An Experiential Approach to Teaching Students About Usability and HCI

George Hackman Jr., Thomas W. Ferratt, Fantine M. Kerckaert

Introduction

As computers are utilized more in the workplace, the need for developing "easy-to-use" computer hardware and software has become more important. This has led to increased interest in testing the ease of use of software, i.e., software usability testing, and in researching the nature of human-computer interactions. Today's management information systems and computer science students will be designing and implementing the software systems of the future and, therefore, will need to understand the procedures of usability testing. They will also need to learn and appreciate questions addressed by human-computer interaction (HCI) research. Students in a scholars section of an Introduction to Computers and Information Systems class (MIS 103) at the University of Dayton, in Dayton, Ohio, were given the opportunity to increase their knowledge about usability testing and HCI research by participating in a "hands-on" project. This article describes and evaluates their experience, thereby providing guidance for incorporating usability principles in course work via direct experience in usability testing.

Overview of Usability Testing

Software usability testing is the means of assessing the ease of use of software packages. Usability testing allows the designer of the software to test empirically whether the current software package meets certain set usability goals. For example, if a software designer has designed a new word processor to reduce the time necessary to create and print out a simple business letter by half, usability testing can determine whether or not this goal has been achieved. Because of increased usability testing in the software development cycle, the need for computer scientists and management information systems specialists who understand usability testing has also increased. However, a detailed understanding of usability testing and its value cannot be obtained solely from the classroom. It requires active participation in a usability test.

Thinking out loud is typically used in usability testing to help gain useful information from the user about the software. Thinking out loud is useful because it helps the software developer understand the user's preconceived notions and final opinions about the software. The standard procedure used to elicit responses from users is simply to ask them to think out loud while they are working on specified software tasks. However, users often have difficulty verbalizing their opinions because of the increased workload imposed by thinking out loud. Bell Northern Research (BNR) Labs (Kennedy, 1989) have advocated using teams of two users in a version of usability testing called *Co-Discovery Learning*. In Co-Discovery Learning a pair of users work together to solve certain tasks. The verbalized information shared by two users as they work on the software together as a team represents a "natural thinking out loud" for the typical user. Research currently being conducted at the University of Dayton's Center for Business and Economic Research is empirically investigating BNR's claim that such a dyad approach actually yields more useful information than the standard single user thinking out-loud paradigm.

Traditionally, usability testing is done after the product is complete as a final quality check, but Whiteside and Holtzblatt (1988) have emphasized the importance of integrating usability testing early in the design cycle. After the product is complete, usability testing has only limited application because it is

often difficult to make changes after the fact. Usually, instead of making the necessary improvements, developers opt for a quick fix by implementing changes to the user manual. Unfortunately, these changes typically result in only marginal improvements in user performance.

However, by using tools such as rapid prototyping during software development, a reduced functionality model can be built that simulates the functioning of the final system. This process allows for testing and modification of the interface earlier in the development cycle. With the increased acceptance of this approach to usability testing in which the users' needs are emphasized from the initial stages of development, management information specialists and computer scientists will often be asked to interpret the results of usability testing. They will also need to be able to successfully incorporate these results into the products they are developing; thus it is important that they understand the source of this information and its meaning. By having students design and conduct a usability test in which they set up the test scenario, record problems encountered by participants, and put together a report indicating problems, the University of Dayton hopes to provide its students with a better understanding of this phase of the development cycle.

Project Overview

Students in the class were divided into five groups of six participants. Two of the participants served as users/learners of the software while the other four were observers. They were given three questions to answer through both a pilot and a formal test conducted at the Information Systems Laboratory (ISL), a unique, behavioral observation facility designed specifically for usability testing (described below). The questions were:

- What are the most frequent errors people make when seeking to build personal competency with a specific software tool?

- When attempting to learn a specific information processing tool in small groups, what do people ask and tell each other?

- How much time does it take to complete specific learning exercises when working in small groups?

The students were also asked to use their test observations to make recommendations about how people should learn to use software. It was expected that these research results would be applied to future MIS 103 students.

Each group was expected to present their results in the format of a journal article with sections including Introduction, Methods, Analysis and Discussion, and Conclusions. They had the option of producing their reports either on paper or on videotape. They were also told that they could combine the mediums in any way they wished. Furthermore, they had the capability of using taped episodes from the ISL session to prepare a short, video executive summary.

Procedure

Students were directed to prepare for the formal usability test in the ISL by reviewing prior research, investigating the use of the ISL, and conducting a pilot study.

To begin their review of prior research, the students were given articles by Catrambone and Carroll (1987) and Carroll (1987). The article by Catrambone and Carroll (1987), *Learning a Word Processing System with Training Wheels and Guided Exploration,* explores the utility of training novice users with a reduced functionality system designed to minimize common errors made when learning a new system. The article by Carroll, (1987) entitled *Minimalist Design for Active Users,* deals with ways of enhancing the active computer user's interaction with the computer by offering a "Minimal Manual" which supports active learning by providing concise instruction focused on easily understood goals. The article introduces the mini-manual as a middle ground between the "mechanized" following of an instruction manual and the "sometimes quite chaotic" method of guided exploration in which the user simply explores the system independently without reference to any user's guide. The students were directed to use the articles as a starting point for their research review.

The students were also directed to investigate the use of the Information Systems Lab before conducting their studies. To accomplish this the students were given a tour of the lab and were acquainted with the equipment that they would be using in the study. Each group also met individually with the graduate student in charge of the lab to address each group's unique concerns and questions about using the lab.

The final step before initiating the lab study was to conduct a pilot study. The pilot study was conducted in the computer labs where the students typically completed their assignments for the class. The students were instructed to videotape the pair of users/learners with a portable video camera as they worked with the software. The groups then analyzed the videotapes, looking for errors made by the participants and for any communications that occurred between the participants. They were required to submit a formal report on the pilot study. It was suggested that the students use the pilot study to help prepare them for the final lab study and formal report.

The formal studies were conducted at the University of Dayton's Information Systems Lab. This facility contains both an observation room and control room, and enables unobtrusive viewing of users' interactions with software via four cameras and a one-way mirror. All interactions are captured on videotape and can be subsequently analyzed to gain more information about the tests. During the testing process one camera is dedicated to the keyboard, another to the monitor screen, a third to the documentation, and a fourth to capture the nonverbal reactions of the user. The facility employs a quad-split device which enables the pictures from all four cameras to be displayed simultaneously on one 3/4" or VHS tape. The lab also has a complete 3/4" editing facility that can be used to produce a video summary report highlighting critical events which occurred during testing. The students used the facility

to produce and edit a videotape of the highlights of their study.

During the lab session, groups worked on either Lotus or Smart software. Smart is an integrated package that includes word processing, spreadsheet, graphics, and data base management capabilities. Lotus is a commonly used spreadsheet program. In both the pilot test and the actual test users of the software worked on class assignments. The assignments consisted of carrying out typical word processing, database management, spreadsheet and graphing functions with the software, and were designed to facilitate improving competency with the software.

Results

Conclusions about the project were drawn from the student papers and presentations, from their comments and suggestions about the studies, and from the observations of the graduate student who supervised all of the testing. The student papers followed the prescribed format. Papers were well written and interesting from a research point of view.

In terms of increasing their familiarity with usability testing, the students not only became familiar with the usability testing equipment, they also employed several techniques in their analysis of the data that are typically used in usability testing. Students marked the times of problems encountered during the formal session by assigning two members of the group to observe a specific computer user/learner. They then categorized these problems into different classes which included: keystroke errors, syntax errors, and conceptual errors. Another usability technique effectively utilized by the students was the construction of a videotape which included highlights from the usability test. For these tapes the students picked segments that illustrated the most frequent problems encountered by the users and incorporated them into the report. They later presented this report to the MIS class as a summary of their findings.

In answer to the question of what the students said to one another while working on the tasks, the class consensus was that the students primarily consulted with each other when they encountered problems. For example, one group wrote, "Considering the length of time the students worked side by side, their conversations were minimal... Most questions that they did discuss either regarded the directions in the text or which command they should choose for a certain operation." And, after commenting that one of the group members worked ahead at his own pace, despite the fact that the other member of the group lagged behind, they further noted, "This infers that when people work together the completion time can be decreased because they do not significantly hold each other back, but do act as easy references when there is a question."

Another group commented that the primary interaction was one of checking results. In one group, the student experimenters observed, "... at this time Brian became a very important asset to the study. He was able to look over Mike's formulas and help him see what went wrong. They (the participants) frequently compared their screens to see if their values matched what the other person had." The same group also pointed out, "Most of the major comprehension errors that the two subjects ran into were quickly solved with the use of selective communication."

In answer to the question of how to improve the learning process, several useful comments were made. Most group papers emphasized that improving the clarity of the instructions in the documentation manual would be helpful. One group pointed out that it was important to understand the basic commands before beginning the assignment: "The authors of the Learning Smart text have done this specifically in the Get Oriented section, by including several beginning steps which familiarize the user with the system before the actual assignment is started." A second group also similarly commented, "Theoretical errors may be reduced by studying the SMART manual more thoroughly." A third group suggested, "It is also a good idea to always choose a comfortable steady pace and not try to rush through the entry of material. In the end, rushing only leads to 'dumb' mistakes that only use up time." These observations and analyses made by the groups indicate that the students were actively considering research questions in human-computer interaction.

When asked about what recommendations they would make for people who were learning to use software tools, the students made several interesting comments. One was to use assignments that did not give such specific instructions. Another student pointed out that the study demonstrated that it was important to "follow the logic. Don't just punch buttons." In a related comment a student wrote, "Choose assignments that are more open. Step-by-Step guidance in the book creates a monkey see, monkey do attitude that detracts from learning." Another student noted, "Taking the time previously to learn instructions and functions helps to save time in the future."

Conclusions

The primary goals of this exercise were to introduce students in an introductory MIS class to usability testing and to give them experience in conducting human-computer interaction research. From the answers given by the students to the research questions it is clear that they gave these research questions considerable thought and that they effectively used the data available to them. In terms of improving the students understanding of usability, the study enabled the students to work in a usability lab and to experience and participate in the process of usability testing. They gained an understanding of usability testing by actually conducting a usability test. Most of the students enjoyed the experience, and several students asked about how they could get more information about usability testing and human-computer interaction research.

Given the benefits of the experience, the cost of putting together the study was relatively low. The students invested about three hours on a given day for the actual study and about seven hours to complete the other requirements for the study. In terms of resources utilized, it took several hours per week of graduate student time, the use of a facility for con-

ducting the study, one VHS tape, and two 3/4" tapes per group. However, it would not require a high technology facility similar to the ISL to duplicate the experience for students at other universities. A much simpler set up, similar to the one employed for the pilot study, would still give the basic usability and HCI research experience.

References

Catrambone, Richard and Carroll, John M., *Learning a Word Processing System with Training Wheels and Guided Exploration*, Proceedings of SIGCHI + SIGGI, 1987, pp. 169-174.

Carroll, John M., *Minimalist Design for Active Users*, Human-Computer Interaction – Interact '84 B. Shackel (ed.), Amsterdam: North-Holland, 1985, pp.34-44.

Kennedy, Susan (1989), *Using Video in the BNR Usability Lab*, SIGCHI Bulletin, 21 (2), 68-71.

Whiteside, J., and Holtzblatt, K. (1988). *Usability Engineering: Our Experience and Evolution*, in M. Helander (Ed.) *Handbook of Human-Computer Interaction* (791-817). North-Holland: Elsevier.

About the Authors

George Hackman, Jr., is currently a Human Factors Engineer with IBM in Bethesda, Maryland. He earned a B.A. in Psychology and German in 1988 and an M.A. in Experimental Psychology/Human Factors in 1993 at the University of Dayton. His areas of interest include Usability Testing Methodologies, Multimedia Applications and User Interface Design for Production Worker Environments.

Thomas Ferratt is Professor of Management Information Systems in the Department of MIS and Decision Sciences at the University of Dayton. He earned his Ph.D. in Business Administration and M.B.A. from Ohio State University and his B.B.A. from the University of Notre Dame. He has worked as a computer programmer, systems analyst, and project manager. Besides an interest in usable systems, particularly to support managerial work, he has active research interests in the management and education of information systems professionals and users of information technology. He is currently the Chair of ACM's Special Interest Group on Computer Personnel Research (SIGCPR).

Fantine Kerckaert is the Associate Director of the University of Dayton's Center for Business and Economic Research. Her primary responsibility includes administration of grant and contract research in the areas of usability testing and marketing. She has a B.A. in Psychology from Miami University in Oxford, Ohio, and an M.B.A. from the University of Dayton.

Addresses:

George Hackman, Jr., ImagePlus Systems Development, IBM Corporation, 10401 Fernwood Rd., Bethesda, MD 20817, USA.

Thomas Ferratt, Department of MIS and Decision Sciences, University of Dayton, Dayton, OH 45469-2130, USA; email: ferratt@udavxb.oca.udayton.edu.

Fantine Kerckaert, University of Dayton, Center for Business and Economic Research, Anderson Center, 300 College Park, Dayton, Ohio 45469-2110.