Increasing Productivity and Reducing Errors through Usability Analysis: A Case Study and Recommendations.

Constance M. Johnson, B.S.N.[1,2], Todd Johnson, Ph.D.[2], Jaijie Zhang, Ph.D.[2]
The University of Texas M.D. Anderson Cancer Center[1] and The University of Texas School of Allied Health Sciences, Department of Health Informatics[2], Houston, Texas.

ABSTRACT

The usability problems of a system often occur due to inattention to well-documented and well-established design guidelines and heuristics. These problems often lead to increased errors, user dissatisfaction, and often user abandonment. Although there are a plethora of design principles, programs are still being constructed without integration of these principles. One family history-tracking program was examined for usability compliance. In addition to a user analysis, a task analysis was conducted comparing the designers' conceptualization of tasks with the users' conceptualization of these tasks. A cognitive walkthrough was then conducted on these tasks. Finally, a keystroke level model was used to show the differences between the execution times of these tasks. This model showed a serious mismatch between the designers and users conceptions of the task. The suggested redesign showed timesaving for each of these tasks.

INTRODUCTION

The prevailing goal in the design of user-centered software is to create a system that not only has utility, but also usability. A program that fully actualizes both of these factors gives rise to a system that is easy to learn and use, increases user acceptance and utilization, increases user productivity and satisfaction, decreases user errors, and decreases user frustration and decreases training requirements. Regrettably, due to appreciable time and financial constraints, system developers are not always able to consider all of the attributes that are critical in the design of functional and usable interfaces. In addition, many designers are unaware of the importance of user-centered software design. Yet, poorly designed systems and confusing user interfaces result in frequent user errors, a decrease in productivity, and an increase in user frustration. FDA data collected between 1985 and 1989, demonstrated that 45 to 50 percent of all device recalls stemmed from poor product design (including problems with software) [1]. Furthermore, the FDA recognizes that a poorly designed user interface can induce errors and operating inefficiencies even when operated by a well-trained, competent user. In response, the FDA has revised its Good Manufacturing Practice regulations to include specific requirements for product usability [1]. They have also published guidelines for interface design and usability testing [2] and produced a continuing education article that specifically covers human factors issues [3].

Although the Agency for Health Care Policy and Research [4] cite diagnostic inaccuracies and system failures as two sources of errors in the healthcare system, they also suggest "computer-assisted technologies" as a means to improve patient safety, prevent errors and enable clinicians to make better treatment decisions.

To err is human, even in systems with exemplar design. Nonetheless, systems that are designed to match users' capabilities, requirements, and expectations will result in a system that minimizes human error. This is most important when designing medical decision support software that depends upon capable human intervention when making patient care decisions.

Given the importance of user-centered software design, it is imperative that healthcare administrators and professionals, and software designers adopt appropriate procedures and guidelines to ensure that all healthcare software follows sound user-centered design principles. These include guidelines on the purchase of new software, development of in-house software, analysis and redesign of software already in use, and user education on the reporting of usability problems.

This paper demonstrates, through a case study, one approach to analyzing and redesigning healthcare software. In particular, it shows how a system that was developed without considering usability principles can be redesigned to create a system where the users' mental model can match the designers' conceptual model utilizing good human user-centered design guidelines, heuristics, and theories.

ANALYSIS OF ONE SYSTEM

In 1997, a cancer genetics family history-tracking program was designed at The University of Texas M. D. Anderson Cancer Center (UTMDACC) for conducting cancer genetics studies as a part of an academic program. Microsoft Access 97 was used to construct the back-end database and Microsoft Visual Basic 5.0 was used to construct the front-end interfaces. Its main function was to provide practitioners with the tools to create readable and consistent pedi-

1067—5027/00/$5.00 © 2000 AMIA, Inc.

grees (family trees). It combined the functionality of a relational database with data storage, manipulation, and retrieval, and a pedigree drawing component. The system included four main components, the data entry interfaces, predefined reports, data editing screens, and a link to a pedigree drawing program (Cyrillic 2.13, Cherwell Scientific Publishing Inc., UK). Although this tracking program has much functionality, an initial user survey and usability analysis revealed important missing functions and a host of usability problems. The tracking program needs more externalization of information and perceptual cues for operating procedures so as to increase the directness of the interface. Without good functionality and usability, it will remain limited in its utility in clinical, research, and educational settings.

In 1999, user and task analyses were conducted on the cancer genetics family history-tracking program to determine its usability and functionality problems. The users were analyzed on both the horizontal and vertical dimensions, as discussed below in the user analysis. The task analyses included an analysis of usability heuristics including Shneiderman's [5] Eight Golden Rules of Design. In addition, the tasks were analyzed by a first time user's and the designers' conceptualization of the task; a verbal protocol analysis; and a Keystroke Level Model [6] to show actual execution times. These were compared to a cognitive walkthrough [7] of specific tasks. Finally, recommendations for change were determined, and a final Keystroke Level Model [6] was used to show the incorporated changes with the predicted execution times. The final execution times were then compared with the designers' execution times and time saved was determined.

USER ANALYSIS

Users are categorized along a horizontal dimension, according to their different types of tasks, and a vertical dimension, according to their different levels of experience for specific types of tasks. Along the horizontal dimension, existing users of the tracking program are health care professionals (nurses, genetic counselors, researchers/scientists, medical doctors), research assistants, students, data managers, project managers, programmers, system administrators, and secretaries. Along the vertical dimension, existing users are computer literate at the novice, intermediate and expert levels, males and females with education levels ranging from high school to post graduate degrees, and over the age of twenty-one. Frequency of use of the system ranges from casual to intermittent to regular usage. A comparison of the user characteristics showed that only the nurses, genetic counselors, data managers, and data entry personnel are regular users of the system. With the exception of the data entry personnel, all other regular users have a high level of domain knowledge. Although this same group showed a limited amount of time to learn the system, they were very motivated to learn the system. Finally, all of this subgroup had intermediate general computer knowledge and at the very least were familiar with other software programs such as Microsoft Office.

The health care professionals are using the family history tracking program for direct patient care (e.g., familial genetic risk assessment, genetic counseling, etc.) and indirect patient care (e.g., research). The nurses and/or genetic counselors use the system to input family history data, print pedigrees, edit data, and query these data for requested reports. The physicians, scientists/researchers and other medical personnel only read these data, use the data for direct and indirect patient care, print pedigrees, edit data, and query these data for research. The data manager and project manager are involved in every aspect of system installation and maintenance, data editing/cleaning, and querying these data for requested reports. The system network administrator simply controls who has access to the program under the direction of the project manager. The data entry personnel only enter these data and the secretaries are responsible for only sending out family history questionnaires upon request.

Although this is a rather homogenous population in terms of computer knowledge, most problems have arisen with this population when the tracking program was not consistent with other programs they have used in the past. In this regard, there have been user errors, complaints, and loss of productivity.

TASK ANALYSES

Task analysis [8,9] is the process of identifying system functions that have to be performed, procedures and actions that have to be carried out to achieve task goals, and information to be processed. The purpose of task analysis is to ensure that only the necessary and sufficient task features that match users' capacities that are required by the task will be included in system implementation. Features that do not match users' capacities or are not required by the task will only generate extra processing demand for the user and thus make the system harder to use.

HEURISTICS VIOLATIONS

The cancer genetics family history-tracking program violates many design heuristics. According to Shneiderman [5] the rule of design: "Strive for Consistency" is the most commonly disregarded of all of the rules. Consistency not only concerns screen design, but should also address whether the users' conceptualization matches the designers' conceptualization. The family history-tracking program does not echo consistency with other programs a user may have encountered in the past. Although the program is consistent in its layout, use of color, and fonts, the main problem lies between the users' expectations and the designers' conceptual model. For example, the user, must perform actions without prompts; there are non-functional menus, and misleading labels.

"Informative feedback" is another design guideline which, proposes that a system should offer frequent positive and negative feedback for user actions [5, 8]. Although the tracking program does have some edit checks, they are not consistent with other applications. For example if a user leaves mandatory fields blank, a dialog box does not appear informing the user of their error. In three of the date fields, if the user enters the wrong number of characters, the cursor will not move from that field, but the program does not inform the user what error was committed. Unfortunately, this type of feedback forces the user to guess what they did wrong. Use of system feedback that has a positive tone, consistent terminology and appearance of feedback close to the source of the error will increase user satisfaction and improve user performance.

Constructive error messages are another design heuristic that should give users clear instructions regarding an error they may have committed. In the family history tracking program, error messages that do appear in dialog boxes are not presented with "user-centered phrasing" [5]. Although, they are specific to the problem, the terminology is not understandable and they often contain a code number. For example, the following message appeared in a dialog box after the user performed an action; 'Form_load ERROR. [3251] Operation is not supported for this type of object." Negative words such as ERROR, or INVALID, according to Shneiderman [5] can "heighten user anxiety and increase the chances of further errors". Additionally, Shneiderman [5] states that upper case messages should only be used for very serious problems. The use of messages without a clear syntax as in the previous example does not allow the user to feel in control of the application. According to Shneiderman [5] the following guidelines should be used to construct satisfactory error messages: positive tone, natural language, clarity, understandable format, and consistent terminology and placement.

"Permit easy reversal of actions", is another design heuristic suggested by Shneiderman [5] which should allow users to perform actions that they know they can completely reverse in the event of an error. In the cancer genetics family history tracking program there is no undo menu feature in the Edit menu. This can lead to user anxiety and fear of making a mistake. This omitted feature can also be costly in terms of productivity.

Finally, the system should not force the user to employ what Norman [10] terms more "knowledge in the head" than "knowledge in the world". When an application requires more "knowledge in the head" according to Norman [10], the learning time is lengthened, ease of use initially may be low and the information may not be easily retrievable. This can be a source of great frustration to the user. This tracking program does not offer any on-line help, and forces the user to remember many actions that are not consistent with other applications they may have used in the past. In short, it makes too many mental demands on the user.

This analysis showed that the system does not allow enough worker flexibility. Although it should be a closed system, it does not give the user directions on how the task should be performed in the event of a disturbance. While it is very detailed, it does not include enough constraints, does not give enough detailed direction to the user, thus leading to more data entry error. Finally, it does not allow the user the variability to deal with new situations. Many constraints need to be added to the system to decrease data input error, thus allocating less time for the user to find and correct errors and more time to care for patients.

USERS' AND DESIGNERS' CONCEPTUAL MODELS

A total of four tasks were analyzed at the designers' conceptualization of the task and the first-time users' conceptualization of the task. The first time user's conceptualization was determined through a verbal protocol analysis. These tasks were then analyzed using the Keystroke Level Model [6] to show differences in execution times by summing up the time taken for each keystroke, pointing, clicking, thinking, waiting, and deciding. Additionally, a cognitive walkthrough [7] was conducted on each specific task. The main point of the cognitive walkthrough is to determine what steps the user will take to complete a goal, to identify potential usability problems, and to determine the ease of learning the system. Finally, recommendations for change were determined, and a final Keystroke Level Model [6] was used to show the predicted execution times of the suggested redesign. All of the execution times were then compared and time saved was determined. The tasks analyzed were Loading the Cancer Genetics Database Application, Opening the Family History Data Entry Interface, Beginning Entry of Data (use of the add button), and Entering

New Family (Proband only). Each of these tasks had either significant problems or merely visibility problems. Table 1 shows the differences in the execution times in seconds at the keystroke level.

Table 1. A Comparison of Execution Times for Analyzed Tasks

Tasks	Designers' Concept (seconds)	Users' Concept (seconds)	Redesign Concept (seconds)
Load Program	9.22	27.02	8.90
Open Interface	8.80	16.90	4.50
Begin Data Entry	3.65	3.65	0
Data Entry of Proband	141.64	161.36	128.06
Total Time	**163.31**	**208.93**	**141.36**

The total time required at the keystroke level according to the designer conception to perform the four tasks shown in Table 1 was 163.31 seconds or 2.72 minutes. The actual time shown by the first-time user through a verbal protocol analysis was 208.93 seconds or 3.48 minutes. The execution time according to the suggested redesign of the interface was 141.46 seconds or 2.35 minutes. The redesign saves the user 21.85 seconds (designers' conception minus the redesigned conception) per proband entered. If the user enters 200 probands per year, total timesaving would be 4382 seconds or 1.21 hours. However, not only probands are entered into this database, family members are entered as well. If 2000 additional accompanying family members were entered, 27,160 seconds or 7.54 hours would be saved for a total savings of 8.75 hours plus additional time saved from not having to modify data. Redesigning this database not only gives the users a more usable interface, but also saves man-hours for entry of family history information. Although, this is a small amount of time over one year, it reflects only the time savings of an expert user. We expect much more significant time savings will be gained in practice, because new users will more easily learn to use the software, and the occasional user will find the software easier to use.

The cognitive walkthrough analysis (CW) of the cancer genetics family history-tracking program disclosed many problems that a first-time user would encounter with functionality, ease of use, and system learnability. These problems began with loading of the application through data entry of the first family member (proband). The severity of the problems showed how much the users' progress was impeded, and how the repair of these problems could save the users' time and affect the users' performance. This walkthrough defined the conjectured steps the user would take with each task. The CW proved to be a very valuable process. It is suggested that if this usability test is conducted early in the design process on a prototype before release of the application, numerous problems can be identified and corrected to make the interfaces within the application, easier to use and certainly more functional. Further study of the application is needed to address the other problems with the interfaces not examined in the study.

RECOMMENDATIONS FOR REDESIGN

These usability analyses revealed important missing functions and a host of usability problems. The cancer genetics family history-tracking program needs more externalization of information through visualization and perceptualization of procedures through perceptual operators. With funding from the Texas Higher Education Coordinating Board under the Advanced Research Program, the functionality and usability of the tracking program will be improved by employing human factors engineering techniques. This project is unique due to the nature of the software and its emphasis on human factors engineering, which is the study of the interaction between the user, the device, and the task in a certain environment.

This project will improve the usability of the family history-tracking program and generalize it through a five-step process that utilizes human factors engineering methods to iteratively refine and test the cancer genetics family history-tracking program. There are well-published guidelines and principles for designing user friendly systems. Incorporation of these user-centered design methodologies [5, 8] will lead to the widespread use of an application. The redesign will begin with a complete usability analysis, which will consist of a user analysis, an environmental analysis, a comparative analysis, and a comprehensive task analysis. The results of the usability analyses will be used to iteratively develop and test a prototype of the new version of the tracking program. The prototype will be tested using GOMS and Keystroke Level Models [6]. These types of task analyses will be used to determine whether the user goals are being met and what the user will need to learn to accomplish their goals within the interface. The prototype will then be evaluated by conducting small-scale usability studies, which will include the cognitive walkthrough, think-aloud verbal protocol, usability survey, and heurisitic evaluation methodologies. These types of usability evaluations will uncover functional and interface design flaws and measure subjective user satisfaction. These steps overlap with the modification of the program because the technical limitations revealed during program modification may require revisions to proposed interface changes. Any proposed changes will need to be tested before being incorporated into the final program. The existing program will then be modified to incorporate the user-interface and functionality changes identified and evaluated in the previous steps. Finally the modified program will be compared to the old program using a controlled experiment to determine if the redesign decreases the error rate, increases productivity, and increases user satisfaction ratings. These dependent variables will be measured with two dependent samples. This within-subject design will control for individual variability and reduce subject bias. This will be followed up with a survey to assess the perceived usability of the application, the ease of use, speed of task performance, user productivity, and users' perceptions of errors. The results of this redesign will be generalized so that they can be applied to the development of other clinical information systems with similar features.

CONCLUSION

This paper shows one method for a systematic approach to improving system usability. These methods can have important benefits toward user acceptance and use of a program. In addition to usability analyses and studies, two other recommendations are proposed to improve system usability. Number one, information technology groups need to be educated on the principles of good design. Number two, the user culture needs to be educated to report all usability problems. It is easy for a designer to develop an application without consideration of usability and functionality issues, without consideration of whether the system is easy to learn, easy to navigate, easy to remember, or efficient. According to Norman [10], the user's mental model, which is acquired through his contact with the system, should mirror the designer's conceptual model displayed through system's model. A breach in this communication between the designer and user results in usability and functionality problems. Using known usability and functionality guidelines, heuristics, and theories, we have proposed how one system with many functional and usability problems can be redesigned into a system that will not only have utility, but also usability, and thus user acceptance.

ACKNOWLEGEMENT

This research is supported by a grant from the Texas Higher Education Coordinating Board under the Advanced Research Program, Grant # 011618-0077-1999.

REFERENCES

1. Food and Drug Administration: Center for Devices and Radiological Health (April, 1998). Human Factors Implications of the New GMP Rule. Overall Requirements of the New Quality SystemRegulations.[On-line].Available: http://www.fda.gov/chrh/humfac/hufacimp.html

2. Food and Drug Administration (1996). Do It By Design: An Introduction to Human Factors in Medical Devices. [On-line]. Available: http://www.fda.gov/chrh/humfac/doit.html.

3. Food and Drug Administration (1997). Improving Patient Care by Reporting Problems with Medical Devices. A MedWatch Continuing Education Article. [On-line]. Available: http://www.fda.gov/medwatch/articles/mdr/mdr.pdf.

4. Agency for Healthcare Research and Quality. Medical Errors: The Scope of the Problem. Fact Sheet, Publication No. AHRQ 00-P037, Rockville, MD. [On-line]. Available: http://ahrq.gov/clinic/errback.htm

5. Shneiderman, B. (1998). Designing the User Interface: Strategies for Effective Human-Computer Interaction. Reading, MA: Addison Wesley Longman, Inc.

6. Kieras, D. (1993). Using the Keystroke-Level Model to Estimate Execution Times. The University of Michigan, Manuscript.

7. Nielsen, J., & Mack, R. (1994). Usability Inspection Methods. New York: John Wiley & Sons, Inc.

8. Nielsen, J. (1993). Usability Engineering. New York: AP Professional.

9. Vicente, K.J. (1999). Work Domain Analysis and Task Analysis: A Difference That Matters. Manuscript.

10. Norman, D.A. (1998). The Design of Everyday Things. New York: Doubleday.