Is Usability Evaluation Important: The Perspective of Novice Software Developers

Fulvio Lizano
Aalborg University
Dept. of CS
Selma Lagerlöfs Vej 300
Aalborg East, Denmark
fulvio@cs.aau.dk

Maria M. Sandoval
National University
Informatics School
PO-Box 86-3000
Heredia, Costa Rica
msandova@una.cr

Anders Bruun
Aalborg University
Dept. of CS
Selma Lagerlöfs Vej 300
Aalborg East, Denmark
bruun@cs.aau.dk

Jan Stage
Aalborg University
Dept. of CS
Selma Lagerlöfs Vej 300
Aalborg East, Denmark
jans@cs.aau.dk

In this paper we present the results of a study which aims to explore the perspective of novice software developers about usability evaluation. It is important for a software organization to understand how novice developers perceive the role and importance of usability evaluation. This will permit development of effective methods and training programs that could potentially increase the application of usability evaluation. The results suggest that the perspectives of novice software developers about usability are characterized by a clear understanding about what usability evaluation is and a clear awareness about obstacles and advantages. However, our study also reveals certain shortcomings in the "usability culture" of novice developers, especially about the users' role in usability evaluation. Despite this limited "usability culture", novice developers' understanding of usability evaluation reflects a positive opinion about their participation in these activities. In addition, novice developers think that usability, in a general sense, is an important aspect of their work.

Usability evaluation, usability evaluation perspectives, novice software developers

1. INTRODUCTION

Usability evaluation is an important and strategic activity in software projects (IEEE Computer Society, 2004). Its relevance had been recognized in the context of the user (Lindgaard & Chattratichart, 2007) and the software organization (Bak et al., 2008). However, several studies had identified important obstacles to its applicacion in software developement process (Bak et al., 2008; Ardito et al., 2011). Some of these obstacles are related to the understanding of the usability concept, resource demands, the lack of suitable methods, availability of users and the software developers' mind-set (e.g. it is difficult to think like a user, lower acceptance of usability evaluations, and developers' emphasis in implementing efficient code)..

Alternatively, Rosenbaum and Rohn & Humburg (2000) reported other obstacles such as resource constraints, resistance to "User-Centered Design/usability", lack of understanding/knowledge about the usability concept, lack of better ways to communicate the impact of work and results, and lack of trained engineers in usability/HCI. In a similar way, Seffah & Metzker (2004) identified problems such as misunderstanding the concept of usability, lack of coupling between User-Centered Design techniques and software development life cycle, the gap between software development and usability, and the fact that education about software development is not coupled with usability. In addition, Gulliksen et al. (2004) argued that the main obstacle is the lack of respect and support for usability issues and its practitioners. Finally, Ferre & Juristo & Moreno (2006) argue that a diffuse positioning of HCI techniques in the software development process is the main obstacle presented to usability.

All of these studies have considered software developers as one homogeneous group. However, there are obviously clear differences between novice and expert software developers. Usually, an expert developer has several years of experience not only in technical activities as for instance coding, but also in other roles, e.g. architect, project manager, etc. (Berlin, 1993; Roff & Roff, 2001). The professional growth process of novice developers is characterized by a continuous learning process both in their formal education at college and their new professional roles in organizations. However, in their academic process it is remarkable the absence of training for soft skills which are a major component in the new jobs. (Begel & Simon, 2008). This fact could explain why according Taft (2007) there are some particular

problems of these new college graduates such as the lack of communication and team work skills, as well as limited experience in complex development processes, legacy code, deadlines, and working with limited resources.

The literature presented above conveys a good understanding of specific soft skills of novice software developers. Yet none of the studies have dealt with novice software developers' perception of usability. This information is crucial in order to develop adequate methods, which enable effective participation of novice developers and facilitate their interaction with more experienced developers. Such knowledge could also help in the design of adequate training programs for novice developers.

This paper presents the results of a study that explored the perspective about usability evaluation of novice software developers. We studied the understanding of the concept of usability evaluation and the obstacles and advantages for apply usability evaluation as they were seen by novice software developers. To complement this particular perspective, the study also explored the importance given by novice developers to usability in a general sense. This paper presents the method used, the results, an analysis section, and finally our conclusion.

2. METHOD

Our study used an online questionnaire with participation of advanced students of a System Engineering undergraduate course.

2.1 Participants

We focussed on advanced students enrolled in the last core course of System Engineering. These students have 18 months of real experience working in a software project with real users. In addition, because of particular characteristics presented in the context where the study was made, 87% of these students normally have a job related with software development processes (Lizano & Sandoval & García, 2008). Finally, the lack of training for soft skills presented in academic organizations (Begel & Simon, 2008), equally affects both advanced students and novice software developers. Combination of previous courses and modest real professional experience has produced in these participants a particular perspective that we were interested in explore.

We contacted the participants through the official list of students and projects. The questionnaire was submitted to 141 students included in the official register of the course. 72 completed it (51%). The average age is 22.2 (SD =2.17). 21 females (29%) participated in the study. All participants lived and worked in the Central Valley, which is the most developed zone in Costa Rica. The organizations where the participants had their jobs or where they carried out their project, had the following sizes: 26% (1-10 employees), 26% (11-50 employees), 17% (51-250 employees) and 31% (>250 employees).

2.2 Procedure

We contacted all the professors who lectured on the last core course of System Engineering in order to explain the motivation behind the study and request their collaboration. All professors then relayed the information to the students. Each student received instructions on filling in the questionnaire with focus on their role as software developers in an organization or as members of a software team that developed a software system in an organization during the previous 18 months.

2.3 Data collection and analysis

The questionnaire was divided into sections such as demographic and general information, importance of usability, understanding of the usability concept, obstacles and advantages of usability evaluation.

The importance of usability issues given by participants was measured using questions grouped in five concept pair. Each pair was formed by two topics, one of them related to software development activities, e.g. "identify potential software problems and bugs" and the other related to usability activities, e.g. "identify potential usability problems". For each pair of concepts, the participants had to select which topic was more important. The concepts were defined based on the main contents of a course in systems engineering (software development topics) and of a course in design, implementation and evaluation of user interfaces (usability topics). The order of the pair of concepts and the position of each concept into the pair, were randomly defined. Two-alternative forced choice was used in order to contrast usability and software development matters. In this sense, our aim was focussed on the relation of usability and common software development matters in the context of novice software developers, which is the logical alternative considering limitation of experience of such developers.

Data on obstacles to usability evaluation were collected by using a combination of open/closed questions. First, an open question was used to allow participants to express an obstacle, using their own words. These open questions allowed us triangulate the results obtained in the closed questions cited above and, in some way, reduce bias of such ipsative questions by offering opportunity to participants to clarify or express in a different way their opinions. Next, a closed

question with several options of commonly known obstacles was presented. The idea was to offer alternative obstacles that the participants had not considered before. The common obstacles were defined based on Bak et al. (2008) and Ardito et al. (2011). We used a similar approach to collect data about advantages of usability evaluations.

We used two different approaches to analyse the data collected. A quantitative analysis was used for the closed questions, while we used the grounded theory approach for the open questions (Strauss & Corbin, 1998).

3. RESULTS

3.1 The importance of usability

We wanted to know how the novice software developers perceived the importance of usability in a broad sense. We presented to the participants several pairs of concepts in order to inquire which one they found most important. The results are presented in Table 1.

Table 1: Perceptions of the importance of usability versus software development activities

#	Detail	# Res	%	Dif.
1	Usability. of soft (U)	41	57%	↑14
	Dev.Quality code (S)	31	43%	
2	Des.bas. U. needs(U)	47	65%	↑30
	Des.bas.requer. (S)	25	35%	
3	Identify usab. prob.(U)	26	36%	
	Identify bugs (S)	46	64%	↑28
4	HCI (U)	10	14%	
	SQA (S)	62	86%	↑72
5	Des.consid. VDP (U)	15	21%	
	Des.consid. patterns(S)	57	79%	↑58
AVG	Usability concepts (U)	28	39%	
	Soft.Dev. concepts (S)	44	61%	↑22

U: Concept/activity related with usability
S: Concept/activity related with other software process

Despite preference on usability in the first two pairs of concepts it is evident that for the novice software developers, technical quality is the primary goal in software development. Overall, the novice software developers find software development activities more important than usability activities (the overall average of perceived importance was 61% versus 39%). The differences are largest in the pairs where usability is contrasted with some software activity related to quality, i.e. the largest difference is in pair 4 where the usability topic presented was HCI. This fact could be originated by certain unawareness about HCI, but it seems as it is mostly related to the preference that novice software developers have for software matters especially by software quality.

3.2 Understanding of the usability evaluation concept

An additional aim of this research was to explore the understanding of the usability evaluation concept among novice software developers. We decided to use an open question in order to obtain these data. 44 of the 72 respondents expressed their understanding in a way clearly related with a generally accepted definition of usability evaluation. In their answers it is possible to find references to concepts such as "user", "test" and "usability". For example, an answer that could illustrate this understanding is: *"It is tests that measure how well a user can use a program, without requiring any external intervention"*.

Fewer participants (11 of 72) responded using concepts more related to functionality, e.g. *"It is the tests made with the end user to verify the functionality of the software, to find and fix errors"* Some other participants (8 of 72) expressed understandings related to other types of testing.

After this open question, we presented to the participants with a definition of usability evaluation based on the ISO-9241 standard. The idea was to explore if the novice developers really found that they had participated in or worked with usability evaluation, according to that particular definition. In general, most novice developers found that they had participated in conducting a usability evaluation; 40 of 72 participants (56%) expressed a high level of agreement on that. Only 2 of 72 participants (2.8%) expressed a high level of disagreement. This result corresponds to the clear understanding that participants have about the usability evaluation concept. In their definitions about what usability evaluation is, the novice developers present concepts or ideas that know by first hand due their participation in these kinds of evaluations.

3.3 Obstacles in conducting usability evaluations

We used a combination of open/closed questions to identify perceived obstacles to the application of usability evaluation according to the novice developers. First, an open question was presented inquiring about obstacles or problems that the respondents had experienced during a usability evaluation. They were requested to write down one or more obstacles or problems. One participant mentioned 2 obstacles, while the rest only mentioned one. Thus the total number of obstacles or problems mentioned was 73. The primary obstacle detected is related with users' behaviour/problems. We identified this obstacle in 23 of 73 items. Next example illustrate this result: "The software is not accepted by the user, even considering that this software is what he had

requested". As it is possible to see in this example, the users' behaviour is presented in a negative context in the software development process. In the second place, we found two different obstacles not necessarily related to usability evaluations. We identified both obstacles in 13 of 73 items. In the first case, participants mentioned problems in software that are directly or closely related to its design (e.g. "There are design factors that the user does not like, or technical details that the user wants in the system"). The second case is related to technical and organizational issues (for example "Problems in the software (bugs), problems with the data (e.g. clean databases)")

In the closed question we presented to participants several obstacles previously identified in the literature. Here the most selected obstacle was "too many resources" (28 of 122). This result justify our intention to offer other options of common obstacles that might have gone unnoticed in the open question; this obstacle was not mentioned in the open question, but in the closed section it was the most selected option. The second most selected obstacle was the difficulty to get customers/users to participate in usability evaluations (23 of 122). These results are closely related to the first obstacle detected in the open question (users' behaviour or problems). In third position we found two obstacles: my software does not have any problems (17 of 122) and no usability problems (16 of 122). These obstacles, which are connected each other, show that for a considerable number of the novice developers, their software does not have problems.

3.4 Advantages in conducting usability evaluations

We applied same combination of open/closed questions to identify perceived advantages to the application of usability evaluation according to the novice developers. The total number of advantages mentioned was 79. The primary advantage mentioned by the respondents was an increase of quality in the software (26 of 79) for example "Allows for fixing problems that could become more serious if they are not repaired on time". In this case, it is possible to reconfirm the participants' perspective about the relevance of quality (see section 3.1). The second most mentioned advantage is a guarantee benefit of usability evaluations: it improves the software development method. 24 of 79 responses were related to this advantage, e.g. "Creation of a system that will be controlled by and adapted to the enterprise processes in an easy way". According these comments, novice software developers seem at usability evaluations as a way of to identify any potential usability problems and incidentally, improve other relevant aspects of the software development process. Other advantages cited by

participants were users' satisfaction (16 of 79), improve the design of the software system (6 of 79), and professional growth (5 of 79).

The closed question, which contained several options of commonly accepted advantages, generated results closely related to the previous ones. The two primary advantages were "increase user satisfaction" and "quality improvement". They were widely selected (68 and 61 respectively of 211). The third most selected advantage was "increase competitiveness" which was selected by 33 participants. Another relevant advantage is to increase competences, which was selected by 31 respondents. In this case, novice software developers think that their participation in a usability evaluation could help them increase their professional competences. This is another new finding of this study.

4. DISCUSSION

The aim of this study was to explore the perspectives of novice software developers on usability evaluation. We focussed on the perceived importance of usability for novice developers, on their understanding of the usability evaluation concept and on obstacles to and advantages of conducting usability evaluations.

Concerning the importance of usability, our study showed that 39% of the novice developers perceive usability topics as being more important than software development topics. Given the situation with a lack of "usability culture" and the perceived obstacles to applying usability evaluations, it is interesting that more than one third of the novice developers still find usability most important. This indicates that usability has an impact on the mind-set of some of the novice software developers. In comparison, software development topics are considered more important by 61% of the respondents. Our results shows that sometimes usability activities are perceived as being more important than software development activities (see Table 1, pairs 1 and 2). By contrast, software development is more relevant than usability when contrasting usability activities against other quality activities (see Table 1, pairs 3, 4 and 5). In general, usability activities received relatively much emphasis; however, software quality is still the main focus for novice developers. This is particularly clear in concept pairs 4. Here, 86% of the participants considered software quality assurance as being more relevant than human-computer interaction. Thus we conclude that although usability is perceived by novice developers as being important, quality in software is even more so.

Our findings also show that novice developers can define usability evaluation quite well, i.e. they

understand the concept of usability evaluation. This is clear from the considerable number of participants who provided definitions of usability evaluation using concepts such as evaluation, user and usability in their answers. Moreover, only a few respondents used concepts related to functionality or various kinds of technical or functional tests. This proper understanding of the concept of usability evaluation can explain why novice developers are highly convinced of the relevance of their participation in this kind of evaluation. The clear understanding of the concept of usability evaluation contradicts results found in other studies (Ardito et al., 2011; Bak et al., 2008; Seffah & Metzker, 2004; Rosenbaum & Rohn & Humburg, 2000). Even if we consider that those previous studies had been made with more experienced actors (mainly from software organizations), the novice developers' clarity on these concepts originate from the education programs they have followed. Nowadays, HCI topics are common in many software development curricula. Contrastingly, we found a low level of understanding of the HCI concept, which has also been reported in other studies (Rosenbaum & Rohn & Humburg, 2000; Ferre & Juristo & Moreno, 2006)

Our findings regarding the perceived obstacles to applying usability evaluation show that the main obstacle is the users' behaviour and other problems related with users. Confirming this, the novice developers consider that their software does not have usability problems. These results indicate a lack of "usability culture". On the surface, this contrasts our findings of a high level of understanding of the usability evaluation concept; but as noted by Rosenbaum & Rohn & Humburg (2000), a clear understanding of the usability concept is not enough to understand what usability evaluation implies. Yet Nielsen (2005) reports contradicting findings by showing that users are strongly engaged in the usability testing process. Our findings indicates a manifestation of a well-known obstacle which is the software developers' mind-set.(Ardito et al., 2011; Bak et al., 2008). Another obstacle identified by novice developers relates to the perceived high cost of usability evaluations, which is also found in other studies (Ardito et al., 2011; Bak et al., 2008; Rosenbaum & Rohn & Humburg, 2000).

In addition, some new obstacles are suggested in our study. This is particularly the case with design of the software and other technical and organizational problems, e.g. software bugs, lack in "usability culture", etc. In some way, this new finding contradicts the decoupling of usability from software engineering reported by Seffah & Metzker (2004); for novice developers there are an evident relation between usability and other software development activities. However, the concern of novice developers for software bugs, illustrates the importance they contribute to software quality at the cost of usability issues (See Table 1)

The ability of usability activities to help improving software quality is considered to be the main advantage. This result confirms the importance of software quality for novice developers. This is an expected result considering the general opinions related to the aims of the testing process; it is generally accepted that testing is performed, among other major aims, to evaluating product quality (IEEE Computer Society, 2004). In addition, improved user satisfaction is another advantage identified by novice developers. These advantages are supported in the study of Ardito et al. (2011). This particular opinion of novice developers, related to one of the most relevant advantages of usability evaluation, seems to contradict their own perspective about the main obstacle: the user. This also indicates the lack of "usability culture" among novice developers.

Extending the findings of Ardito et al. (2011) about advantages of usability evaluation, our study identifies two new advantages: it could improve the software development method and developers' participation in usability evaluations could allow them to increase their professional competences. Certainly, usability evaluation has a clear purpose in identifying usability problems that would otherwise affect the software usability negatively. This could be considered the major aim of usability evaluations. However, it is interesting that the novice developers emphasize other benefits related to software development. With this, the novice developers present themselves as persons who try to see beyond obvious and expected results of a particular process as, in this case, usability evaluations. The increased competencies of novice developers allow us to understand that these professionals have criteria to recognise the knowledge is important for their future careers.

5. CONCLUSION

This paper presented the perspective of novice software developers on usability evaluation. This included several elements such as the importance, meaning, obstacles and advantages of usability evaluations. We have contrasted our study with other previous studies, which emphasize organizational perspectives.

Our study showed that usability activities are considered important by more than one third of the novice developers. Compared to usability activities, software quality activities have a higher priority. Despite this, usability still appears to be important. Emphasis on usability activities could be increased, e.g. with training programs that diminish the lack of "usability culture" detected in this study. In contrast

to the lack of "usability culture", our results also show that novice developers have a clear understanding of what usability evaluation is as well as they an advanced ability to express obstacles and advantages. Our findings about obstacles and advantages are supported by other studies. In addition, we have also found new ones. The role of design in usability evaluation is noteworthy, something that is relevant for novice developers as an obstacle and also as an advantage.

In general, the novice software developers' perspective could be contradictory with the belief of their emphasis in implementing efficient code. Our conclusion is that both approaches usability and efficient codification seems to be relevant for novice developers such is showed in their vision about role of usability and software quality.

For future work we would like to study the potential synergies between usability evaluation and design activities in order to help in the coupling efforts of software engineering and HCI.

ACKNOWLEDGEMENTS

References

Ardito, C., Buono, P., Caivano, D., Costabile, M.F., Lanzilotti, R., Bruun A. and Stage, J. (2011). Usability Evaluation: a survey of software development organizations. In *Proceedings of 33 International Conference on Software Engineering & Knowledge Engineering*. Miami, FL, USA, 7-9 July 2011.

Bak, J.O., Nguten, K., Risgaard, P. and Stage, J. (2008). Obstacles to Usability Evaluation in Practice: A Survey of Software Development Organizations. In *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges*, Lund, Sweden, 20-22 October 2008. ACM, New York, NY, USA, 23-32.

Begel, A., and Simon, B.(2008). Novice software developers, all over again. In *Proceedings of the Fourth international Workshop on Computing Education Research (ICER '08)*, Sydney, Australia, 6-7 September 2008. ACM, New York, NY, USA, 3-14.

Berlin, L. M. (1993). Beyond program understanding: A look at programming expertise in industry. In *Empirical Studies of Programmers: Fifth Workshop*. Ablex Publishing Corp., 6–25.

Ferre, X., Juristo, N. and Moreno, A.M. (2006). *Obstacles for the Integration of HCI Practices into Software Engineering Development Processes*. In C. Ghaoui & Idea Group Reference (Eds.). Encyclopedia of Human Computer Interaction, 422-42.

Gulliksen, J., & Boivie, I., & Persson, J., & Hektor, A. and & Herulf, L.(2004). Making a difference: a survey of the usability profession in Sweden. In *Proceedings of the third Nordic conference on Human-computer interaction (NordiCHI '04)*. Tampere, Finland, 23-27 October 2004. ACM, New York, NY, USA, 207-215.

IEEE Computer Society (2004). *SWEBOK Guide to the Software Engineering Body of Knowledge*. Available from: www.swebok.org [01 May 2013]

Lindgaard, G., and Chattratichart, J. (2007). Usability Testing: What Have We Overlooked?. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. San Jose, CA, USA, April 28-May 3 2007. ACM, 2007. p. 1415-1424.

Lizano, F., Sandoval, M.M. and Garcia, M.A. (2008). Generación de una cultura organizacional en el aula: "un caso específico: la cultura hacia la calidad en el desarrollo del software, en la cátedra de Ingeniería de Sistemas. En *I Congreso Intern.Computación y Matemática.* UNA, Heredia, Costa Rica.

Nielsen, J., (2005) *Authentic Behavior in User Testing*. Jakob Nielsen's Alertbox. Available from: www.useit.com/alertbox/20050214.html [01-05-2013].

Roff, J. and Roff, K (2001). *Careers in E-Commerce Software Development.* The Rosen Publishing Group.

Rosenbaum, S., & Rohn, J.A., & Humburg, J. (2000). A toolkit for strategic usability: results from workshops, panels, and surveys. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '00),* The Hague, The Netherlands, April 1-6, 2000. ACM, New York. 337-344.

Seffah, A. and Metzker, E. (2004). The obstacles and myths of usability and software engineering. *Communications of the ACM*, 47(12), 71-76.

Strauss, A. and Corbin, J. (1998). *Basics of qualitative research. Techniques and Procedures for Developing Grounded Theory*, 2.edition. SAGE Publications.

Taft, D. K.(2007). Programming Grads Meet a Skills Gap in the Real World. *eWeek.com*. Available from: www.eweek.com/c/a/Application-Development/Programming-Grads-Meet-a-Skills-Gap-in-the-Real-World/ [01 May 2013].