AN EMPIRICAL STUDY OF PERSPECTIVE-BASED USABILITY INSPECTION*

Zhijun Zhang, Victor Basili, and Ben Shneiderman
Department of Computer Science
University of Maryland
College Park, MD 20742, USA
{zzj, basili, ben}@cs.umd.edu

Abstract

Inspection is a fundamental means of achieving software usability. Past research showed that during usability inspection the success rate (percentage of problems detected) of each individual inspector was rather low. We developed perspective-based usability inspection, which divides the large variety of usability issues along different perspectives and focuses each inspection session on one perspective. We conducted a controlled experiment to study its effectiveness, using a post-test only control group experimental design, with 24 professionals as subjects. The control group used heuristic evaluation, which is the most popular technique for usability inspection. The experimental results are that 1) for usability problems covered by each perspective, the inspectors using that perspective had higher success rate than others; 2) for all usability problems, perspective inspectors had higher average success rate than heuristic inspectors; 3) for all usability problems, the union of three perspective inspectors (with one from each perspective) had higher average success rate than the union of three heuristic inspectors.

INTRODUCTION

Usability inspection (Nielsen & Mack, 1994) is an important approach to achieving usability. Different usability inspection techniques have been practiced, including heuristic evaluation, cognitive walkthrough, and pluralistic walkthrough, etc. (Nielsen & Mack). Empirical studies (Desurvire, 1994) (Jeffries, Miller, Wharton, & Uyeda, 1991) (Nielsen, 1992) showed that when using these techniques the percentage of usability problems detected by each inspector was rather low. We proposed the perspective-based usability inspection technique to improve the effectiveness of usability inspection. In this paper, we describe an empirical study conducted recently and its results.

PERSPECTIVE-BASED USABILITY INSPECTION

With the observation that it is difficult for inspectors to capture all different dimensions of usability issues at the same time, we propose perspective-based usability inspection, where each inspection session focuses on a subset of usability issues covered by one of several usability perspectives. Each perspective provides the inspector a point of view, a list of inspection questions that represent the usability issues to check, and a specific procedure for conducting the inspection. Our assumption is that with focused attention and a well-defined procedure, each inspection session can detect a higher percentage of the problems relating to the perspective used, and that the combination of different perspectives uncovers more problems than the combination of the same number of inspection sessions using the same and general inspection technique.

This idea is supported by studies on defect-based reading (Porter, 1995) and perspective-based reading (Basili, 1996). These two studies showed that when inspecting software requirement documents, it is more effective to let each inspector focus on one class of de-

*This work was supported by the National Science Foundation under grant NSF UMIACS-01-5-23394.

fects or inspect from one particular perspective than to let each inspector have the same and general responsibility.

Currently we use the following usability perspectives: novice use, expert use, and error handling. "Error handling" has been separated from "novice use" and "expert use" situations, so that inspections on "novice use" and "expert use" will only deal with interactions along the correct paths. A unique procedure is used for inspections on "error handling".

In developing the technique, we built a model for human-computer interaction by extending Norman's "seven stages of action" model (Norman, 1988) with error handling. A set of usability goals were defined for each perspective. The inspection questions for each perspective were derived by going through the model and see if the respective usability goals can be achieved based on the usability context, which includes the characteristics of the user interface, the users, the tasks, and the users' working environment.

The abstract form of the human-computer interaction model is as follows:

1. Form the goal
2. Form the intention
3. Form the action
4. Execute the action
5. Perceive the feedback
6. Interpret the results
7. Understand the outcome
8. Deal with errors that may have occurred

METHOD

An experiment was conducted at a government organization with 24 professionals. We used a post-test only control group experimental design. The control group used heuristic evaluation. The experiment group was further divided into three sub-groups along the three perspectives.

Each subject was assigned to use one technique to inspect two alternative interfaces of a Web-based data collection form, namely interface H and J. Interface H had two variations, H/N and H/M. The subjects were randomized and assigned to different techniques and different interface orders. The layout of the experimental design is shown in Table 1, where the numbers indicate the number of subjects in each slot.

Table 1: The experimental design

	Heuristic (12)	Novice (4)	Expert (4)	Error (4)
H/N, H/M, J	3	1	1	1
H/M, H/N, J	3	1	1	1
J, H/N, H/M	3	1	1	1
J, H/M, H/N	3	1	1	1

The 24 subjects in the main study were familiar with both the interface domain and the task domain. They were either programmers, domain experts, technical researchers, or cognitive researchers. Efforts were made to evenly distribute participants of different backgrounds to different groups. Before the main study, we conducted a pilot study with 7 graduate students of Computer Science major to test out the instruments. We also asked two external usability experts to review the interfaces and report the usability problems they found.

In the main study, each participant first watched a video introduction of the project background and the inspection technique to be used. After signing the consent form, each inspector spent up to 100 minutes in one of the two "cognitive lab" rooms to conduct the inspection. All inspection sessions were observed from an adjacent room through one-way mirrors. The sessions were also videotaped, with two views: one of the computer screen and the other of the inspector's facial expression and upper-body movement.

The usability heuristics used were:

1. Speak the users' language
2. Consistency
3. Minimize the users' memory load and fatigue
4. Flexibility and efficiency of use
5. Use visually functional design
6. Design for easy navigation
7. Validation checks

8. Data entry
9. Provide sufficient guidance

Each heuristic had a detailed explanation about the related usability issues. For example,

> *8.* **Data entry:** *Easy to enter data. Data are visible and clearly displayed. Allow the users to change data previously entered. Easy to find data already entered. Necessary entries are clearly defined. Entries are in correct format.*

For "novice use" perspective, inspectors were asked to think of novice users with a list of characteristics: being able to use a keyboard and a mouse, without visual disabilities, etc., which were defined based on the context of the application. Inspectors were given the description of the application and the user tasks. For each task, they were asked to think about whether a novice user would be able to choose the correct action, execute it successfully, and understand the outcome. They were provided with a list of detailed usability questions. For example, for data entry,

> *Are formats for data entries indicated?*

For "expert use", inspectors were asked to think about expert users and check the interface for efficiency, flexibility, and consistency in supporting the user tasks. They were given a list of usability questions relating to these issues. For example, for data entry,

> *Are possible short-cuts (e.g. using the Tab key to switch to the next field) available?*

> *Are possible default values used?*

For "error handling", inspectors were given a classification of user errors. They were also given the characteristics of the users as the "novice use" inspectors were. For each user task, inspectors were asked to list the possible user errors and check the following questions for each user error:

> *Does the user interface prevent the error as much as possible?*

> *Does the user interface minimize the side effects the error may cause?*

> *When the error occurs, will the user realize the error immediately and understand the nature of the error from the response of the user interface?*

> *When the error occurs, does the user interface provide guidance for error recovery?*

RESULTS

Altogether 82 problems were detected for interface H, 61 for interface J. These problems were collectively identified by the 24 experiment subjects, 7 pilot subjects, and 2 external expert reviewers. The results for interface H were the combined results for H/M and H/N. A large portion of the problems (33 for H and 20 for J) were specific to the application domain of form design, and not covered by the usability heuristics or the usability questions for the perspectives. Since they would have negative influence on the use of the forms, they were counted as usability problems.

The performance of the 24 experiment subjects are presented and discussed as follows.

The primary independent variable was the inspection technique. But another independent variable, the interface order, was confounded in the experimental design. An ANOVA test failed to reveal a significant main effect by the interface order or its interaction with the inspection technique. The following statistics test the effect of inspection technique as the only source of differences.

The dependent variables include the number of all usability problems identified by each individual inspector and the number of certain types of usability problems identified by each individual inspector.

The comparison of the individual detection effectiveness between the two technique groups is given in Table 2. It shows the mean and standard deviation (in parentheses) of the percentage of problems found by each individual inspector in each technique group. The improvement of perspective-based technique over heuristic evaluation is about 30% on the average. The p-values were the results of the t-test.

Table 2: Individual effectiveness (overall)

	Heur.(%)	Pers.(%)	Improve(%)	p-value
H	8.4(3.8)	10.4(3.4)	23.8	0.094
J	9.4(6.4)	13.4(6.0)	42.5	0.064
Both	9.1(3.8)	11.9(3.6)	31.2	0.037

Table 3: Individual effectiveness on different types of problems

	Category	Heur.(12)(%)	Pers.(4)(%)	p-value
H	Novice	8.0(6.6)	18.5(9.0)	0.012
	Expert	15.9(10.3)	20.5(8.7)	0.22
	Error	14.3(12.2)	33.9(6.8)	0.0046
J	Novice	11.7(9.1)	26.3(11.1)	0.010
	Expert	14.9(11.1)	28.0(18.5)	0.052
	Error	9.0(10.9)	29.3(13.5)	0.0043

Table 3 shows the individual detection effectiveness with respect to usability problems categorized by the three perspectives. For usability problems related to each perspective, the average percentage of such problems detected by the 4 inspectors using that perspective is compared against the average percentage by the 12 heuristic inspectors. The standard deviations are in parentheses. It shows that the use of the "novice use" and "error handling" perspectives greatly improved the inspector's detection effectiveness for problems within the category. The use of "expert use" perspective brought about improvement to a lesser extent, possibly because that the inspectors themselves were all experts in the application domain and user interface domain. Thus they were able to capture a large portion of the "expert use" problems even without help from the "expert use" perspective.

Although all inspectors conducted the inspection individually, we were interested in comparing the aggregated results of multiple inspectors. For example, we compared the number of unique problems identified by 3 perspective inspectors (one from each of the three perspectives) and 3 heuristic inspectors. The results are shown in Table 4. There were 220 positive aggregations for heuristic evaluation and 64 for perspective technique. Since the data points in each group were not independent from each other, no statistical test was performed.

We also did a permutation test (Edington, 1987) of simulated 12-person teams. This involves con-

Table 4: Aggregated problems found by 3 inspectors

	Tech.	Problems(%)	Improve(%)
H	Heuristic	21.8(5.0)	
	Perspective	27.7(4.4)	26.5
J	Heuristic	24.1(7.2)	
	Perspective	32.8(7.4)	35.7

Table 5: Permutation tests for team performance

	#Possible teams	Rank of pers. team	p-value
H	2,704,156	262,577	0.097
J	2,704,156	122,993	0.045

structing all possible 12-person teams and see how the un-diluted perspective team ranked among all possible 12-person teams in terms of number of unique problems detected. Whether or not we can claim that the perspective-based technique had a beneficial effect on team performance depends on how the un-diluted perspective team (with all 12 perspective inspectors) appears towards the top of the ranking. The p-value is the rank of the un-diluted team divided by the total number of teams. There were 2,704,156 possible 12-person teams out of the 24 subjects. The results of this test are given in Table 5. It shows that at $p < 0.10$ level, the perspective-based inspection technique significantly improved the effectiveness of an inspection team.

DISCUSSION AND FUTURE DIRECTIONS

The experimental results showed about 30% improvement at both the individual level and for the aggregation of 3 inspectors. As we had expected, the perspective inspectors found much more usability problems covered by the assigned perspective, as shown in Table 3. Furthermore, the average number of all the problems found by each perspective inspector was also higher than that of each heuristic inspector. In summary, the focused attention enabled inspectors to find much more problems of certain categories without suffering the total number of problems found in general.

To generalize the results, the following issues need to be considered:

- Generating specific inspection questions. To apply perspective-based usability inspection effectively, the generic inspection questions for each perspective need to be instantiated to the specific context of the project. In the experiment, this was done by the experimenter. One question that remains is whether a usability practitioner can follow the provided guideline to develop the specific inspection questions that are equally effective.

- Domain experts vs. usability experts. The subjects in this experiment were all experts in the application domain, with some knowledge in usability. We need to know how the technique works for inspectors who are usability experts with some knowledge in the application domain, as well as for inspectors who are experts in both usability and the application domain.

- Inspection time. In the experiment, each inspector was given a time limit of 100 minutes to inspect the two interfaces. Although most participants finished inspection within the time limit, there was one case where a perspective inspector said that given the time limit she was not able to follow the technique very well. In some studies, inspectors were asked to conduct the inspection, besides doing their daily work, within two weeks. It would be interesting to test how many more problems the inspectors can detect when they are given more time.

- Experience with the technique. In this experiment, both techniques were new to the subjects. We would like to know how the learning curve is like for each technique.

We plan to conduct more empirical studies to address some of these issues. A lab package is being built to facilitate replications of the experiment by other researchers. We also plan to build an application package so that practitioners can learn and use the technique and provide some feedback that may address the other issues mentioned above.

Acknowledgements

We thank Elizabeth Nichols, Barbara Sedivi, Nancy VanDerveer, Ellen Soper, and Al Paez of the US Bureau of the Census for their cooperation in the study. We thank all the subjects for their participation. We thank Pawan Vora of US WEST and Chris North of University of Maryland for their expert reviews of the interfaces.

References

Basili, V., Green, S., Laitenberger, O., Shull, F., Sorumgard, S., & Zelkowitz, M. (1996). The empirical investigation of perspective-based reading. Empirical Software Engineering, 1, 133-164.

Desurvire, H. W. (1994). Faster, Cheaper!! Are Usability Inspection Methods as Effective as Empirical Testing? In J. Nielsen & R. L. Mack (Eds.), Usability Inspection Methods (pp. 173-202). New York, NY: John Wiley & Sons.

Edington, E. S. (1987). Randomization Tests. New York, NY: Marcel Dekker.

Jeffries, R., Miller, J. R., Wharton, C., and Uyeda, K. M. (1991, April). User interface evaluation in the real world: A comparison of four methods. CHI'91 Conference Proceedings. 261-266.

Nielsen, J. (1992, May). Finding usability problems through heuristic evaluation. CHI'92 Conference Proceedings. 373-380.

Nielsen, J., & Mack, R. L. (Eds.). (1994). Usability Inspection Methods. New York, NY: John Wiley & Sons.

Norman, D. (1988). The Design of Everyday Things. New York, NY: Basic Books.

Porter, A., Votta, L., Basili, V. (1995, June). Comparing Detection Methods for Software Requirements Inspections: A Replicated Experiment. IEEE Transactions of Software Engineering, 21, 563-575.