

Open UX design practice: the case of Firefox

Paula M. Bach and John M. Carroll

College of IST, Penn State University

University Park, PA 16802

(pbach, jcarroll}@ist.psu.edu

ABSTRACT

We describe open UX design using the Firefox web browser as a case study. We analyze the social complexity of integrating UX practices into an open source project using activity awareness, a framework for describing team performance. Through this analysis we found how Firefox UX practitioners use UX theories and methodologies.

Author Keywords

User experience, practice, open source, FLOSS, activity awareness, Firefox.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

Open UX design practice refers to UX design in free/libre/open source (FLOSS) contexts. FLOSS projects are characteristically managed on the web and therefore much of the process is open to the public. Not many open source projects engage in UX design, however, some projects do have particular UX strategies in place. For example the Firefox web browser employs UX practitioners.

Open UX design is a recent phenomenon. FLOSS development is clearly different from traditional software development approaches, thus, it is a question whether and how existing UX theories and methods apply to open UX. FLOSS developers find bugs, submit features, write code, review code, and coordinate code integration, in fast iterations that are released often [4]. Because their work is distributed over the Internet, they use email, web forums, and inter-relay chat (IRC) for communication. Developer work is merit-based and developers who are highly skilled and knowledgeable hold leadership positions and make decisions while they gain trust from other developers [6]. Any developer can find a FLOSS project to work on as long as she adheres to the project's social and technical structures. The social and technical structures make integrating traditional UX theories and methods challenging.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2009, April 3–9, 2009, Boston, MA, USA.

Copyright 2009 ACM 978-1-60558-246-7/08/04...\$5.00

We investigate the Firefox web browser to understand how UX theories and methodologies integrate with a FLOSS development environment. To frame our analysis, we use a theory that describes team performance using four facets required for information sharing in collaborative activities.

ACTIVITY AWARENESS

Activity awareness is a theoretical framework used for analyzing and understanding coordinated team performance [1]. Because UX practitioners often have to coordinate activities among various stakeholders (e.g. users, developers, and managers), social interactions can be challenging and particularly in FLOSS environments where UX activities are unfamiliar. In the highly distributed FLOSS development environment, activity awareness provides facets that aid in capturing the dynamics of social interaction. Each of the facets provides insight into how FLOSS UX practitioners bring theory and methods to the coordinated efforts of FLOSS development. The four facets of activity awareness are common ground, communities of practice, social capital, and human development.

Common ground

Common ground is a communication protocol for checking and indicating shared knowledge and beliefs. This technique is particularly critical for multidisciplinary teams with differing knowledge sets and disciplinary perspectives. In addition, distributed groups have to continually work at and monitor common ground; they cannot ever take it for granted the way that face-to-face teams sometimes can.

Communities of practice

Communities enact activities that they share through practice. These activities are specific to the community members and also share a tacit understanding of how to participate in the community. Developers wishing to join an open source community must understand how members enact activities and figure out social practices. This poses problems for UX designers in open source communities, because sharing practices with developers involves a process of enculturation, learning a rich set of moves and expectations, a variety of signals that members may not even be able to readily articulate but which they regularly and fluently enact. When UX practitioners join a FLOSS community of practice they must achieve a high level of awareness – they must know and recognize they can do the same things the other members do.

Social capital

Complex teamwork requires successful interactions. When continued beneficial interactions build trust among team members and other networks toward a social good, teams weather adversity. These favorable interactions toward a persistent social good build social capital. Open source developers build networks of social capital to help them solve problems. UX designers have a tougher time engaging in open source projects because building trust, social networks, and beneficial interactions with developers can be challenging.

Human development

When people engage in open ended, highly interactive complex problem solving, in team environments, over spans of time, they change. Human development favors change and bringing UX designers to FLOSS communities can capitalize on that.

CASE STUDY: UX IN FIREFOX

Firefox UX started in 2006 when Mozilla hired a UX director to lead design. Subsequently two more UX practitioners have been hired. While much of the design process is open and available on the web, some decisions are made behind the scenes with the management teams, board of directors, and core development team. The process reported here is taken from an interview with the UX director¹ conducted in October 2007 and analysis of the online documents and conversations as part of an ongoing research project. A wiki document outlining the planning and design for Firefox 3 was created in late May 2006. This wiki contained a requirements document and feature list, among other information. The community was encouraged to participate in the planning and design through two discussion groups. The development team (including the UX team) participated in the discussions addressing concerns amongst each other and with the community of users. The feature list was continually updated based on the discussions in the list and at some point a ‘bug’ was created to initiate work on the feature and depending on the complexity of the work, a feature requirements document was created. Much discussion, including design decisions, also occurs in Bugzilla, the bug tracker used for Firefox development. When features are complete, the bug is closed and status marked as complete in the requirements document.

For example a lively discussion from May to September 2007 took place over a proposal to change the location bar in the browser that displays the URL. A Firefox developer put forth the proposal to see how the community would react to such a change. The motivation for the proposal was security-based and suggested providing the user with information about “who they’re dealing with online,”

¹ The UX director is now the director of front-end development, user experience, and product delivery at Mozilla.

according to the Mozilla security developer. The changes included removal of the URL icon because it offers no reliable information about the security of the site, and to highlight portions of the URL. The discussion about URL highlighting was summarized by the UX lead and entered into the wiki. This change, however, did not make it into the requirements document and hence Firefox 3 because it was unclear how much highlighting would help the user. However, developers and users posted different mockups for review, and one of the Mozilla UX practitioners suggested that even if they had an eye tracker available, reading highlighted text would probably be only milliseconds quicker when parsing the URL to determine if it was familiar.

A discussion in Bugzilla about information in the security tab, in the preferences dialog, occurred about how to present security information to users because information in the Firefox 2 security tab dialog was too technical. The lead security developer and two other developers submitted patches to a redesign, the UX director conducted a design review, and feedback from five other developers/users guided the design until another bug was created titled “Clean up Security Page Info visuals” to address the layout. Some discussion along with patches and UI review resolved this and patches were submitted and subsequently both of these bugs were closed.

At first glance UX design is not easy to recognize. Design happens in discussion lists, bug trackers, and requirements documents. No obvious design space exists, whereas, code exists in repositories. Requirement and feature specifications are, however, written from a user-centered perspective. A unique aspect of open UX design is the participation of the community in the design and development of Firefox. The Firefox community consists of about² forty core developers, 100 daily contributors, 1000 contributors, 10,000 nightly testers, and 100,000 beta testers, and 30 million daily users. As such, the UX team has a considerable amount of information to integrate into UX design. When interacting with the community and their suggestions and feedback, the lead UX director weighs two different philosophies for how to interject UX knowledge into the community. The first way is to be the expert. This approach states that the Firefox UX practitioners are experts and they know what is better for the user experience, just like developers are experts about code. The other approach is to provide research and data, to back the UX design with science. Commenting on these two approaches, the UX director states that,

What needs to happen is that we need to say that our opinions are rooted in observational science, perceptual science, but there are foundations for our expertise. And that we need to build credibility with these kinds of

² These numbers are from spring 2007.

expertise, but we should be given a free rein to play around with things. And we should be trusted a little more.

To this end, one of the Mozilla UX practitioners maintains a blog about UX to share information with the community and for example, one blog entry about quantitative design talks about cognitive performance modeling and why ‘your mom³’ is not statistically significantly, or more formally, why it could be a mistake to reply on single cases, or worse imagined single cases. According to the UX director, the downside of providing data all of the time for design decisions is that the community is afraid to commit to changes unless they are backed up by science and that what the UX team is striving for is to have the community accept that some design changes can be playful and open for discussion. But he also states that,

[The Mozilla community is] highly motivated and users care more. Paranoia and nervousness to protect the user experience result in conservatism.

Given the complexities surrounding open UX design, awareness of UX activities in and by the Mozilla community is essential for understanding benefits and challenges of UX methods and theories.

UX ACTIVITY AWARENESS IN FIREFOX

Building trust is a key activity and a precursor to understanding the effectiveness of UX theories and methods in open design. Using the four facets of activity awareness we analyze the Firefox 3 UX process.

Common ground in Firefox UX

In the location bar discussion both the security lead developers and the UX director provided summaries of the discussion to check that information was being understood appropriately. Also the UX blog post about quantitative design provides a mechanism in the comments section where the UX team can see how community members are sharing common knowledge and beliefs, if any, or where breakdowns might occur.

Communities of practice in Firefox UX

The Firefox UX practitioners have negotiated the socio-technical structures by integrating their activities, for example, design reviews and rationale for changes into existing structures. Firefox UX practitioners work in the bug tracker to monitor and guide the design changes and provide design reviews for final changes before a bug is closed. Furthermore, the UX team provides research-based rationale among opinions. In the location bar discussion, a Firefox UX practitioner posted a link to a study exploring how users responded to toolbars with information about phishing and the legitimacy of a website.

³ In open source communities, developers often justify UX design decisions based on how their mom or their grandma might easily use the software.

Social capital

Despite the frustrations experienced by the Firefox UX team regarding conservatism with design explorations, the team can leverage community passion to build social capital. UX participation in many ways throughout the community provides opportunities for successful interactions. Thus with time, the community will be able to understand that design proposals are explorations and not planned changes to get upset about. In addition, building social capital through interactions builds trust in UX expertise, which is an alternative to demanding respect because of expertise.

Human development

The integration of UX in Firefox includes bringing new knowledge to the discussion forums where developers interact with UX practitioners and both learn from each other. An indication of change is the promotion of the UX director to director of front-end development, user experience, and product delivery. This position provides an opportunity for human development across the Mozilla organization because the UX perspective is being perpetuated from a broader position.

CONCLUSION

A salient aspect of open UX design is using the community for new ideas and feedback. This is somewhat akin to participatory design. One difference is that in Firefox users care a great deal about the software they are helping to design and build, whereas users of business applications, for example, are empowered to help with the design of software that they usually are required to use. FOSS communities usually choose to use the software. We have seen efforts by the UX team to bring design theories to the community. Although pointing to science does not always bring a discussion to consensus, it does bring UX activity awareness to the community along with the other ways Mozilla UX practitioners have worked on building common ground, a community of practice, social capital, and human development.

ACKNOWLEDGMENTS

We thank the Mozilla UX team for sharing their practices with the world.

REFERENCES

1. Carroll, J. M., Rosson, M. B., Convertino, G. and Ganoe, C. Awareness and teamwork in computer-supported collaborations. *Interacting with Computers*, 18 2006, 21-46.
2. Crowston, K., Annabi, H., Howison, J. and Masango, C. *Effective Work Practices for Software Engineering: Free/Libre Open Source Software Development*. City, 2004.
3. Scacchi, W. *Socio-Technical Interaction Networks in Free/Open Source Software Development Processes. Software Process Modeling*. Springer, Netherlands, 2005.