



Open Source Usability and User Experience

Mikko Rajanen , University of Oulu

The purpose of this article is to highlight the challenges of user-centered and usability processes and methods in a open source software development context as well as to provide some answers for these challenges.

being 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.²

Usability has been addressed especially by the field of human–computer interaction, which has also introduced lots of methods and processes aimed at improving usability. Usability is an important quality characteristic of software, systems, and services, and it is vital for facilitating rich interactions between users and technology. Usability is defined as one of the main software product and system quality attributes in many international standards, such as ISO 9126. In this standard, *usability* refers to the ability of a product to be understood, learned, used by, and attractive to the user when used under specified conditions.¹ The second common definition for usability is in standard ISO 9241-11, where *usability* is defined as

USABILITY AND USER EXPERIENCES

The importance of usability has been acknowledged, for example, in information systems research, where the constructs of usefulness (the degree to which a person believes that using a particular system would enhance his or her performance) and ease of use (the degree to which a person believes that using a particular system would be free of effort) have been identified as essential factors for successful technology adoption. As a result of changes in the sociotechnical landscape, the concept of usability was further expanded in the early 2000s with the user experience aspect—at first, as the satisfaction component of the usability definition itself and, later, as its own distinct domain of design and evaluation to also cover the feelings and experiences of individual users (see, for example, Bevan et al.³). The relationships between user interface design, usability, user experience, and service design are presented in Figure 1.



Usability evaluation is a crucial part of usability methods. Usability evaluation should be started very early in the design and development process so that the results can truly be taken into account in the design and so that it will not be too expensive to make changes because the design will have not yet been developed too far. Finding issues only in the late part of the development process is problematic because making changes to a finished or almost finished system is too expensive and time-consuming. At the early stage of design and development, the role of usability evaluation is to gather qualitative feedback and requirements, whereas, in the later phases of the development, the focus should be on if these requirements are met. Therefore, next, we take a look at the benefits that better usability and user experience provide.

BENEFITS OF USABILITY AND USER EXPERIENCE

There are many benefits for all stakeholders from using user-centered design for the development process and aiming for better usability in both the traditional software development context and the open source software (OSS) development context.⁴ Users can benefit from better usability through higher productivity when the most frequent tasks take less time, and users make fewer errors. The development organization can benefit from better usability through a reduction in the time and resources needed for product development due to a reduced need for changes in later development phases because the usability methods help find needs for changes in requirements and design early on.⁵

In the OSS development context, good usability can help OSS communities get more nontechnical users and happier users because their software is easy to use and has a good reputation. Increased user satisfaction from

a good user interface and usability is an important factor for the image and also a competitive factor in the company's OSS development context. Furthermore, user-centered iterative and incremental development that aims at better usability also gives OSS projects more time for redesign and facilitates the involvement of nontechnical community members as active

issues in the most frequent tasks can cause frustration to peer-support members of the community.

CHALLENGES OF BRINGING USABILITY AND USER EXPERIENCE INTO AN OSS DEVELOPMENT CONTEXT

OSS has become widely used in everyday life, and the user base of OSS and

Finding issues only in the late part of the development process is problematic because making changes to a finished or almost finished system is too expensive and time-consuming.

participants in the design and development, providing feedback and providing redesign solutions in the project forums. This might also encourage nontechnical community members to provide peer support in the project forums, which would help both community-based OSS projects and company OSS projects.

Furthermore, better usability would likely decrease the need for users to contact the peer-support channels with usability-related issues, and there is less pressure for redesign. Repeated support requests related to usability

open source systems has expanded from technology-oriented developers into nontechnical users. This has caused more problems related to the level of usability of OSS because a software that is developed by technically oriented developers for their own use is likely not that easy to use from the perspective of a less technology- or development-oriented user, who compares the OSS to commercial software and expects similar levels of ease of learning and ease of use.

The status of usability and user experience is problematic in OSS

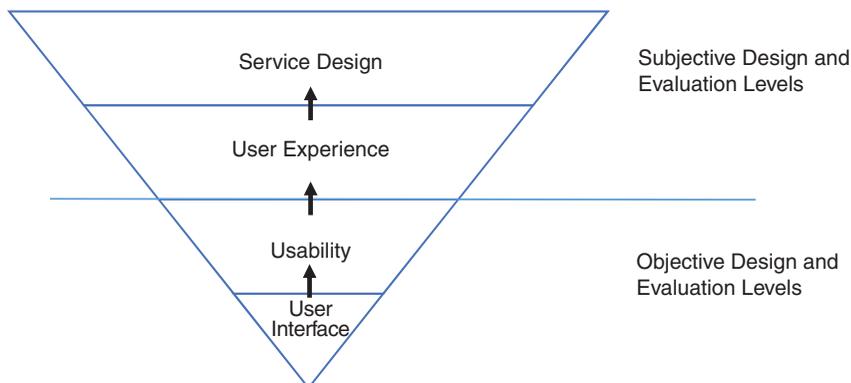


FIGURE 1. The relationships among user interface design, usability, user experience, and service design. (Adapted from Rajanen⁸; used with permission.)

FROM THE EDITOR

Open source programmers, as the saying goes, develop open source to scratch their own itch. Thus, many open source programmers are also users of their software and feel there is no need to involve product managers and usability experts. After all, that's them. Some of the time, this works. More often, it does not, and many users are wondering who came up with obviously inadequate user interfaces and workflows. In this article, Mikko Rajanen discusses how to integrate user experience design into open source software development, a long overdue addition. Let's hope more projects will pick up thoughts like this! Be happy, be healthy, be safe! —Dirk Riehle

development because most OSS projects do not know about the importance of usability, and, therefore, there are no usability experts helping. This is especially true of smaller OSS projects during their crucial first years, when most of the design is made and after which it is more and more difficult to make radical changes to the user interface, task flows, and the usability. Furthermore, OSS development originates from the needs of the developers themselves, who do not typically consider other users or their characteristics, mental models, and needs until the software has become popular among nontechnical users.

OSS just for their own use, to scratch their own itch, but also for a vast number of nontechnical users with limited skills in technology use. This nontechnical user base should be taken into account when developing OSS solutions so that OSS will continue its increase in popularity and use. Nontechnical users expect OSS to have the same level of ease to learn, ease to use, and intuitiveness as commercial software solutions. Now, most OSS users are these kinds of nontechnical users, who do not want to experience or report bugs or usability problems but just expect a certain level of usability and a good user experience.

This nontechnical user base should be taken into account when developing OSS solutions so that OSS will continue its increase in popularity and use.

The quality of an OSS system is mostly reliant on the experience of the developers and their ability to collaborate and make compromises because, contrary to closed source software development, OSS development is more free form and meritocratic.

The increased number of users due to successful OSS solutions has led the user base of OSS solutions to grow in ways that were not imaginable a couple of decades ago, when OSS was more on the fringe and used by few very technology-oriented hackers. Now, developers no longer design and develop

As the popularity of OSS solutions grows and the number of nontechnical users increases, so do the requirements and expectations for better user interface design, good usability, and good user experience. Nontechnical users are not typically involved with the development of the OSS, as they consider software to be tool that should fulfill their needs; they are not interested in participating in the development of this software tool, or they do not even know that it would be possible to do so. OSS developers do not usually reach out to these nontechnical users to

gather their requirements or to test design solutions. However, usability design and evaluation as well as user-centered design processes are rare among OSS projects. Often, OSS developers do not know the concept of usability; can mix it in with graphic design; and can consider usability as an “add-on” that could be easily added mechanically to the next software version, if so desired.

In OSS development, contributing something that the community views as valuable—usually software code—translates into prestige, which makes it possible to influence the design and the future direction of an OSS project. However, it has been shown to be difficult for usability and user experience experts to contribute and gain merit within the OSS development context. As OSS developers and communities do not know about usability, it is very difficult for usability experts to gain enough merit in the development community to have an impact. Often, any attempts by usability experts to provide their expertise for OSS projects and make the OSS easier for nontechnical users to use is met with indifference or even hostility—the developers claiming, for example, that usability is just a matter of taste, that usability is not relevant for OSS, or that “this OSS is not meant for girlfriends.”⁴ Contributions by usability experts have been ignored; they have been excluded outside of the community; and, even when usability experts have been successful in contributing, their contributions have been reverted by an individual developer.⁶

One issue that user interface designers and usability practitioners face is that developers may consider usability and user interface design to be largely a matter of subjective taste. Because of this view, developers might disregard even the absolutely objective usability measurements as merely opinions of usability practitioners and users, and they discard these measurements as not having objective validity in reality, no matter how much objective usability measurement data and



FIGURE 2. The phases of a usability expert contributing to an OSS project.

analysis are presented.⁷ Therefore, there have been calls to replace the subjective satisfaction component in the concept of usability with objective factors of harmony and symmetry of design as well as to have the subjective user satisfaction component in the concept of user experience, which is, by definition, subjective and based on the feelings and emotions of individual users.⁸

The decentralized and development-driven OSS context does not fit very well with user-centered usability processes. Building trust for usability experts and showing the merits of better usability are the key ways for usability experts to be able to effectively contribute to OSS projects; different strategies may be utilized, depending on the size and culture of the OSS project. Usability experts might try to establish their authority and trust by showing their competence in usability with facts and data or by trying to slowly integrate into the community through other means.

While usability should be an integral part of the development process from the very beginning, often, there is a lot of development already done before a usability expert gets involved with an OSS project. Task flows and functionalities have been set, and the user interface has been developed without design or testing with real users. Thus, it is very difficult for a usability expert to jump on an already moving train and fix the problems in the user interface, task flows, and functionalities. However, the skills and expertise of usability experts are needed in OSS development projects to ensure the continued success of OSS among nontechnical users. Therefore, we should take a

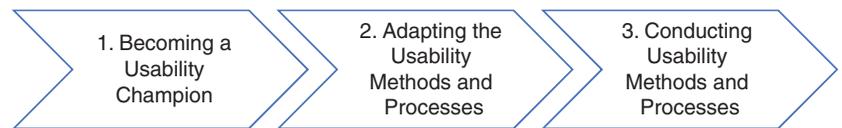


FIGURE 3. The phases of an OSS developer contributing usability to an OSS project.

look at the possible solutions of integrating usability experts into the OSS development context.

SOLUTIONS AND BEST PRACTICES FOR INTEGRATING USABILITY AND USER EXPERIENCE INTO OSS DEVELOPMENT

While OSS projects are generally very different from each other with regard to their size, culture, ways of working, organizational structure, and goals, there are some general solutions and best practices for integrating

To improve the status of usability in OSS, OSS projects should actively seek persons who have the skills and motivation to improve usability. Software developers do not typically have these required skills or motivation, as they have a more technological outlook, and they concentrate on improving the software behind the scenes. One of the greatest challenges to be solved is how usability experts can gain merit and become recognized within OSS communities.

The phases of a usability expert contributing to an OSS project are

One of the greatest challenges to be solved is how usability experts can gain merit and become recognized within OSS communities.

usability processes and methods into OSS projects. OSS projects could actively seek nontechnical contributions from the very start of project creation. Finding experts who could do user interface design and development; usability design and evaluation; and graphical assets, such as icons and visual design, would be very important for small and starting OSS development projects. There could be a forum or social media platform where OSS projects could pitch their design visions and try to attract nontechnical contributors, especially usability experts.

presented in Figure 2. Each of these phases presents a challenge for the usability expert if there is no support from the OSS project community—for example, in the form of a usability champion among the developers.⁹ OSS communities need to understand how usability experts can help to increase the popularity and reputation of their OSS among nontechnical users, how developers should learn about the importance of usability, and how important it is to design and evaluate user interface designs and task flows as well as get feedback from nontechnical

users. This is a challenge, especially in small- and medium-sized OSS projects, where developers might not have any idea what usability is. Also, OSS is typically developed from small, separately developed modules, so it can be difficult to outline an overall uniform design among all modules.

Furthermore, usability design and evaluation are relatively easy and straightforward to learn and do, and observing real end users interacting with the system is an eye-opening experience that reveals problems at both the user interface and the task flow level very fast. Therefore, OSS projects not having usability experts involved with their project would do well to organize even informal tests where they could observe nontechnical typical users trying out the most common tasks with their OSS.

While, ideally, there would be usability experts available and welcomed to each OSS project, there are not enough usability experts to contribute to all of the numerous OSS projects. One solution would be for more usability- and user-oriented OSS developers to step in, become usability champions for the OSS projects they are already contributing to, and adapt usability methods and processes to the context and ways of working of the particular OSS project they know very well. While the usability expert would have to gain enough merit one way or another in an OSS project to actually have an impact, learn about the culture and customs of that OSS project, and assess how usability methods and processes would have to be adapted to this context, the OSS developer likely already has merit and is recognized within the project community. Furthermore, this OSS developer is already familiar with the culture and customs and, therefore, could adapt the usability methods and processes on the go. Figure 3 outlines the phases of an OSS developer contributing usability to an OSS project.

In distributed and collaborative environments, such as an OSS project,

good usability is enabled through five important factors:

- ▶ a collaborative environment and suitable tools to enhance communication between project stakeholders, including nontechnical users and contributors
- ▶ an established decision-making process that helps when communicating with nontechnical users and contributors, such as usability experts
- ▶ a good and trust-based relationship among developers, users, and usability experts
- ▶ a usability expert who knows the user-centered design process and can apply it from the very start of a new OSS project
- ▶ usability champions among the developers, who know the importance of good usability to the success of the OSS project from the very beginning.

Users expect similar levels of usability and user experience from OSS as they have experienced using commercial software, and better usability and user experience are crucial for increasing the popularity of open source solutions, so open source developers and communities as well as researchers and practitioners should find new ways of bringing better usability methods and practices into the OSS development context. 

REFERENCES

1. *Software Product Evaluation—Quality Characteristics and Guidelines for the User*, ISO/IEC 9126, International Organization for Standardization, Geneva, Switzerland, 2001.
2. *Ergonomics Requirements for Office With Visual Display Terminals (VDTs)*, ISO 9241, International Organization for Standardization, Geneva, Switzerland, 1992.
3. N. Bevan, J. Carter, and S. Harker, “ISO 9241-11 revised: What have we

learnt about usability since 1998?” in *Proc. Int. Conf. Hum.-Comput. Interact.*, Cham, Switzerland: Springer-Verlag, Aug. 2015, pp. 143–151, doi: 10.1007/978-3-319-20901-2_13.

4. M. Rajanen, “Applying usability cost-benefit analysis - Explorations in commercial and open source software development contexts,” Ph.D. dissertation, Univ. of Oulu, Oulu, Finland, 2011.
5. M. Rajanen, “Different approaches to usability cost-benefit analysis,” in *Proc. Eur. Conf. Inf. Technol. Eval.*, 2006, pp. 391–397.
6. M. Rajanen, N. Iivari, and A. Lanamäki, “Non-response, social exclusion, and false acceptance: Gatekeeping tactics and usability work in free-libre open source software development,” in *Proc. IFIP Conf. Hum.-Comput. Interact.*, Cham, Switzerland: Springer-Verlag, 2015, pp. 9–26, doi: 10.1007/978-3-319-22698-9_2.
7. M. Rajanen and N. Iivari, “Power, empowerment and open source usability,” in *Proc. ACM SIGCHI Annu. Conf. Hum. Factors Comput. Syst. (CHI)*, Seoul, South Korea, 2015, pp. 3413–3422, doi: 10.1145/2702123.2702441.
8. M. Rajanen, “De gustibus non est disputandum, but usability is not a matter of taste,” in *Proc 7th Int. Workshop Socio-Tech. Perspective IS Develop.*, 2021, pp. 189–197.
9. M. Rajanen, N. Iivari, and E. Keskitalo, “Introducing usability activities into open source software development projects: A participative approach,” in *Proc. 7th Nordic Conf. Hum.-Comput. Interact., Making Sense Through Des.*, 2012, pp. 683–692, doi: 10.1145/2399016.2399120.

MIKKO RAJANEN is a postdoctoral researcher and university lecturer at the University of Oulu, 90014 Oulu, Finland. Contact him at Mikko.Rajanen@oulu.fi.