Assignment #1

Usability Article Reviews

by

Ellen F. Glazer

An assignment submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

School of Computer and Information Sciences
Nova Southeastern University

2001

Buur, J. & Bagger, K. (1999, May). Replacing usability testing with user dialog. *Communications of the ACM*, 42(5), 63-66.

Usability testing consists of examining products before customers take ownership. This type of testing allows research and development departments to obtain objective data on the use of products. Conventional usability testing procedures conducted by impartial usability specialists used the think aloud method in an isolated testing facility with testing experts and users separated.

Buur and Bagger made several exciting breakthroughs in usability testing and share these findings with the reader. Previous usability testing methods hampered thorough understanding of communication between designers and users while new testing breakthroughs have decreased or eliminated problems with product use and technology between designers and users.

This article outlines four examples of how conventional usability testing, updated to a modern format, increases dialog between users and designers. This new usability-testing concept was implemented in Denmark's largest manufacturing company with 18,000 employees worldwide. Prior to new testing methods, user interfaces were typically composed of small liquid crystal displays and a set of designated push buttons. In 1992, a usability lab was established and successfully adopted methods from the human computer interaction community in usability testing to improve products.

Through this research, the authors have developed five steps to replace usability testing with user dialog. Placement of the test facilitator in the lab created tension and nervousness for the user. By making the test facilitator an active and attentive partner for the user, the testing atmosphere proved to be more conducive to high-level communication. User activities observed and documented by video recordings allowed testers to communicate their findings to the research and development department. Videos convinced management that their company was

not sufficiently designing easy-to-use products and that they needed user-centered methods for success.

For companies to be customer oriented, designers need training to learn how to listen carefully to users and abandon their traditional intimidation tactics that proved inadequate in obtaining positive feedback. The authors developed a solution to this dilemma by educating designers to act as test facilitators and efficient observers by conducting training sessions and gaining critical feedback about the process. Some designers became frustrated, but with additional training and analyzing session videos, they were able to overcome communication problems between facilitators and users, and they were able to discuss how to improve the overall usability testing procedures.

Research showed that single user sessions did not yield ample feedback with users only answering direct questions. The testing sessions developed into workshops with several users participating in pairs. The traditional think aloud format changed to "co-discovery learning," while the facilitator listened and observed. Workshops consisting of eight participants increased user confidence allowing close contact with several members of the design team. All day workshops permitted product testing and strategy discussions with designers. These workshops contributed to organized, significant discussions between users and designers, which led to improvements in product design.

Having workshops early in the design phase with user participation increases product success and allows flexibility of the prototype. Other advantages of dialog include: revealing user priorities and practices, accelerating the design process, and allowing manufacturers to move toward innovative designs. The authors expect to relocate workshop discussions to the user’s real world environment making them more comfortable thereby allowing added

interaction and disbursement of knowledge. This is an excellent article outlining procedures to secure increased user testing dialog.

Reference List

Buur, J., & Nielsen, P. (1995). Design for usability: Adopting HCI methods for the design of mechanical products. *International Conference on Engineering Design,* (Prague), Heurista.

Dumas, J., & Redish, J. (1993). *A Practical Guide to Usability Testing*. Norwood, NJ: Ablex Publishing Corporation.

Rubin, J. (1994). *Handbook of Usability Testing*. New York, NY: John Wiley & Sons.

Hook, K. (1999). Designing and Evaluating Intelligent User Interfaces. *Conference Proceedings of the 1999 International Conference on Intelligent User Interfaces.* ACM, Redondo Beach, California, 5-6.

Many users think systems should adapt to their needs instead of users adjusting to systems. The author believes that, to be of real use, intelligent user interfaces must consider usability issues. Methods for design and evaluation of intelligent user interfaces should be prepared from the usability perspective. The author provides insight about usability issues for intelligent user interfaces such as making systems individualized or personalized, thereby increasing the systems flexibility and appeal. This article approaches the usability issues for intelligent user interfaces by discussing usability criteria applicable to adaptive systems and methods used to design and to evaluate them.

A system capable of adapting to users rather than requiring users to adapt to the system is an appealing concept to both users and researchers. Hook warns the reader that an intelligent interface behaving like a human being, by smoothly changing its behavior to fit with user's knowledge, abilities, and preferences, is far from current technology capabilities. Very few intelligent user interfaces have succeeded commercially, but the ones that have succeeded use elementary adaptations based on simple knowledge of users. Researchers for human-computer interaction fear that intelligence at the interface will breach usability principles and obscure responsibility issues.

Usability issues worried the intelligent user interface community and prompted better problem definitions and design criteria. Hook argues that the problems within intelligence at the user interface may violate quality usability principles developed for direct manipulation systems. Assigning system controls users, maintaining standard responses to similar input, and designing transparent systems are solid usability principles. Adaptable systems that automatically

customize to individual needs violate the principal of predictability and may lose transparency, thereby obstruct user system control.

In the design phase of a system, redesigning the target system may eliminate common problems such as trust and privacy. Hook stresses the necessity to develop a system design that meets the needs of users by incorporating the adaptive or intelligent parts globally throughout the system. The decision to include adaptive or intelligent parts in the system, according to Hook, is rarely part of the modern design process, and he stresses it must be decided early in the design phase. Hook cautions readers and researchers that adaptive systems making claims about user needs have little to do with what will actually be of real help to users.

The challenge of evaluation is to prove adaptive behavior improves user interaction. Proper design and thorough evaluation methods will assure that designers are adapting to the needs of users. Hook warns that evaluating systems are difficult tasks, and even more so when testing an adaptive system.

The author states her disappointment that very few studies on evaluating systems exist, and even less on adaptive systems. Hook concludes that proper analysis of users and their tasks and needs is a necessary part of any development of an adaptive system. This reviewer has limited knowledge on usability testing, but Hook outlines her views on designing and evaluating intelligent user interfaces with clarity and provides sound standards to conduct these tasks.

Reference List

Benyon, D. (1993). Adaptive systems: A solution to usability problems. *Journal of User Modeling and User-Adapted Interaction*, 3 (1), 1-22.

Hook, K. (1997). Evaluating the utility and usability of an adaptive hypermedia system, *Proceedings of 1997 International Conference on Intelligent User Interfaces*. ACM, Orlando, Florida.

Suchman, L. (1997). From interactions to integrations, *Proceedings of Human-Computer Interaction*, INTERACT.

Marold, K., Larsen, G., Shaw, K., & Robertus, P. (1999). Usability testing for a computer skills WBT (Web Based Training) program. *SIGCPR '99 Conference Proceedings on Computer Personnel Research,* ACM, New Orleans, LA, 304-309.

End-users are using Web Based Training (WBT) courses to update their skills and to become computer literate. This type of training delivers instructional content through a Web browser on the Internet or an intranet. The authors outline implementation of basic computer skill WBT from its initial development through end-users usability testing. The usability-testing phase for this project contained two parts. A multimedia survey, presented to students before entering college, did not yield sufficient information. Due to the lack of feedback from the first survey instrument, an additional survey, administered to current Introduction to Computers students, measured WBT effectiveness.

Web based training needs to teach basic minimum computer literacy skills necessary for students to complete course work and graduate college. It is common for entry-level college students to have the basic computer skills, but variations of those skills differ greatly. Three instructors and several student workers from Metropolitan State College of Denver spent two years designing a WBT model to developed student basic computer skills using an eight-module WBT system. Before college wide deployment of the new system, the design team conducted a two-phase usability study. WBT courses can be delivered Just in Time (JIT) and taken any time, any place (ATAP) for the convenience of the modern student.

Development of this WBT system demanded new instructional design content. To ensure user-centered systems, end-users were involved in all phases of usability testing. The authors recommend conducting usability tests during all phases of system development including prototype stages, subsystem testing, and testing the final system. Marold, Larsen, Shaw, and Robertus stress that end-users should appraise the system using formative and summative

evaluations and testing should incorporate both covert and overt observations. The usability testing instruments used in this study determined the end-users' perception of WBT significance. The design team performed usability analysis in the initial phase and again in the full implementation phase using a small group to pilot test the course. Before widespread deployment, the end-user group made several valid recommendations for changes to the system. Testing included a usability survey at the course midpoint providing the design team with general WBT interface information. This survey instrument assisted in gathering quality data as to the likes, dislikes, desired changes, and recommendations from the end-users, but did not contribute to content evaluation of the eight modules.

With lack of significant data from the first survey, the design team distributed an effectiveness survey. This survey instrument included demographics, student status, enrollment hours, and whether the subject had ever previously taken an online course. Data analysis from the second survey showed that the students received the new system favorably and showed that learning basic computer concepts was effective and enjoyable for students. The authors caution the reader not to draw any conclusions as to the effectiveness of student learning. Analysis showed that WBT courses could be stand-alone courses, when introduced earlier in the term have more success, and are an effective alternative to traditional classroom instruction.

The project outlined in this article includes many advantages that most researchers do not have at their disposal. The amount of human resources, financial resources, and time availability gave this project an opportunity for system development with thorough usability evaluation for increased success.

Reference List

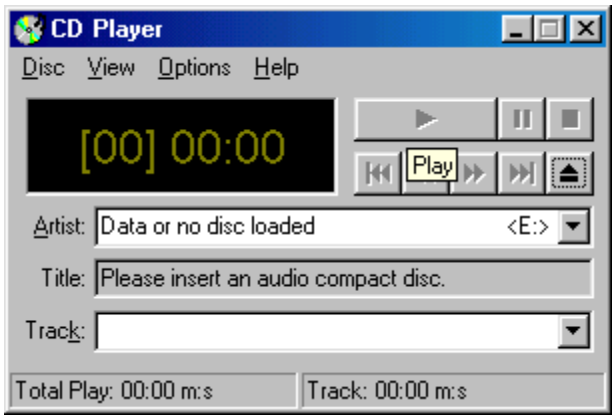
Neilson, J. (1994). *Usability Engineering*. Boston, MA: Academic Press.

Shneiderman, B. (1998). *Designing the User Interface*. Reading, MA: Addison-Wesley.

Prates, R., de Souza, C., & Barbosa, S. (2000, January). A method for evaluating the communicability of user interfaces. *Interactions,* ACM, 7 (1), 31-38.

Messages sent from designer to user are called user interfaces. The designer customizes a message for specific users according to their needs and expectations through interactive dialog. These messages are an interactive language embedded in the software. This article evaluates the effectiveness of communication interfaces in various situations.

Communicability is how software efficiently and effectively communicates the design intent and interactive principles to users. The communicability evaluation method obtains responses from users rating how well they receive intended messages and identifies areas of collapse in the communication flow. The user interface helps users communicate what they mean by what they say. Communicability can be of high design or low design depending on the physical display used to tell users what they must do. An example of high design is the CD Player software, which tells users what to do to playback compact discs on the computer with visual buttons similar to that of traditional non-computerized CD players. An example of low communicability design is a pull-down menu.



The communicability evaluation method provides users and designers with a process for direct two-way communication. This method identifies areas that are not clear or not understandable to users. Users participating in communicability evaluations are encouraged to express their feelings, expectations, and approval or rejection attitudes toward the interface design anytime during the evaluation phase. Several articles stress that designers and experts should not be used to evaluate software, for these types of users are able to interpret error

messages that the typical end-users could not, and therefore their expertise and background should eliminate them from participating in the evaluation process. Communicability evaluation, conducted at different stages throughout the design phase, allows continual modification of the product for the duration of the development stage. Formative evaluations help designers compare alternatives to the system during the design phase, and summative evaluations are conducted at the conclusion of product design to identify necessary changes that must be incorporated in the next release.

This article emphasizes the importance of users as evaluation testers in their role as user-to-system and designer-to-user communication feedback providers. Tagging, interpretation, and profiling are the three major steps in the communicability evaluation method. Tagging target will identify where communication has collapsed, interpretation charts the breakdowns, and profiling generates a depiction of the message communicated by the system.

Communicability evaluations generate vast amounts of information used by designers and HCI specialists in different ways. Engineers and HCI experts use this method to evaluate interface design, identify problems, and recommend redesigning options. Designers need communicability evaluation results for predictions and detection of interaction problems. Formative evaluations in the early stages allow designers to compare different design options or to assess the choices they have made. To solve or minimize problems, designers tag user errors in the application. Summative evaluations in later stages identify features to be changed or added in future software releases. Extensive evaluations should include a combination of usability, communicability, interviews, and interface inspection methods to maximize feedback. Several article stress that using designers or users with broad computer software experience as testing subjects will corrupt evaluation results.

Reference List

Norman, D., & Draper, S. (1986). *Cognitive Engineering. User-Centered System Design*. Hillsdale, NJ: Erlbaum Associates, 411-432.

Preese, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., & Carey, T. (1994). *Human-Computer Interaction*. Reading, MA: Addison-Wesley.

Sellen, A., & Nicol, A. (1990). *Building user-entered online help. The Art of Human-Computer Interface Design*. Reading, MA: Addison-Wesley.

Nodder, C., Williams, G., & Dubrow, D. (1999). Evaluating the usability of an evolving collaborative product – changes in user type, tasks and evaluation methods over time. *SIGGROUP '99 Conference Proceedings on Supporting Group Work,* ACM, Phoenix, AZ 150-159.

Engineers and designers often use and evaluate new technology before end-users. The functionality and interface that is acceptable to engineers and designers is not acceptable for the typical customer. This article focuses on the difficulties of evaluating an evolving product while it is in the development and design stages.

Microsoft Corporation tries to make products appealing to their target customers throughout the product life cycle from first release through the current release. The authors warn the reader that typical textbook approaches to usability testing do not always work in real world situations. In actual testing environments user types, user tasks, and usability evaluation methods change as the product matures.

This article documents actual obstacles incurred by the design team in the development and testing phase of Microsoft's product NetMeeting. Product usability must adapt as the needs of the design team changes, and engineers must develop a relationship with the team before implementing user-centered concepts successfully. As a product develops, designers must negotiate technology changes as well as modifications to the Internet environment. Testing includes visiting end-users at home, at their place of work, and in controlled laboratory environments to learn how they interact with the product in their daily lives, specifically focusing on functionality issues. The development team consisted of 12 people from Microsoft at the onset of the project in August 1995. Throughout the development cycle of the product, one usability engineer worked exclusively with the NetMeeting team. The entire team consists of 40 developers, program managers, testers, and support staff.

Microsoft developed the code for NetMeeting and compiled it every night in order for the most recent changes to be available for testing the next morning. The article outlines specific steps in developing and designing the user interface, their extensive review of literature, and reports on each set of beta tests. Usability evaluation methods had to adapt to match user groups and the technology upgrades in the product. A question posed early in the project was "Who are the intended users?" The design team focused on early adopters of technology and experienced Windows users who had the hardware capable of running NetMeeting.

Early in the evaluation phase, it was difficult for team members to work together comfortably, but compromise insured continued involvement, and as the team recognized the importance of usability contributions, engineers become more flexible in developing improved ways for evaluating the product. After a short period of adjustment, team cohesiveness led to the development of innovative evaluation techniques. As the team conducted evaluations, members had to be aware of the constant changes as the software continued to develop and mature. Over the life cycle of the product, the team noted an interesting change: the development stage focused on technology, but later in the project, the focus shifted to user scenarios.

This article outlines a critical element of usability: to develop ongoing communication between product team members and to stay focused on the end-user. This focus will ensure that the team is in a position to contribute the necessary user-centered information at the beginning of each design cycle. Selecting a representative test group from ordinary end-users will produce superior outcomes and as discussed in several of these articles, problems occur from testing technology savvy users, early adopters, or programmers.

Reference List

Chapanis, A., Ochsman, R., Parrish, R., & Week, G. (1972). Studies in interactive communication: The effects of four communication modes on the behavior of teams during cooperative problem solving. *Human Factors*, 14, 487-509.

Gentner, D., & Grudin, J. (1996). Human interface design models: Lessons for computer human interfaces. *IEEE Computer*, 29 (6), 28-35.

Grudin, J. (1991). Interactive systems: Bridging the gaps between developers and users. *IEEE Computers*, 24 (4), 59-69.