Training Software Development Practitioners in Usability Evaluations: An Exploratory Study of Cross Pollination

Anders Bruun
and Jan Stage

Department of Computer Science, Aalborg University,
Selma Lagerlöfs Vej 300, DK-9220, Aalborg Oest, Denmark
{bruun, jans}@cs.aau.dk

Abstract. Successful integration of usability evaluation into software development processes requires software companies to employ personnel that possess skills within both usability and software development. However, the sheer lack of usability specialists and their cost are two limiting factors for software companies wanting to integrate usability evaluation. A possible solution to these problems is to cross pollinate by training existing personnel in conducting usability evaluations and analyzing the collected data. This exploratory study extends previous research by showing that it is possible to provide software development practitioners from industry with key knowledge on usability evaluation. Results show that a pair of practitioners can identify the same number of problems as one usability specialist after 14 hours of training. Furthermore, software practitioners are better at providing clear and precise problem descriptions than at describing the impact, cause, user actions and providing data support for observations.

Keywords: Usability evaluation, training, software development practitioners, problem identification, problem descriptions.

1 Introduction

For the past decade software companies have increased their focus on integrating usability engineering (UE) into development processes. A considerable challenge for these companies is the limited supply of usability specialists in the industry, which leads to integration problems caused by missing key knowledge [12], [16]. Another challenge, which especially relates to small software companies, is that these have to cope with the constraint of low budgets. In practice this means that small companies do not have the funds to pay for comprehensive consultancy or staffing of usability specialists [9], [13]. A survey conducted by Gulliksen et al. supports this by showing that usability specialists are primarily employed by medium-sized or large companies [8]. The fact that small companies usually do not have staff that possesses usability knowledge is expressed as one of the main barriers for integrating UE into software development processes [14], [17].

One way of solving these problems may be to cross pollinate disciplines by increasing usability knowledge across existing personnel, an approach which previously has provided positive results. Metzker and Offergeld for instance describe a software project in which developers participated in contextual task analysis, which motivated them to produce components with a high level of usability [14]. However, a recent literature review presented in [2] shows that the majority of related studies are applying university students as the empirical basis, which leaves room for further studies on software development practitioners' ability to apply UE methods. Another point for consideration is the fact that the majority of related work focus on measuring quantitative aspects of e.g. usability evaluations such as the number of problems identified. Thus in the case of usability evaluation, there is also a need to report findings on aspects such as the quality of problem descriptions and in particular which parts of the descriptions that software practitioners find difficult to fulfill.

This exploratory study extends previous research by studying how software development practitioners from industry perform in identifying usability problems and by providing insights in the quality of their problem descriptions. We have chosen to train practitioners in user based evaluation methods, as such methods have proven to be effective in creating the wake-up calls necessary for companies to start focusing on UE or to increase the awareness of developers [11].

The paper is structured in the following way. First we provide a description of the experimental method applied after which we present our findings and discuss these with respect to related work. Finally we provide the conclusion and point out avenues of future work.

2 Method

In this section we describe the scientific method applied which consisted of a training course that provided key usability knowledge and an evaluation experiment that assessed software practitioners' performance in analyzing usability evaluation data. We start by presenting the participants of the training course and experiment.

2.1 Participants

Software Development Practitioners. Five software development practitioners (henceforth mentioned as "SW-P" or "practitioners") employed in a small software company participated in the experiment. Table 1 shows an overview of their job functions within the company and experience with usability work in general. SW-P 1 had 1.5 years of job experience as a systems developer and did not have any experience with usability evaluation during his employment at the company. However, as part of his education he had previously participated in a HCI course and in the conduction of 4-5 usability evaluations (7 years back). SW-P 2 was a test manager with 8 years of job experience in the company and did not have any practical experience in applying usability methods. She had read a single chapter on the subject during her education. SW-P 3 had 2 years of experience as project manager and systems developer, but had no previous experience with usability work. SW-P 4 had

3.5 years of experience as a systems developer in the company and did not have any experience with usability work before this study commenced. SW-P 5 had worked as a systems developer for 2 years in the company. Additionally he had participated in a HCI course during his education and had experience from conducting a single usability evaluation 13 years back.

Table 1. Overview of the software development practitioners' (SW-P) job functions within the company and experience with usability.

SW-P no.	Function	Usability Experience
1	Systems developer	HCI course + 4-5 evaluations
2	Test manager	Through literature
3	Project manager + systems developer	None
4	Systems developer	None
5	Systems developer	HCI course + 1 evaluation

Trainers. The two authors prepared and held a usability training course for the practitioners (see course description in section 2.2 below).

External Raters. Three usability specialists acted as external raters of the problem lists produced during the evaluation experiment as we did not want to evaluate the outcome of our own training (see section 2.3 for further details). None of these raters had taken part in the training or the conduction of the usability evaluation and are thus considered to be unbiased.

Test Users. Six test users were recruited for the evaluation experiment, all of which were representative end users of the evaluated system.

2.2 Training Course

The authors conducted a two-day training course (14 hours) on user based usability evaluations. The course was held as a combination of presentation and exercises. At the end of the course we gave the practitioners a homework assignment in which they were asked to analyze five video clips from a previous usability evaluation of an e-mail client. We collected the resulting problem lists and gave the participants feedback on how they could improve their problem descriptions.

2.3 Evaluation Experiment

The emphasis of this study is based on the usability evaluation conducted by the 5 practitioners after completing the training course. Due to planning time and busy participant calendars this was executed one month later.

System. The system evaluated was a web application that citizens may use when they move from one address to another. The system was partly developed by the software company in which the 5 practitioners were employed but none of the practitioners had participated in the development of the particular system.

Setting. The evaluation was conducted in the usability laboratory at the university which consists of a test room with cameras and a microphone and an observation room behind a one way mirror. During each session a test user was sitting at a table in the test room using the web application. Next to the user a practitioner acting as test monitor would be positioned.

Procedure. All practitioners took part in planning the test while three of these (SW-P 1, 2 and 3, see Table 1) conducted the evaluation. Afterwards, all 5 analyzed the obtained video material and described the usability problems. The usability evaluation was conducted in one day where SW-P 1, 2 and 3 acted as test monitor two times each. After completing the evaluation all 5 practitioners analyzed the video material from the lab individually. One of the authors analyzed the same video material and this person is mentioned as the "HCI specialist" from this point on. The practitioners and the HCI specialist used the same template for describing problems in order to promote a consistent format. The three unbiased external raters were then asked to evaluate the quality of the problem lists created by the 5 practitioners and the HCI specialist. Finally, the HCI specialist held a meeting with the five practitioners in which the 6 individual problem lists were merged into a total list of usability problems, which served as a white list to calculate the thoroughness in identifying problems. At the same meeting a debriefing interview with each of the developers were conducted.

Analysis of Problem Description Quality. The three unbiased external raters were asked to evaluate the quality of the problem lists created by the 5 practitioners and the HCI specialist. To measure the quality of the lists, the raters were asked to first read each problem list and then provide a rating on a scale from 1 – 5 (1 = "Not fulfilled", 2 = "Scarcely fulfilled", 3 = "Partially fulfilled", 4 = "Almost fulfilled" and 5 = "Fulfilled"). These ratings were given on the following attributes (based on the research presented in [3]):

1. Be clear and precise while avoiding wordiness and jargon
2. Describe the impact and severity of the problem
3. Support your findings with data
4. Describe the cause of the problem
5. Describe observed user actions

Finally the external raters were asked to provide a qualitative assessment of each list, i.e. to provide arguments of the ratings given and examples from the problem lists.

4 Results

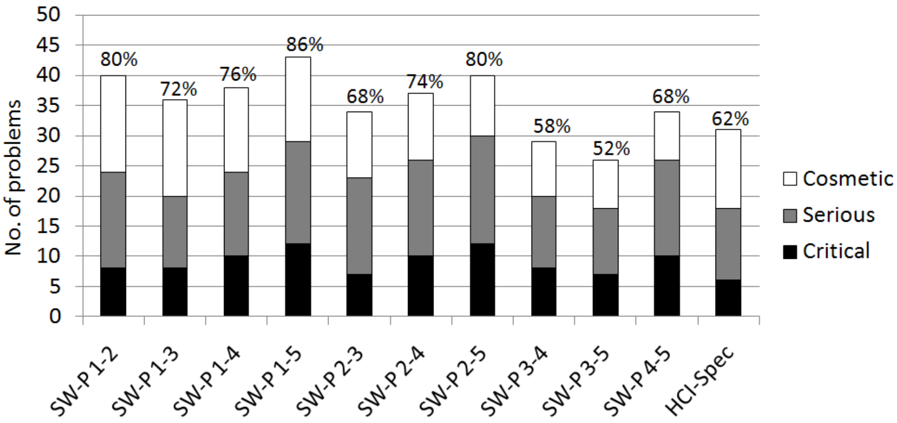
This section presents our findings and is divided in two subsections where the first describes practitioners' ability to identify problems while the second provides qualitative details on their ability to describe usability problems.

4.2 Identification of Usability Problems

Results show that a total of 50 usability problems were identified of which 12 are critical, 19 serious and 19 cosmetic, see [15] for elaboration of severity categorizations. The HCI specialist identified 31 of the problems (62 %) and the practitioners identified between 14 (28 %) and 33 (66 %), the mean being 24.2 (SD=8.1), or 48.4 %. On average practitioners identified 78 % of the problems found by the HCI specialist. Considering the amount of critical problems practitioners identified a mean of 6.8 (57 %) (SD=2.6) where the most and least thorough found 83 % (SW-P 1) and 25 % (SW-P 3) respectively. In comparison the HCI specialist identified 6 (50 %). Considering the serious problems practitioners found 10 (SD=3.9) on average (53 %), the highest being 79 % (SW-P 2) and the lowest 21 % (SW-P 3). The HCI specialist found 12 serious problems (63 %). In the case of cosmetic problems the average is 7.4 (SD=3.2), or 39 %, where SW-P 1 identified most (63 %) and SW-P 4 fewest (21 %), while the HCI specialist found 13 (68 %).

Pair wise Identification. In practice it can be too resource demanding to utilize five evaluators in analysis of usability data, thus in the following we study the effectiveness of each pair of practitioners. Figure 1 provides an overview of the number of problems identified by all pairs of practitioners. All pairs identifies an average of 35.7 (SD=5.2) of all problems (71.4 %), where SW-P 1 and SW-P 5 was the pair that identified most problems (86 %) and SW-P 3 and SW-P 5 identified fewest (52 %). In comparison, the HCI specialist identified 62 %.

Figure 1. Overview of the number of problems identified by all pairs of practitioners.



It should be mentioned that the best performing pair (SW-P 1 and 5) had previous practical experience with conducting usability evaluations, see Table 1**.**
By removing all pairs consisting of SW-P 1 or 5 we see that the average number of identified problems is lowered to 33.3 (SD=4), which amounts to 66.7 % of all problems. Considering the severity categorizations we find that the average number of critical problems identified for all SW-P pairs is 9.2 (SD=1.9), 14.8 (SD=2.4) for serious problems and 11.7 (SD=3.1) for cosmetic problems, or 77 %, 78 % and 62 % respectively.

4.3 Quality of problem descriptions

This subsection describes the software development practitioners' ability to describe usability problems according to the five quality attributes of clarity, impact, data support, cause and user actions, which are derived in [3].

Table 2 provides an overview of the median quality ratings given by the three external raters where higher ratings indicate a higher level of fulfillment according to the quality attributes (1-5 scale). The table shows that problem descriptions written by practitioners 1 and 5, who received the median scores of 4 and 3 respectively, described their usability problems with a quality comparable to that of the HCI specialist (median = 4). The other three practitioners scored lower as their median rating was 2. The table also shows that practitioners are better at being clear and precise (clarity) in their problem lists than any of the other attributes, which is elaborated upon below along with qualitative comments made by the external raters.

Table 2. Median quality ratings given by the three external raters to the problem lists written by the software development practitioners (SW-P) and the HCI specialist.

Participant	Clarity	Impact	Data	Cause	Actions	Overall median
SW-P 1	4	3	4	4	4	**4**
SW-P 2	2	2	2	2	2	**2**
SW-P 3	2	2	2	2	1	**2**
SW-P 4	3	2	2	2	2	**2**
SW-P 5	4	2	3	3	3	**3**
Overall median	**3**	**2**	**2**	**2**	**2**	**2**
HCI specialist	**4**	**3**	**4**	**3**	**5**	**4**

Clarity. Table 2 shows that the practitioners were better at fulfilling the clarity attribute than any of the other attributes as they scored an overall median of 3. In comparison the HCI specialist received the median rating of 4 by the external raters. This was also the case for practitioners 1 and 5. As an example on the qualitative comments given, one of the raters mentioned that 5's list provided *"Good insights in the problems experienced"*. Practitioners 2, 3 and 4 scored the lowest median ratings on this attribute where one rater mentioned the following about practitioners 3's list: *"Extremely short and imprecise descriptions. Actually the descriptions are so poor that you in most cases cannot find out what the problem is"*.

Impact. Table 2 also shows that lower median ratings were given with respect to the impact attribute compared to clarity, which is the case for both the practitioner and HCI specialist descriptions. Practitioners got an overall median of 2 and the HCI specialist 3. Practitioner 1 performed on par with the HCI specialist on this matter and got a higher median rating than the remaining four. One of the external raters commented that practitioners in some problems describe the impact on the user's task but other elements such as business effects and affected system components are left unmentioned. This is also the case for descriptions provided by the HCI specialist.

Data Support. Practitioners' descriptions received an overall median rating of 2 by the external raters where practitioner 1 and 5 scored highest (4 and 3 respectively). In comparison the HCI specialist received the median rating 4 on this quality attribute. One of the raters commented that practitioners in general describe how many test users that experience given problems and that they in certain descriptions state whether or not the task was a success or a failure. Another mentioned that: *"Many problems are not clearly connected to observations"*, thus this rater found that practitioners did not always consider objective data. The same rater additionally mentioned that practitioners made use of vague statements such as: *"The user does not understand"* or *"the user is in doubt"*, statements which are of a speculative nature. However, the practitioners did describe how many test users that experienced the problems and whether or not the tasks were completed, which is similar to the information provided by the HCI specialist. Additionally it was commented that the HCI specialist provided *"good descriptions of the critical incidents"*.

Problem Cause. On this attribute an overall median rating of 2 was given on practitioners' descriptions and the HCI specialist received a median of 3. Practitioners 1 and 5 once more scored higher median ratings than the other three. One of the external raters mentioned the following about practitioner 1's descriptions: *"The list is ok with good descriptions that to a great extent describe causes"*, which was agreed upon by another rater. The third rater, however, found that this practitioner was guessing on the users' thoughts and the cause of the problem in some of his descriptions. Practitioners 2, 3 and 4 were given the lowest ratings in which case all three raters agree that no causes or arguments are provided.

User Actions. Finally Table 2 shows that practitioners and the HCI specialist received median ratings of 2 and 5 respectively on this attribute. Two of the raters mentioned that several of the practitioner descriptions provided examples on users' navigational flow, but that reactions are sometimes described implicitly by stating that users *"are in doubt"* or *"overlooks"* certain elements in the interface. However, according to one of the raters, practitioners 2 and 3 do not describe user reactions at all. Yet again practitioners 1 and 5 scored the highest ratings compared to the other practitioners, where they received medians of 4 and 3 respectively. Two raters found that the descriptions written by the HCI specialist contained detailed information on users' navigational flow and reactions.

5 Discussion

Findings from this study suggest that practitioners are able to identify 48.4 % of all usability problems where the one who identified most problems found 66 % and the one who identified fewest found 28 %. Considering related work, the studies presented in [1], [7] and [21] show that university students are able to identify between 11 % and 33 % of all problems. We additionally found that practitioners on average discovered 78 % of the problems identified by the HCI specialist. In comparison study presented in [20] show that students identified a mean of 37 % of the problems identified by specialists. Thus, in our study we see that the performance of software development practitioners performed closer to the HCI specialist compared to findings in related work.

As mentioned previously, it can be too resource demanding in practice to utilize five evaluators in analysis of usability data, which is why we also examined how many problems each pair of practitioners identified. Our study shows that the most effective pair found 86 % and the least effective found 52 %, where the average was 71.4 %. Also, looking at the number of problems in each severity category, we found that, on average, all pairs identified 77 % of the critical problems, 78 % of the serious and 62 % of the cosmetic. In comparison the HCI specialist identified 50 %, 63 % and 68 % of the critical, serious and cosmetic problems respectively. Thus, we see that two software development practitioners from this study are able to identify more critical and serious problems than the HCI specialist while they have comparable performance with respect to cosmetic. To validate the performance of the HCI specialist in our experiment we found a study conducted by Jacobsen and colleagues which shows that four specialists conducting video based analysis identified an average of 52 % of all problems [10]. This is comparable to the 62 % identified by the specialist in our study. In relation to this it should be mentioned that SW-P 1 and SW-P 5 was the pair that identified most problems (86 %), a finding which may be explained by the fact that they had practical usability experience from their education (7 and 13 years ago respectively, see Table 1). Thus, it could be argued that these practitioners are not novices compared to the participants applied in related work. However, our results indicate, that even by removing all pairs consisting of SW-P 1 or 5 we still find that a pair of practitioners on average perform better than the HCI specialist in terms of number of identified problems.

In the above we have compared the performance of the software development practitioners in this study to that of students', which are used as the empirical basis in related work. The higher level of thoroughness of the practitioners in our study could be caused by differences in the amount of training given and in [7] the students received 6 – 9 hours of training in the form of reading instructions of the methods to be applied. The 14 hours given in our two-day course as a combination of theory and exercises differs considerably from this. On the other hand it is reported in [20] that students received 40 hours of training as a combination of lectures and exercises. Another cause for the differences may be motivational factors, as software development practitioners, due to a competitive market, are more dependent on increased sales of their software products than university students. Also, students may lack incentive in cases where they do not receive payment or if the experiment is part of a mandatory course, a notion which is supported in [20].

Findings also revealed that practitioners on average were unable to fulfill the quality attributes in their problem descriptions to the same degree as the HCI specialist. Exceptions to this, however, were SW-P 1 and 5 who provided a quality comparable to the specialist, which as mentioned earlier may be caused by their previous experience with usability evaluations. Still, the average result corresponds to the findings in [20] in which it is reported that qualitative aspects of the problem descriptions written by students are poorer than that of HCI specialists. Our study extends this quality assessment by dividing it into the five quality attributes mentioned in [3]. This enables us to express that practitioners were better at providing clear and precise problem descriptions than they were at describing the impact, cause, user actions and providing data support for observations. A reason for this may be located in the fact that some of the software development practitioners in our study are used to provide code comments in their software. During one of the debriefing interviews a practitioner mentioned: *"I find it important to write understandable comments because it's easier to get back into the code if you've had one or two weeks of vacation"*. Thus, clarity as a quality attribute is important to industry practitioners in a different context which could indicate why they fulfill the clarity attribute better than any of the other.

6 Conclusion

This exploratory study indicates that cross pollinating usability and software development disciplines may be accomplished by training software development practitioners. Findings show that the practitioners after a two-day training course gained key knowledge on how to conduct usability evaluations as they were able to identify a mean of 48.4 % of all usability problems and that two practitioners are able to identify 71.4 %. This exceeded the performance of an HCI specialist, who identified 62 % of all problems. We also observed that practitioners were better at providing clear and precise problem descriptions than they were at describing the impact, cause, user actions and providing data support for observations. Their problem descriptions, however, were of a lower quality compared to the specialist with the exception of two practitioners.

Findings from this study should be backed up by further studies based on more participants. Also, as our study is conducted at a fixed point in time, we still need studies of long term effects of letting such practitioners do the testing in order to validate that such cross pollination will be carried out in everyday work situations. Also, it would be interesting to conduct further studies on learning retention, e.g. how knowledge within the area increases or diminishes over time.

References

1. Ardito, C., Costabile, M.F., De Angeli, A., Lanzilotti, R.: Systematic evaluation of e-learning systems: an experimental validation. In Proc. NordiCHI 2006, pp. 195-202. ACM Press, New York (2006).

2. Bruun, A.: Training Software Developers in Usability Engineering: A Literature Review. In Proc. NordiCHI 2010, pp. 82-91. ACM Press, New York, NY, USA.
3. Capra, M.G.: Usability Problem Description and the Evaluator Effect in Usability Testing. Virginia Polytechnic Institute & State University, Blacksburg (2006).
4. Edwards, A., Wright, P., Petrie, H.: HCI education: We are failing–why? In Proc. HCIEd, pp. 127-129. Springer (2009).
5. Fonseca, M., Jorge, J., Gomes, M., Gonçalves, D., Vala, M.: Conceptual design and prototyping to explore creativity. In IFIP, vol. 289, pp. 203-217. Springer (2009).
6. Frøkjær, E., Hornbæk, K.: Metaphors of human thinking for usability inspection and design. In TOCHI, vol. 14, issues 4. ACM Press, New York (2008).
7. Frøkjær, E., Lárusdòttir, M.K.: Prediction of Usability: Comparing Method Combinations. In Proc. IRMA. Idea Group Publishing, Hershey (1999).
8. Gulliksen, J., Boivie, I., Persson, J., Hektor, A., Herulf, L.: Making a difference: a survey of the usability profession in Sweden. In Proc. NordiCHI 2004, pp. 207-215. ACM Press, New York (2004).
9. Häkli, A.: Introducing user-centered design in a small-size software development organization. Helsinki University of Technology, Helsinki (2005).
10. Jacobsen, N.E., Hertzum, M., John, B.E.: The Evaluator Effect in Usability Studies: Problem Detection and Severity Judgments. In Proc. of HFES, pp. 1336-1340. HFES, Santa Monica (1998).
11. Høegh, R. T., Nielsen, C. M., Overgaard, M., Pedersen, M. B., Stage, J.: The Impact of Usability Reports and User Test Observations on Developers' Understanding of Usability Data: An Exploratory Study. In Int. Journal of Human-Computer Interaction, vol. 21, issues 2, pp. 173-196. Taylor & Francis (2006).
12. Ji, Y. G., Yun, M. H.: Enhancing the minority discipline in the IT industry: A survey of usability and User-Centered design practice. In Int. Journal of Human-Computer Interaction, vol. 20, issue 2, 117-134. Taylor & Francis (2006).
13. Juristo, N., Moreno, AM, Sanchez-Segura, M. I.: Guidelines for eliciting usability functionalities. In IEEE Transactions on Software Engineering, vol. 33, issues 11, pp. 744-758. IEEE Computer Society Press (2007).
14. Metzker, E., Offergeld, M.: An Interdisciplinary Approach for Successfully Integrating Human-Centered Design Methods Into Development Processes Practiced by Industrial Software Development Organizations. In Proc. IFIP International Conference on Engineering for Human-Computer Interaction, pp. 19-34. Springer-Verlag, London (2001).
15. Molich, R. Usable Web Design. Nyt Teknisk Forlag (2007).
16. Nielsen, J.: Finding usability problems through heuristic evaluation. In Proc. CHI, pp. 373-380. ACM Press, New York (1992).
17. Rosenbaum, S., Rohn, J. A., Humburg, J.: A toolkit for strategic usability: results from workshops, panels, and surveys. In Proc. CHI 2000, pp. 337-344. ACM Press, New York (2000).
18. Rubin, J., Chisnell, D.: Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests, 2nd. Edition. John Wiley & Sons, Inc., Indianapolis (2008).
19. Seffah, A., Gulliksen, J., Desmarais, M.: An Introduction to Human-Centered Software Engineering. In Seffah, A., Gulliksen, J., Desmarais, M.C. (eds.), Human-Centered Software Engineering — Integrating Usability in the Software Development Lifecycle, Human-Computer Interaction Series, Vol. 8, pp. 3-14. Springer, Netherlands (2005).
20. Skov, M. B. and Stage, J., 2008. Direct Integration: Training Software Developers and Designers to Conduct Usability Evaluations. In Proc. of the First Workshop on the Interplay between Usability Evaluation and Software Development. CEUR-WS.org.
21. Wright, P.C., Monk, A.F.: The Use of Think-Aloud Evaluation Methods in Design. In ACM SIGCHI Bulletin archive, vol. 23, issue 1, pp. 55 - 57. ACM Press, New York (1991).