2008

Usability in Open Source Software Development: An Interpretive Case Study

Netta Iivari
University of Oulu, netta.iivari@oulu.fi

USABILITY IN OPEN SOURCE SOFTWARE DEVELOPMENT – AN INTERPRETIVE CASE STUDY

Iivari, Netta, Department of Information Processing Science, University of Oulu, P.O Box 3000, 90014 Oulu, Finland, netta.iivari@oulu.fi

Abstract

This paper analyzes usability in open source software (OSS) development. OSS development is a current phenomenon both in Information Systems (IS) research and practice. Also the Human Computer Interaction (HCI) community has started to acknowledge OSS development as an interesting object of study. The importance of usability is accepted within the both fields of IS and HCI. Usability of the OSS tends to be poor, but usability is becoming a relevant topic of research also in the OSS context, even though, to date, it has not been examined much. Therefore, an interpretive case study is carried out to examine how usability is understood and dealt with in an OSS project. The results show that different kinds of meanings have been attached to usability in the project, and some action has also been taken to ensure it in the development. However, clear targets for improvement can also be identified. Implications both for theory and practice are discussed.

Keywords: Open Source Software Development, Usability, User Involvement, Interpretive Case Study

1 INTRODUCTION

This paper analyzes usability discussions in an open source software (OSS) development project. OSS development is selected as the object of analysis due to its current nature both in Information Systems (IS) and Human Computer Interaction (HCI) research (Benson et al 2004, Cetin et al 2007, Fitzgerald 2006, Frishberg et al 2002, Nichols & Twidale 2003, Niederman et al 2006, Zhao & Deek 2006). In OSS development source code needs to be 'available for anyone who wants to use or modify it' (Niederman et al 2006: 131). There are differences in the licensing agreements, however, due to which there actually is a 'continuum of openness' (Niederman et al 2006: 131).

Usability is an important quality characteristic of a software products and systems. Especially the field of HCI has also addressed the development of usable software products and systems in approaches such as UE (Usability Engineering) and UCD (User-Centered Design) (Kujala 2003, Mayhew 1999, Nielsen 1993, Rosson & Carroll 2002). Within the field of HCI, usability is typically defined as 'the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use' (ISO 9241 1998). Another widely cited definition defines usability to consist of learnability, efficiency, memorability, errors and satisfaction - it should be easy to learn to use the system, the use should be efficient, it should be easy to remember how to use it, the users should not make errors or they should easily recover from them, and the use should be satisfactory (Nielsen 1993).

In IS research, both usefulness and ease of use have been praised as important elements in technology adoption (Davis 1989) and diffusion (Rogers 1995). The Technology Acceptance Model, widely utilized in IS research, defines as relevant aspects in the acceptance process to be perceived usefulness – "the degree to which a person believes that using a particular system would enhance his or her job performance"; and perceived ease of use – "the degree to which a person believes that using a particular system would be free of effort" (Davis 1989: 320). The theory of diffusion of innovations is another widely used theory within the IS field. The perceived usefulness is very similar to the relative advantage construct, the perceived ease of use resembling very closely the complexity construct in the theory of diffusion of innovations (see Rogers 1995).

One can conclude that the importance of usability (as well as usefulness) is accepted within the both fields of IS and HCI, the field of HCI providing more practical guidance also for developing usability. Therefore, in this research effort, particularly HCI literature on usability is reviewed, and contrasted with the findings of this research. Related to OSS development, it has already been acknowledged that usability of OSS tends to be poor, but usability is becoming a relevant topic of research also in the OSS context (Andreasen et al. 2006, Benson et al 2004, Cetin et al 2007, Feller & Fitzgerald 2000, Nichols & Twidale 2003, Nichols & Twidale 2006, Twidale & Nichols 2005, Zhao & Deek 2005, Zhao & Deek 2006). In this research effort an interpretive case study on a usability oriented OSS project is carried out to understand how usability is understood and dealt with in OSS development.

The paper is organized as follows. The next section reviews usability and OSS development literatures. The third section presents the research method utilized, the case involved in this study and the procedures of data gathering and analysis. The fourth section outlines the results of the empirical examination. The fifth section summarizes the results and discusses their implications.

2 LITERATURE ON USABILITY AND OSS DEVELOPMENT

2.1 Usability Literature

In the field of HCI, it is emphasized that that usability is always related to specific users, and to their specific goals and tasks that are done in a specific context of use. It is important to acknowledge that usability can be developed only after one understands and specifies who are the intended users and their work practices and tasks. In addition, one needs to understand and specify the intended context of use and its effects. (ISO 13407 1999.) Based on the understanding, afterwards, one needs to carefully redesign the work practices and tasks. After redesign, interaction solutions are to be produced. Finally, usability evaluation is an essential step that should be started early in order that the results affect the design and the changes are not expensive (ISO 13407 1999). Different kinds of Usability Engineering (UE) and User-Centered Design (UCD) methodologies (e.g. Mayhew 1999, Nielsen 1993, Rosson & Carroll 2002) offer specific methods and techniques for developing and ensuring usability. Many differences can be identified in them (se e.g. Iivari & Iivari 2006), but despite that they all emphasize the importance of gaining a detailed understanding the user, his/her tasks and work practices, and the context of use as a basis, careful redesign of the tasks or work practices based on that understanding, gathering of user feedback, and iterating the design solution based on the user feedback. To carry out these activities, the literature suggests hiring a group of experts, variably labeled as usability, UCD or UE specialists. (Iivari 2006, Mayhew 1999, Nielsen 1993, Rosson & Carroll 2002.)

Active user involvement is essential when developing usability (e.g. Iivari 2006, ISO 13407 1999, Rosson & Carroll 2002). In order to understand the users, their work practices and tasks and the context of use, one needs to be in contact with the users. When evaluating the developed solution, representative users should take part. However, user involvement can be of different types. It can be classified as informative, consultative or participative (Damodaran 1996): in informative type users act only as providers of information and as objects of observation, in consultative type they are allowed to comment on predefined design solutions, while in participative type they actively participate in the design process and have decision-making power regarding the design solution (Damodaran 1996). Especially IS tradition has advocated user involvement of participative type, while in HCI literature, where usability specialists are expected to carry out the usability activities, users role tends to be informative or consultative at the most (Iivari 2006). In this situation, it is interesting to consider the role of the usability specialists; it can also be classified as informative, consultative or participative in the similar sense as is the case with the role of users (cf. Damodaran 1996). It might be that these usability specialists are only allowed to act as providers of information or as commentators of predefined design solutions. However, they might also be allowed to be in participative role actively taking part in the design process having decision-making power regarding the design solution (cf. Damodaran 1996). In this paper, all these types and situations are acknowledged.

Existing literature has revealed that very divergent and even surprising meanings have been attached to usability in practice; usability can be used only as a buzzword or slogan without any proper understanding of it (e.g. Bloomer & Croft 1997, Iivari 2006). It might be difficult to get usability accepted in the development, since the management may view it as a threat to budgets and schedules and the developers may perceive it to mainly increase the development time (Bloomer & Croft 1997). It is problematic that usability methods might be applied in isolation without having any effects on the solution. The usability specialists might also be viewed as police and their findings as unwelcome. (Aucella 1997, Iivari 2006.) In all, one usability activities have proven to be quite challenging in the development, and many different meanings can be attached to usability in practice.

2.2 Usability and Open Source Software Development Literature

In OSS development the developers typically produce software for themselves. For that reason, however, OSS development has been argued of utilizing a truly ‘user-driven’ approach (Nichols & Twidale 2006, Zhao & Deek 2006) and user involvement has been emphasized as an important element of OSS development (e.g. Feller & Fitzgerald 2000, Nichols & Twidale 2003, Nichols & Twidale 2006, Zhao & Deek 2005). However, it has already been noticed that in OSS development the distinction between user and developer is blurred (Zhao & Deek 2005). The OSS community is now starting to acknowledge that ‘we are not our users’ (Frishberg et al 2002). People are now acknowledging that from the point of view of naïve, non technical, non computer professional users, usability of the OSS is poor, and the development process anything but 'user centered' (Andreasen et al. 2006, Benson et al 2004, Bødker et al 2007, Cetin et al 2007, Feller & Fitzgerald 2000, Nichols & Twidale 2003, Nichols & Twidale 2006, Twidale & Nichols 2005, Zhao & Deek 2005, Zhao & Deek 2006). Therefore, usability is becoming a relevant topic of research in the OSS context, even though, to date, it has not been examined much (Andreasen et al. 2006, Benson et al 2004, Cetin et al 2007, Nichols & Twidale 2003, Nichols & Twidale 2006, Zhao & Deek 2005).

The existing literature already reveals many issues, which cause problems for usability in OSS development. Altogether, the developers do not have knowledge about the non technical users, their tasks and the contexts of use. In addition, communicating usability problems to the developers has proven to be challenging. Furthermore, no usability methodology is typically employed, because this type of approach can be seen as being in contrast with the 'open source philosophy'; it is assumed that in OSS development there is no possibility for systematic methodologies or formal process models. (Benson et al 2004, Bødker et al 2007, Cetin et al 2007, Nichols & Twidale 2003, Nichols & Twidale 2006, Twidale & Nichols 2005, Zhao & Deek 2005, Zhao & Deek 2006.)

Some attention to the non technical users has been paid also in OSS development. Extensive gathering of user feedback and bug reporting have been highlighted. However, it has also been reported that non technical users may be intimidated or incapable to use the bug reporting systems. (Nichols & Twidale 2003, Nichols & Twidale 2006, Zhao & Deek 2005, Zhao & Deek 2006) The field of HCI, furthermore, emphasizes trained usability specialists to contribute to the development. Related to that, it is problematic that usability specialists do not typically participate in OSS development, and the OSS developers do not normally have the knowledge and skills needed. The OSS developers do not necessarily even know that there is a HCI community and usability specialists, who could contribute to the development. Furthermore, the users can not act as usability specialists, because they are not trained for developing and ensuring usability, even thought they encounter the usability problems while using the system. (Benson et al 2004, Cetin et al 2007, Frishberg et al. 2002, Nichols & Twidale 2003, Nichols & Twidale 2006, Twidale & Nichols 2005, Zhao & Deek 2005, Zhao & Deek 2006).

Particularly three pieces of research are very relevant in relation to the topic of this paper. First, an empirical study of OSS developers' understanding of usability has been carried out by utilizing a questionnaire and interviews. Data from OSS development but also from a company providing usability support to OSS development was gathered (Andreasen et al. 2006). The results show that the OSS developers viewed usability as highly important, but they had a rather limited understanding of it. Usability was perceived as an add-on property. Usability specialists were given only an advisor position in the development; the developers were reluctant to give them decision making power. However, usability evaluations were deemed as useful. (Andreasen et al. 2006.) This paper concentrates on the understandings of usability and on how it is dealt with in practice by analysis of the discussions in a usability discussion forum, not by asking the developers their opinions about it. In addition, the focus is also on usability activities other than usability evaluation.

Second, in an action research project on OSS development, researchers have carried out context of use analysis, set up a HCI discussion list and developed a list of heuristics to be used in OSS development (Bødker et al 2007). Related to the HCI discussion list, a problem was that the discussions did not have effects on the solution; no decision making was done in the list (Bødker et al 2007). The authors also argue that OSS ideology and discourse emphasize that OSS developers do not need to care about the end users and that OSS culture is about doing, not about analyzing/understanding (Bødker et al 2007). However, the HCI discussion list contributed by making usability a topic of discussion, and by making the developers realize that there are also non technical users (Bødker et al 2007). This paper contributes by proving data of an OSS community initiated usability discussions; i.e. no researcher intervention has been involved in this case. Instead, the developers of this OSS project have become interested in this issue by themselves. However, a research intervention could have been useful also in this project. This will be discussed further in the discussion part of this paper.

Furthermore, another interesting research effort has examined usability discussions in bug reporting discussion lists, focusing on discussions explicitly mentioning usability (or HCI or interface) (Twidale & Nichols 2005). The researchers note a multitude of challenges related to usability in OSS development. First of all, they argue that usability specialists try to gain legitimacy in OSS development, but many times there are very few of them and they tend to be isolated and overcrowded by the developers in the discussion lists (Twidale & Nichols 2005). In addition, usability bugs might be very complicated to fix, they may be subjective, they may be very difficult to explain textually, and there is a need of many trade-offs, analyses, etc. before reaching any solution for fixing them (Twidale & Nichols 2005). Finally, the authors state that discussion lists, altogether, are not enough for usability discussions, but instead some sort of design area for brainstorming and discussing user studies, HCI literature, design guidelines etc. would be useful (Twidale & Nichols 2005). This paper contributes by proving data of usability dedicated discussion list, focusing on what is understood to be 'usability', altogether, in the OSS development. The focus is not limited to messages mentioning 'usability', but instead all messages posted to the usability discussion list are included. The focus is on meanings attached to usability, instead of trying to understand what a 'usability bug' is and how to fix it.

3 RESEARCH DESIGN

In this research effort an interpretive case study method was utilized. Interpretive case studies attempt to understand and make sense of the world, not to explain in the predictive sense, which is the aim of case studies of positivist nature. In interpretive case studies, theories are used only as sensitizing devices; the aim is not to test or falsify them. The meanings attached to the phenomenon studied are focused upon. Qualitative data is gathered, and the active role of the researcher as an interpreter of the data is emphasized. The aim is to understand the phenomenon from the viewpoint of the people involved in the activities analyzed, to produce thick descriptions, and to gain thorough understandings of particular cases. (cf. e.g. Denzin & Lincoln 2000, Klein & Myers 1999.)

The case involved in this research is an OSS development project developing an OSS solution to be used by people for entertainment purposes. The desktop application does not necessitate specific knowledge or education of its users. One could assume that people without any technical knowledge or programming skills could be interested in using the solution. The project has also shown clear interest in developing usability of their solution, them e.g. being listed in a website of OSS projects asking for usability support from usability specialists. In addition, a usability discussion forum was established over three years ago in the project website, asking the users of the OSS: "We would like to invite our users to participate in further improving the program. Do you have a suggestion how usability could be refined? Is there some peculiarity in the GUI that really annoys you? This is the place to discuss!" (Developer). Over 1600 posts and nearly 400 topics had emerged in this discussion forum during the time of data gathering. In addition, this OSS project is an active one with around 20 000 posts and several thousand members actively contributing to different forums in the project's website. In OSS projects, generally, interaction among the community members relies very heavily on web based communication tools, i.e. on discussion forums, IRC, news, FAQs, how-to guides etc. (Lakhani & von Hippel 2003, Ye & Kishida 2003, Scacchi 2002). The website of this OSS project provides the source code of the OSS in question, a user manual, information related to mailing list and IRC, and six discussion forums for different subject matters (e.g. for bug reporting, feature requests, development discussions etc.). In five of the forums, messages had been posted within the past 24 hours. In all, this OSS project is an example of an active, usability oriented OSS project.

All the posts of the usability discussion forum were saved and printed out for the analysis purposes. In interpretive research one seeks emic meanings – meanings held by the people within the case studied. The ideal in interpretive research is that there is no theory under consideration during the data gathering and analysis. This, however, is impossible, since researchers' prior knowledge and assumptions always shape the investigation. Actually, it is argued that prejudice, prejudgment, and prior knowledge are a necessary starting point for our interpretations. Nevertheless, one should try to avoid the restriction of the data gathering and analysis by the existing theories, since they may bias and limit the researcher's focus in the first place and then inevitably also the findings. Altogether, the researcher should try to be aware of his/her a priori assumptions. Interpretive case studies involve both inductive (data-driven) and deductive (theory-driven) reasoning, with the emphasis, nevertheless, on the former. (Denzin & Lincoln 2000, Klein & Myers 1999, Walsham 1995.)

Generalizations and abstractions are aimed at also in interpretive research. It is argued that the particularities in the empirical material should be related to more abstract categories, and that the 'unique instances' observed in the cases are to be related to the 'ideas and concepts that apply to multiple situations'. (Klein & Myers 1999: 75.) Generalizations in interpretive research can be of different types, but generally one can state that the 'generative mechanisms' identified in the interpretive case studies are 'tendencies,' implying that they are explanations, but not predictive ones. On the whole, the generalizations from interpretive case studies are to be viewed as 'explanations of particular phenomena derived from empirical interpretive research in specific IS settings, which may be valuable in the future in other organizations and contexts.' (Walsham 1995: 79.)

The data analysis proceeded as follows. All the posts of the usability discussion forum were printed and read through by the researcher. Related to each topic and post in it, the sender of the message was documented as well as all the repliers to form an understanding of the active members of this OSS community in this discussion forum. Regarding the content of the messages, it was soon realized that it varied significantly. While continuing the reading, it became apparent, however, that quite large proportion of the posts could be assigned to few categories that inductively emerged during the data analysis. Afterwards, a more detailed analysis regarding the meanings attached to 'usability' was carried out, focusing on two aspects: what is meant by usability and how it is dealt with.

4 EMPIRICAL ILLUSTRATIONS

4.1 Introducing the Usability Discussion Forum

First some descriptive data of all the messages and message senders (including both OSS developers and OSS users) of the analyzed OSS usability forum is offered in table 1.

Message senders	**Number**	**Percentage**
Newbies	282	48.70 %
Guests	274	47.32 %
Members (inc. developers)	23	3.97 %
Developers	2	0.12 %
Guests' or newbies' messages	1102	67.70 %
Members' messages (inc. developers)	526	32.30 %
Developers' messages	192	11,79 %
Content		
Topics	399	
Messages	1628	
Feature requests related to the OSS	499	30.65 %
Problems related to the OSS	300	18.42 %
Helping	177	10.87 %
Thanking the developers and praising the OSS	113	6.94 %

Table 1. The message senders and the message contents in the usability discussion forum

Most of the message senders are labeled as 'newbies' or 'guests', meaning that they are beginners or just visiting the site. Later on, these people can become (junior, senior, full) members, and in the OSS context, also OSS developers. However, in this forum, only two people were defined as OSS developers, some of the members also having positions as ‘administrators’ or ‘moderators’. Interesting to notice is that even though most of the messages have been posted by the ‘newbies’ and ‘guests’, the members and especially the developers have been very active posters compared to their number. Each member has posted avg. 23 messages, and each developer even avg. 96 messages to this forum, while the ‘newbies’ and ‘guests’ only avg. 2 messages. However, those newbies and guests, who have participated in more than one topic, have posted avg. 7 messages each to the discussion forum. Most of the topics have been initiated by the ‘newbies’ and guests’, the members replying to them.

Regarding content, as mentioned, the usability forum begins with the invitation from one of the developers to users to participate in further improving the program. Usability and user interface (UI) improvements and problems are explicitly mentioned. Therefore, one could argue that the messages posted to this forum all deal with UI and usability issues. However, usability was explicitly mentioned only in 56 messages out of 1628. Furthermore, concepts closely related to usability (easy to use, easy to understand, efficient to use, user friendly, intuitive, consistent, accessible, ergonomic, user experience) were mentioned in additional 27 messages. These messages were analyzed in detail. However, as mentioned, all the messages were included in the analysis and their content was categorized through inductive analysis. Data driven reasoning led us to identify four categories into which a large proportion of content of the messages fitted nicely. The categories are: problems, feature requests, thanking, and helping, which are briefly described next.

In the feature request category, naturally, different kinds of features were requested. Interestingly, only a couple of times were the senders asked to post their messages to the feature request or bug forums, which were designated places for this kind of posts. The features requested varied from appearance issues (requesting certain kinds of tabs, bars, buttons etc. in certain places, or certain kinds of colors, themes, etc.) to behavior issues (possibilities to do certain kinds of things or suggestions how certain features should behave) and integration issues (integrating the OSS with different kinds of software and hardware). The possibilities to have certain appearance or behavior as options or customizable were also requested numerous times. Sometimes a number of senders had backed up the first sender, aiming to convince the developers of the importance of a particular request.

In the problem category, different kinds of problems were expressed. The sender did not know how to use a particular feature or the OSS altogether, or he/she did not like a particular feature or the OSS altogether. Sometimes the OSS had behaved in an unexpected or unpleasant way. Other times the OSS lacked certain functionality needed by the users to accomplish certain tasks. Few times the sender provided a lengthy description of the situation and the context of use in which the problem occurred, as well as the different steps he/she had taken in trying to accomplish something.

The helping category was related to either problems or feature requests, the sender offering advice for the original sender. Either the original sender did not know how to use the OSS or the original sender was requiring some sort of functionality that was available somehow already in the OSS. The sender of the helping message described what the original sender should do in order to accomplish what he/she wished to accomplish. This type of action seems to be typical in OSS development. It has already been argued that OSS development is user-centered in this sense: the OSS users get user support and help in OSS discussion forums (Lakhani & von Hippel 2003).

Finally, the thanking category constantly grew and deserved its place as a category of its own. Many senders started or ended their message by thanking the developers and/or praising the OSS as the best one available. This might also be characteristics to the OSS discussion forums, since the OSS developers are not paid for their work, but instead their own needs and the reputation gained motivates them. The literature has indicated that a large user base is important socially and psychologically, and it motivates the developers, since it is a flattery and reward (Ye & Kishida 2003). The thanking messages were clearly sent to motivate the OSS developers.

4.2 Exploring Usability

Based on the previous section, one can conclude that usability is related to the following issues in this discussion forum. Usability discussions tend to be discussions, in which newbies and guests post their messages (questions, comments, suggestions) and more experienced members reply (help, comment) to them. Therefore, the discussion forum is used for describing how one can and is supposed to use the OSS (cf. Lakhani & von Hippel 2003). Furthermore, usability is related to different kinds of features of the OSS. Usability improvement is related to certain kind of appearance and behavior improvements suggested by the senders. Related to appearance issues, the senders recommend e.g. certain kind of tabs, buttons, icons and their places to improve usability. Related to behavior, on the other hand, different kinds of solutions and requirements e.g. for sorting, browsing, navigation, defining access rights, prediction of user behavior, mouse clicking and removing are requested. Some of them are totally new features, the others only extending or changing the current behavior.

Related to these requests, the senders make clear distinction between taste and usability. For example, a newbie requests to change the way information is currently organized, related to which a member replies that “I think that you should find [an OSS] that suits your taste, instead of trying to make a highly successful [OSS] conform to you.” Especially if another software was mentioned, the developers seemed to react quite strongly, and negatively, to the suggestion. However, the original sender continued that “Sorry if I've been misunderstood, but this isn’t about taste, it's about usability. This is just a remark that may be considered to enhance usability. Simpler, smaller and not so nested [information organizations] in [a view] may make it just a little bit easier to find what you need.” Therefore, it is emphasized that usability is not a matter of opinion, but it is related to facts concerning fast and easy use of the OSS.

Furthermore, it is also emphasized that usability is not related to nice appearance: “[Another software] has an awesome new feature called [a feature]. I'm sure at first most will write it off as simply more eye candy; however, I really like this new [feature]” (Newbie) “The one thing I really love about [the feature] is the possibility to [do something] by double-clicking. (…) It's just more "Real-Feeling" than searching (…) by scrolling through titles. (…) I think it's not just eye-candy-stuff, but it's definitely a cool usability-feature.” (Member) This comment also brings up an interesting issue of calling usability improvements as features: “The thing is, this is a usability feature - it means my usage of [a feature] can be adjusted to the assumptions (…) made by [a community]” (Newbie) “For me [the OSS] is a nice looking [software] with some nice features, but I am deinstalling it on a regular basis, because for me it lacks some basic usability features. Please make it possible to [do something] and to [do something] by using the cursor keys. (…) Right now (…) you can only assign keys to this function, which does not work for my situation.” (Newbie)

The last comment emphasizes also the problems reported by the senders related to their use of the OSS. Therefore, one can say that usability is clearly also related to the problems in using a particular feature or the OSS altogether, or to disliking a particular feature or the OSS altogether. Related to disliking, the language used is sometimes very harsh. Certain appearance and behavior issues are criticized to diminish or even sacrifice usability: a “glaring usability problem” is discussed in numerous messages. Also “using [the OSS] with a screen resolution of 1600x1200 renders this user interface quite unusable. It looks plain ugly, and is one of the worst things that ever happened to [the OSS]” (Guest) “I don’t like the way things are now arranged in [a version], I never used to need to scroll that much (…) I felt like downgrading immediately but I'll just not show [a panel], unless I REALLY need to because it looks rather ugly.”(Newbie) “The new placement for [a feature] is a mistake!!!!! Please go back to the previous version! 😕 Or let the users have a possibility to switch the layout. I want [different kind of appearance], it looks ugly! (Newbie) “Seriously, this was supposed to be the [another software] killer. The only problem is, when one installs [another software], (…), he [can do something], the UI is logically laid out, etc. [This OSS] is a hacked up pile of turd.” (Newbie)

Furthermore, intuitiveness and consistency of the UI come up several times when discussing usability. Numerous messages criticize the UI as unintuitive, since some action produces unexpected results. The users assume that the OSS behaves differently than it actually does. Consistency, on the other hand, is reported in relation to certain functions that behave differently in different places of the OSS. For example, one sender had found four different ways of deleting in the OSS, and sometimes this could be canceled but sometimes not. There seems to be agreement that consistency and intuitiveness are highly important issues from the viewpoint of usability.

Some of the messages report clear problems in understanding how to use the OSS. The senders or their friends have tried to use the OSS unsuccessfully and report their frustration. However, part of the messages deal with rather minor issues, arguing that "there are a few annoying interface quirks that make little sense to me (…) [The OSS] makes some rather silly and common mistakes like a lot of Linux apps." (Guest) On the other hand, some of the senders provide lengthy descriptions of the situation and the context of use in which the problem occurred, as well as of the different steps he/she had taken in trying to accomplish something, providing thus valuable data to the OSS development. Some of the senders also position themselves as users of the OSS, arguing them being either typical or novice users. It is explained how novice users can't find certain functionality or just don't know how to use the OSS. Friends and relatives are also mentioned several times as examples of people, who should be able, but currently can't use this OSS. In addition, it is acknowledged that "there are three types of computer users in the world. The first is by far the majority. The first doesn't understand how the software in front of them works and doesn't want to. They just want it to work. For them, everything needs to be spelled out. This is the group that you will never hear from on forums like these and in bug/wish list reports." (Guest) "Well, it is the type 1 user that we keep in mind." (Developer)

In all, the importance of usability is no doubt accepted in this OSS community, since the discussion forum altogether has been established and is still active. It is also emphasized that: "Happy users make coders happy 😉" (Developer) The developers also seem to wish usability specialists to take part in the development. Several messages mentioning usability specialists have been posted: "I'm not a usability expert, so I am not qualified to respond to the rest of your post" (Member), "[A button] is likely to stay unless some usability expert can suggest something better." (Member) "Using this approach, some problems appear. I am no usability expert (rather the opposite). (…) So yeah, my solution isn't the best." (Member) However, there are no usability specialists contributing in this discussion forum, or at least they do not identify themselves as such in this discussion forum.

Regarding decision making related to fixing the usability problems and improving usability, 16 times the developers, after a while they have received a message reporting a problem or feature request, report of having fixed the problem already. In addition, in a number of situations the replyers report that the requested feature is already available somehow in the OSS, the reported problem can be avoided by using the OSS in a certain way or that the feature will be included in some future version of the OSS. However, in a number of situations the action to be taken is negotiated. In some situations it is decided that the difference in the opinions can be solved by voting in the discussion forum. Many times the replies include messages, in which support is given to the original message. However, the developers can always just decide that nothing will be done. In certain situations the developers also refer to the issue that there might only be very few users wishing the feature, due to which it will not be implemented. However, many other arguments are also used, including the developers just not liking or prioritizing the feature, or the requested feature somehow conflicting with the OSS philosophy. 'Open source philosophy' argues for free software, gift giving and code quality, and opposes the world of commercial software development (see e.g. Bergquist & Ljungberg 2001, Steward & Gosain 2006). This type of discourse was clearly identifiable also in this discussion forum.

5 CONCLUDING DISCUSSION

Table 2 summarizes the results related to how usability is understood and dealt with in an OSS project.

Aspect	Empirical Example
Defining usability - Effectiveness, efficiency and satisfaction - Learnability, efficiency, memorability, errors and satisfaction	Usability is … - Effectiveness, efficiency, satisfaction, learnability, consistency and intuitiveness - Related both to appearance and behavior - Related both to usefulness and ease of use - Feature to be implemented
Developing usability - Understand the users, their tasks and the context of use - Redesign the tasks - Evaluate the solution - Involve the users (informative, consultative, participative) - Involve the usability specialists (informative, consultative, participative)	Usability is developed by … - Understanding the users, their tasks and the context of use through the discussion forum, but few users describe these issues and they resemble technical users, since they take part in a usability discussion forum in an OSS project web site - No explicit user task redesign - Evaluation through gathering (technical) user feedback - Involving the (technical) users in informative and consultative roles through the discussion forum - Calling for usability specialists to contribute in informative or consultative roles through the discussion forum

Table 2. Understandings of usability and actions taken to develop it in an OSS project.

Related to the definitions of usability, nobody in the discussion forum explicitly provided it. However, issues related to achieving ones goals (effectiveness), achieving them fast and easy (efficiency), liking or enjoying the use of the OSS (satisfaction) and being able to use the OSS from the viewpoint of novice users (learnability) were compatible with the existing definitions of usability (cf. ISO 9241 1998, Nielsen 1993), and in accordance with the existing OSS research (Andreasen et al. 2006). In addition, consistency and intuitiveness were called for numerous times. Intuitiveness is related to learnability and memorability (Nielsen 1993). Consistency is also an accepted characteristic of quality user interfaces. Interesting observation is the positioning of usability as a feature to be implemented. This view seems to position usability as a technical property of the system (cf. Andreasen et al. 2006). A related matter is the view of usability as functionality of the system (cf. Andreasen et al. 2006), evident also in this discussion forum. For OSS developers, it might be natural to view usability as a technical property or functionality to be implemented. However, clearly this is a limited understanding of usability, which might impede proper usability enhancements. In addition, it might cause confusion that the senders did not differentiate the constructs of usefulness and ease of use. However, this is no wonder; since it has been criticized that also in the HCI research there is a tendency to broaden the usability concept to cover also utility and usefulness (Iivari & Iivari 2006).

Related to the existing OSS research, also this study indicates that the developers do not have much knowledge of the 'non technical' users, their tasks and the contexts of use. The scarce knowledge they have, furthermore, seems to be based on the discussion forum, in which it is unlikely that the 'non technical' users ever participate. Therefore, it might be that also in this case the developers, even though realizing there are also 'non technical' users using their OSS, do not know much about them and their needs (cf. Bødker et al 2007). Related to the already identified lack of usability specialist involvement in OSS development, also this study provides evidence that the usability specialists do to participate in OSS projects, even though the project had requested it several times.

On the other hand, related to communicating usability problems to the developers, the message senders seemed to be quite capable of expressing themselves, even though one can suppose that some sort of design area for brainstorming and discussing user studies, HCI literature and design guidelines could also be of use (cf. Twidale & Nichols 2005). Researchers have argued that usability bugs tend to be very complicated to fix, difficult to explain textually, and there is a need of many trade-offs and analyses before reaching any solution for fixing them (Twidale & Nichols 2005). In this case, the problems and feature requests were not necessarily related only to usability, since clearly both usefulness and ease of use were addressed in these messages. However, many times the textual description was very informative. Sometimes the senders had also provided some kind of graphical information as addition to the textual description. However, clearly this necessitates a lot of effort from the user to provide this, due to which some sort of support could be of value.

Researchers have criticized that discussions in the HCI discussion forum do not necessarily have any effects on the solution. They also argue that OSS ideology and discourse emphasize that OSS developers do not need to care about the users. (Bødker et al 2007.) This case, however, contradicts these results. First of all, the OSS community clearly seemed to care about the users. In addition, decision making was done in the discussion forum, even though the OSS developers were the ones making the final decisions. This is characteristics to OSS development altogether: the technically capable and active 'core team' of OSS developers makes all the decisions regarding the OSS (Ye & Kishida 2003). However, based on the results of this study, one can argue that if the interest in usability is initiated by the community, not by outsider researchers, there might be no problems related to decision making or related to having effects on the final solution.

However, a related issue is the role given to users and usability specialists in the development. Only informative and consultative roles have been planned for the users and the usability specialists. It is assumed that they provide information and comment the existing solutions in the discussion forum. However, clearly an open question is that whether and how the usability specialists could be allowed a participative role, actively taking part in the design process having decision-making power regarding the solution. Interesting data related to that could be gathered, if usability specialists started to participate in the discussion forum. Research intervention could be organized related to this, and one could try to test the hypothesis that in OSS development the usability specialists are isolated and overcrowded by the OSS developers in the discussion forums (Twidale & Nichols 2005). The current results seem actually to contradict this, since the usability specialists seem to have a lot of respect and authority. However, currently they do not yet participate in the development. A related issue is active, 'non technical' user involvement, which was lacking in this case. An interesting path for future work would be to examine how more active, 'non technical' user involvement could be combined with OSS development. This study provides evidence of one, limited sort of end user involvement in OSS development, since the users taking part probably do not represent the 'non technical users'. However, this study provides evidence that usability is dealt with and users involved in OSS projects, but without commonly acknowledged terminology and methodology. In the future, more empirical data related to usability and user involvement in OSS development should be gathered, also by interviewing the OSS developers to gain an understanding from these 'natives'' point of view.

References

Andreasen, M., Nielsen, H., Schrøder, S. & Stage, J. (2006). Usability in Open Source Software Development: Opinions and Practice. Information Technology and Control 25(3A): 303-312.

Aucella, A. (1997). Ensuring Success with Usability Engineering. Interactions 4(3): 19-22.

Benson, C., Müller-Prove, M., Mzourek, J. (2004). Professional usability in open source projects: GNOME, OpenOffice.org, NetBeans. In Extended Abstracts of the Conference on Human Factors in Computer Systems, pp. 1083-1084, ACM Press, New York.

Bergquist, M. & Ljunberg, J. (2001). The power of gifts: organizing social relationships in open source communities. Information Systems Journal 11: 305-320.

Bloomer, S. & Croft, R. (1997). Pitching Usability to Your Organization. Interactions 4(6): 18-26.

Bødker, M., Nielsen, L. & Orngreen, R. (2007). Enabling User-Centered Design Processes in Open Source Communities. In Proc. Human Computer Interaction International, Part I: Usability and Internationalization (Aykin, N. Ed.), pp. 10-18, LNCS 4559.

Cetin, G., Verzulli, D. & Frings, S. (2007). An Analysis of Involvement of HCI Experts in Distributed Software Development: Practical Issues. In Proc. Human Computer Interaction International: Online Communities and Social Computing (Schuler, D. Ed.), pp. 32-40, LNCS 4564.

Damodaran, L. (1996). User involvement in the systems design process – a practical guide for users. Behaviour & Information Technology 15(16): 363-377.

Davis, F. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly 13(3): 319–340.

Denzin, N. & Lincoln, Y. (2000). Introduction: The Discipline and Practice of Qualitative Research. Handbook of Qualitative Research (Denzin, N. & Lincoln, Y. Eds.), pp. 1-29, 2nd Edition, Sage Publications Inc, Thousand Oaks.

Feller, J. & Fitzgerald, B. (2000). A Framework Analysis of the Open Source Development Paradigm. In Proc. 21st International Conference on Information Systems, pp. 58-69, December 10-13, Brisbane, Australia.

Fitzgerald, B. (2006). The Transformation of Open Source Software. MIS Quarterly 30(3): 587-598.

Frishberg, N., Dirks, A., Benson, C., Nickel, S. & Smith, S. (2002). Getting to know you: open source development meets usability. In Extended Abstracts of the Conference on Human Factors in Computer Systems, pp. 932-933, ACM Press, New York.

Iivari, N. (2006). Discourses on 'culture' and 'usability work' in software product development. Acta Universitatis Ouluensis, Series A, Scientiae rerum naturalium 457.

Iivari, J. & Iivari, N. (2006). Varieties of User-Centeredness. In Proc. 39th Annual Hawaii International Conference on System Sciences. IEEE.

ISO 13407 (1999). Human-centered design processes for interactive systems. International standard.

ISO 9241-11 (1998). Ergonomic requirements for office work with visual display terminals (VDT)s - Part 11 Guidance on usability. International standard.

Klein, H. & Myers, M. (1999). A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems. MIS Quarterly 23(1): 67-94.

Kujala, S. (2003). User involvement: a review of the benefits and challenges. Behaviour & Information Technology 22(1): 1-16.

Lakhani, K. & von Hippel, E. (2003). How Open Source Software Works: "Free" User-to-User Assistance. Research Policy 32(6): 923-943.

Mayhew, D. (1999). The Usability Engineering Lifecycle: A practitioner's handbook for user interface design. Morgan Kaufmann Publishers, San Francisco.

Nichols, D. & Twidale, M. (2003). The Usability of Open Source Software. First Monday 8(1), 21 pp.

Nichols, D. & Twidale, M. (2006). Usability Processes in Open Source Projects. Software Process Improvement and Practice 11: 149-162.

Niederman, F., Davis, A. Greiner, M., Wynn, D. & York, P. (2006). A Research Agenda for Studying Open Source I: A Multilevel Framework. Communication of the Association for Information Systems 18: 129-149.

Nielsen, J. (1993). Usability Engineering. Academic Press, Boston.

Rogers, E. (1995). Diffusion of Innovations. 5th Edition. Free Press, New York.

Rosson, M. & Carroll, J. (2002). Usability Engineering: Scenario-based Development of Human-Computer Interaction. Morgan-Kaufman, San Francisco.

Scacchi, W. (2002). Understanding the requirements for developing open source software systems. IEE Proceedings – Software 149(1): 24-39.

Steward, K. & Gosain, S. (2006). The Impact of Ideology on Effectiveness in Open Source Software Development Teams. MIS Quarterly 30(2): 291-314.

Twidale, M. & Nichols, D. (2005). Exploring Usability Discussions in Open Source Development. In Proc.38th Annual Hawaii International Conference on System Sciences. IEEE.

Walsham, G. (1995). Interpretive case studies in IS research: nature and method. European Journal of Information Systems 4: 74-81.

Ye, Y. & Kishida, K. (2003). Toward an Understanding of the Motivation of Open Source Software Developers. In Proc. 25th International Conference on Software Engineering, pp. 419-429, IEEE.

Zhao, L. & Deek, F. (2005). Improving Open Source Software Usability. In Proc. 11th Americas Conference on Information Systems, pp. 923-928, August 11-14, Omaha, USA.

Zhao, L. & Deek, F. (2006). Exploratory inspection: a learning model for improving open source software usability. In Extended Abstracts of the Conference on Human Factors in Computer Systems, pp. 1589-1594, ACM Press: New York.