Usability Inspection Methods

Jakob Nielsen

Bellcore
445 South Street
Morristown, NJ 07960

Email: nielsen@bellcore.com (primary), nielsen.chi@xerox.com (backup)

ABSTRACT

Usability inspection is the generic name for a set of cost-effective ways of evaluating user interfaces to find usability problems. They are fairly informal methods and easy to use.

Keywords: Usability engineering, heuristic evaluation, cognitive walkthroughs, pluralistic walkthroughs, feature inspection, consistency inspection, standards inspection.

INTRODUCTION

Software inspection [1][7] has long been used as a method for debugging and improving code. Similarly, usability inspection [25] has seen increasing use since about 1990 as a way to evaluate user interfaces. The four basic ways of evaluating user interfaces are automatically (usability measures computed by running a user interface specification through some program), empirically (usability assessed by testing the interface with real users), formally (using exact models and formulas to calculate usability measures), and informally (based on rules of thumb and the general skill and experience of the evaluators). Under the current state of the art, automatic methods do not work and formal methods are very difficult to apply and do not scale up well to handle larger user interfaces.

Empirical methods are the main way of evaluating user interfaces, with user testing probably being the most commonly used method. Often, real users can be difficult or expensive to recruit in sufficient numbers to test all aspects of all the versions of an evolving design, leading to the use of inspection as a way to “save users.” Furthermore, project schedules or budgets sometimes impose restrictions that make informal methods like inspection desirable as a “discount usability engineering” solution [16][21] since they are highly cost-effective [8]. Several studies have shown that usability inspection methods are able to find many usability problems that are overlooked by user testing but that user testing also finds some problems that are overlooked by inspection, meaning that the best results can often be achieved by combining several methods [5][6][11].

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

INSPECTION METHODS

Usability inspection is the generic name for a set of methods that are all based on having evaluators inspect the interface. Typically, usability inspection is aimed at finding usability problems in a design [13], though some methods also address issues like the severity of the usability problems and the overall usability of an entire design [28]. Many inspection methods lend themselves to the inspection of user interface specifications [17] that have not necessarily been implemented yet, meaning that inspection can be performed early in the usability engineering lifecycle [19].

- **Heuristic evaluation** [23][27] is the most informal method and involves having usability specialists judge whether each dialogue element follows established usability principles (the “heuristics” [15][18][21][22]).
- **Cognitive walkthroughs** [12][29][30] use a more explicitly detailed procedure to simulate a user’s problem solving process at each step through the dialogue, checking if the simulated user’s goals and memory content can be assumed to lead to the next correct action.
- **Formal usability inspections** [9] use a six-step procedure with strictly defined roles to combine heuristic evaluation and a simplified form of cognitive walkthroughs.
- **Pluralistic walkthroughs** [3][4] are meetings where users, developers, and human factors people step through a scenario, discussing each dialogue element.
- **Feature inspection** [2] lists sequence of features used to accomplish typical tasks, checks for long sequences, cumbersome steps, steps that would not be natural for users to try, and steps that require extensive knowledge/experience in order to assess a proposed feature set.
- **Consistency inspection** [31] has designers representing multiple projects inspect an interface to see whether it does things in the same way as their own designs.
- **Standards inspection** [31] has an expert on some interface standard inspect the interface for compliance.

Heuristic evaluation, cognitive walkthroughs, feature inspection, and standards inspection normally have the interface inspected by a single evaluator at a time (though heuristic evaluation is based on combining inspection reports from a set of independent evaluators to form the list of usability problems). In contrast, pluralistic walkthroughs
and consistency inspections are group inspection methods. Finally, formal usability inspections combine individual and group inspections. Many usability inspection methods are so easy to apply that it is possible to have regular developers serve as evaluators, though better results are normally achieved when using usability specialists [20].

Acknowledgments

This summary is based on the author's own work, presentations at the CHI'92 workshop on usability inspection methods [14], and chapters in the recent Usability Inspection Methods book [25]. Workshop and book contributors include Brigham Bell, Randolph Bias, Louis A. Blatt, Janice Bradford, Patricia Brooks, George Casaday, Heather Desurvire, Robin Jeffries, Sandra Jones, Michael J. Kahn, Clare-Marie Karat, James A. Knutson, Thomas Lanzetta, Clayton Lewis, Robert L. Mack, Allan MacLean, Frank Montaniz, Peter Polson, Amanda Prail, John Rieman, Linda Tse, Cathleen Wharton, and Dennis Wixon. The present discussion of usability inspection is the responsibility of the author and does not necessarily represent the positions of these other people.

References

Like usability testing, inspection methods have evolved and have benefited from years of usage. They have their own rich history that I feel are undervalued as an effective way of identifying problems in an interface. They’re more than a second-rate usability test. One of the primary sources for understanding inspection methods is from a 1994 book by Nielsen and Mack. A more recent contribution is from Chauncey Wilson who put together an approachable and comprehensive review of inspection methods in his 2014 book.