Alternatives to Traditional Laboratory-based Usability Testing in Free/Libre/Open Source Software

Celeste Lyn Paul
KDE
www.kde.org
celeste@kde.org

ABSTRACT

In this workshop position paper, two case studies of alternative ways to conduct usability testing in Free/Libre/Open Source Software (FLOSS) projects are described. The first case study involves making the usability test a university project and using students to conduct the testing. The second case study involves using members of the open source user community to help organize the usability test, recruit participants, and conduct the usability test. These two case studies provide a beginning point for discussing ways of adjusting traditional usability methods for FLOSS practices.

Author Keywords

free/libre/open source software, usability testing.

ACM Classification Keywords

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

INTRODUCTION

Accomplishing good design in free/libre/open source software (FLOSS) has been notoriously difficult. Issues affecting design range from methods which do not fit FLOSS development methods, lack of understanding of users, too few contributors, and so on [3].

Recently there has been a focus on integrating usability testing into FLOSS development. Several FLOSS groups have attempted to follow industry's example and set up usability labs and conduct laboratory-based usability testing. Even then, laboratory-based testing is not a reasonable solution for grass-roots FLOSS projects which primarily operate on volunteers and have minimal budgets.

However, even in industry usability testing has been difficult to integrate into the development cycle. Traditional rigorous, laboratory-based methods have evolved into less-formal methods in order to accommodate the speed of modern development practices [4, 5]. In order to get more usability testing done in FLOSS, better usability testing methods must be developed. These methods must fit the unique characteristics of the FLOSS development environment, including resources, culture, and technology.

FLOSS USABILITY TESTING CASE STUDIES

This workshop position paper describes two cases studies of usability testing in FLOSS projects. The university laboratory-based usability testing study and the pilot of the community-based usability testing study have also been summarized in [2].

University Laboratory-based Usability Testing

The first example is a case of laboratory-based usability testing for Kubuntu's Ubiquity [6] software. Ubiquity is the installation software for the Kubuntu Linux distribution. It helps guide the user partition the disk drive, set up a user account, and install system files to the computer. The Kubuntu development team was concerned about the usability of the software, and was interested in getting feedback via usability testing.

The usability test was set up as a semester project at the University of Baltimore in Maryland in an interaction design methods graduate course. Students in the course helped develop the usability testing protocol, recruit participants, run testing sessions, and analyze the data. The organizer reviewed the usability testing protocol, set up the test in the usability laboratory, compiled the results of the analyzed data into a single report, and presented a single report to the Kubuntu development team. The graduate students were separated into three groups to help make working together easier. Each group recruited three participants for a total of nine participants for the entire study. A small stipend was provided by the project's sponsoring company which was provided to participants as a small stipend.

Overall, the project was a success. Nearly every recommendation made by the graduate students were eventually fixed over the course of several releases. What might have contributed to the project's success was the organizer's involvement in Kubuntu. The organizer had an opportunity to give a formal briefing of the usability testing results at an Ubuntu Developer Summit and speak directly to the Ubiquity developers. The organizer was also available for consult during the following release development cycles to answer questions about updated and new features.

Benefits

- Access to the university usability laboratory was free because it was part of a class project.

- Campus provided a resource for participants similar to a Kubuntu target audience.
- The project's sponsoring company provided a small amount of money to use as a recruiting incentive and participant stipend.

Drawbacks
- Usability testing took much longer to coordinate as a semester project than if it had been planned and executed by one organizer.
- There are possible inconsistencies in the data collection because of the user of multiple (and some inexperienced) moderators.
- The students did not take the time to get to know the project in detail (such as the technology used and people managing and developing the software) which may have affected their recommendations.

Community-based Usability Testing
The second example involves the usability testing of a popular FLOSS music management software called Amarok [1]. The Amarok project was ready to begin a major development cycle. Usability was one of the goals for the next major release and the project was interested in getting usability feedback from real users. At the same time, a Linux users group (LUG) near the usability test organizer was interested in finding ways non-technical Linux enthusiasts could contribute to FLOSS.

The organizer went to a LUG meeting and gave a presentation about usability testing and the Amarok software. LUG members were then encouraged to participate in the Amarok usability testing in two ways. First, members could recruit a friend or family member (someone besides a highly technical Linux geek) to participate in the testing. Second, members could act as the "moderator" and guide their guest through the usability test requirements. Meanwhile, the organizer created a simple usability testing "protocol" which provided an activity for participants to complete in 15-20 minutes. The LUG leader also found a location at a local library to hold the testing sessions.

During the testing, the organizer took note of participants actions, comments, body language, and reactions to different elements in the UI. After testing, the organizer compiled notes from testing and comments from the LUG members into a usability testing report.This report was delivered to the Amarok development team in time for their developer meeting. The usability issues and design recommendations in the report were discussed during the developer meeting and incorporated into future release cycles.

Benefits
- The "friends and family" recruit found participants who were not as technical as most FLOSS-oriented volunteers and closer to "normal" users.
- The informal nature of the usability test allowed for the activity to be organized and conducted in a shorter period of time than traditional usability testing.
- The participants did not expect a stipend since they were either friends or family of the moderator; snacks and stickers were provided as refreshments for anyone who participated..

Drawbacks
- Testing sessions were limited to 15-20 minutes which allowed for only a small amount of data collection.
- There are possible inconsistencies in the data collection because of the user of multiple and inexperienced moderators.
- This case was lucky to find a library which would allow them to run the activity; finding such a location may be difficult in other circumstances.

CONCLUSION
These two case studies of FLOSS usability testing have shown alternative ways of conducting usability testing in FLOSS projects. Issues discussed included logistics regarding where and how to test, recruiting and cost of participants, and turn around time of results to developers. These issues, and more, will be discussed during the workshop.

REFERENCES
1. Amarok. http://amarok.kde.org/. Last accessed December 14 2009.
2. Paul, C.L. (2009a). Examples of Low-Budget, Rigorous Usability Tests. Sidebar in *The Art of Community* by Jono Bacon. Sebastopol, CA: O'Reilly, 199-201.
3. Paul, C.L. (2009b). User Research Challenges in FLOSS Usability Projects. In proc. OSS 2009, 264-273
4. Sy, D. (2007). Adapting Usability Investigations for Agile User-Centered Design. UPA Journal of Usability Studies, 2(3):112-132
5. Tullis, T., Fleischman, S., McNulty, M., Cianchette, C., and Bergel, M. (2002). Empirical Comparison of Lab and Remote Usability Testing of Web Sites. In proc. UPA 2002.
6. Ubiquity. https://wiki.kubuntu.org/Ubiquity. Last accessed December 14 2009.