Association for Information Systems
AIS Electronic Library (AISeL)

SIGHCI 2003 Proceedings

Special Interest Group on Human-Computer Interaction

2003

When Information Technology Design Favors Form over Function: Where is the Value-Added "Tipping Point"?

Rita M. Vick
University of Hawaii, vick@hawaii.edu

Brent Auernheimer
California State University Fresno, brent@csufresno.edu

When Information Technology Design Favors Form over Function: Where is the Value-Added "Tipping Point"?

Rita M. Vick
University of Hawaii
vick@hawaii.edu

Brent Auernheimer
California State University Fresno
brent@csufresno.edu

ABSTRACT

Performing usability analysis early in the design process results in lower overall development, deployment, and maintenance costs. Pre-development user and task analysis through questionnaires, observation, low-fidelity prototyping, and usability testing enables productive interactive testing of subsequent operable system prototypes. This helps assure a positive return on investment in information technology. When user-centered design assessment is supplanted by assumptions about user, task, and work environment, the result is often production of applications embellished with functionality unrelated to the user's task. Surveys were administered to elicit user perception of system usability and usefulness and of satisfaction with intra-team interaction. This was the first step in determining the relationship between form and function for users of a Synchronous Distributed-Decision Support System (SD-DSS). It was anticipated that the teamwork process would be most troublesome while the SD-DSS would be perceived as easy to use and functional. The reverse proved to be the case.

Keywords

Distributed teamwork, decision modeling, usability, cognitive fit, task analysis.

INTRODUCTION

A frequent assumption of end-user system designers is that creating systems that are usable and easy to use will help guarantee satisfied users and provide organizations with commensurate returns on investments in information technology (IT). This is true provided that designers do not make additional assumptions about what constitutes usability and ease of use in a given work context. Lack of careful user, task, context, and coordination analysis often leads to systems that provide more form than function. In their quest to present products that make a task easy and promise to reward business investment in the technology with competitive advantage, designers may inadvertently embed required functionality behind an interface that is overly simple, adorned with exotic features, or does not fit the cognitive requirements of the task. In either case, the result is a disappointed user and a disappointing return on investment. In the quest for usability, function has been sacrificed for form resulting in inadequate attention being devoted to providing cognitive fit between the user and the task and to assuring that the user receives support for understanding the task. The described study was conducted in order to find out more about how human-computer interaction (HCI) design principles can be leveraged to counter this design trend.

GDSS FUNCTIONALITY

An important aspect of solving complex multicriteria problems is the use of software support to structure the decision process and assist groups of decision makers with assigning preferences and weights to criteria. Decision Support Systems (DSSs) tend to be complex, and the use of software that provides modeling capability can result in a perceived increase in the complexity of the task (Limayem & DeSanctis, 2000). Attempts have been made to design explanation and automated decision guidance into Group Decision Support Systems (GDSSs) to enhance decision models building by providing cues to direct decision makers toward correct structuring and implementation of model components.

Dennis, Haley, and Vandenberg (1996) and Benbasat & Lim (1993) found that, although groups benefit from decision modeling as evidenced by improved decision quality, model building is time consuming and difficult. According to Limayem & DeSanctis (2000), both of those studies determined that use of GDSS technology tended to reduce consensus, decision confidence, and overall satisfaction despite the fact that decision quality improved. Decision makers tend to avoid decision aids because they reveal conflict and place a cognitive load on the user.

In accord with the findings of the present study, Tuttle and Stocks (1997) believe that most software puts too much emphasis on ease of use and too little emphasis on decision maker understanding of the models they are building. The suggested solutions include embedding explanations that require little cognitive effort and provide more problem-structuring support for group cognition. The suggestion has been that cognitive feedback could provide information about preferences and model structure (Te'eni, 1991) by, for example, calling attention to inconsistencies in decision-maker judgments. Bjorkman (1972) suggested that cognitive feedforward might provide explanation before each step of the model-building process. The premise is that "feedforward" assistance may "attenuate cognitive strain by providing decision makers with information that otherwise would have been learned through feedback." (Limayem & DeSanctis, 2000, p. 388).

Beyond Interface Design

Adaptive Structuration Theory (AST) (DeSanctis & Poole, 1994) maintains that the productive potential of a GDSS is only partly determined by the features designed into the system. More importantly, adoption and continued use of a collaborative technology depends on how the features of the system are applied by users in the work setting. Disengagement between intended and actual use can be caused by the way the system is introduced into the organization, inadequate knowledge of the system's purpose and functionality, or use of the system for unintended (e.g., political) purposes. If this happens, the system will fail to gather a sufficient number of users and will not become an integral part of the organization.

AST was deployed in a case study in a natural work setting to study the technology adaptation process in virtual teams engaged in new product development (Majchrzak, Rice, Malhotra, King, & Ba, 2000). It was found that changes in the alignment of work structures as initially set by the team, flexibility of structures, and occurrence of discrepant events serve to mediate the pre-existing structure/appropriation relationship (Majchrzak et al., 2000, p. 595). This study extended AST to include these mediating factors between existing structures and appropriation of the technology by users.

In order to produce useable systems, designers need to model users' task knowledge and represent this knowledge in a way that provides a good cognitive fit between the user's problem-solving strategies and the nature of the tools provided by the technology. The problem representation and tools must match the characteristics of the task (Umanath & Vessey, 1994; Vessey & Galetta, 1991). Multiple converging design techniques need to be deployed to develop a working understanding of the individual field of practice in order to model the cognitive and interactive nuances that account for what constitutes expert knowledge of a given domain (Potter, Roth, Woods, & Elm, 2000). If the user is not a domain expert, then additional system and training support is required to avoid high rates of error or low usage levels. Usability is a concept that is often misunderstood and so is often oversimplified. It is a complex, multi-faceted concept that represents individual elements of user capability and task demand that impact one another and take on emergent properties in complex work environments. "Usability" must be understood to be inclusive of multiple independent concepts including user satisfaction, system effectiveness, context of use, applied task knowledge domain, and the level of expertise of the user (Frøkjær, E., Hertzum, M., & Hornbæk, K., 2000).

THE STUDY CONTEXT

The author designed and taught a course titled "Computer Supported Collaborative Work (CSCW) in Practice" for four semesters to a total of 74 senior-level undergraduate computer science and engineering students. Class sizes ranged from 18 to 27 students randomly assigned to self-directed teams of three to five participants for the virtual teamwork part of each class meeting. Teams were dispersed throughout a computer lab and communicated only through NetMeeting chat. Lab sessions were 60 to 80 minutes in duration. Teams remained intact throughout the semester and were free to assign members to particular tasks or to work on the task as a group through application sharing, sending files to other team members via file transfer, and accessing information as required from e-mail, the course Web site, or the Internet. Because the nature of the task scenario was complex and did not have a single "right" answer, teams were also free to exercise creativity and critical thinking in pursuit of appropriate responses to the series of variations on the overall task that were presented to them at the beginning of each lab session.

The four courses ran from 10 to 15 weeks in length. Observational, experimental, and survey methods were used to assess group process and outcome. Surveys were administered at the midpoint and at the end of each course to elicit information from participants regarding their satisfaction with their teams and with the software used for communication support (NetMeeting) and the software used for decision modeling (TeamEC™).

Participants generated ideas and determined their relevance, planned the problem solution, determined which criteria were of prime concern, developed alternative ways of meeting the criteria, and assigned weights to each model element using NetMeeting chat. This enabled capture of time-stamped transcripts of team interaction. Participants used the NetMeeting whiteboard to share information, visualize solutions, and as a form of team memory to capture the progress of decision model development.

During the lab sessions, each team assumed a real-life role within an assigned scenario. Roles were rotated so that each team was exposed to each role. The teams were expected to complete a decision model within the timeframe of the lab period. Each class worked consecutively with two scenarios. The first scenario was designed to familiarize the students with working as a team in a simulated distributed Group Decision Support System (GDSS) environment devised by using NetMeeting connectivity to support TeamEC™ as a shared application. This first scenario dealt with evaluation of alternative solutions for an ill-defined policy issue ("How to Revive Hawaii's Economy"). Participants assumed the perspectives of government, business, education, and organized labor. In the second scenario, participants assumed the roles of employees of a "tech startup" company where teams worked as a "task force" responsible for the design, development and marketing of a collaborative system and assumed corresponding roles.

The first scenario presented a broad policy-based decision problem that was designed specifically to be removed from the technical computer science and engineering learning domain of the participants. This was done to focus students' learning on decision-making as a process and as a particular type of problem solving. Removing learning to a domain in which students were not expected to be expert also enabled experiential learning unencumbered by the need to excel. In this context, students were free to move forward and backward within the problem context as they experimented with learning to think critically about decisions as unique problems. The problem for Scenario 1 was one that is common to all locations and cultures (revitalizing the local economy) so that it could be readily understood by all participants regardless of individual demographic differences. The second scenario focused on a collaborative system design, development, and deployment decision problem specific to the participants' domain of expertise.

The primary measure of group performance was decision model quality. The decision modeling software (Expert Choice, Inc., http://www.expertchoice.com/) is designed for analyzing, synthesizing, evaluating, and justifying complex decisions in a group setting. The software brings structure, organization, and coherence to the decision-making process and supports a multi-objective decision making method based on the Analytic Hierarchy Process (AHP) methodology (Saaty, 1980) in which elements in a non-binary tree structure are subjected to a series of pairwise comparisons to assess their relative value, likelihood, or desirability. It is one of several optimization methods that decision makers can use to reconcile problems having multiple conflicting objectives.

At the end of each scenario, teams competed in a "Face-Off" to determine which team could produce the best decision model. During the lecture portion of the class session immediately following each Face-Off, all the models were discussed, the team with the best model was proclaimed the winner of the competition, and the members of winning team were awarded a small prize. At this time, students completed the surveys that revealed their perceptions of the usefulness and ease of use of TeamEC™ and NetMeeting. A separate survey focused on individual team members' perceptions of and satisfaction with teamwork. The surveys served as benchmarks to gauge team progress (team perceptions of the software support and the effectiveness of their work as a team in solving the assigned problems) and to provide feedback to the instructor. They also provided information intended to help understand more about how the design of interactive collaborative systems helps or hinders the user.

EFFECTIVENESS OF THE GDSS USED IN THE STUDY

It is important to discuss the extent to which TeamEC™, as a type of GDSS software, performed in terms of the above considerations. Although the interface is clear and relatively easy to use, it resulted in a perceived increase in complexity for participants in all four courses. TeamEC™ has embedded explanation into the design of the interface. This is accomplished through online help, explanatory comments that identify functions, and model element definitions. The only automated decision guidance, the "inconsistency ratio" indicator, appears at the end of the pairwise comparison process. This index assists structuring and implementation of model components by indicating whether inconsistent judgments have been entered into the model during the pairwise comparison process. Teams benefited from the guidance provided by these elements as well as the decision modeling process, itself, and all teams' models improved over time.

In the present study, the instructor compensated for lack of built-in feedforward or feedback support in TeamEC™ by (a) being constantly available during lab sessions for consultation and (b) using e-mail to make comments and suggestions to assist teams' understanding of the model-building process. E-mail feedback was sent after each lab session to each team. Since students referenced this e-mail feedback during subsequent lab sessions, the effect was to provide problem-structuring support for group cognition by providing team-specific help with model structuring and content problems as teams progressed. This form of feedback became feedforward assistance since it provided explanation for each subsequent model-building session and so alleviated cognitive strain on teams. Three general types of e-mail were sent to teams: (1) maintenance (file naming/saving, crash recovery), (2) structuring (tree structure validity), and (3) content (tree content validity).

Microsoft NetMeeting was used to simulate synchronous distributed teamwork in a computer laboratory. It provided communication support via text chat and enabled application sharing so that teams members could work simultaneously on collaboratively building decision models. Students communicated only via text chat. Other features of NetMeeting available for use by the teams were the whiteboard, shared clipboard, and file transfer. Teams also had access to course notes on the class Web site and to e-mail for referencing instructor feedback. Web access enabled searches for external information that might assist problem solution. This use of NetMeeting resulted in multiple windows open simultaneously on each participant's desktop.

PARTICIPANT SATISFACTION SURVEY RESULTS

Surveys were administered at the benchmark points (immediately following each Face-Off) of each of the four courses yielding two sets of survey results per course. Students were asked their opinions of TeamEC™, NetMeeting, and their experiences of working in a team. Results of the surveys are shown in Table 1. IT system use has been found to be strongly correlated to perceived usefulness and perceived ease of use (Mahmood, Hall, and Swanberg, 2001). The survey instrument used for NetMeeting and TeamEC™ was "Measurement Scales for Perceived Usefulness and Perceived Ease of Use" (Davis, 1989), a 7-point Likert scale ranging from "Agree" to "Disagree" or "Satisfied" to "Dissatisfied" where lower scores indicate greater satisfaction. A modified version of an instrument for measuring meeting success (Davison, 1997) was used to assess participants' perceptions of their team's effectiveness.

It had been anticipated that the participants in this study would assign preference for the three major survey subjects in the following order: (1) TeamEC™, (2) NetMeeting, and (3) Teamwork. The assumption was that computer science and engineering students would find the structure and elegance provided by TeamEC™ cognitively compatible with their technical domain of expertise. NetMeeting would also be compatible with their skills and interests, but to a lesser extent because its communication support functions were less straightforward. NetMeeting's whiteboard, used for brainstorming, required original thinking. NetMeeting's chat feature contained the usual inefficiencies associated with text-based communication – time delays causing confusion in sequencing of chat entries and the read-think-respond requirement for expression of thoughts. It was predicted that the least liked aspect of the assignment would be having to work in groups. The participants were unaccustomed to this form of teamwork. Additionally, there were cross-cultural and other demographic differences that teams had to manage. Personality conflicts were a constant concern for the team members.

Survey	NetMeeting	Group Work	TeamEC™
1-1	27.4	25.4	35.4
1-2	23.1	24.4	34.9
2-1	26.4	30.7	40.1
2-2	23.7	25.0	39.6
3-1	22.6	26.5	31.6
3-2	21.9	25.9	31.9
4-1	26.3	25.3	35.8
4-2	21.7	24.4	34.1
Total	193.1	207.4	283.4

Table 1. Survey Results

As can be seen in the Table 1 Totals line, results were contrary to expectations. The most striking aspect of these results is the remarkable similarity across all four classes for all eight survey dates. Without exception, the order of preference was the same: (1) NetMeeting, (2) Group Work, and (3) TeamEC™. The difference between NetMeeting and group work was small, but consistent. The difference between TeamEC™ and both NetMeeting and group work was notably large and also consistent. While participants' opinions of the software support tools were more favorable at the time of the second administration of the surveys (with the exception of the third class, which held an even less favorable opinion of TeamEC™ in the second survey), the parallel results maintained.

On the surface, these survey results do not seem favorable to TeamEC™. A guiding HCI principle is that software must be useful as well as easy to use. An additional often-imposed requirement is that software should make the task more intelligible to the user. If software fails to meet these requirements, it is not likely to be accepted. TeamEC™ suffered from some of these drawbacks. However, there are at least two more specific explanations that mitigate unfavorable response to use of TeamEC™ in this study.

First, the network through which the students shared the TeamEC™ application was frequently unstable. Each team required complex multitasking support to concurrently use external NetMeeting server connectivity, chat, whiteboard, application sharing for TeamEC™, Internet, and e-mail access. The load on the support system was compounded because six teams were working simultaneously from the same lab. Although the network often crashed, and the TeamEC™ client-server application often crashed, NetMeeting was robust. Therefore, chat transcripts and whiteboard records did not fall victim to these regularly occurring episodes. The output of TeamEC™, an independent application, was not automatically archived. While students blamed TeamEC™ when they lost their decision models during a crash, they had repeatedly been advised to frequently save their models. Some teams followed that advice while others forgot. They usually remembered after they experienced their first crash and lost their models.

Second, TeamEC's strongest feature was, from the participants' point of view, the most troublesome. The TeamEC™ software allows fairly wide latitude as to what is inserted into the decision tree. However, the software does give the user an indication of whether all pairwise assessments were made consistently. When there are inconsistencies, the software suggests that model elements be reassessed, although, in some cases, the inconsistencies are not important to the overall outcome. Therefore, reassessment is left to the discretion of the user. In general, participants in this study were unaccustomed to dealing with the level of precision required by the software. The software places high value on fine-grained analysis based on critical thinking skills at a conceptual level that was foreign to most participants.

Results from other studies of GDSS technology use found that GDSSs tended to reduce consensus, decision confidence, and overall satisfaction despite the fact that decision quality improved. It was earlier noted that decision makers tend to avoid decision aids because they reveal conflict and place a cognitive load on the user. In addition to these negative effects, software often emphasizes ease of use to the detriment of decision makers' understanding of the models they are building.

CONCLUSION

In the present study, the software's design emphasized understanding the task but users did not have a clear conceptual model of how to structure decisions. A mismatch occurred between the software's form and functionality and the users' ability to bridge their own knowledge gap. It is necessary to provide functionality for timely access to information and to present that information in a format that is easy to find and use. It is even more important to assure that task-specific domain knowledge is represented in a way that matches user understanding of a task to avoid misleading the user into believing that the functionality of the system supplants the need for the user to think critically about the task.

It is essential to identify which stakeholders are to benefit from a specific usability analysis (Mayhew and Mantei, 1994). The present study provides evidence that the value-added "tipping point," where form confounds function, may be reached for multiple reasons. An excess of features can lead to confusion. A paucity of features may result in insufficient guidance for novice users. Regardless of cause, if negative outcomes result from deployment of a technology, the lack of return on the investment in the technology will be highly detrimental to the strategic capabilities of the organization.

The study described in this paper was situated in an academic context. In order to carry this work forward, it will be necessary to further analyze distributed team decision making through case analysis of virtual teams in a real-world setting within and across organizations.

REFERENCES

1. Benbasat, I., & Lim, L.-H. (1993). The effects of group, task, context, and technology variables on the usefulness of group support systems: A meta-analysis of experimental studies. *Small Group Research, 24* (4), 430-462.
2. Bjorkman, M. (1972). Feedforward and feedback as determiners of knowledge and policy: Notes on a neglected issue. *Scandinavian Journal of Psychology, 13*, 152-158.
3. Davis, Fred D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology, *MIS Quarterly, 13* (3), pp. 319-340.
4. Davison, R. M. (1997). An instrument for measuring meeting success. *Information and Management, 32* (4), 163-176.
5. Dennis, A. R., Haley, B., & Vandenberg, R. J. (1996). *A meta-analysis of effectiveness, efficiency, and participant satisfaction in group support systems research.* 17th International Conference on Information Systems, Dec. 16-18, Cleveland, OH.
6. DeSanctis, G., & Poole, M. S. (1994). Capturing the complexity in advanced technology use: Adaptive structuration theory. *Organization Science, 5*(2), 121-147.
7. Frøkjær, E., Hertzum, M., & Hornbæk, K. (2000). Measuring usability: are effectiveness, efficiency, and satisfaction really correlated? In *Proceedings of SIGCHI.* (The Hague, The Netherlands, April 1-6), pp. 345-352. New York: ACM Press.
8. Limayem, M., & DeSanctis, G. (2000). Providing decisional guidance for multicriteria decision making in groups. *Information Systems Research, 11* (4), 386 - 401.
9. Majchrzak, A., Rice, R. E., Malhotra, A., King, N., & Ba, S. (2000). Technology adaptation: The case of a computer-supported inter-organizational virtual team. *MIS Quarterly, 24* (4), 569-600.
10. Mahmood, M. A., Hall, L., & Swanberg, D. L. (2001). Factors affecting information technology usage: A meta-analysis of the empirical literature. *Journal of Organizational Computing and Electronic Commerce*, 11 (2), pp. 107-130.
11. Mayhew, D. J. and Mantei, M. (1994). A basic framework for cost-justifying usability engineering. In R. G. Bias and D. J. Mayhew (Eds.) *Cost-Justifying Usability* (pp. 9-44). Boston: Harcourt Brace & Co.
12. Potter, S. S., Roth, E. M., Woods, D. D., & Elm, W. C. (2000). Bootstrapping multiple converging cognitive task analysis techniques for system design. In J. M. Schraagen, S. F. Chipman, & V. L. Shalin (Eds.), *Cognitive task analysis* (pp. 317-349). Mahwah, NJ: Lawrence Erlbaum Associates.
13. Saaty, T. L. (1980). *The analytic hierarchy process*. New York: McGraw-Hill.
14. Te'eni, D. (1991). Feedback in DSS: Experiments with the timing of feedback. *Decision Science, 22*, 644-655.
15. Tuttle, B., & Stocks, M. H. (1997). The effects of task information and outcome feedback on individual insight into their decision models. *Decision Science, 28* (2), 421-442.
16. Umanath, N. S. & Vessey, I. (1994). Multiattribute data presentation and human judgment: A cognitive fit perspective. *Decision Sciences, 25* (5/6), pp. 795-824.
17. Vessey, I. & Galetta, D. (1991). Cognitive fit: An empirical study of information acquisition. *Information Systems Research, 2* (1), pp. 63-84.