significantly reduce the time spent on observation. However, if the user tests are observed only by some of the members of the development team, they will face the challenge of disseminating their experience to the members who did not observe the user test.

The second fundamental problem with observation is that is does not facilitate systematic gathering of all the relevant usability problems. In our case, the two members who observed the user tests remembered some of the problems they had experienced. However, other problems were only remembered when the developers read the usability report, and there were problems in the usability report that they could not even remember having observed.

The results from our two empirical studies emphasize that the usability report is an effective means for systematic processing of usability problems. Once the usability problems and their importance are understood, the report is a useful tool for working on the problems. The usability report effectively complements observation. Thus a viable approach would be to have a few key members of the development team observe the user tests and then present their impressions to the rest of the team. All members of the team should then have access to the report.

Some of the impressions that are acquired through observation can, of course, be shared by written texts or conversation, but to provide the same strong experience may require the use of video. Video recordings would allow the entire development team to see the evaluation either in its full length or in smaller parts. The participants suggested that the usability report should be combined with such edited video clips.

> It was really necessary that you actually saw her, and how she moved the mouse, and how she scanned the software with her eyes. You could see what she was trying to, but could not do. The entire situation provides input to an understanding. The report does not describe the browsing with the eyes and the movement with the mouse. That information adds to the understanding.

The two observers in our case study expected that the development team could get similar benefits from seeing the video or video clips illustrating the problems: "They should all have seen it, right. We could afterwards have distributed the tapes, so that the different groups could have sat down and analysed them, and looked into single situations and so on."

It is questionable if it is beneficial for the whole team to view the entire amount of recorded video, as there are often periods without usability problems, although it may a very effective experience to see the most critical usability problems repeated by every user. So for the first impression, the entire video may be a powerful tool to underline the usability issues. In our case study the observers initially saw lack of user competence, as opposed to system defects, as the source of the prob-

lems. The video clips might prove useful when a developer is trying to solve a usability issue and needs to see where and how the problem that he is working on actually occurred.

Video and text feedback might be combined in a multimedia product as evaluated at IBM (Fath, Teresa, & Holzman, 1994), where text describes the usability problems and video clips show examples of the ways in which users experience the problem. Traditional reports are still utilized, but video clips could "provide compelling evidence to developers who are reluctant to correct usability problems" (Fath et al., 1994). A drawback associated with the use of video clips is that it is very time consuming to edit the tapes from all the user tests (Borgholm & Madsen, 1999).

## 6. CONCLUSION

In this article we presented results from an exploratory study of two kinds of feedback from usability evaluation to user interaction design. The study involved a case study of observation of user tests as well as a field experiment with usability reports. Observation impacted the developers essentially by facilitating a rich understanding of usability problems, their severity, and the use situations in which they occur. Observation also created empathy with the users and their process of using the system. The usability report in the traditional form had a strong impact on the developers' understanding of the specific usability problems that the users experienced with the system. The report also supported a systematic approach to deal effectively with the problems. In this process, all of the key elements of the report were deemed necessary.

The results also uncovered fundamental problems with both kinds of feedback. An appealing solution is to combine observation and usability reports. Moreover, the developers stressed that the usability evaluation has a long-term impact beyond the current project. The understanding they acquired of the users and their work would impact the next version of the system as well as other systems that they would work on in the future.

The two empirical studies are qualitative and they have been conducted in the natural setting. This is a well-suited approach when the aim is to acquire knowledge from practitioners and develop theories from it (Benbasat et al., 1987). A main drawback of this approach is limited generalizability of the results (Wynekoop & Conger, 1990). Therefore, quantitative experiments are necessary to support our conclusion further. A different avenue of further work is to explore strengths and weaknesses of other kinds of feedback from usability evaluation to interaction design.

## REFERENCES

American National Standards Institute. (2001). *The common industry format (ANSI/NCITS-354-2001)*. New York: Author.

Bachrach, C. & Newcomer, S. F. (2002). Addressing bias in intervention research. *Journal of Adolescent Health*, *31*, 311–321.

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly, 11*, 368–386.

Bærentsen, K. B., & Slavensky, H. (1999). A contribution to the design process. *Communications of the ACM, 42*(5), 72–77.

Baillie, L. (2003). Future telecommunication: Exploring actual use. *Proceedings of the Interact 2003 Conference*, 697–700.

Blatt, L., Jacobsen, M., & Miller, S. (1994). Designing and equipping a usability laboratory. *Behaviour and Information Technology, 13*, 81–93.

Borgholm, T., & Madsen, K. H. (1999). Cooperative usability practices. *Communications of the ACM, 42*(5), 91–97.

Cockton, G. (2004) From quality in use to value in the world. *Proceedings of the Conference on Human Factors in Computing Systems (CHI) 2004*, 1287–1290.

Dolan, W. R., & Dumas, J. S. (1999). A flexible approach to third-party usability. *Communications of the ACM, 42*(5), 83–85.

Dumas, J. S., & Redish, J. C. (1993). *A practical guide to usability testing*. Norwood, NJ: Ablex.

Ehrlich, K., Beth, M. B., & Pernice, K. (1994, January). Getting the whole team into usability testing. *IEEE Interface*.

Fath, J. L., Teresa, L. M., & Holzman, T. G. (1994). A practical guide to using software usability labs: *Lessons learned at IBM. Behaviour and Information Technology, 13*, 94–105.

Fowler, C., Stuart, J., Lo, T., & Tate, M. (1994). Using the usability laboratory: BT's experiences. *Behaviour and Information Technology, 13*, 146–153.

Frøkjær, E., & Hornbæk, K. (2004). Input from usability evaluation in the form of problems and redesigns: Results from interviews with developers. *Proceedings of the Workshop on Improving the Interplay Between Usability Evaluation and User Interface Design, NordiCHI 2004*, 27–30.

Frøkjær, E., & Hornbæk, K. (2005). Comparing usability problems and redesign proposals as input to practical systems development. *Proceedings of the Conference on Human Factors in Computing Systems (CHI) 2005*, 391–400.

Frøkjær, E., & Korsbæk, H. (1992). Informatization policies in Denmark. In P. H. A. Frissen et al. (Eds.), *European public administration and informatization* (pp. 25–47). Amsterdam: IOS Press.

Hart, S. G., & Staveland, L. E. (1988). Development of a multi-dimensional workload rating scale: Results of empirical and theoretical research. In P. A. Hancock & N. Meshkati (Eds.), *Human mental workload*. Amsterdam: Elsevier.

Hartson, H. R., Shivakumar, P., & Pérez-Quiñones, M. A. (2004). Usability inspection of digital libraries: A case study. *International Journal of Digital Libraries, 4*, 108–123.

International Organization for Standardization. (1998). *ISO 9241-11 ergonomic requirement for office work with visual display terminals (VDTs)—Part 11: Guidance on usability*. Switzerland: Author.

Kjeldskov, J., Skov, M., & Stage, J. (2004). Instant data analysis: Conducting usability evaluations in a day. *Proceedings of NordiCHI 2004*, 233–240.

Kvale, S. (1997). *Interview: An introduction to the qualitative research interview (In Danish)*. Copenhagen, Denmark: Hans Reitzel.

Landauer, T. K. (1996). *The trouble with computers: Usefulness, usability, and productivity*. Cambridge, MA: MIT Press.

Lund, A. M. (1994). Ameritech's usability laboratory: From prototype to final design. *Behaviour and Information Technology, 13*, 67–80.

Mathiassen, L. (1998). *Reflective systems development*. Unpublished doctoral thesis, Aalborg University, Aalborg, Denmark.

Mayhew, D. J. (1999a). Strategic development of the usability engineering function. *Interactions, 6*(5), 27–34.

Mayhew, D. J. (1999b). *The usability engineering lifecycle*. San Francisco: Morgan Kaufmann.

Molich, R. (2000). *User-friendly computer systems (In Danish)*. Copenhagen, Denmark: Teknisk Forlag.

Molich, R., Ede, M. R., Kaasgaard, K., & Karykin, B. (2004). Comparative usability evaluation. *Behaviour and Information Technology, 23*, 65–74.

Moran, T. P. (2002). Everyday adaptive design. *Proceedings of DIS 2002*, 13–14.

Muller, M. J., & Czerwinski, M. (1999). Organizing usability work to fit the full product range. *Communications of the ACM, 42*(5), 87–90.

Murphy, J., Howard, S., Kjeldskov, J., & Goschnick, S. (2004). Location, location, location: Challenges of outsourced usability evaluation. *Proceedings of the Workshop on Improving the Interplay between Usability Evaluation and User Interface Design, NordiCHI 2004*, 12–15.

Nielsen, J. (1993). *Usability engineering*. San Francisco: Morgan Kaufmann.

Nielsen, M. C., Overgaard, M., Pedersen, M. B., & Stenild, S. (2004). *A review of literature on usability evaluation methods for mobile systems*. Department of Computer Science, Aalborg University, Aalborg, Denmark.

Norman, D. (2002). *The design of everyday things*. New York: Basic Books.

Pace, S. (2004). A grounded theory of the flow experiences of Web users. *International Journal of Human-Computer Studies, 60*, 327–363.

Palmiter, S., Lynch, G., Lewis, S., & Stempski, M. (1994). Breaking away from the conventional "usability lab": The customer-centered design group at Tektronix, Inc. *Behaviour and Information Technology, 13*, 128–131.

Perfetti, C. (2003). *Usability testing best practices: An interview with Rolf Molich*. Retrieved May 10, 2004, from http://www.uie.com/articles/ molich_interview

Radle, K., & Young, S. (2001). Partnering usability with development: How three organizations succeeded. *IEEE Software, 18*, 38–45.

Redish, J., Bias, R. G., Bailey, R., Molich, R., Dumas, J., & Spool, J. M. (2002). Usability in practice: Formative usability evaluations—Evolution and revolution. *Proceedings of the Conference on Human Factors in Computing Systems (CHI) 2002*, 891–897.

Rohn, A. J. (1994). The usability engineering laboratories at Sun Microsystems. *Behaviour and Information Technology, 13*, 25–35.

Rubin, J. (1994). *Handbook of usability testing: How to plan, design, and conduct effective tests*. New York: Wiley.

Salzman, M. C., & Rivers, S. D. (1994). Smoke and mirrors: Setting the stage for a successful usability test. *Behaviour and Information Technology, 13*, 9–16.

Scholtz, J., & Morse, E. (2002, May/June). A new usability standard and what it means to you. *SIGCHI Bulletin*, pp. 10–11.

Spencer, R. (2000). The streamlined cognitive walkthrough method, working around social constraints encountered in a software development company. *Proceedings of the Conference on Human Factors in Computing Systems (CHI) 2000*, 353–359.

Strauss, A., & Corbin, J. (1998). *Basics of qualitative research. Techniques and procedures for developing grounded theory* (2nd ed.). Thousand Oaks, CA: Sage.

Sy, D. (1994). Bridging the communication gap in the workplace with usability engineering. *Proceedings of the 12th Annual International Conference on Systems Documentation*, 208–212.

Zirkler, D., & Ballman, D. R. (1994). Usability testing in a competitive market: Lessons learned. *Behaviour and Information Technology, 13*, 191–197.