Quick and clean,

fast and efficient usability testing for software redesign

M.A. de Best
m.a.debest@student.tudelft.nl
K. van Deurzen
kvdeurzen@gmail.com
M. Geels
geels@hotmail.com
F.P.J. van Oostrom
f.p.j.vanoostrom@student.tudelft.nl
J.F.F. Pries
frenspries@gmail.com
B.R.V. Scheele
bruno@ch.tudelft.nl
Quick

Faculty of Industrial Design Engineering, Technical University of Delft,
2628 CE, Landbergstraat 15, Delft, Netherlands

ABSTRACT

In this paper, we present and reflect on a test method used to uncover serious and critical usability errors in consumer software in order to prepare qualitative requirements for a full redesign. This technique is suitable for usability testers with little experience, such as beginning researchers or students. The emphasis lies on the adaptation of existing techniques in both the data gathering and analysis phase of the usability test to create fast and efficient results. The validation of this method will also be discussed.

Keywords

Computer human interaction, usability test methods, consumer software

INTRODUCTION

In today's world, the influence of consumers in design is becoming stronger than ever. The increase of choice in product alternatives forces products to distinguish themselves most notably in user experience in order to rise above their competitors. Because of this, usability testing has become more important for designing successful products. However, usability testing is a lengthy and costly part of the redesign, and all too often it is performed improperly.

For the Design For Interaction master's course Usability Testing and Redesign at the Delft University of Technology, we were tasked to perform a usability test on a software application for creating and ordering photo albums to prepare for a full redesign of that software. In order to achieve this, we devised a research method based on several known methods. We present this method as a fast and efficient way for inexperienced researchers and students to obtain information on serious and critical usability errors of consumer software.

The method will be discussed in three parts. First the literature detailing the used methods will be introduced, in order to prepare the reader for. Then we will expand on the data gathering phase of our research, explaining about our research setup. After this we will discuss the data analysis phase.

We will also discuss the validity of our method for this research goal at length, by doing a comparison study of the same data by separate researchers. And finally, we will give our consideration on the value of the data in creating requirement for a software redesign.

RELATED WORK

Since usability testing is now becoming the new standard in product development, the need to reduce its costs has increased as well. The last decade, many researchers have tried to develop methods which are faster and cheaper than existing testing methods. One very important and widely accepted change in usability evaluations is the implication of the Nielsen/Landauer formula [4] which describes the percentage of errors found as a product of the number of test users and the probability a participant finds a specific problem. This mathematical model shows that five participants are sufficient to find about 85% of the errors and eight participants is sufficient for almost 95% of the existing errors in regular circumstances. Limiting the number of test subjects greatly reduced the time and costs needed for a usability test. There is some criticism about the formula though, which is mainly aimed at the fact that a small focus group is unlikely to properly represent the entire population. Another criticism is that the fact that not every error has the same probability to be found by a user is not represented in this model.

A way to improve the insight into the cognitive process around a product can be improved by the thinking-aloud protocol (TAD) as developed by Lewis [3] and justified by Ericsson and Simon [1]. Using this method participants are asked to talk during the tests and tell the observers "what they are trying to do, questions that arise as they work" and "things they read". Although TAD enables the designers to get a more qualitative insight in how their design is used and reflected upon, it has been argued that continually

talking during the test is not natural for the participants and might therefore influence the way a subject handles the product. A variation on TAD was suggested to minimize this effect; the retrospective protocol.

To reduce the amount of time needed to analyze the data, Kjeldskov, Skov and Stage [2] suggested Instant Data Analysis (IDA). In their article they describe a time efficient way of data analysis which is performed during the usability test itself, opposed to afterwards as done with Video Data Analysis (VDA). The obvious disadvantage of IDA is that the observer might fail to see some problems. The turn side is that the most critical errors will be seen and the errors that the observer fails to notice tend be cosmetic. Therefore Kjeldskov et al. point out that IDA might be more than adequate when identifying only the critical and serious problems.

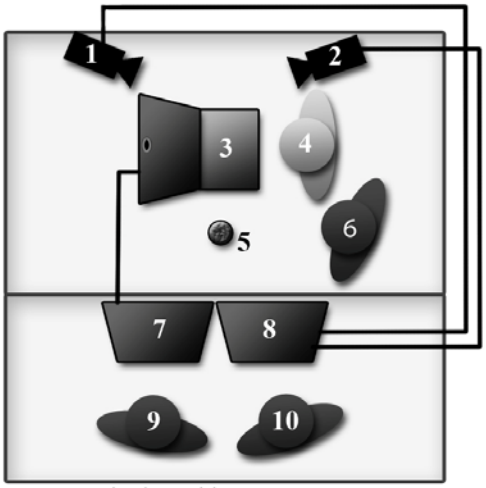
Above methods mostly describe the way data is gathered, but fail to answer how the raw data should be analyzed and presented. Zapf, Brodbeck, Frese, Peters and Prümper [6] split problems into four classes; functionality, usability, inefficiency and interaction problems. With these classes the nature and location within the 'man-machine system' of errors are described. In their conclusions Zapf et al. argue that each class within his empirical taxonomy can give a hint as to what kind of solution is necessary to reduce the errors. Although specific guidelines for these solutions are not given general directions are mentioned in their work.

As Zapf et al. try to describe the nature of errors, the company Ruigrok|Netpanel [5] has designed a graphical way to display the location of errors and comments of participants. Their application 'Tag-It' is an online instrument which enables subjects to add comments to certain locations on a website. This results in a heat map (a visual representation of data through colored areas) of the application tested, where all comments are represented by colored dots. This way of data representation gives an easy quantitative representation of 'hot zones' (places of interest) of an internet page, which is generated and supported by qualitative data.

STUDY

The current study was performed according to an adapted version of observational research setup in the IDA method. IDA proposes the use of two researchers, one sitting next to the test participant and the other in a separate observation room, supported by camera's showing the user's actions. Both observers take notes on the performance of the participant.

Our setup is detailed in Figure 1. An interesting fact about the setup is the addition of a researcher in the observation room. Now, one of the researchers can take notes based on computer screen showing the user's actions and the other takes notes while paying attention to the camera system recording the user's physical actions and expressions. The researcher sitting next to the user is relieved from note taking and performs as an active listener for the test participant.



1. Camera pointed at participant
2. Camera pointed at participant's hands and the keyboard
3. Laptop with screencapturing software and a webcam
4. Participant
5. Microphone
6. Active listener
7. Screen clone from laptop
8. Screen showing both camera's.
9. Notetaker one
10. Notetaker two

Figure 1: Data gathering research setup

The advantage of this is that the researchers in the observation room are completely dedicated to note taking, while the active listener can concentrate on keeping the participant thinking aloud to support the note takers instead of taking notes himself. This improves the gathering of data, since the second dedicated note taker will be less likely to miss observations than an active listener who also has to focus on note taking. The disadvantage of this method is however the need of an additional researcher for a longer period, which can complicate scheduling.

A note should be made about the number of test participants. While Nielsen states [4] that five participants are sufficient, we have opted to use ten participants. This was done because Nielsen assumes the tests are done according to a scenario, while we have opted to let the participants roam freely in the software to design their own calendar. This approach is less likely to consistently find the maximum of possible errors per participant, creating a need to test with a larger number of participants.

ANALYSIS

The data analysis was performed in three steps. The first was specifying problems from the gathered notes and ranking these on their severity. Afterwards the problems were categorized and finally the problems were labeled on a heat map to give us a visual representation of the problem areas.

Specifying problems

The IDA method proposes a brainstorm session after each test, facilitated by an extra researcher who has not participated in the test. This session will result in a list of usability errors. The facilitator will then rank these errors and afterwards all researchers will discuss this ranking.

Our method simplified and shortened this process, by performing the brainstorm sessions with all three present researchers. The note takers could discuss their findings, while the active listener could perform the role as facilitator more efficiently, since he was present and interacting with the test participant.

The brainstorm session focused on converting all the notes into problems or observations and ranking the problems on a scale of 1 (cosmetic) to 5 (critical). The ranking was based on how severe the problem is in the eyes of the test participant. For example, problems ranked with a 5 have the potential to make the participant quit using the software altogether. This severity ranking is very useful for prioritizing problems in order to determine which concessions should be made during the redesign, if any.

Categorizing problems

All problems were then categorized using the taxonomy proposed by Zapf et al. [6] Because most of the problems we found are closely related and could be fit in several categories at once, we have assigned the problems to the category we found to fit the problem the most.

Type of problem	*Amount*	*Percentage*
Functionality Problems	12	20%
Fatal errors (fixation)	1	2%
Efficiency Problems	6	10%
Usability problems		
Knowledge errors	11	18%
Habit errors	8	13%
Thought errors	18	30%
Judgment errors	4	7%

Table 1: Percentages of problems in categories

This made it easy to determine which category has the largest portion of problems (Table 1), which helped us to identify what problem category should be focused on during the redesign. The categories themselves also supported the creation of design requirements that, when followed, would avoid most of those problems in the redesign.

Creating a heat map

By labeling the problems and observations in screenshots, we were able to create a visual heat map (Figure 2) influenced by Ruigrok|NetPanel [5]. This heat map was an excellent way to have a visual representation of the areas and steps in which the most usability problems occur.

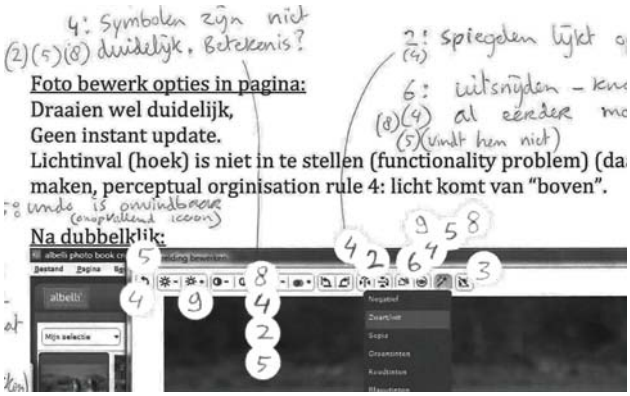


Figure 2: Heat map of usability problems

It gave us directions in choosing which software elements require the most attention during the redesign.

VALIDITY

Although the validity of IDA has been proven before [2], we tested the validity again as our setup was different. This was achieved by comparing the data of one participant gathered in two, independent ways.

First of all the outcome of one observer was compared to the results of the second observer to find overlap and differences between the observers, proving internal validity. This was done to see if the individually observed errors are also seen by other observers. Comparing the two scripts with raw data from two different observers, we found an overlap of 79% (22 of all 28 identified problems were seen by both observers). This was deemed to be sufficient to prove that the errors seen were really there, and IDA was proven suitable for us to gather data with.

The previous test proves that the data collected was valid. Our main concern using IDA was whether enough data was collected and no serious errors were missed. To analyze this, two observers who were not involved at time of the original usability test performed a VDA on the videos taken during the session. The video data existed of a recording of the actual screen, a frontal view of the participant, a close-up of the laptop, and an overview of the situation. All video material was supported by audio recordings.

The two VDA observers made an analysis which could then be compared to the results of the brainstorm session of the IDA. When comparing these two lists an overlap of 54% was found (13 out of 24 identified problems were summarized during the brainstorm of the IDA). This was far lower than expected, but might be explained by the fact that during the brainstorm sessions the important problems were summarized but minor problems were discarded as not important for the redesign. The next step was to compare the VDA list to the raw data collected during the

IDA. Here a respectable overlap of 76 % was found (19 out of 25 identified errors were seen during the IDA). When further analyzing the differences it was found that most of the errors missed during the IDA were cosmetic errors as predicted by Kjedskov et al. [2]. Splitting the errors into the three suggested categories (Critical, Serious and Cosmetic) gave the following results (Table 2).

Category	*IDA*		*VDA*		*Total*
Critical	3	100%	3	100%	3
Serious	9	82%	10	91%	11
Cosmetic	7	63%	10	91%	11
Total	19	76%	23	92%	25

Table 2: Number of usability problems identified using IDA and VDA of 1 participant

This shows that both IDA and VDA detected all three critical errors. Nine out of eleven serious problems were observed with IDA and seven out of eleven cosmetic errors were observed. As discussed earlier the main goal of a usability test as a mean for a redesign is to give an overview of the most serious errors in the existing product. In this case it is therefore not dramatic that only 63% of the cosmetic errors were identified using IDA.

VALUE OF DATA

The usability test serves to create the requirements that are needed to completely redesign the software package in question. Instead of merely letting the designers fix every problem to the best of their ability within relatively the same design, these requirements let them create software that is better suitable for users, since it addresses the causes of problems rather than the problems themselves. In the remainder of the text, we discuss how the data gathered is of value in creating these requirements.

As mentioned above, the heat map shows in which steps and areas of the software a user has problems and how many. It is a clear indication on where to focus when redesigning. The fundamental cause of these problems can be determined using the categories in which the problems are sorted. This seems to be sufficient for the redesign to avoid all these problems.

When looking over the redesign of the software one can use the problems determined during the severity to determine if problems that have been found still occur in the redesign. If this is the case, the ranking of the problem can help determine whether it should still be fixed or if a concession concerning the issue can be made.

CONCLUSION

We have presented our research method for acquiring qualitative requirements for a complete redesign of consumer software. We view this method as useful for inexperienced researchers, such as beginning researchers and students.

Our data gathering was done by an observation test, based on the IDA method, but has been changed slightly to support inexperienced researchers. The result is a valid setup in which the researchers are less likely to miss serious and critical usability problems.

We elaborated on three successive ways to analyze the data gathered during the observation. The first step was to take the data and specify problems from them. These problems were ranked according to severity. Afterwards, the problems were categorized to gain insight into which kind of problems occur the most and they were also labeled in a heat map to create a visual representation of the problem areas and the amount of problems in that area.

Then, requirements for the software redesign were created. The heat map supported the researchers into determining what areas in and steps of the software to concentrate on. The categories gave insight into which causes of problems should be addressed and how. The severity ranking of the problems supported the researchers during the redesign to determining in what way concessions should be made, if they should be necessary.

Using this method can therefore be seen as a time saver for creating requirements for redesigns.

ACKNOWLEDGMENTS

We want to thank Prof. Dr. Huib de Ridder and Dr. Ingrid Mulder for encouraging us to participate in the CHI Nederland Conferentie 2009 with this paper.

REFERENCES

1. Ericsson, K.A., Simon, H.A. *Protocol analysis: verbal reports as data* (1984), MIT Press, Cambridge MA.
2. Kjeldskov, J., Skov, M.B., Stage, J. Instant data analysis: conducting usability evaluations in a day. *Proceedings of NordiCHI '04* (Tampere, Finland, October 23-27), ACM Press, 233-240.
3. Lewis, C., Rieman, J. *Task-centered User Interface Design: a practical introduction* (1993). Available at http://oldwww.acm.org/perlman/uidesign.html. Retrieved at April 09, 2009.
4. Nielsen, J. Why you only need to test with five users. Jakob Nielsen's alertbox (March 19, 2000). Available at http://www.useit.com/alertbox/20000319.html. Retrieved at April 09, 2009.
5. Woerden. M., Os, S. van. Research 2.0, Tag-it. *Ruigrok|NetPanel* (2008). Available at http://www.usabilityweb.nl/2008/07/research-20-tag-it/. Retrieved at April 09, 2009.
6. Zapf, D., Brodbeck, F.C., Frese, M., Peters, H., Prümper, J. Errors in Working with Office Computers: A First Validation of a Taxonomy for Observed Errors

in a Field Setting. International Journal of Human-Computer Interaction 4 (1992). 311-339.