Markus Bengts

USABILITY AS A CONSTITUENT OF END-USER COMPUTING SATISFACTION

Master's thesis in
Information Systems
7.7.2004

University of Jyväskylä
Department of Computer Science and Information Systems
Jyväskylä

ABSTRACT

Bengts, Markus Kristian
Usability as a Constituent of End-User Computing Satisfaction / Markus Bengts
Jyväskylä: University of Jyväskylä, 2004.
87 p.
Master's thesis

This thesis studies usability as a constituent of end-user computing satisfaction. Usability is a complex concept that consists of many different aspects. Ease of use as well as efficient and pleasurable interaction are some indicators of usability. User satisfaction, for its part, is defined in this thesis as an affective attitude that emerges as preference when alternative systems are equally well known and freely available. The goal of this study is to create a better understanding of the concept of usability and to understand the relative importance of the different aspects of usability as constituents of user satisfaction. The focus is on human-computer interaction, and the study is restricted to experienced and skillful computer users and their utilization of software systems to achieve real-world goals.

Different measurement instruments and rating scales for user satisfaction have been created; however, the relationship between satisfaction and usability remains unclear. Usability as a constituent of user satisfaction is investigated through a field experiment. A web-based system with three different user-interface alternatives was implemented and the system was used by information technology students to practice SQL-queries in a university course. 43 students reported their preference and the underlying reasons by answering both structured and open-ended questions in a web-based questionnaire. The study combined quantitative and qualitative analysis.

The most important contribution of this thesis is a conceptual framework for usability studies. The results also indicate that availability of desired features, simple interaction and user-control are as constituents of satisfaction more important than simple screen design and error-free usage.

KEYWORDS: usability, user satisfaction, preference, conceptual framework.

CONTENTS

1 INTRODUCTION 5

2 USABILITY.....9
- 2.1 Background theory 9
- 2.2 A conceptual framework for usability studies.....14
- 2.3 Affective aspects.....16
- 2.4 Utility aspects.....18
- 2.5 Cognitive aspects.....20
- 2.6 Evaluation.....23

3 SATISFACTION AND PREFERENCE.....25
- 3.1 Attitude function and structure.....25
- 3.2 Measures of satisfaction.....29
- 3.3 Rational versus affective choice.....32
- 3.4 Satisfaction measured as preference.....33
- 3.5 Evaluation.....34

4 USER-INTERFACE DESIGN THEORY.....36
- 4.1 User centered design.....36
- 4.2 Usability heuristics.....37
- 4.3 Conceptual models.....39

5 METHOD.....40
- 5.1 Research strategy.....40
- 5.2 Tasks.....44
- 5.3 User-interface alternatives.....44
- 5.4 Participants and procedure.....48
- 5.5 Data collection.....50
- 5.6 Analysis.....51
- 5.7 Evaluation.....52

6 RESULTS.....54
- 6.1 Quantitative analysis.....54
- 6.2 Inductive qualitative analysis.....56
- 6.3 Deductive qualitative analysis.....61
- 6.4 Evaluation.....62

7 CONCLUSION.....66

REFERENCES.....69

List of figures

FIGURE 1. The research model.....7

FIGURE 2. A model of the attributes of system acceptability (Nielsen 1993).....10

FIGURE 3. Layered model of usability (van Welie et al. 1999).....14

FIGURE 4. A conceptual framework for usability studies.....15

FIGURE 5. An example of HTA of general creative activities.......19
FIGURE 6. Functions of attitudes toward software systems.......26
FIGURE 7. Hierarchical model of attitude (Ajzen 1989, 246).......27
FIGURE 8. A causal chain model of attitude.......28
FIGURE 9. The gene-interface.......45
FIGURE 10. The visi-interface.......46
FIGURE 11. The conv-interface in the stage of selecting exercise.......47
FIGURE 12. The user-interface alternatives.......48
FIGURE 13. Perceived usability.......60

List of tables

TABLE 1. Preference distribution for the user-interface alternatives.......54
TABLE 2. Test group cross-tabulated with preference.......55
TABLE 3. Gender cross-tabulated with preference.......55
TABLE 4. Association between the nominal variables and preference.......56
TABLE 5. Mean rank for the preference groups on the plastic interval variables and significance of the Mann-Whitney U test.......56
TABLE 6. The constituents of user satisfaction.......61
TABLE 7. The constituents of user satisfaction summarized.......62
TABLE 8. SUS-scores for the user-interface alternatives.......65

Appendixes

APPENDIX 1 User-interface alternatives.......76
APPENDIX 2 Usability tests of the user-interface alternatives.......84
APPENDIX 3 Usability tests of the questionnaire.......87

1 INTRODUCTION

Human-computer interaction (HCI) is a multidisciplinary field that has emerged in the 1980's from underlying disciplines such as computer science, software engineering and information systems. HCI often makes use of theories from its reference disciplines, for example, anthropology, cognitive science, education, psychology, sociology and industrial design. The Special Interest Group on Computer-Human Interaction (SIGCHI) was formed in the Association for Computing Machinery (ACM) in 1982. SIGCHI has developed curricula for HCI (Hewett, Baecker, Card, Carey, Gasen, Mantei, Perlman, Strong and Verplank 1992) and defined HCI as “a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.” HCI focuses especially on understanding the user-interface requirements resulting from the cognitive restrictions of humans and the implications of these requirements for user-interface design.

The goal of HCI research is often to enable development of usable systems. Usability is nowadays a major factor of competition in software business. Furthermore, it is of growing importance as computer-supported work becomes more common. Usability is usually defined as a collection of attributes or aspects. Nielsen (1993, 25), for example, defines usability as consisting of learnability, efficiency, memorability, errors and satisfaction. This is one of the most famous definitions; however, other researchers (e.g. Dix, Finlay, Abowd, and Beale 1998, 162-175; Preece, Rogers, and Sharp 2002, 13-20) have presented different sets of usability attributes. An analysis of these sets is needed to gain a better understanding of the concept of usability.

End-user computing satisfaction has been a popular topic in HCI throughout the short history of the discipline (Harrison and Rainer 1996). It is an obvious factor determining customer loyalty in software business. Furthermore, user satisfaction probably also contributes to the quality of work life for people who spend their days in front of a computer screen. Different measurement instruments and rating scales for user satisfaction have been created and much research has been done in this subject area (e.g. Doll and Torkzadeh 1988; Harrison and Rainer 1996; Tractinsky 1997; Hassenzahl, Platz, Burmester, and

Lehner 2000; Chin and Lee 2000; McHaney, Hightower, and Pearson 2002; Lindgaard and Dudek 2003). However, the relationship between satisfaction and usability remains unclear (Lindgaard and Dudek 2003).

The goal of this study is to understand how user satisfaction is formed. The importance for software businesses to achieve user satisfaction and the vague understanding of the relationship between satisfaction and usability motivate this study. In contrast with Nielsen's (1993, 25) definition, satisfaction is not seen as an attribute of usability, but rather as an external goal that can be achieved through good usability and other constituents. Despite of many available rating scales, user satisfaction is seen as a complex construct that is very difficult to operationalize (cf. Lindgaard and Dudek). Doll and Torkzadeh (1988) define user satisfaction as an "affective attitude towards a specific computer application." Many other researchers agree that satisfaction is an attitude (e.g. Melone 1990; Harrison and Rainer 1996; Keinonen 1998, 89). In this study *user satisfaction* is defined as an affective attitude that emerges as preference when alternative software systems are (1) equally well known, (2) very well known, (3) freely available and (4) specific real-world goals are to be achieved through the use of the system. Hence, satisfaction is operationalized through preference.

The experiments of Lindgaard and Dudek (2003) show that perceived usability is a major factor in user experience. This study assumes that usability is an important constituent of user satisfaction. Therefore, a good understanding of the different usability attribute collections and a clarification of this concept is needed before studying user experience. The objective of this study is to achieve a clear understanding of (1) the concept of usability and (2) the relationship between usability and user satisfaction.

More specifically, the goal is to identify the different aspects of usability and to present them in a consistent concept system. In addition, indicative answers are expected to the main research question: *What is the relative importance of the different aspects of usability as constituents of user satisfaction?* The study is also open to other constituents of user satisfaction; therefore, an additional research question is stated: What other attributes of software systems (i.e. non-usability attributes) contribute to user satisfaction? FIGURE 1 depicts the research model that illustrates the research problem and the setting in which satisfaction is measured as preference.

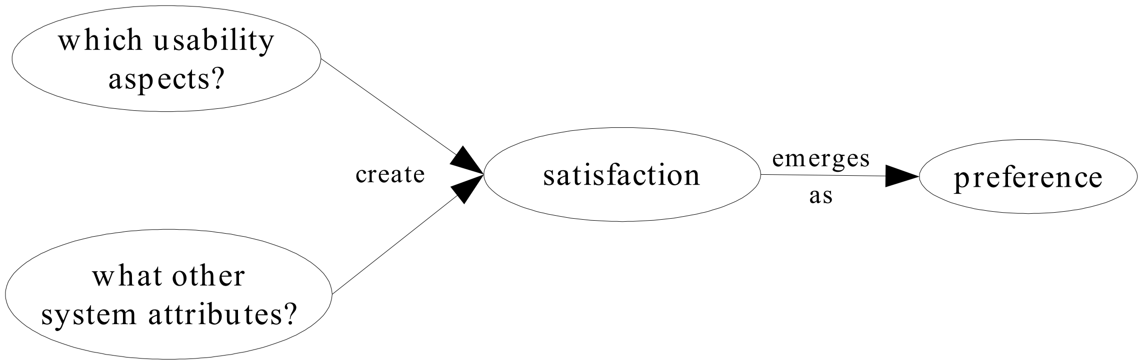


FIGURE 1. The research model

This study focuses on human-computer interaction and especially on web-based computer software systems. However, the results might be applicable also to other kinds of interactive products. Furthermore, the study is restricted to personal use of a software system. Therefore, the social and organizational aspects of computer systems are not considered. The definition and operationalization of satisfaction sets one more restriction. This study can only consider software products that are used to achieve some real-world goals. Games and other products designed purely for entertainment and amusement are not studied.

Strauss and Corbin (1998, 33) point out that a researcher is seldom able to enter into a project with concepts and a well structured design established in advance. They explain that the design also must be permitted to emerge during the research process. The design of this study was not clear in advance, but now afterwards it can be described as consisting of two stages. The process started with a conceptual-analytical study of different definitions of usability. This first stage of the research process was based on literature and its contribution is a conceptual framework for usability studies. The research model presented above was developed after the first stage at the beginning of the second stage. The second stage of the process was empirical and it combined both a qualitative and a quantitative approach (see Patton 2003, 558) to understand how users perceive the usability of a system and to elucidate how user satisfaction is formed. The design of the empirical part resembles a field experiment and it was carried out in conjunction with the course "ITK135 – Databases and Information Management" in the University of Jyväskylä. A web-based system with three user-interface alternatives was designed and implemented for the field experiment. Students of the course tried out the alternatives and then used one of them to practice SQL-queries during the course. Afterwards, their preference was recorded with closed questions and the underlying reasons were probed with open-ended questions in a web-based questionnaire. The results of the field experiment include some modifications to the framework developed in the first stage and a rating of the relative importance of some aspects of usability. The most important contribution of this second stage is a better understanding of perceived usability and the constituents of user satisfaction.

Chapters 2-4 of this thesis are based on literature and they present some background theory for the field experiment that is reported in chapters 5-7. Chapter 2 was written during the first stage described above. It investigates the concept of usability and presents the framework that is later used in the analysis of the empirical data. Chapter 3 deals with the concept of satisfaction and it justifies the definition and operationalization used in the field experiment. The user-interface design theory that was applied in the design of the user-interface alternatives is presented in chapter 4. Then, in chapter 5 the study moves on to describe the empirical part. It provides the details of the user-interface alternatives and explains the design and procedure of the field experiment. The results are presented in chapter 6, and the study ends with a concluding summary in chapter 7.

2 USABILITY

To be able to study how usability is related to satisfaction a clear understanding of the concept of usability is needed. Section 2.1 discusses some previous definitions of usability together with some background theory to justify the modifications in which satisfaction is placed outside the concept of usability and usability is viewed as consisting of affective aspects, utility aspects and cognitive aspects. The main contribution of this chapter – a conceptual framework for usability studies – is based on this background and presented in section 2.2. The following sections (2.3, 2.4, and 2.5) explain different parts of the framework and present definitions for the concepts used in the framework. These sections also discuss alternative definitions and explain why a particular definition was chosen. The framework is evaluated in the last section (2.6) and some ideas for future work are also presented.

2.1 Background theory

Usability is commonly viewed as a very broad concept that includes completely different aspects; therefore, it is usually conceptualized as a collection of separately defined attributes. The International Organization for Standardization has given the following definition of usability in ISO 9241-11 (Donyaee 2001):

> Usability: the extent to which, a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

We can immediately recognize three attributes of usability from this definition: effectiveness, efficiency, and satisfaction. Frøkjær, Hertzum, and Hornbæk (2000) used this definition in an analysis of the CHI-conference proceedings from 1997-99. From a total of 19 usability studies suitable for the analysis only 42 % covered all three aspects. The remaining 58% assumed that their limited choice of measured aspects were sufficient to capture overall usability. Frøkjær et al. also conducted an experiment in which 87 students completed 20 information retrieval tasks with 5 alternative user-interfaces. They measured efficiency as task completion time, effectiveness as quality of solution, and satisfaction as preference. Their results show that overall usability cannot be obtained by a single measure because different usability aspects are not always correlated. Instead, the different aspects should be measured separately and all aspects should be included to be able to make statements about overall usability.

van Welie, van der Veer and Eliëns (1999) have also examined different definitions of usability. They argue that the attributes of the ISO-definition are still too abstract to be applicable in practice; consequently, the concept needs to be analyzed further. Another argument for further examination of the concept is that effectiveness, efficiency and satisfaction might not cover all relevant aspects of usability.

Nielsen (1993, 25) presents usability as an attribute of system acceptability. His model of the attributes of system acceptability is shown in FIGURE 2. The literature of the field usually makes use of Nielsen's (1993, 26) definition that was already mentioned in the introduction. It presents usability with five attributes: learnability, efficiency, memorability, errors, and satisfaction. Shneiderman (1998, 15) talks about measurable human factors, when he presents the same five aspects using a different terminology in a chapter called “Goals of User-Interface Design.”

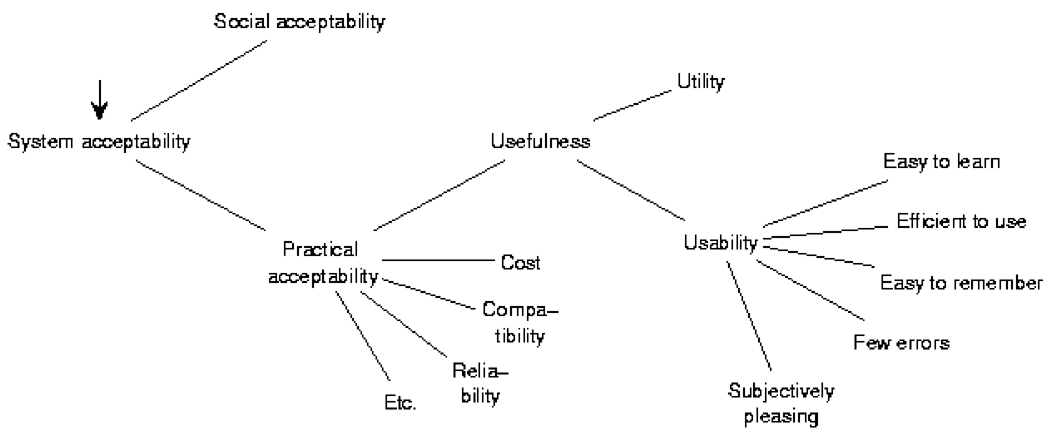


FIGURE 2. A model of the attributes of system acceptability (Nielsen 1993, 25)

Satisfaction is essentially different from the other aspects presented in Nielsen's definition of usability. The other attributes of Nielsen's definition can be operationalized and objectively measured. Data on satisfaction is based on users' opinions and can only be extracted by asking the users, through questionnaires or interviews. Bevan (1995) claims that satisfaction data can also be obtained indirectly, for instance by counting positive and negative comments during use. He also suggests that such data could be obtained by investigating absenteeism or health problem reports; however, that would be quite far-fetched and not so easily applicable in usability studies. Nielsen (1993, 34) discusses psycho-physiological measures such as EEG, pupil dilation, heart rate, skin conductivity, blood pressure, and the level of adrenaline in the blood as possible measures of satisfaction, but he believes these are often inappropriate for usability studies because of poor ecological validity. Satisfaction is measured differently compared to the other attributes of Nielsen's definition, furthermore, it is not in the system but rather an external goal that can be achieved indirectly through other usability aspects that influence perceived usability and the user's affective attitude towards the system.

Now that satisfaction has been placed outside the concept of usability, we can move on to the background theory supporting the next modification of the previous definitions. Eysenck and Keane (2000, 489) report that affect and emotions have been recognized as important factors for cognitive functioning but still deliberately ignored in cognitive science to avoid unnecessary complication. They explain that affect covers many different kinds of experiences such as emotions, moods, and preferences. Further, they define emotions as referring to short-time and intense experiences while mood refers to long-time low-intensity experiences. Norman (2004, 11) defines affect as the judgmental system including both conscious and subconscious processing. Emotion according to Norman is the conscious experience of affect including attribution of its cause and identification of its object. This study makes use of a broader definition that includes the different kinds of experiences according to Eysenck and Keane without restriction to only conscious experiences.

In an interview in the ACM Ubiquity magazine Norman (2002a) argues that humans have two different processing systems: one is cognition and the other is affect. He explains that the cognitive system interprets and understands the world, while affect evaluates and makes quick value judgments. He also stresses that the affect system has a strong impact on how the cognitive system processes information. Humans solve difficult problems better when they are happy. Happy people are global, breadth-first thinkers, while stressed people are local, depth-first thinkers. Consequently, a more pleasant interface or device is also easier to learn and use. Norman (2002b) also presents the same theory in his own article.

Actually Norman (2002a) also mentions reflexive reactions as the lowest level of processing in addition to the evaluating middle level and the analytical highest level. The reflexive level might be of interest for instance when developing computer games that require short reaction times from end-users. However, the reflexive level is not considered in this study because it is irrelevant in most usability studies.

Tomkins (1981) was perhaps the first psychologist to claim that affect and cognition are independent and interrelated systems. He offered his theory in the early 1950s to rescue psychology from an "overly imperialistic cognitive theory." Eysenck and Keane (2000, 492) have compared many studies of affect and cognition. They conclude that emotion can be independent of conscious cognitive processes. This conclusion confirms the nowadays widely accepted theory of affect and cognition as independent and interrelated systems.

The way in which Preece, Rogers and Sharp (2002, 13-20) view the concept of usability can be analyzed with respect to the affect-cognition theory. Preece et al. divide the goals of interaction design into usability goals and user experience goals. "Usability goals are concerned with meeting specific usability criteria (e.g., efficiency) and user experience goals are concerned with explicating the quality of the user experience (e.g., to be aesthetically pleasing)" (Preece et al. 2002, 14). Their explicitly defined usability goals include effectiveness, efficiency, safety, utility, learnability, and memorability. They explain that the user experience goals are less clearly defined but these goals are about creating systems that are satisfying, enjoyable, fun, entertaining, helpful, motivating, aesthetically pleasing, supportive of creativity, rewarding, and emotionally fulfilling. In this way Preece et al. have divided the goals of interaction design by grouping together the goals that are directly connected to the users' affect system into user experience goals, and by calling the other goals usability goals.

Preece et al. (2002, 14) include utility in their usability goals. However, in Nielsen's definition, utility is seen as an attribute of system acceptability outside the scope of usability which only addresses the system aspects that are directly related to the users' cognitive or affective information processing. The question of whether utility should be regarded as an attribute of usability or a separate factor affecting system acceptability is vain. It is crucial, though, to recognize the importance of utility when considering system acceptability from the HCI perspective. Including the proper functionality improves the utility of a system. Jacobson, Booch and Rumbaugh (1999, 5) describe the unified software development process as being use case driven. By this they want to stress the importance of capturing the relevant functional requirements of a system. Through the concept of use cases they force developers to think in terms of value to users and not just in terms of features that might be good to have.

Gibson (1986, 127) defined the term affordance as something the environment “offers the animal.” Affordances doesn't have to be visible or perceivable in any way. They just exist in the environment and they “have to be measured relative to the animal.” Torenvliet (2003) expresses his frustration about the wide misuse of this term. He explains that affordances are the functionality that exists in a system. Thus, while other aspects of usability may require that the affordances are made visible or perceivable in another way, developing for utility is concerned with getting the appropriate affordances into the system. This is better known as specifying the functional requirements of the system. Torenvliet (2003) thinks this is a good way for usability experts to bring value to development projects.

When striving for a high level of utility, developers do not have to consider the users' cognitive or affective information processing. To achieve utility developers must understand the work that the system is intended to support. Another viewpoint would be to understand the needs that the system is intended to satisfy. Because utility differs in this way from the other usability goals listed by Preece et al. (2002, 14), utility is not viewed as one of the cognitive aspects but as a separate high level aspect. From the HCI perspective utility is especially important when evaluating alternative general purpose systems and comparing their usability in a specific work context. In software engineering utility is normally understood as supporting a specific business process in a company.

Now that satisfaction has been left out and utility has been placed outside the cognitive aspects of usability, one more dimension can be added to this discussion. van Welie, van der Veer and Eliëns (1999) propose a layered model, in which usability at the highest level is defined according to the ISO 9241-11 standard. The next level consists of the attributes of usability according to Nielsen's definition. The attributes at this level are called *usage indicators* and van Welie et al. (1999) claim that these can be observed in practice when users are at work. The following level in their model consists of *means* that should be understood as design guidelines. Developers can improve usability through the means that influence the usage indicators. FIGURE 3 illustrates the layered model of van Welie et al. (1999).

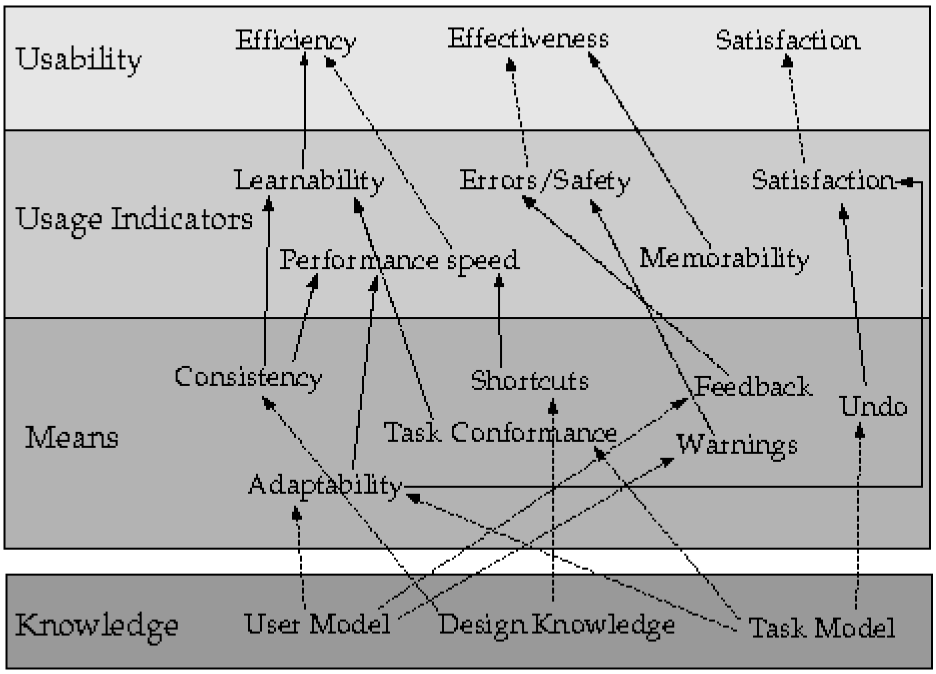


FIGURE 3. Layered model of usability (van Welie et al. 1999)

2.2 A conceptual framework for usability studies

By explaining affect and cognition as two different processing systems according to Norman (2002a; 2002b) and utility as a factor of system acceptability that is not directly related to users but rather to the work that users perform, I have tried to justify my conception of usability in terms of affective aspects, utility aspects and cognitive aspects. I have combined this conception with the layered model of van Welie et al. and further expanded it with central concepts of the HCI literature to propose a conceptual framework for usability studies. It is presented as a tree-like concept hierarchy and illustrated in FIGURE 4. It is important to note that strong boundaries between the branches do not exist. This framework is provided as an aid for understanding and analysis but it would definitely be misleading to divide the domain by closing out one or more branches of the tree. “Attractive things work better” (Norman 2002b) and understandable things are more attractive, because the affect system influences the cognitive system and the cognitive system can to some extent decide about affects. Consequently the domain of usability concepts has to be considered as a whole.

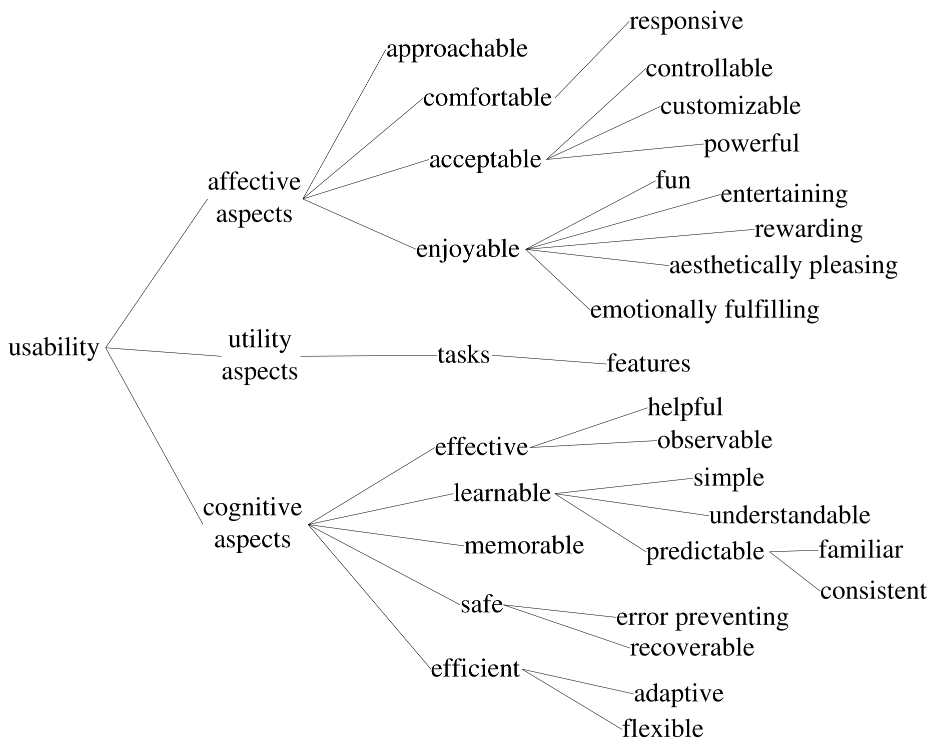


FIGURE 4. A conceptual framework for usability studies

The cognitive aspects presented in FIGURE 4 are easily divided into usage indicators and means in the same way as van Welie et al. (1999) have done. When users are at work, it can be observed whether a system is effective, learnable, memorable, safe, and efficient. However, some of these indicators are operationalized through lower level components. Satisfaction, for its part, is more difficult to observe. It relies on opinion and has to be asked while the cognitive aspects can be measured. The system properties presented in the branch of affect should be seen as goals and subgoals that developers need to keep in mind. Developers also need to understand which of the goals are important and which can be ignored depending on the nature of the system being developed. Achievement of these goals is verified by asking the users for their opinion. Utility, for its part, can be determined without test users by comparing the requirements of the work and the functionality of the system provided that the requirements are known.

2.3 Affective aspects

Nielsen (1993, 34) says that users sometimes refuse to use a program just because the manual is too big. He explains that the approachability of a system means how difficult users think it would be to learn the system when they have first seen it but not yet tried to use it. “Attitudes guide behavior... away from aversive events” (Breckler and Wiggins 1989, 481).

Bevan (1995) defines satisfaction as composed of comfort and acceptability of use. Further, he explains that comfort is about what the user feels when using the system, while acceptability is about the user's perception of the cognitive aspects of the system's usability and appears as the user's overall attitude towards the system. Satisfaction is in this study seen as a complex concept as described in the introduction and further discussed in the next chapter; however, Bevan's descriptions of comfort and acceptability are accepted here. Analyzing these descriptions in the light of the theory proposed by Norman (2002a; 2002b) shows that comfort is purely affective while acceptability also requires reflective thought. Still acceptability as a value judgment is supposed to be based very much on affective processing and hence placed in the branch of affective aspects. The formation of acceptability is supposed to be an example of the case in which the cognitive system to some extent can decide about the resulting affective judgment. Controllable, customizable and powerful systems are also supposed to be judged as acceptable.

Responsiveness is seen as one of the means to design comfortable systems. Dix et al. (1998, 172) define responsiveness as “how the user perceives the rate of communication with the system.” Responsiveness gives a feeling of security when the user knows what is going on in the system. Observable and rewarding systems might be perceived as responsive and comfortable. Still these means are seen to fit better in other branches. “Visibility of system status” is one of Nielsen's (1994b, 30) refined heuristics and that kind of responsiveness is probably always experienced positively. Making a system particularly rewarding might be important when developing games that are intended to require special skills from the user; however, in the case of work-related systems it is likely that users would feel annoyed if the system would praise them whenever they get their work done.

Systems have to be enjoyable especially if they are intended to be used in the free time, but in many cases there might be a trade off between properties in the branch of enjoyability and efficiency for example. Acceptability strictly as the perception of the cognitive aspects might also be decreased with an overly entertaining system. Acceptability is probably achieved simply by improving the cognitive aspects and utility.

In the context of affective aspects of usability it is also relevant to note that user-interface properties can influence the user's motivation and creativity. Shneiderman (2000) believes that the results of creative work can be improved by making the creative process more social. He describes creative processes as consisting of four phases: collecting information, relating, creating, and disseminating results, and he recognizes eight activities during these phases:

- searching and browsing digital libraries,
- consulting with peers and mentors,
- visualizing data and processes,
- thinking by free associations,
- exploring solutions – what-if tools,
- composing artifacts and performances,
- reviewing and replaying session histories, and
- disseminating results.

He believes that software systems supporting these activities can greatly improve the outcome of creative work, and he also points out that other aspects of usability are essential in such systems so that the user is not interrupted by the system and so that the user's attention can be completely devoted to the task. He considers the cognitive aspects of creative work and recognizes functional requirements that can support creativity.

Norman (2002a; 2002b) explains that positive affect influences the cognitive system to perform better when broad thinking and creative problem-solving is required. Perhaps creativity could also be supported by properties that create positive affect. Shneiderman's (2002, 2) concept of “new computing” involves understanding human needs to be able to develop useful technologies. These technologies should give users a feeling of security, mastery, and accomplishment and thus enable users to relax, enjoy and explore. This surely is a noble goal, and when achieved would probably lead to systems that are both motivating and supportive of creativity as well as satisfying.

Norman (2004, 135-136) explains that we have a tendency to attribute human motivations, beliefs and feelings to inanimate objects we may interact with. This might be one reason why it is so tempting to develop software agents that mimic human performance. Shneiderman (2002, 62) gives many examples in which people have rejected products that were developed to mimic human behavior. He argues that “most users don't want a relationship with their computer, they want control over it.” Furthermore, he claims that successful computer systems rarely imitate humans but rather empower people.

2.4 Utility aspects

Understanding the requirements of the work and getting the right affordances into the system can be a very complicated task and it is unquestionably an important part of system development. According to Preece et al. (2002, 16) “utility refers to the extent to which the system provides the right kind of functionality so that users can do what they need or want to do.” There is probably not much disagreement on the meaning of this term. Nielsen (1993, 25) gives a very similar definition: “utility is the question of whether the functionality of the system in principle can do what is needed.”

The branch of utility did not get much attention in FIGURE 4. It could be expanded according to general categories of work or human needs, but then it would not fit into the dimension of usage indicators and means. I suggest that it is expanded differently depending on the considered work domain using a method called hierarchical task analysis (HTA) to model the relevant tasks in the case of a specific system. HTA is described for example in Preece et al. (2002, 131-134) and Dix et al. (1998, 262-268). The tasks supported or not supported would then be practical measures of utility, and features that enable those tasks would be means to improve utility. In addition to the simple on/off evaluation of tasks supported, each task could also be evaluated along the dimension of performance. It does make a difference how well a desired task is performed. For example, the quality of a slide presentation might be as important as the possibility to create a presentation in the first place.

Shneiderman (2002, 76) explains that when people have the security to go beyond basic needs “they can become creative in... and they enjoy participating in...” Creating and participating could be seen as general categories of activities. Shneiderman (2002, 87) combines his four phases of creative work with four circles of relationships to form an activities and relationships table (ART). He shows that it can be used to analyze many different kinds of activities. FIGURE 5 gives an example of how the branch of utility could be expanded with Shneiderman's four phases of creative work.

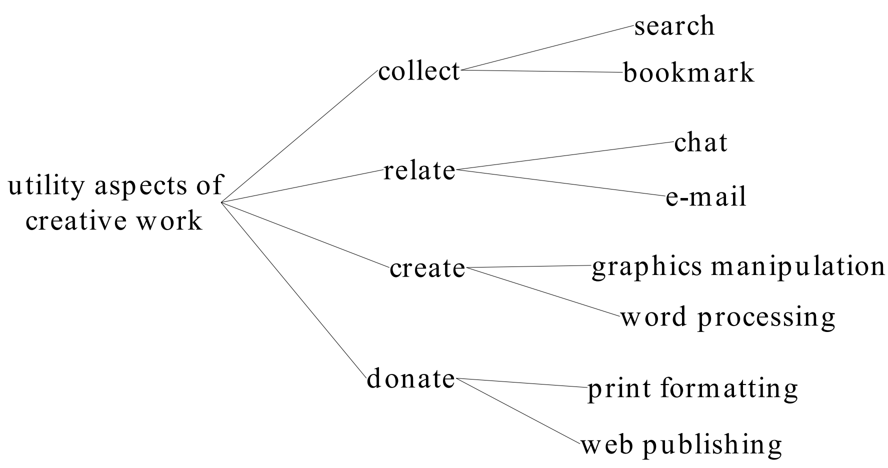


FIGURE 5. An example of HTA of general creative activities

Utility considerations should not be restricted only to the level of the whole system. It should also be consider at a lower level. When striving for simplicity and observability during the design process, the utility of every visible object on the screen should be carefully evaluated.

Developing a highly customizable system is sometimes seen as a way of addressing the problem of getting all the necessary affordances into the system. The UNIX-philosophy – a bunch of simple tools that can be combined in a multiplicity of ways to accomplish almost any task – has actually been used to create very successful implementations of this approach. But the task of developing software systems does not belong to most work environments, and component based software development is not easy if one does not have the proper education.

2.5 Cognitive aspects

Preece et al. (2002, 14) give the following definition for effectiveness: "effectiveness is a very general goal and refers to how good a system is at doing what it is supposed to do." It appears from their example that they see effectiveness as a combination of utility, learnability, and efficiency. A more explicit and practical definition is required for a usage indicator. The ISO 9241-11 standard gives a definition that is better suited: "effectiveness: the accuracy and completeness with which users achieve specified tasks" (van Welie et al. 1999). One of the means to improve effectiveness is to design a helpful system, that is, a system that provides different kinds of help functions, tutorials and manuals. Effectiveness can also be improved through observability. Dix et al. (1998, 172) define observability as the "ability of the user to evaluate the internal state of the system from its perceivable representation." In addition to the state of the system this study also includes observability of functionality under this term. "Visibility" is regarded as a special case of observability and it is used in this study instead of observability if the considered system cannot be observed through other senses than vision. Furthermore, all the means that are placed in the branch of learnability might also be good ways to improve effectiveness.

According to Nielsen (1993, 26) efficiency means that "once the user has learned the system, a high level of productivity is possible." The definition of the ISO

9241-11 standard: “efficiency: the resources expended in relation to the accuracy and completeness with which users achieve goals” (van Welie et al. 1999) could complement Nielsen's definition but efficiency has to be restricted to users that have learned the system to distinguish it from learnability. Efficiency can be operationalized as the time spent to complete a task in the same way as Frøkjær et al. (2000) have done.

Adaptability, customizability, and flexibility could improve efficiency. Dix et al. (1998, 162) see flexibility as a top level category of usability principles and define it as “the multiplicity of ways the user and the system exchange information.” Further, they define customizability as “modifiability of the user interface by the user or the system” (Dix et al. 1998, 168). Bevan (1995) defines flexibility as “the extent to which a product is usable for different types of users and tasks.” This study distinguishes between adaptability and customizability by defining that an adaptable, or even better, an adaptive system changes its behavior on the system's initiative for instance by recognizing usage patterns, while a customizable system provides possibilities for the user to tune the system and tailor its functionality. In this study flexibility means that the system constantly provides different interaction alternatives like shortcuts and wizards. When striving for adaptivity, it is important to find the proper level of automation so that one of Nielsen's (1994b, 30) refined heuristics, “user control and freedom,” is not violated.

Shneiderman (1998, 83) believes that the degree of automation will increase in the course of time and he predicts that the role of human users in future automated systems will be to deal with unexpected situations and to improve the system. He advocates eliminating human action when no judgment is required, but at the same time he warns that users become anxious about automatically changing, unpredictable systems that cannot be restored to their previous state. He also suggests development of systems in which users may start at the most simple level and choose to increase complexity if they need more functionality or more advanced features.

Dix et al. (1998, 162) define learnability as “the ease with which new users can begin effective interaction and achieve maximal performance.” Others seem to agree quite well with this. Developing a good conceptual model, like Norman

(1988, 12-17; 1998, 173-180) teaches, might be the key to making a system more understandable and predictable, and thus easier to learn. Dix et al. (1998, 163) show that familiarity is slightly different as it depends on earlier experience and knowledge. They define familiarity as “the extent to which a user's knowledge and experience in other real-world or computer-based domains can be applied when interacting with a new system.” Things can be understandable, like Norman's (1998, 177) example shows, even if they are not familiar. Dix et al. (1998, 163) also define predictability as being dependent on past interaction history. Their definition “support for the user to determine the effect of future action based on past interaction history” fits in here somehow as it is clearly distinguished from familiarity, but I would like to add the possibility of predictability independent of past interaction history. Consistency surely contributes to predictability if we accept the definition of Dix et al.

Metaphors are often used to map familiar to unknown and to create a feeling of familiarity in this way. Nielsen (1993, 128) warns that metaphors can be misleading or that they can imply too much. Norman (1998, 180) bangs the drums even more about how bad metaphors can be. He admits that metaphors can help in learning a new system especially when the properties of the metaphor and the system are very closely related, but still he advises developers to forget about metaphors. Preece et al. (2002, 55-60), on the other hand, discuss the critique of interface metaphors and provide convincing arguments for careful use of metaphors.

Most system properties that contribute to learnability also make the system easier to remember. A good conceptual model as well as carefully designed visual signs might be important ingredients of memorability. Eysenck and Keane (2000, 223) note that stimulus with emotional involvement or rarity value are memorable. Preece (1994, 109) might give a more practical piece of advice by explaining that meaningful command names and icons are easier to remember. It is also much easier to recognize visible objects than to recall command names for example (Eysenck and Keane 2000, 175). Familiarity and especially the part that concerns knowledge and experience in real-world domains could also be an important contributor to memorability. Nielsen's (1993, 26) definition of memorability, “ the system should be easy to remember, so that the casual user is able to return to the system after some period of not having used it, without having to learn everything all over again,“ fits in here quite well and there is not much special about it. Preece et al. (2002, 17) go along the same lines: “memorability refers to how easy a system is to remember how to use, once learned.”

Shneiderman (1998, 15) claims that error handling is a critical component of system usage. Shneiderman (1998, 76) also explains that users make far more mistakes than expected. His experiments show that better recoverability, lower error rates and improved satisfaction can be gained with constructive and specific error messages. Further, he points out that preventing errors in the first place is more effective and actually even attainable in many cases.

Talking about safety feels a bit dramatized when limited to normal computer software systems as one does not really need to worry about the screen exploding in one's face if one presses the wrong button. In real safety critical systems, such as in nuclear power- or air traffic control, errors are a very big concern. Preece et al. (2002, 14) explain that safety is not limited to external conditions but it also involves helping the users avoid the dangers of carrying out unwanted actions accidentally. In this thesis an error-preventing and recoverable system is labeled as being safe. A safe system can be described in the same way as Nielsen (1993, 26) defines the usability attribute called errors: “The system should have a low error rate, so that users make few errors during the use of the system, and so that if they do make errors they can easily recover from them. Further, catastrophic errors must not occur.” Dix et al. (1998,172) give a good definition of recoverability: “ability of the user to take corrective action once an error has been recognized.”

2.6 Evaluation

Creating new theory always involves uncertainty. Where evidence is missing it is replaced by intuition or subjective interpretation. Many of the properties that were presented on the level of means in my concept system could be placed in different branches of the tree because they influence many different usage indicators or higher level aspects. Further, there is no evidence that affect, cognition, and utility would be the proper categories to cover the scope of relevant usage indicators in usability studies, and it seems almost ridiculous to combine utility with affect and cognition. Still there is a good reason to separate utility from cognitive aspects and to include it as an important viewpoint in usability studies. The concept hierarchy could surely be developed further by adding more usage indicators and means that are not synonyms to the terms that already exist in the system.

The description of human information processing in two systems – affect and cognition – is very simple and easy to understand, therefore it was suitable to be used as a starting point for this chapter. This kind of study could also be based on a more advanced description of human information processing. The interacting cognitive subsystems that Barnard and May (1999) propose is an example of a model that could be used as a framework for this kind of study. Still I think the simple framework proposed in this chapter is quite usable.

The cognitive backgrounds of the different aspects of usability could be studied further to provide a better understanding of the system properties that influence usability. Memorability, especially, did not get much attention in this chapter. Studying different theories about human memory and trying to recognize what makes a system memorable might bring more value to this study. A better understanding of the human affect system is also important. For example the study of satisfaction, motivation, and creativity in relation to usability is an application area for that knowledge. Different lists of heuristics and design guidelines could also be studied with intention to improve the conceptual framework proposed in this chapter.

This theoretically developed framework is used as a starting point both for data collection and analysis in this thesis. The framework is basic in such a way that with proper customization it can probably be applied in a much larger range of problems than has been discussed here. Some examples of application areas are user interface development, usability inspections, usability testing, and basic HCI research.

3 SATISFACTION AND PREFERENCE

The goal of this chapter is to justify the operationalization of satisfaction as preference. Satisfaction was defined in the introduction as an affective attitude, and the concept of attitude has been investigated very much in psychology. This concept is discussed in section 3.1 which is mostly based on the compilation “Attitude structure and function” of Pratkanis, Breckler and Greenwald (1989). Section 3.2 analyzes some previously known measurement instruments of satisfaction and explains why these are not appropriate for this study. Some literature on decision making and choice is discussed in section 3.3. Section 3.4 draws conclusions based on the previous sections of this chapter and presents the main point of this chapter – a justification of the operationalization of satisfaction as preference. This chapter also ends with an evaluation in the final section.

3.1 Attitude function and structure

Keinonen (1998) deals with consumer decision-making in chapter three of his Ph.D. thesis. He notes that satisfaction and attitude are very similar concepts. Keinonen (1998, 68) points out the difference between these concepts in the context of buying behavior by noting that satisfaction is formed after the purchase while attitudes also exist before the purchase. Satisfaction has been of interest to researchers both in marketing and HCI. Kotler (1997, 40) presents a marketer's view of how satisfaction is formed. He explains that satisfaction is formed in a post-purchase comparison of expectations and product performance. The financial transactions and ownership is not of interest in HCI and therefore satisfaction is considered to be formed as “the subjective sum of the interactive experience” in HCI (Lindgaard and Dudek 2003). Melone (1990) suggests that user satisfaction research could benefit from the stronger theoretical frameworks on attitude developed in the behavioral sciences as satisfaction and attitude are so similar.

Shavitt (1989, 311-314) analyzes the function of attitudes and distinguishes functions that he labels knowledge, utilitarian, object-appraisal, social adjustment, value-expressive and ego-defensive. Abelson and Prentice (1989, 361-362) simply organize these functions into two categories: instrumental and symbolic functions. For the purpose of this study it is useful to distinguish at least four different functions that the attitude toward a software system can serve (FIGURE 6). The knowledge function helps to organize information by categorizing computer systems and making judgments about a specific system based on values assigned to the categories (Snyder and DeBono 1989, 340). Attitudes may serve this function if users have earlier experiences with similar systems; however, it is supposed that this function will not count for much once the systems are known well enough. The utilitarian and object-appraisal functions, from now on collectively referred to as utilitarian, are very similar. They imply that attitudes guide users to choose systems to avoid punishments and maximize rewards. The social adjustment and value-expressive functions are also quite close to each other and henceforth referred to simply as value-expressive. Users can, for example, to express themselves choose a more efficient system even if it brings along more punishments. The ego-defensive function, for its part, implies that users might form otherwise unjustified negative attitudes toward an efficient system just to protect themselves from facing their inability to learn the system. The knowledge and utilitarian functions are instrumental according to Abelson and Prentice while the value-expressive and ego-defensive functions are symbolic.

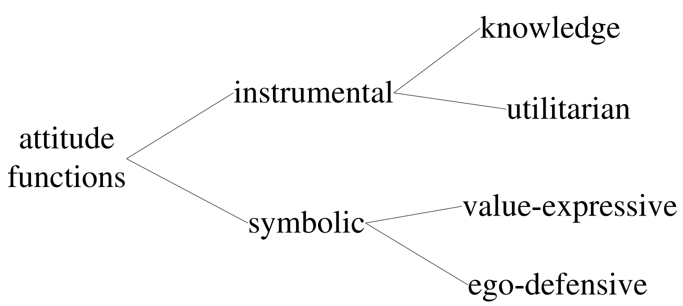


FIGURE 6. Functions of attitudes toward software systems

McGuire (1989, 38-44) discusses seven alternative models of the structure of individual attitudes. The most interesting are the cognitive-affective-conative model, the attributes-evaluation model and the serial sufficing-selections model. The attributes-evaluation model assumes that an attitude toward an object is the sum of attribute-products that are calculated as the subjective value of the attribute multiplied by the subjective probably of its existence in the object. McGuire (1989, 42) notes that this model depicts humans as completely rational value maximizing decision-making machines. The serial sufficing-selections model also emphasizes the utilitarian function of attitudes, but it proposes a simplified version of the attitudinal preference ordering process. It requires less cognitive work as the alternatives are evaluated first only by the most salient attribute and then those alternatives that exceed a sufficing value are evaluated by the next most salient attribute on the following round.

The cognitive-affective-conative model suggests that an attitude is composed of three components. Ajzen (1989, 242-244) explains that an attitude is a latent construct that can be inferred from verbal and nonverbal responses which can be categorized according to the cognitive-affective-conative model. His description of the three response categories gives a good understanding of this model even though he does not talk about distinct component but only about responses of different nature. Cognitive responses are for example expressions of beliefs about the attitude object or lower thresholds for the perception of positive or negative stimuli. Affective responses are expressions of feelings toward the attitude object or bodily reactions such as facial expressions. And conative responses are actions or expressions of intended behavior with respect to the attitude object. Ajzen (1989, 245-247) also describes a hierarchical cognitive-affective-conative model which is depicted in FIGURE 7. The idea of three components has also been criticized (e.g. Cacioppo, Petty and Geen 1989, 277-295); however, it is still useful and widely accepted (e.g. Huskinson and Haddock 2004).

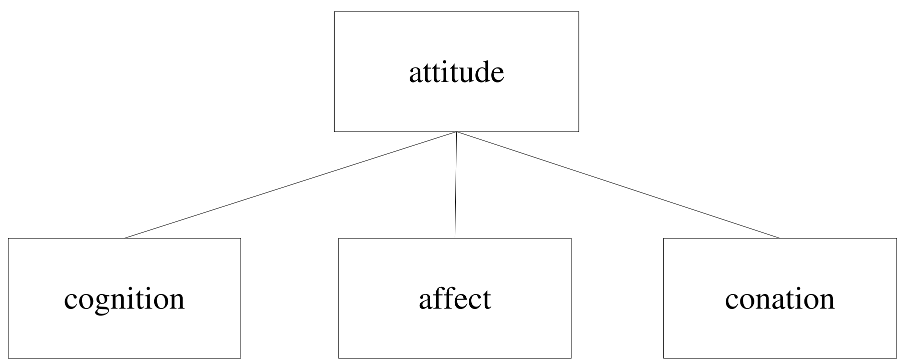


FIGURE 7. Hierarchical model of attitude (Ajzen 1989, 246)

After a discussion of the cognitive-affective-conative model Ajzen (1989, 248) presents a complementing causal chain perspective in which cognitive beliefs cause attitudes, which in turn cause intentions and behavior. He also notes that the influence can occur in the opposite direction from behavior to attitudes and from attitudes to beliefs for example in the case of habitual or forced behavior. In a recent paper Huskinson and Haddock (2004) choose to concentrate on the affective and cognitive components of attitude. They suggest that affective and cognitive information guide attitudes and their research shows that there are individual differences in the relative importance of these two components. Furthermore, their results indicate that cognitive information influences the attitudes of cognition-based individuals more, while the attitudes of affect-based individuals are more influenced by affective appeal. Based on these findings a slightly modified version of Ajzen's causal chain model is proposed and presented in FIGURE 8 to describe the formation and function of attitude. According to this model both affective appeal and cognitive beliefs influence a person's attitude toward an object, and the attitude in turn influences intension, decision making and behavior.

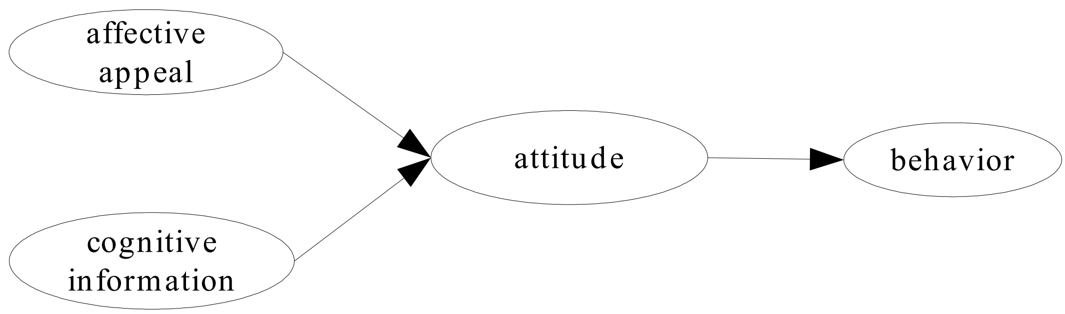


FIGURE 8. A causal chain model of attitude

Breckler and Wiggins (1989, 408) review different definitions of attitude and emphasize that attitudes are learned, imply evaluation and predispose action. Pratkanis (1989, 70-92) discusses the cognitive representation of attitudes. He views an attitude as “a person's evaluation of an object of thought” and he stresses that an attitude is stored in memory and then used as a simple heuristic in cognitive processes such as reasoning and decision making. The evaluative characteristic of attitude is emphasized in this study and the model presented in FIGURE 8 is considered a good description of attitude as a learned predisposition to behave consistently in line with one's evaluation of the attitude object. As satisfaction is a special case of attitude (Keinonen 1998, 68) and choice is a special case of behavior, the similarity between the research model (FIGURE 1) and the causal chain model of attitude (FIGURE 8) is obvious.

3.2 Measures of satisfaction

Various standardized questionnaires have been proposed as measures of end-user computing satisfaction. This section analyzes and criticizes some of the most frequently used operationalizations of user satisfaction and explains why they are not suitable for this study.

Measurement instruments of user satisfaction can be derived from the technology acceptance model (TAM) that Davis (1989; 1993) proposed to describe the formation of attitude and acceptance or rejection of information systems. TAM is based on an earlier theory of reasoned action by Fishbein and Ajzen (1975) and it can also be described as a special case of Ajzen's (1989, 248) causal chain model as it suggests that system characteristics influence perceived ease of use and perceived usefulness, these in turn influence attitude which influences intentions and finally intentions influence actual use.

Davis (1989) proposes two rating scales as measurement instruments of perceived usefulness and perceived ease of use which are fundamental determinants of user acceptance according to TAM. These rating scales have been used as operationalizations of user satisfaction (e.g. Keinonen 1998, 48); however, an analysis based on the framework developed in chapter two reveal that these rating scales address only a limited number of aspects that are supposed to influence user satisfaction. The ease-of-use scale contains items that make statements about the system being learnable, understandable, flexible and controllable, while the usefulness scale contains items about efficiency, effectiveness and utility. All the items of the usefulness scale are connected to the job a user is supposed to do with the system and therefore utility is an additional precondition for high scores on all these items. In this study satisfaction is supposed to be a very complex construct and the user's perception of this limited set of usability aspects is not regarded as good measure of satisfaction.

Another famous operationalization of satisfaction is proposed by Doll and Torkzadeh (1988). They conducted a thorough study to develop the end-user computing satisfaction instrument (EUCSI). They collected questionnaire items from previous research and added items concerning ease of use because it had previously been neglected in satisfaction inquiries. In this way they created a 40-item questionnaire that was tested in personal interviews with 96 computer users. Their analysis was based on the assumption that the sum of the questionnaire items and two general criterion items properly reflected user satisfaction, hence they modified the questionnaire by dropping out those items that were not correlated with the sum of the other items and the criterion items. They continued with an 18-item questionnaire that was tested again through personal interviews but this time with 618 users. A factor analysis of the questionnaire items revealed that the concept of user satisfaction could be interpreted as a structure consisting of five factors that Doll and Torkzadeh named content, accuracy, format, ease of use, and timeliness.

Doll and Torkzadeh (1988) checked the correlation of individual items with the corrected item-total and a criterion item also in their second study. This time all the correlations were high but Doll and Torkzadeh still deleted six more items to improve the distinction between the factors. Finally they proposed a 12-item instrument (EUCSI) as a standard measure of application-specific user satisfaction.

McHaney, Hightower and Pearson (2002) carried out a survey in Taiwan to assess the construct validity, internal validity and reliability of EUCSI in cross-cultural settings. Their analysis gave similar results as Doll and Torkzadeh (1988) report, and they also mention a number of studies that have replicated these results. Harrison and Rainer (1996) used EUCSI as a general measure of computing satisfaction in a survey with the salaried personnel of a university. Their results also support the reliability of EUCSI and the validity of the five-factor construct even when EUCSI is not used as an application-specific instrument.

Even though many studies have confirmed that the individual items of EUCSI are correlated with the total and that the factor structure exists in the answers, the validity of EUCSI as a measure of user satisfaction can be questioned.

Satisfaction is defined as an affective attitude also by Doll and Torkzadeh themselves, still EUCSI does not include any items to measure affective appeal or attitude. A purely theoretical analysis of EUCSI reveals that it concentrates on utility aspects and further it assumes that the task of generating reports is central in all end-user computing. This view might have been appropriate in the 1980s; however, today a typical end-user would probably be confused with a questionnaire focusing on the quality of information output.

The items of EUCSI can also be analyzed with respect to the framework developed in chapter two. The items of the content factor measure perceived utility simply as tasks supported or not supported, while the items of the accuracy, format and timeliness factors focus on the system's performance on the supported tasks or how well the tasks are supported. The items in the ease-of-use factor as well as one of the items in the format factor address the user's perception of the cognitive aspects of the system's usability.

Both EUCSI and TAM concentrate on utility and ease of use. Legris, Ingham and Collerette (2003) conducted a meta-analysis of empirical studies that utilize TAM. They reviewed a number of scientific periodical publications from 1980 to 2001 and found 22 studies that met their criteria. Their meta-analysis indicate that this model does not explain more than 40% of the variance in actual use of information systems. And therefore, they conclude that TAM should be integrated into a broader model. McHaney, Hightower and Pearson also note that a comprehensive instrument for measuring user satisfaction does not exist and they regard EUCSI as a good substitute.

Chin and Lee (2000) have also contributed with a questionnaire for measuring user satisfaction. They first present a model that explains the formation of satisfaction. In this model satisfaction is composed of "expectation based satisfaction" and "desire based satisfaction." Expectation based satisfaction is formed by evaluating the discrepancies between prior expectations and post-hoc perceptions, while desire based satisfaction is formed by evaluating the gap between prior desires and post-hoc perceptions. The questionnaire of Chin and Lee is based on EUCSI but it contains additional comparative questions, for example, "To what extent does the ease of use of the system meet your original expectations?" This questionnaire is of course limited to the same aspects as

EUCSI.

The conclusion of this discussion is that satisfaction is very difficult to measure. Standardized questionnaires are always limited to some predefined aspects and the relevant aspects might differ depending on the nature of the attitude object and prior interaction with it. Unstructured interviews with open-ended questions might be the best way to find out whether the attitude is positive or negative; however, results would not be comparable and it would be extremely difficult to estimate the actual degree of satisfaction.

3.3 Rational versus affective choice

Keinonen (1998, 67) notes that the choice between alternative products is a problem-solving process in which the consumer aims to optimize the results. He also mentions more advanced models in which focus is on the relationship between the effort of the process and the quality of the results.

Eysenck and Keane (2000, 483-488) start their section on decision making by presenting a normative theory which assumes that humans always try to maximize the subjective value of the outcome of a choice. Then they continue by presenting a number of phenomenons in which people do not behave in line with this theory of optimal decision making. Eysenck and Keane mostly focus on rational decision making and do not pay attention to how affective processing influences decisions.

Norman (1993, 128-130) describes the decision making in high-level business meetings that he has attended in the American industry. He points out that decisions are prepared with presentations of facts, computations and estimations, but then, highly placed executives usually tell some personal stories to add deep understanding of the context and emotional aspects that are connected to the decisions. Norman thinks it is very important to combine both rational logic and affective value judgments in decision-making settings.

In his textbook Norman (2004, 12) stresses that “affect and emotion are crucial for everyday decision making.” He refers to Damasio's (1994) study of people with brain injuries impairing their emotional systems and explains that tight choices between alternatives, which are rationally estimated to provide equal value, are purely based on emotions. Furthermore, he explains that the affective system aids in decision making by reducing the number of things to be considered. Goleman (1995, 27-28) also refers to Damasio and explains that the decision making of patients, whose emotional systems are impaired,is terribly flawed even though their cognitive systems work perfectly well.

Etzioni (1988) suggests that most choices are based on emotional involvement and value commitments. He proposes a theory in which normative-affective factors define the relative weight of normative-affective and logical-empirical considerations in decision making. For example grading a thesis is a decision in which the normative-affective factors require only logical-empirical factors to be considered. Etzioni's theory and Damasio's findings suggest that decisions are based on both affective and cognitive information.

3.4 Satisfaction measured as preference

The intension of this chapter was to justify the operationalization of satisfaction as preference. Comparing the literature on attitude reviewed in section 3.1 and the literature on choice reviewed in section 3.3 shows that the formation of attitude and the decision making involved in choice are very similar processes. Furthermore, as the attitude toward a software system evaluates and functions as a heuristic in decision making it is reasonable to assume that user satisfaction emerges as preference when alternative systems are (1) equally well known, (2) very well known, (3) freely available, and (4) specific real-world goals are to be achieved when using the system.

Keinonen (1998, 90) defines preference as “the relativistic behavioural response to the object, i.e. selecting one option rather than another, without considering the costs.” Keinonen (1998, 66) also explains that consumer choice behavior varies depending on the degree of involvement. The operationalization of satisfaction in this study is restricted to the case of high-involvement, in which the evaluator is prepared to acquire enough information about each option to be able to give a fair judgment. The high-involvement choice, in which the user knows all alternatives equally well and very well is considered to be close to post-purchase evaluation. The alternatives must also be freely available so that the evaluator does not need to consider the costs. Real-world goals in the final restriction refer to goals that are not in virtual reality nor in the computer system itself. This restriction was added simply because the constituents of satisfaction are supposed to be very different when considering computer games for example.

3.5 Evaluation

The contribution of this chapter is limited to a synthesizing model of attitude (FIGURE 8) and a justification of the operationalization of satisfaction through preference. The validity of this operationalization is based on assumptions about attitude formation, decision making in choice and the role of attitude in choice. All these assumptions can be questioned. Most of them are based on empirical research results and some are theoretical speculations. However, this operationalization seems to be acceptable as other researchers implicitly assume a relation between satisfaction and preference. Nielsen and Levy (1994), for example, investigate preference and make conclusions about satisfaction and the study of Frøkjær, Hertzum and Hornbæk (2000), already introduced in the previous chapter, operationalized satisfaction as the preference of user-interface alternatives without any further doubt or justification.

All the restrictions connected to the operationalization of satisfaction as preference make it somewhat limited. When user satisfaction is measured for a specific application which is in use in an organization, questionnaires like EUCSI might be the most practical measure. Furthermore, operationalization through preference is not possible when we do not have alternatives or when one alternative is learned and well known while the others are unfamiliar to the evaluators.

Hufnagel and Conca (1994) note that comparing questions yield more consistent answers than direct questions because they provide an anchor point against which judgments can be made. Alternative systems surely provide good anchor points for a comparison; therefore, very high reliability can be assumed if the restrictions mentioned above are properly taken into account. The operationalization of satisfaction as preference is assumed to be useful in research and the discussion of this chapter as a justification of this operationalization can be considered as a small contribution.

4 USER-INTERFACE DESIGN THEORY

User-centered design is widely recognized as a tried and trusted method to achieve usable systems (e.g. Mayhew 1999). It is shortly described in section 4.1 as it was practiced in the development of the user-interface alternatives for the field experiment described in the following chapters. As it is impossible to contrast all aspects of usability in a single experiment, only a few interesting aspects were chosen for implementation. A short discussion of the usability heuristics (4.2) and conceptual models (4.3) related to the chosen aspects is provided in this chapter as relevant background theory for the design and implementation of the user-interface alternatives described in the next chapter. This chapter does not attempt to provide a comprehensive survey of either of these topics. It simply concentrates on those specific heuristics and conceptual models that are necessary to be aware of in order to understand the differences between the user-interface alternatives that were developed for the field experiment.

4.1 User centered design

Gould and Lewis (1985) present three principles of user-centered design:
- early focus on users and tasks,
- empirical measurement (i.e. usability testing), and
- iterative design.

Vredenburg, Mao, Smith and Carey (2002) conducted a survey with experienced usability professionals to find out what usability principles are most widely used in the industry. Their survey shows that iterative design and usability evaluation are clearly the most frequently deployed principles and these are also perceived to have a strong impact on the quality of the user-interface. The textbooks on the topic also seem to agree on the importance of iterative design and usability testing (e.g. Nielsen 1993; Dix et al. 1998; Mayhew 1999; Preece et al. 2002).

As involving users is regarded important, the number of users needed is the next problem. Nielsen (1989) conducted a survey study with experienced computer users about the relative importance of various usability aspects. He also presented a list of programs and asked the respondents to evaluate those that they were familiar with. He gave four predefined statements to be rated and calculated standard deviations for each of them. The results show that users agree quite well on “overall user friendliness” while there was much more disagreement on whether the programs included the needed features. Nielsen concluded that one can get reliable data about overall user friendliness by asking only a small number of users. In a later study Nielsen and Landauer (1993) found that about 80% of the usability problems in an application are found with five test users and additional users will mostly just repeat the same problems without giving much further insight (cf. Nielsen 2000). Other researchers have got contradictory results; however, in a recent panel discussion in the CHI-conference Nielsen argued that the ideal would be to do testing with 3-4 users and save resources for more iterations instead (Barnum, Bevan, Cockton, Nielsen, Spool and Wixon 2003).

4.2 Usability heuristics

Nielsen (1994a) compared several sets of usability principles and design guidelines. He rated the explanatory power of each principle and guideline on 249 different real-world usability problems and developed through factor analysis a synthesizing set of usability heuristics. This set is perhaps the most famous guidance for user-interface design and evaluation. However, the factor analysis bunched together usability principles that address quite different aspects of usability. These factors are probably fine as evaluation heuristics for usability experts. However, an analysis of some of these rough bunches is necessary to explain how visibility and user control was contrasted with simplicity in this study.

Nielsen's (1994b, 30) factor:

> Visibility of system status: The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

consists of principles and guidelines mostly about feedback and how to design responsive systems. However, it also includes principles like “do not hide features" and "provide status information" (Nielsen 1994a) which more accurately reflect what visibility means in this study.

The evaluation of different programs in Nielsen's (1989) survey shows that users feel that outliners and spreadsheets are more pleasant to work with than computer-oriented and business graphics programs. Nielsen concludes that users probably like the outliners and spreadsheets more because they feel more in control of the interaction with these programs. Nielsen also suggests that a feel of directness is likely to be valued by users. Directness according to Nielsen means a "feeling of minimal resistance between user intentions, user actions, and achieved results."

Nielsen's (1994b, 30) heuristic:

> User control and freedom: Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

focuses on aspects that are labeled "recoverability" in this study; nevertheless, it also includes "modelessness: allow users to do what they want" and "allow user[s] to initiate/control actions" (Nielsen 1994a) which properly describe what user control means in the user-interface alternatives that were implemented for the field experiment. User control was implemented as: allow users to do what they want whenever they want.

Visibility and user control as described before are contrasted with simplicity in this study. Nielsen's (1994b, 30) "aesthetic and minimalist design:"

> Aesthetic and minimalist design: Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

describes quite well how simplicity is implemented in the user-interfaces designed for this study. Simplicity is implemented here particularly by excluding rarely needed information and functionality.

4.3 Conceptual models

Visibility, user control and simplicity as described above are implemented in this study by the means of designing different conceptual models. A conceptual model is here defined as the fundamental design decisions that determine how human users are supposed to understand a system. Preece, Rogers and Sharp (2002, 41-55) review different kinds of conceptual models. They divide the models into two categories: (1) models based on activities and (2) models based on objects. Further, they present four kinds of activities

- instructing,
- conversing,
- manipulating and navigating, and
- exploring and browsing

as the basis for conceptual models. They point out that these are not mutually exclusive and systems often use a combination of different interaction styles. This study uses instructing in the most simple user-interface while the others combine instructing either with conversing or navigating.

5 METHOD

The following three chapters report a field experiment that was conducted in conjunction with the course “ITK135 – Databases and Information Management” at the University of Jyväskylä. Section 5.1 of this chapter discusses the research strategy, choice of methods and design of the field experiment with respect to the research questions of this thesis. Section 5.2 analyzes the task of practicing SQL-queries which the participants were supposed to do in the experiment, and the user-interface alternatives that were implemented for the experiment are described in section 5.3. Issues concerning Internet research are discussed together with the description of the procedure of this experiment in section 5.4. The data collection and analysis are explained in sections 5.5 and 5.6 respectively. And finally the chapter ends with an evaluation of the empirical work in section 5.7.

5.1 Research strategy

von Wright (1971) describes two main traditions of science. He calls them the Aristotelian and the Galilean tradition. He points out that the Galilean tradition actually has an ancestry that dates back to Plato's times, hence it is as a tradition not as much younger as the name might suggest. The Aristotelian tradition aims to understand and describe, while the Galilean tradition aims to explain and predict. The Aristotelian tradition emphasizes descriptive aspects that aim to make facts teleologically or finalistically understandable, and it can be connected with modern qualitative research. The Galilean tradition, on the other hand, takes a causal-mechanistic view and it can be connected with quantitative research that most often is of hypothetical-deductive nature. These two traditions have generated a paradigms war, or at least a fierce debate, in which advocates of qualitative and quantitative methods try to prove the superiority of their own paradigm. Patton (2003, 66-73) takes a pragmatic approach to this issue and stresses the importance of matching the method to the situation and to the goals of the study. A qualitative approach is suitable when there is no theory available from which to deduce hypotheses or when deep understanding of a specific case is pursued. In contrast, a quantitative approach is suitable to test existing theory or to probe only a few predefined issues from many respondents. Thus, a quantitative approach yields knowledge about the average state of affairs instead of understanding of the reasons for a particular state of affairs. Strauss and Corbin (1998, 33) advocate “a true interplay of methods” in which qualitative results direct quantitative research and vice versa in a circular and evolving process. They stress that creating theory is the ultimate goal and the method is only a way to accomplish this goal. Hence, the primacy of either mode of doing research would not be reasonable.

When planning strategies for design, data collection and analysis it is useful to check how others have investigated similar topics. Nielsen's (1989) study “What do users really want?” was a first serious attempt to create a relative ranking of some different aspects of usability according to how important they are to users. He conducted a survey study with the Copenhagen Macintosh user group, which he felt was quite representative of experienced computer users. However, a small controversy in Nielsen's study is that he aimed to understand what users themselves find important in user-interfaces and still he used a method that Patton (2003, 56) would call “pigeon holing.” That is, he had the users rate some predefined aspects of usability on a Likert scale. His questionnaire surely did not cover all possible aspects of usability and he did not present any kind of framework or classification of the aspects so that he could have covered at least all classes of usability aspects. This study uses open-ended questions to let the users emphasize what they feel important and a framework of usability aspects is developed and applied in the analysis.

If Nielsen's (1989) study was pioneering work to understand the constituents of user satisfaction, then the study of Lindgaard and Dudek (2003) as well as Norman's (2004) textbook represent the latest advances in this subject area. Lindgaard and Dudek conducted several experiments in each of which some 20 participants evaluated two different websites. The participants knew that their task was to assess the websites and they had 10 minutes to explore each site. After this they evaluated the sites in unstructured interviews and filled in an assessment form. Lindgaard and Dudek found that rating scales and interviews often give similar results, but the qualities they emphasize differ. In their study a highly attractive and somewhat unusable site got lower scores on the rating scales compared to the interviews.

The design of this study was inspired by the study of Lindgaard and Dudek. The design is referred to as a field experiment, even though it is not a neat test for linear correlation between two continuous variables with random assignment to groups and measurement before and after treatment. Nonetheless, some characteristics of a traditional experiment can be identified. User-interface properties were the independent variables of this design, while user satisfaction was the measured dependent variable. Also the order in which the user-interface alternatives were tested was counterbalanced.

The field experiment was designed to address two specific sub-questions of the main research problem. The first one contrasts simplicity and additional features: Do users prefer simple generic tools or feature-rich task-specific systems? The second one is about the trade-off between simplicity and visibility: will users prefer an interface consisting of simple screens or one that keeps all the information and functionality visible all the time? In this way, the study focuses on the relative importance of additional features, observability, controllability and simplicity as constituents of user satisfaction. This restriction was necessary because all aspects of usability could not be equally emphasized with a reasonable number of user-interface alternatives. These aspects were chosen to have all three top-level categories involved and because they could easily be implemented as trade-offs without deliberately designing poor alternatives that would appear flawed to the users.

A study can combine elements from both qualitative and quantitative approaches even though qualitative approaches tend to be inductive while quantitative approaches tend to be hypothetical-deductive (Patton 2003, 56-57). In addition to the causal-mechanistic approach of the field experiment, this study also used a qualitative approach in order to recognize other constituents of user satisfaction and to understand their importance.

Patton (2003, 38-41) explains that a researcher should be aware of an overall framework of research strategies when making decisions about a specific study. He presents strategies for design, data collection and analysis in a strategic framework for qualitative inquiry. Patton's design strategies include naturalistic inquiry, emergent design flexibility and purposeful sampling. This study was not a pure qualitative inquiry; instead, the design of this study combines elements of a field experiment and purposeful sampling. Patton (2003, 230-244) further discusses the details of how to design a study based on purposeful sampling. He presents the combination of other types of sampling strategies as the last type in a list of 16 different types of purposeful sampling strategies. The sampling strategy of this study combines two types of purposeful sampling: “opportunistic or emergent sampling” and “intensity sampling.” The ITK135-course has traditionally been very theoretical and all practical training has been done with pen and paper. Constructing SQL-queries is one of the most central topics in the course; however, interacting with a database management system and trying out SQL-queries in practice has been up to the students' own initiative. As a teacher assistant for the ITK135-course I recognized the obvious need to engage students in practical training. Hence, an important aspect in the design of the study was the opportunity to conduct a field experiment in which students would gain from the opportunity to use a software system to practice SQL-queries while participating in the experiment. Intensity sampling is reflected in the choice of a group of IT-students who are experienced computer users and the choice of an application which is used to achieve a real-world goal, that is, to complete course exercises to gain points for a better grade. Furthermore, being familiar with the teacher responsible, course content and schedule provided additional convenience to the study.

After starting with purposeful sampling the qualitative side of this study continued with a data collection strategy that Patton (2003, 40) simply calls “qualitative data.” It aims to “capture direct quotations about people's personal perspectives and experiences.” This study did not presuppose anything in the data collection. Instead it gave the respondents the freedom to elicit and emphasize whatever aspects they felt important.

Among the strategies for qualitative analysis Patton (2003, 55-57) describes one particular strategy that he calls “inductive analysis and creative synthesis.” This strategy stresses that the important dimensions of the analysis should not be nailed down in advance. Instead, they should emerge from patterns found in the data. The analysis process starts by exploring, then confirming and ends with synthesis. Patton (2003, 453-454) also explains that qualitative analysis can start with a theory in a similar way as analytic induction starts with a hypothesis; and, deductive analysis can go on alongside inductive analysis and possible modification of the theory. The analysis of this study starts with a theoretical framework instead of a mental blank state and still it remains open to new insights and possible modification of the original framework.

5.2 Tasks

Johnson and Nardi (1996) discuss user preference for task-specific versus generic application software. They explain that a simple notion of "task-specific" is seldom appropriate because tasks vary very much depending on situation and environment; therefore, one should not pretend to be able to gear software to a specific task but rather to a task domain that can contain many different activities. Users will in any case find new applications for a software product and try to use it in unexpected ways.

The task domain in this experiment can be defined as practicing SQL-queries with a given relational schema. This task domain can be divided into two categories: completing exercises that were given during the course and constructing self-invented queries. In both cases, the purpose of the query must first be decided. In the case of a given exercise, the exercise must be located, read and understood. Then the completion of a single query starts with a problem solving process in which information about the relational schema is needed and possibly also an error-message in the case of a previously failed query. The problem solving should result in the construction of an SQL-query and in the following stage the user types in the query and instructs the system to execute it. After this the user evaluates the results returned by the system and tries to estimate if the query was correct. The students also need to get the queries stored on paper that they can bring with them to the class. Finally, at least seven independent tasks can be identified within this task domain: finding or defining problems, solving problems, typing in queries, executing queries, evaluating results, saving queries and viewing or printing queries.

5.3 User-interface alternatives

In February 2004 a web-based system with three alternative user-interfaces was designed and implemented for the field experiment. The system was designed to support the tasks described in the previous section and to let the students of the ITK135-course practice SQL-queries in an interactive environment. All three user-interface alternatives had the same look and feel in order to standardize the visual appeal and the emotional impact of the alternatives. They differed in conceptual models and interaction style. The user-interface alternatives were carefully designed and implemented in an iterative development process that included video-taped user testing to eliminate usability problems. The usability tests are documented in APPENDIX 2.

The first user-interface alternative, called gene, gave the impression of a generic system because it did not connect the queries to the course or to specific exercises. The conceptual model of the gene-interface was based on instructing and it was very simple and easy to use. One more advantage of the gene-interface was that it did not require the user to create a user-name nor to log in before using the system. A screen-shot of the gene-interface is presented in FIGURE 9. A downside of the gene-interface is that the users have to use one more generic tool, that is, a text-editor, to be able to save and print queries.

FIGURE 9. The gene-interface

The other user-interface alternatives were task-specific by presenting the exercises that were given in the course. They were significantly more complex because of added functionality. They required the user to log in and to find the correct exercise. In addition, they provided the possibility to save tested queries and to browse, view or print all saved queries. One of the task specific interfaces was called visi and the other one was called conv.

The conceptual model of the visi-interface was based on navigating visible objects. This interface kept all functionality and information visible all the time; thus, it emphasized user control and freedom. The downside of this approach was a complex and perhaps even cluttered screen design. FIGURE 10 demonstrates a typical information rich state of the visi-interface.

FIGURE 10. The visi-interface

The conv-interface consisted of several simple screens each of which presented only a single task. The conceptual model of the conv-interface was based on conversing and the interface assumed a predefined order of actions. The video-taped user testing revealed that novice users made less errors with the conv-interface than with the visi-interface. The conv-interface guided the user step by step through the task, while the visi-interface made the overall picture of the whole system more apparent. Typical usage scenarios with each user-interface alternative are presented as sequences of screen-shots in APPENDIX 1 and the conv-interface in the state for selecting exercise is presented in FIGURE 11.

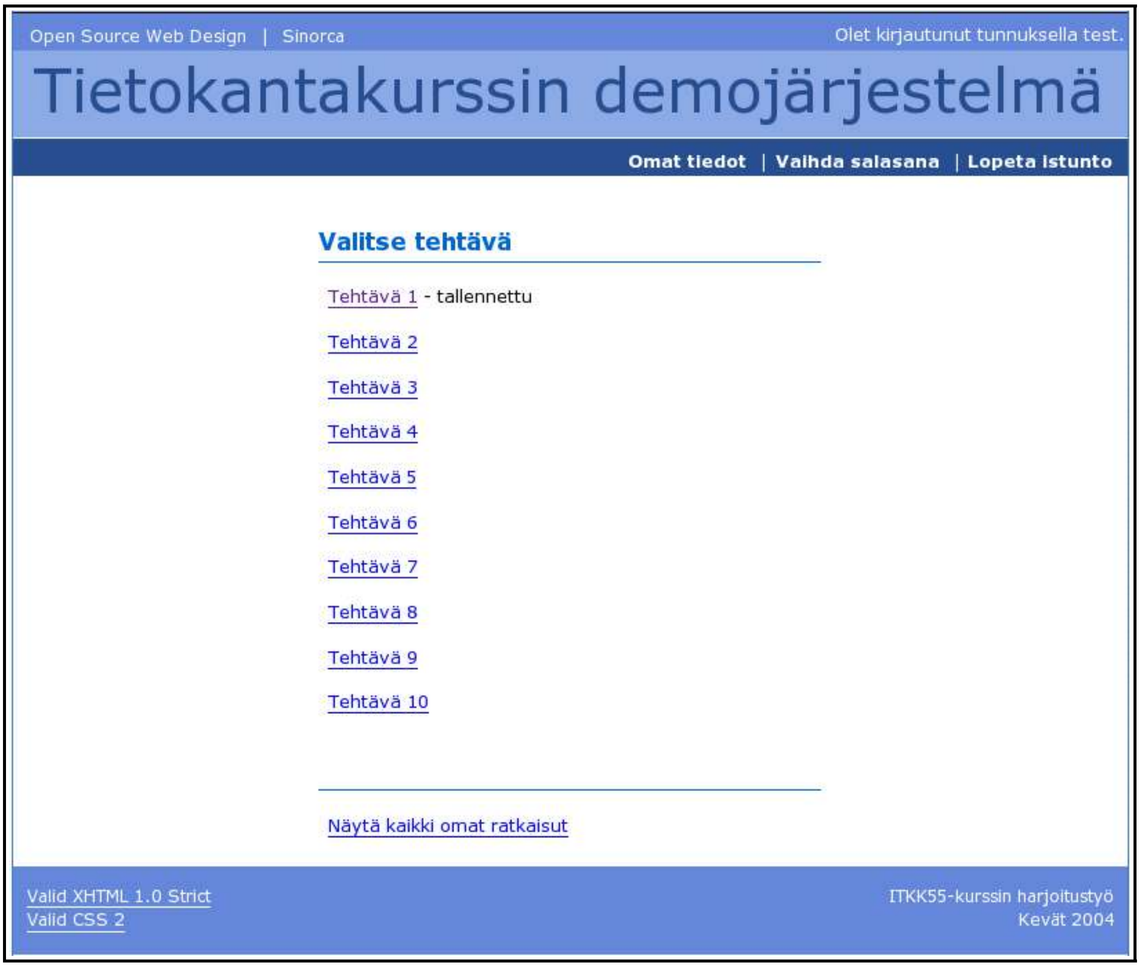


FIGURE 11. The conv-interface in the stage of selecting exercise

The three user-interface alternatives and a summary of their characteristics are presented in FIGURE 12. The user-interface alternatives are depicted as rectangles and the conceptual models as ellipses. The horizontal line separates the generic alternative from the ones that have additional task-specific functionality. The attributes presented to the right in FIGURE 12 are the most salient characteristics of each alternative from the user's perspective.

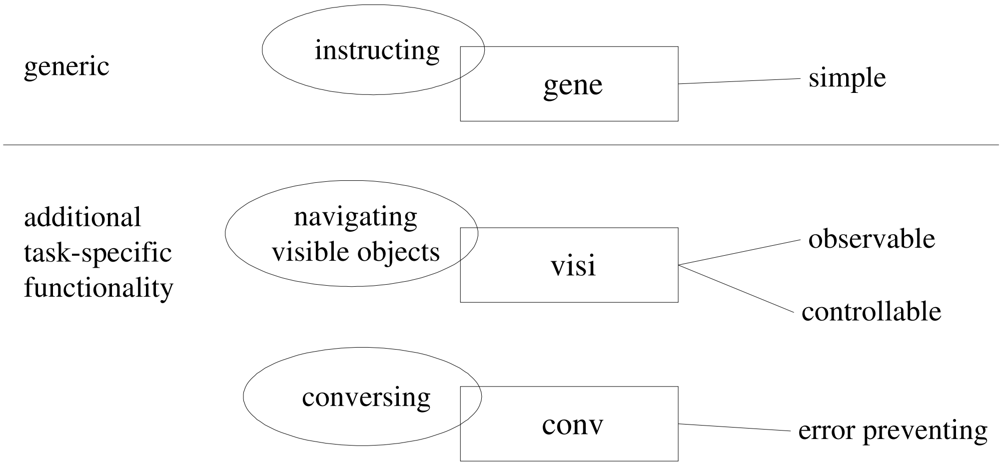


FIGURE 12. The user-interface alternatives

5.4 Participants and procedure

Nosek, Banaji and Greenwald (2002) discuss some special issues related to Internet-based research in comparison with traditional laboratory settings. They warn that people participate Internet-based experiments in very different environments. This is not likely to be a big issue here, since the field experiment was intended to be carried out in a natural environment where other activities, stress, attitude etc. are desired parts of the experiment settings. There are also many advantages with Internet-based research. Some examples are that the anonymity of the participants is easy to preserve and the participants are not likely to get into uncomfortable situations or feel stupid when they can participate in their homes without anyone watching them. The Internet-based design and the absolute anonymity of the participants helped to overcome some difficulties that Eskola and Suoranta (1998, 55) point out in the setting of a teacher studying his students.

Nosek, Banaji and Greenwald (2002) also explain that Internet research puts higher demands on experiment design and instructions because the researcher cannot catch problems and give additional instructions when needed. The instructions for this experiment were sent to the students by e-mail. To make sure that the participants would easily be able to get started with the experiment procedure and follow it the instructions were tested with users for understandability in conjunction with the video-taped usability tests (APPENDIX 2).

All participants of the field experiment were students of the ITK135-course; thus, most participants were experienced computer users and very comfortable with testing new software systems. Participation in the field experiment was voluntary and the advantage of using the system functioned as a motivator for the students. There were some 200 active students on the course and these were divided into three groups according to their last name. All groups were asked to familiarize with all user-interface alternatives and to choose the user-interface that they preferred. The order in which the participants were asked to test the alternatives was varied between the groups. Pure counterbalancing would have required six groups; however, it was assumed that some kind of confirmation bias might favor the interface alternative that was first tested and learned, while the second and the third alternative would be equally treated in the participants' evaluation. The order for the first group was visi, conv, gene; for the second group conv, visi, gene; and for the third group gene, visi, conv.

So called demo-classes were arranged each week on the course. The students prepared for the demos by completing assigned exercises as homework so that they would be able to demonstrate their solutions in the class. SQL-queries were practiced in the demos of the last two weeks before the first intermediate exam. The instructions for testing and using the web-based system were sent to the students in the beginning of the first week of SQL-demos. The system was in use for two weeks until the intermediate exam, and the e-mail message asking the students to answer the web-based questionnaire was sent out right after the exam.

The number of participants cannot be simply stated as one definite figure. 120 user-names were created during the course and 114 of the user-names had distinct e-mail addresses. This shows that some students created two or more user-names. It seems that some user-names were never used because only 101 user-names had both distinct e-mail addresses and saved queries. The questionnaire presented to the students at the end of the experiment was answered by 64 students. However, only 21 of the respondents had tested all user-interface alternatives as the field experiment required. Still 43 respondents answered the open-ended questions and provided valuable data for the qualitative analysis.

5.5 Data collection

Data was gathered with a web-based questionnaire. The questionnaire was developed according to the principles of iterative design and usability testing (see section 4.1). The usability tests (reported in APPENDIX 3) revealed that sometimes participants do not even bother to read instructions that are not directly questions to be answered. The interactive questionnaire was therefore designed in such a way that it forced the respondent to focus on the right questions by disabling some questions or presenting additional questions according to previous answers.

The questionnaire consisted of three parts. The first part gathered background information about the participants with closed questions. The background information included the participant's own assessment of her competence as a computer-user, attitude towards the SQL-exercises in the course, and how difficult she found the course. The second part of the questionnaire was intended to check whether the respondent had followed the instructions of the experiment. It simply asked which user-interface alternatives the respondent had tested and whether she had used the system during the SQL-demos. The third constituted the actual data collection. It probed for the user-interface choice and underlying reasons.

Patton (2003, 21) notes that scientists often have preconceptions that dictate what data they are able to gather. He explains that qualitative researchers should only provide a framework within which the respondents can freely present what they find important. The questionnaire of this study provided this kind of framework by asking the participants to justify their choice with three open-ended questions:

- What are the most important positive attributes of the chosen user-interface?
- What are the most important negative attributes of the other user-interfaces?
- How could the chosen user-interface be improved?

Furthermore, the respondents were asked what they usually find important in user-interfaces and they were also encouraged to give other comments. In addition, the questionnaire included the standard SUS-questions (Brooke 1986) to get a general rating of the user-interfaces.

The questionnaire was quite short and easy to complete; however, the e-mail message that asked the students to answer also prompted them to reserve 10 minutes for responding so that they would have enough time to think carefully about their responses and to write answers to the open-ended questions. After they had submitted the questionnaire they were provided with a simple debriefing page with information about the research and an e-mail address in case they wanted to know more about the research or the use of their responses. The interactive questionnaire and its functionality can be viewed and tested at <http://www.cc.jyu.fi/~mabengts/gradu/itk135/form1.php>.

5.6 Analysis

The quantitative analysis consisted of simple descriptive statistics and some statistical tests. The frequency distribution of the preference for the user-interface alternatives was calculated and then cross-tabulated with different background information. Association between background information and preference was tested for nominal variables with the Pearson χ^2 test and for interval scale variables with the Mann-Whitney U test (see Coolican 1994, 259-274).

The qualitative analysis combined elements from both deductive and inductive strategies. The aim of inductive qualitative analysis is to discover new patterns or categories from the data, while deductive analysis is based on an existing framework (Patton 2003, 453). Both types of analysis can naturally go on alongside when the researcher keeps returning to the data to verify or modify the framework used. The inductive analysis of this study was open so as to allow various new usability aspects to emerge from the data, while the deductive analysis quantified the data by coding the evaluative statements into the categories of the framework originally developed in chapter two and then modified inductively during the qualitative analysis. Inductive and deductive results are presented separately in chapter 6 even though both types of analysis were carried out in parallel.

5.7 Evaluation

Patton (2003, 20-21) describes as an example of the fruit of qualitative methods a study in which open-ended questions in a questionnaire yielded much more informative and credible data when the quantitative part of the study was suspected to be biased. Yet, he explains that there are many problems with open-ended questions in questionnaires. The writing skills of respondents, the impossibility of asking for more description or details and the effort required by the respondents are the limitations that he lists. The respondents of this study were students who were expected to write their own theses within a few years. Hence, it was assumed that many of them would be willing to make the effort of writing their responses, and the writing skills were not a notable limitation in this case. The data collected was very information rich and it showed that many students really had done a careful job in writing their responses. The impossibility of extending responses was of course a limitation; but in any case, it was necessary to preserve the anonymity of the respondents, and interviews would not have been acceptable in the case of a teacher using his own students as informants. Interviews would also have been much more time consuming.

Nosek, Banaji and Greenwald (2002) discuss four essential components of research designs – coherence, simplicity, involvement and consistency – originally presented by Aronson, Ellsworth, Carlsmith and Gonzales (1990). The design of the field experiment in this thesis can also be evaluated with respect to these elements. Coherence means that all activities are integral parts of the study so that they appear meaningful to the participants. Testing user-interface alternatives, choosing one alternative, using it on the course and then answering a questionnaire about it should be coherent activities, and I think that participation felt meaningful for the students. Some comments in the data cited in chapter 6 also reveal that the students perceived the web-based system as a great improvement to the ITK135-course, and one purpose of the questionnaire was to gather data for continued development of the user-interface. Hence, the improvement of the course and the web-based system was a meaningful goal that the students knew they could help to achieve through participation.

Simplicity as an element of research designs means that participants should be spared from unnecessary complication. The steps of this experiment would have been difficult and tedious for novice computer users; nonetheless, participation should have been quite simple for the students of the ITK135-course. All the students should have been experienced computer users and very comfortable with e-mail, web-browsers and testing of new software systems. Furthermore, the user-interfaces were designed to be easy to understand and use. Hence, the task of testing the user-interfaces, understanding the alternative interaction styles and choosing the preferred alternative should have been quite simple. The final questionnaire was also designed not to be too tedious for the respondents. Writing answers to open-ended questions required some effort as discussed above. However, most activities of the field experiment were both simple and easy to integrate with the normal duties on the course.

Involvement – also known as experimental impact – means that the experiment should be engaging and interesting for the participants. The web-based system was introduced to the students in the lectures and demo-classes to get the students involved in the study. The possibility of practicing SQL-queries with the system also improved the involvement of the students, and the students knew in advance what they would gain and be expected to give while participating the experiment.

Consistency means that the design should create the same basic state in all participants and standardize the experience. Nosek et al. present some strategies for creating the same basic state of participants in Internet-based research. However, a natural environment for the use of the system was desired in this experiment. Therefore, no attempt to control the variation in environment or state of participants was made.

6 RESULTS

The results of this study are presented in the first three sections of this chapter. Section 6.1 presents the quantitative analysis and reveals what user-interface alternative was the most popular. The inductive qualitative analysis (section 6.2) creates a new concept system based on the one presented in chapter 2 to provide a better understanding of perceived usability. This concept system is then used in the deductive qualitative analysis (section 6.3) to quantify the data and create a ranking of the different aspects of usability according to the number of users who mentioned each aspect on their own initiative when answering the open-ended questions. When considering these results it is important to keep in mind that the field experiment contrasted only a limited number of usability aspects. The last section (6.4) evaluates some aspects of the research, which can be assessed based on the data gathered from the respondents.

6.1 Quantitative analysis

The instructions for the field experiment required the participants to complete two simple SQL-exercises with each alternative before choosing which alternative to use during the course. The database of saved queries shows that circa 40 participants completed the experiment according to instructions. However, only 21 participants among those 64 who answered the questionnaire had tested all three user-interface alternatives. These 21 respondents are included in TABLE 1 which presents for each user-interface alternative the number of participants who preferred it. Those respondents who did not test all alternatives were excluded from the quantitative analysis.

TABLE 1. Preference distribution for the user-interface alternatives

	Frequency	Percent
conv	6	28,6
gene	2	9,5
visi	13	61,9
Total	21	100,0

Chapter 5 suspected that the order in which the user-interface alternatives were tested might have an impact on the participants' preference. The order was varied between the three test groups and the frequency distribution for each group is presented in TABLE 2. Intuitively there seems to be no association between test group and preference. Because of the low popularity of the gene-interface the analysis will henceforth concentrate only on the differences between visi and conv. The two individuals who chose the gene-interface were thus excluded and respondents who were previously excluded because they had not tested gene were now included if they had tested both conv and visi. After these changes the analysis continues with data from 20 respondents.

TABLE 2. Test group cross-tabulated with preference

	conv	gene	visi	Total
group 1	2		5	7
group 2	1	1	5	7
group 3	3	1	3	7
Total	6	2	13	21

Cross-tabulating gender and preference in TABLE 3 indicates a clear pattern. Female respondents preferred the conv-interface more often while males preferred visi. This result is not of much value statistically because of the small number of female respondents; however, it is in line with Simon's (2001) findings which indicate that women prefer sites that are less cluttered while men prefer sites that make extensive use of graphics and animated objects.

TABLE 3. Gender cross-tabulated with preference

	conv	visi	Total
female	3	1	4
male	3	13	16
Total	6	14	20

The Pearson χ^2 statistic can be computed to test for association between two nominal variables in a contingency table. The χ^2 value and significance are reported for the test group -preference and gender-preference associations in TABLE 4. The gender-preference association yields a significance value below the .05 level; nonetheless, no conclusions can be made based on this because the χ^2 test is not valid if more than 20% of expected cell-frequencies are below 5 (Coolican 1994, 266). This is obvious in TABLE 3 as there would be absolutely no association if only one or two female respondents would have made a different choice.

TABLE 4. Association between the nominal variables and preference

	Pearson χ^2	significance
test group * preference	1.07	.587
gender * preference	4.82	.028

The background part of the questionnaire also contained items on which the respondents evaluated their skill level as computer users and their attitude toward the SQL-exercises on closed questions the answers of which were converted to a so called plastic interval scale from 1 to 5 (Coolican 1994, 193). The Mann-Whitney U test, a non parametric test for two independent samples, can be applied when testing for differences between those who chose conv and those who chose visi. The test converts the ratings to ordinal data and thus the mean rank is also included in TABLE 5. A higher value indicates higher skill level or more ambitious attitude. The Mann-Whitney U test found no significant differences in computer skill or work attitude between the two groups.

TABLE 5. Mean rank for the preference groups on the plastic interval variables and significance of the Mann-Whitney U test

	conv	visi	significance
computer skill	8.67	11.29	0.341
work attitude	9.50	10.93	0.591

6.2 Inductive qualitative analysis

All statements about the user-interface alternatives were coded into one of the categories of the framework presented in chapter 2. Some statements did not fit into any of the categories and thus gave reason to inductively create new categories. New categories were added only if statements from at least three respondents could be grouped to form the new category. Otherwise if the statements were so vague that they could not be grouped at the level of means or usage indicators no new category was formed; instead, the statements were taken into account only at the highest level of the concept system.

One of the most frequently used words in the answers was the Finnish word "selkeä" which means clear. Respondents did not talk about how easy or difficult the system was to learn and remember. Instead "clear" seems to be a high level concept the perception of which can be improved through those means that were placed under learnability in chapter 2. Furthermore, some respondents seem to refer to grouping or organization of interface elements with this concept:

> Respondent #19: "*Selkeys, toiminnot hyvin esillä ja loogisesti järjesteltynä.*"
>
> Translated: Clarity, the functionality well at hand and logically organized.

> Respondent #20: "*selkeyttä, eri osioiden selkeää ryhmittelyä...*"
>
> Translated: clarity, clear grouping of different parts...

The respondents also frequently made statements about ease of use and it seems that "easy" was used as a high level concept in a somewhat similar way as "clear." "Easy" was most frequently used by those respondents who wrote very short answers and therefore it is impossible to cite responses that would describe exactly what was perceived as being easy. However, a female respondent, who chose conv, gives some clues about what she thinks is easy:

> Respondent #5: "*Yksinkertainen, ei liikaa vaihtohetoja näkyvissä. Helppo käyttää.*"
>
> Translated: Simple, not too many options visible. Easy to use.

Both "clear" and "easy" were added to the framework. These concepts were much more useful than “learnable,” “memorable” and “effective” when coding the statements. However, learnability, memorability and effectiveness are all quite straight forward to operationalize and thus probably more useful concepts in some studies. When considering only perceived usability “clear” can be seen as a replacement for “learnable,” “memorable” can be left out because users do not pay attention to it, and “easy” can be conceived as a replacement for “effective” if effectiveness is operationalized as the percentage of users that are able to successfully complete a given task.

Within those statements that were placed in the category of comfortable two separate subcategories could clearly be distinguished. These were comfortable for the lazy:

> Respondent #34: “*sitä voi kättää kotoa helposti, ilman sen kummenpia kikkailuja*”
>
> Translated: it can easily be used from home, without any further gimmickry

> Respondent #37: “*Kyllähän tuo Oraclen kotikäyttöön tarkoitettu versio tuli haettua, mutta ei sitä vielä ole ehtinyt laittamaan kuntoon eikä taida enää olla tarvettakaan..*”
>
> Translated: Yes, I fetched the version of Oracle that was meant for home use, but I haven't had time to set it up yet and I won't likely have use for it anymore..

and comfortable for the suspicious:

> Respondent #56: “*tallennuksen voi tehdä ihan itsekin, eikä tarvi säheltää tunnareidenkaan kanssa.*”
>
> Translated: You can do the saving yourself, so you need not fumble about with usernames.

> Respondent #21: “*Koko systeemin (käyttöliittymä + tietokanta) tulisi toimia siten, että muiden käyttäjien lisäämiä tietoja ei toinen käyttäjä näe*”
>
> Translated: The whole system (user-interface + database) should work in such a way that one cannot see data added by other users

Respondent #49: "*Oli kuitenkin mahdollista tuhota toisen tekemä taulu, mistä olisi voinut tulla ongelmia jollekin..*"

Translated: It was still possible to destroy a table created by someone else, which could have caused problems to someone..

The issues mentioned by respondents number 21 and 49 seem to be obvious bugs. The system was originally meant only for select-queries, and the students could not modify the original tables or their contents. However, it was a mistake that the system allowed students to create tables in the common database and drop tables that had been created. Nonetheless, this mistake provided additional information, and two subcategories of comfortable, comfortable-lazy and comfortable-suspicious, were added to the framework.

Two different kinds of "simple" also emerged from the data. A category of simple interaction was created and placed under efficiency, and the original category of simple was labeled as simple screen design. Statements coded with simple interaction were often about navigation and moving between exercises. Respondents number 17 and 33 describe what simple interaction means:

Respondent #17: "*voi näppärästi vaihdella tehtävästä toiseen esim. katsomaan miten se alihaku taas tehtiin tms..*"

Translated: one can handily change between exercises e.g. to see how that subquery was done etc..

Respondent #33: "*Valitsemassani käyttöliittymässä ei tarvitse lärätä edestakaisin, vaan samalla voi katsella tulosta, hakua sekä korjata hakua, jos (kun..) se on mennyt pieleen.*"

Translated: One needs not flutter back and forth in the user-interface I chose, but it's possible to look at the result, query, and fix the query if (when..) it has gone wrong.

Those respondents who valued simple interaction chose the visi-interface while those who valued simple screen design chose conv. Respondent number 5, cited earlier, and number 64 below talk about simple screen design:

Respondent #64: "*Käyttöliittymässä ei ole mitään turhaa, ainoastaan demotehtävät.*"

Translated: There's nothing needless in the user-interface, only the exercises.

The concept system that resulted from the modifications listed in this section is presented in FIGURE 13. Categories are disconnected in this figure if no statements were coded into them and connected only with a dotted line if statements from less than three respondents fitted in. Furthermore, middle-level categories are emphasized if they contained statements from more than ten separate respondents when subcategories were included.

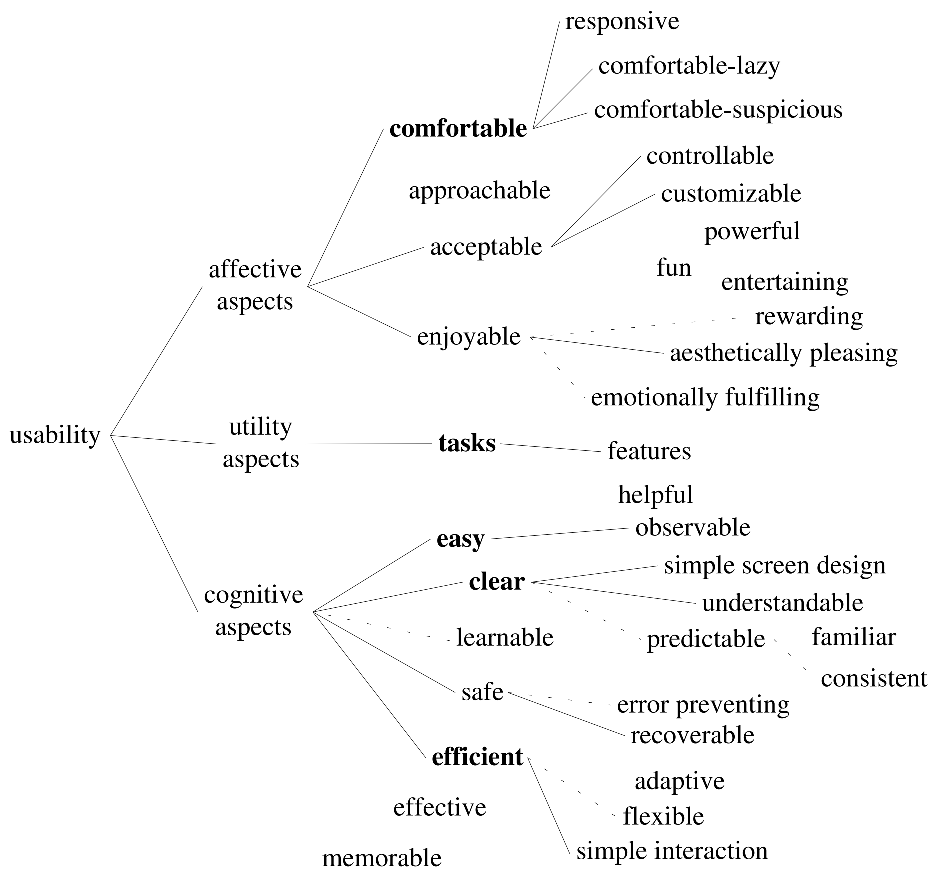


FIGURE 13. Perceived usability

In addition to aspects of usability, the analysis also looked for other constituents of satisfaction. However, only one such category was recognized. It was labeled stable and it contained statements about the existence or absence of bugs. Stable was placed outside the scope of usability because a developer need not know anything about the human information processing or the tasks to be able to write correct programs.

6.3 Deductive qualitative analysis

The concept system of perceived usability (FIGURE 13) was used as a framework and each statement in the data about the user-interface alternatives was coded into one of the categories of this framework. TABLE 6 shows for each category the number of respondents who made statements that were coded into the category in question.

TABLE 6. The constituents of user satisfaction

	frequency
features	25
clear	19
efficient	13
easy	12
simple interaction	12
observable	9
comfortable for the lazy	8
understandable	6
comfortable for the suspicious	5
controllable	5
aesthetically pleasing	4
responsive	4
stable (free from bugs)	4
customizable	3
recoverable	3
simple screen design	3

The results presented in TABLE 6 are summarized at the highest level of the concept system in TABLE 7. The same respondent appears only once per category in both tables. The numbers of TABLE 6 cannot be added up to get the numbers of TABLE 7 because the same respondent can appear in several lower level categories in the same branch.

TABLE 7. The constituents of user satisfaction summarized

	frequency
cognitive aspects	36
utility aspects	25
affective aspects	20

6.4 Evaluation

This section evaluates some aspects of the research based on the data. First, the small number of respondents who had tested all user-interface alternatives was a disappointment. Some measures could have been taken in the field work to avoid this problem. The system with the user-interface alternatives could have been designed in such a way that it would not let the students save any queries before they had tested all alternatives. In addition, a lottery could have been arranged to motivate the students to answer the questionnaire. Now the quantitative analysis was rather superficial because of the small number of respondents.

Secondly, the implementation of the desired differences in the user-interface alternatives as described in FIGURE 12 in the previous chapter seems to have succeeded quite well. Respondent number 18 describes his perceptions of the positive aspects of the visi-interface like this:

> Respondent #18: “*Näkee kerralla kaikki asiat ja pystyy helposti siirtymään tehtävien välillä. Tallennusmahdollisuus ainakin joissain tapauksissa, jotta pystyi ottamaan mallia aikaisemmista tehtävistä.*”
>
> Translated: One sees all things on the same time and can easily move between the exercises. The saving feature at least in some cases, so that one could model after earlier exercises.

The same respondent continues with his perception of conv and gene:

> Respondent #18: "*Keskustelevassa käyttöliittymässä oli liikaa "ohjausta" eli siinä ei itse voinut määrätä mitä halusi tehdä. Yleinen oli hyvä pikaisiin testauksiin, mutta tallentamisen puute voi haitata pitkäaikaisessa käytössä.*"
>
> Translated: There was too much "guidance" in the conversing user-interface or one couldn't have control over what to do in it. The general one was good for quick testing, but the lack of saving can trouble in long-term use.

The comments above show that the intended differences between the alternatives were salient. Respondent number 64, who chose conv, confirms the differences and shows that the "guidance" was not a problem for those who appreciate simple screens:

> Respondent 64: "*Kaikki toiminnot näyttävässä käyttöliittymässä ruudulla oli paljon tavaraa, joka teki siitä hieman sekaisen. Sen navigointipalkit olivat kuitenkin silloin tällöin hyödyllisiä, mutta useimmiten demotehtävien tekemisessä ei tarvinut siirtyä katsomaan aikaisempien demojen tehtäviä, joten valikko oli siinä mielessä myös turha.*"
>
> Translated: There was a lot of stuff on the screen in the user-interface that showed all functions, therefore it got somewhat cluttered. Its navigation bars were sometimes useful, but usually when doing demo-exercises one needed not go back to look at exercises from earlier demos, so the menu was in this regard also useless.

The visi-interface was not deliberately designed in such a way that it would cause the users to make errors; however, a minor usability problem further emphasized the differences between visi and conv, as conv provided error free usage. Respondent number 18 chose to use the visi-interface despite this problem:

> Respondent #18: "*Itse en aina muistanut siirtyä seuraavaan tehtävään vaan vahingossa tein uuden tehtävän vanhan päälle. Jotenkin sitä pitäisi selkeyttää tai muuttaa, ettei niin helposti overwritetä aikaisempaa.*"
>
> Translated: I didn't always remember to move to the next exercise but I overwrote the old exercise with the new one by accident. It should somehow be clarified or changed, so that one wouldn't so easily overwrite an earlier exercise."

In the third place, some aspects of the framework developed in the inductive qualitative analysis can also be evaluated based on the data. Patton (2003, 455-466) provides four criteria for this kind of evaluation. He explains that the set of categories should

- have internal and external plausibility,
- be inclusive of the data,
- be reproducible by another researcher, and
- be credible to the respondents.

The first two criteria can be discussed with respect to the data. Internal and external plausibility means that the set of categories should be internally consistent and comprise a whole picture when viewed externally. In light of the numbers of TABLE 7, the framework seems to be quite balanced, and the evaluative statements were in most cases easily grouped and clearly distinguished form other categories. Furthermore, the absence of evaluative statements that did not fit into any category also shows that the framework is reasonably inclusive.

Finally, it seems that the field experiment did not interfere much with the progress of the course. Instead, the system developed for the course was perceived as a great improvement:

> Respondent #17: "*sql-kyselyiden tekeminen kurssin demoina tuli huomattavasti mielekkäämmäksi webbisliittymän avulla! loistavaa! tämä on oikea tappaja-sovellus!!!*"
>
> Translated: doing sql-queries as course exercises became notably more meaningful with the web-interface! great! this is a real killer-app!!!

> Respondent #39: "*Tämä verkossa käytettävä tietokanta SQL-kyselyjen tekemiseen oli TODELLA LOISTAVA juttu!!!! Ainakaan itse en olisi vaivautunut Agoran mikroluokkiin testaamaan kyselyja. Demoja oli mielenkiintoista tehdä, koska sai heti testattua, toimiiko itse tehty kysely vai ei. Koin oppivani ja ymmärtäväni asioita paremmin, kun pääsin testaamaan niitä käytännössä.*"
>
> Translated: The web-based database for doing SQL-queries was REALLY GREAT!!!! At least I wouldn't have bothered to go to the PC-classes of Agora to test queries. Doing exercises was interesting, because one could immediately test whether one's own query worked or not. I experienced better learning and understanding, when I could test them

in practice.

Respondent #60: "*Hieno parannus ainakin ITK135-kurssia ajatellen, SQL-harjoittelun kynnys madaltui huomattavasti järjestelmän avulla (vrt. Oracle mikroluokissa).*"

Translated: A great improvement at least with regard to the ITK135-course. The threshold for SQL-training was notably lowered with this system (cf. Oracle in the PC-classes).

The results from the standard SUS-scale (Brooke 1986) in TABLE 8 confirm that the usability of all alternatives was at a high level. The SUS-scale yields results between 0 and 100, with 0 for extremely poor perceived usability and 100 for extremely high perceived usability. The study of Nielsen and Levy (1994) shows that a system with average usability gets a score around 64 on this kind of rating scale even though 50 represents neutral. They explain that this might be because users tend to be polite and blame themselves if they have trouble with the system. Nonetheless, all alternatives in this study got a score notably above the expected average and this might be one reason why the testing of the alternatives and the field experiment were not experienced troublesome.

TABLE 8. SUS-scores for the user-interface alternatives

	SUS-score
visi	87.4
conv	87.2
gene	81.8

7 CONCLUSION

This chapter summarizes the main points of this thesis, highlights the theoretical contribution, and draws some conclusions based on the results. The objective of this study was to achieve a clear understanding of the concept of usability and to rank the different aspects of usability according to how important they are as constituents of user satisfaction. The study was carried out in two stages. The first stage was conceptual-analytical and it focused on the concept of usability, while the second was empirical and focused on perceived usability and the constituents of user satisfaction.

The conceptual-analytical stage collected usability attributes from many different sources and arranged them into a hierarchical concept system that shows their assumed mutual relationships. The first level divides the hierarchy into three branches: affective aspects, utility aspects, and cognitive aspects. These were chosen to reflect the two different information processing systems of humans – affect and cognition – according to Norman (2002a; 2002b), and because the utility of the system is not directly related to human beings but rather to their work tasks. The second level of the hierarchy contains usage indicators that can be measured with empirical usability tests, and the third level contains system properties that influence the usage indicators. This hierarchical concept system synthesizes knowledge from many different sources and combines different definitions of usability to give a better understanding of the concept of usability. The proposed concept system can be used as a conceptual framework for different kinds of usability studies and as such it comprises the theoretical contribution of this stage.

The empirical stage combined a quantitative and qualitative approach and it was designed as a field experiment. The field experiment was conducted with a group of information technology students, as the study assumed that they would be representative of skillful and experienced computer users when considering perceived usability and user satisfaction. The conclusions based on the field experiment are restricted to skillful and experienced computer users and they must be taken with some caution because only a group of students was studied.

The field experiment was designed especially to address two specific sub-questions of the main research problem:

- Do users prefer simple generic tools or feature-rich task-specific systems?
- Do users prefer an interface consisting of simple screens or one that keeps all the information and functionality visible all the time?

Despite some weaknesses of the quantitative study these two questions can be answered with some confidence. Most users rather use a task-specific system that integrates more relevant features than a combination of simple generic tools. When considering the second question it is necessary to note that keeping all functions visible also provides simple interaction as everything can be reached with a single selection. Both the quantitative and qualitative results suggest that observability and simple interaction is preferred over simple screen design and error-free usage. Furthermore, it seems that users do not pay attention to errors when they evaluate a user-interface. This last finding might be explained by false self-attribution of errors. Users tend to blame themselves for errors that could have been avoided by different actions and they do not notice that a different system design would have hindered them from carrying out the erroneous action (Norman 1988, 34-35, 40-41).

The empirical study also gave further insight into the concept of usability. The concept system developed in the first stage of the study was used as a framework for the analysis and it was modified based on the data to better reflect perceived usability. The results presented in FIGURE 13 are proposed as a description of how users perceive usability. In addition to the integration of all desired features users want systems that are easy, clear, efficient and comfortable.

The study also made an attempt to understand the relative importance of the different aspects of usability as constituents of user satisfaction. TABLE 6 presents a ranking of the constituents of user satisfaction when users compare the user-interface alternatives that were developed for this study. Here it is important to note that the aspects which were contrasted in the alternatives were of course more salient to the users than other aspects. The inability of this study to equally address all aspects of usability is a shortcoming. Only the contrasted aspects can be ranked according to importance based on this study.

Their order starting with the most important is features, simple interaction, observable, controllable, simple screen design, and error preventing. When comparing to earlier research (Nielsen 1989) one can conclude that efficiency seems to be very important to users while learnability is less important. However, the results are also contradictory on some aspects.

Much more work is required to understand the relative importance of the other aspects of usability. Considering the other aspects, the results presented in TABLE 6 can only be regarded as indicative and they can be used as a starting point for further research. This research can be continued with similar studies that use different types of applications contrasting different aspects to cover the aspects of usability more extensively and to find possible differences between the genders, other groups of users and various application domains.

REFERENCES

Abelson R. P., Prentice D. A., 1989. Beliefs as possessions: A functional perspective. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 361-382.

Ajzen I., 1989. Attitude structure and behavior. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 241-274.

Aronson, E., Ellsworth P. C., Carlsmith J. M., Gonzalez M H., 1990. Methods of research in social psychology. New York: McGraw-Hill.

Barnard P., May J., 1999. Representing cognitive activity in complex tasks. Human-computer interaction 14(1/2), 93-158.

Barnum C., Bevan N., Cockton G., Nielsen J., Spool J., Wixon D., 2003. The “magic number 5”: Is it enough for web testing? In: Cockton G., Korhonen P. (Eds.), 2003. Proceedings of CHI '03 conference on human factors in computer systems: New horizons, Fort Lauderdale, Florida, USA, April 5-10. New York: ACM Press, 698-699.

Bevan N., 1995. Measuring usability as quality of use. Software Quality Journal 4, 115-150. Available at <http://www.usability.serco.com/papers/qusab95.pdf> [referred 16.9.2003].

Breckler S. J., Wiggins E. C., 1989. On defining attitude and attitude theory: once more with feeling. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 407-427.

Brooke J., 1986. SUS – A “quick and dirty” usability scale. Available at <http://www.cee.hw.ac.uk/~ph/sus.html> [referred 2.5.2004].

Cacioppo J. T., Petty R. E., Geen T. R., 1989. Attitude structure and function: From the tripartite to the homeostasis model of attitudes. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 275-310.

Chin W. W., Lee M. K. O., 2000. A proposed model and measurement instrument for the formation of IS satisfaction: the case of end-user computing satisfaction. In: Orlikowski W. J., Weill P., Ang S., Krcmar H. C., 2000. Proceedings of the twenty first international conference on information systems, Brisbane, Queensland, Australia, December 10-13. Atlanta, GA, USA: Association for Information Systems, 553-563.

Coolican H., 1994. Research methods and statistics in psychology. Second edition. London: Hodder & Stoughton Educational.

Damasio A. R., 1994. Descartes' error: emotion, reason and the human brain. New York: Grosset/Putnam Book.

Davis F. D., 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly 13(3), 318-340.

Davis F. D., 1993. User acceptance of information technology: system characteristics, user perceptions and behavioral impacts. International Journal of Man-Machine Studies 38(3), 475-487.

Dix A., Finlay J., Abowd G., Beale R., 1998. Human-computer interaction. Harlow, England: Prentice Hall.

Doll W. J., Torkzadeh G., 1988. The measurement of end-user computing satisfaction. MIS Quarterly 12(2), 258-274.

Donyaee M. K., 2001. Towards an integrated model for specifying and measuring quality in use. Master's thesis in computer science at Concordia University.

Eskola J., Suoranta J., 1998. Johdatus laadulliseen tutkimukseen, 3. painos. Jyväskylä: Gummerus Kirjapaino.

Etzioni A., 1988. Normative-affective factors: toward a new decision-making model. Journal of Economic-Psychology 9(2), 125-150. Reprinted in: Zey M., 1992. Decision making: alternatives to rational choice models. London: Sage Publications.

Eysenck M., Keane M. T., 2000. Cognitive psychology: a student's handbook, 4th edition. New York: Psychology Press.

Fishbein M., Ajzen I., 1975. Belief, attitude, intention and behavior: an introduction to theory and research. Reading, MA: Addison-Wesley.

Frøkjær E., Hertzum M., Hornbæk K., 2000. Measuring usability: are effectiveness, efficiency, and satisfaction really correlated? In: Turner T., Szwillus G., Czerwinski M., Paternò F. (Eds.), 2000. Proceedings of CHI '00 conference on human factors in computer systems, The

Hague, The Netherlands, April 1-6. New York: ACM Press, 345-352.

Gibson J. J., 1986. The ecological approach to visual perception. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Goleman D., 1995. Emotional intelligence. New York: Bantam Books.

Gould J. D., Lewis C., 1985. Designing for usability: key principles and what designers think. Communications of the ACM 28(3), 300-311.

Harrison A. W., Rainer R. K., 1996. A general measure of user computing satisfaction. Computers in Human Behavior 12(1), 79-92.

Hassenzahl M., Platz A., Burmester M., Lehner K., 2000. Hedonic and ergonomic quality aspects determine a software's appeal. In: Turner T., Szwillus G., Czerwinski M., Paternò F. (Eds.), 2000. Proceedings of CHI '00 conference on human factors in computer systems, The Hague, The Netherlands, April 1-6. New York: ACM Press, 201-208.

Hewett T., Baecker R., Card S., Carey T., Gasen J., Mantei M., Perlman G., Strong G., Verplank W., 1992. CHAPTER 2: Human-computer interaction in ACM SIGCHI curricula for human-computer interaction. Available at <http:// sigchi.org/cdg/cdg2.html> [referred 16.9.2003].

Hufnagel E. M., Conca C., 1994. User response data: the potential for errors and biases. Information Systems Research 5(1), 48-73.

Huskinson T. L. H., Haddock G., 2004. Individual differences in attitude structure: variance in the chronic reliance on affective and cognitive information. Journal of Experimental Social Psychology 40(1), 82-90.

Jacobson I., Booch G., Rumbaugh J., 1999. The unified software development process. Reading, Massachusetts: Addison-Wesley.

Johnson J. A., Nardi B. A., 1996. Creating presentation slides: a study of user preferences for task-specific versus generic application software. ACM Transactions on Computer-Human Interaction 3(1), 38-65.

Keinonen T., 1998. One-dimensional usability: influence of usability on consumers' product preference. Ph.D. thesis at the University of Art and Design Helsinki.

Kotler P., 1997. Marketing management. Analysis, planning, implementation, and control, 9th edition. New Jersey: Prentice Hall International.

Legris P., Ingham J., Collerette P., 2003. Why do people use information technology? A critical review of the technology acceptance model.

Information & Management 40, 191-204.

Lindgaard G., Dudek C., 2003. What is this evasive beast we call user satisfaction? Interacting with Computers 15, 429-452.

Mayhew D. J., 1999. The usability engineering lifecycle: a practitioner's handbook for user interface design. San Francisco: Morgan Kaufmann Publishers.

McGuire W. J., 1989. The structure of individual attitudes and attitude systems. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 37-70.

McHaney R., Hightower R., Pearson J., 2002. A validation of the end-user computing satisfaction instrument in Taiwan. Information & Management 39, 503-511.

Melone N. P., 1990. A theoretical assessment of the user-satisfaction construct in information systems research. Management Science 36(1), 76-91.

Nielsen J., 1989. What do users really want? International Journal of Human-Computer Interaction 1(1), 137-147.

Nielsen J., 1993. Usability engineering. Boston: Academic Press.

Nielsen J., 1994a. Enhancing the explanatory power of usability heuristics. In: Adelson B., Dumais S., Olson J. (Eds.), 1994. Proceedings of CHI '94 conference on human factors in computer systems: celebrating interdependence, Boston, Massachusetts, United States, April 24-28. New York: ACM Press, 152-158.

Nielsen J., 1994b. Heuristic evaluation. In: Nielsen J., Mack, R.L. (Eds.), 1994. Usability inspection methods. New York: John Wiley & Sons, 25-62.

Nielsen J., 2000. Why you only need to test with 5 users. Available at: <http://www.useit.com/alertbox/20000319.html> [referred 12.6.2004].

Nielsen J., Landauer T. K., 1993. A mathematical of the finding of usability problems. In: Ashlund S., Mullet K., Henderson A., Hollnagel E., White T. (Eds.), 1993. Proceedings of CHI '93 conference on human factors in computer systems, Amsterdam, The Netherlands, April 24-29. New York: ACM Press, 206-213.

Nielsen J., Levy J., 1994. Measuring usability: preference vs. performance. Communications of the ACM 37(4), 66-75.

Norman D. A., 1988. The psychology of everyday things. New York: Basic Books.

Norman D. A., 1993. Things that make us smart: defending human attributes in the age of the machine. New York: Addison-Wesley Publishing Company.

Norman D. A., 1998. The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the answer. Cambridge, MA: The MIT Press.

Norman D. A., 2002a. Emotion and affect. Ubiquity 3(13). Available at <http://www.acm.org/ubiquity/interviews/d_norman_2.html> [referred 16.9.2003].

Norman D. A., 2002b. Emotion and design: attractive things work better. Interactions 9(4), 36-42.

Norman D. A., 2004. Emotional design: why we love (or hate) everyday things. New York: Basic Books.

Nosek B. A., Banaji M. R., Greenwald A. G., 2002. e-Research: ethics, security, design, and control in psychological research on the Internet. Journal of Social Issues 58(1), 161-176.

Patton M. Q., 2003. Qualitative research and evaluation methods, 3rd edition. Thousand Oaks, CA: Sage Publications.

Pratkanis A. R., 1989. The cognitive representation of attitudes. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 71-98.

Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Preece J., 1994. Human-computer interaction. Wokingham: Addison-Wesley.

Preece J., Rogers Y., Sharp H., 2002. Interaction design: beyond human-computer interaction. New York: John Wiley & Sons.

Shavitt S., 1989. Operationalizing functional theories of attitude. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 311-338.

Shneiderman B., 1998. Designing the user interface: strategies for effective human-computer interaction, 3rd edition. Reading, Massachusetts: Addison-Wesley.

Shneiderman B., 2000. Creating creativity: user interfaces for supporting innovation. ACM Transactions on Computer-Human Interaction 7 (1), 114-138.

Shneiderman B., 2002. Leonardo's laptop: human needs and the new computing technologies. Cambridge, MA: The MIT Press.

Simon S. J., 2001. The impact of culture and gender on web sites: an empirical study. ACM SIGMIS Database 32(1), 18-37.

Snyder M., DeBono K. G., 1989. Understanding the functions of attitudes: lessons from personality and social behavior. In: Pratkanis A. R., Breckler S. J., Greenwald A. G. (Eds.), 1989. Attitude structure and function. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 339-360.

Strauss A., Corbin J., 1998. Basics of qualitative research: techniques and procedures for developing grounded theory, 2nd edition. Thousand Oaks, CA: Sage Publications.

Tomkins S. S., 1981. The quest for primary motives: biography and autobiography of an idea. Journal of Personality and Social Psychology 41(2), 306-329. Reprinted in: Demos E. V. (Ed.), 1995. Exploring affect: the selected writings of Silvan S. Tomkins. Cambridge: Cambridge University Press, 27-63.

Torenvliet G., 2003. We can't afford it! The devaluation of a usability term. Interactions 10(4), 12-17.

Tractinsky N., 1997. Aesthetics and apparent usability: empirically assessing cultural and methodological issues. In: Pemberton S. (Ed.), 1997. Proceedings of CHI '97 conference on human factors in computer systems, Atlanta, GA, USA, March 22-27. New York: ACM Press, 115-122.

Vredenburg K., Mao J.-Y., Smith P. W., Carey T., 2002. A survey of user-centered design practice. In: Terveen L., Wixon D., Comstock E., Sasse A. (Eds.), 2002. Proceedings of CHI '02 conference on human factors in computer systems: changing the world, changing ourselves, Minneapolis, Minnesota, USA, April 20-25. New York: ACM Press, 471-478.

van Welie M., van der Veer G. C., Eliëns A., 1999. Breaking down usability. In: Sasse A., Johnson C. (Eds.), 1999. Proceedings of Interact '99, Edinburgh, August 8 - September 3. Edinburgh: IOS Press, 613-620. Available at <http://www.cs.vu.nl/~martijn/gta/docs/Interact99.pdf> [referred 16.9.2003].

von Wright G. H., Explanation and understanding. London: Routledge & Kegan Paul.

APPENDIX 1

USER-INTERFACE ALTERNATIVES

This appendix presents typical usage scenarios with the different user-interface alternatives: gene, conv and visi. The usage scenarios are explained and illustrated with screen-shots.

The gene-interface

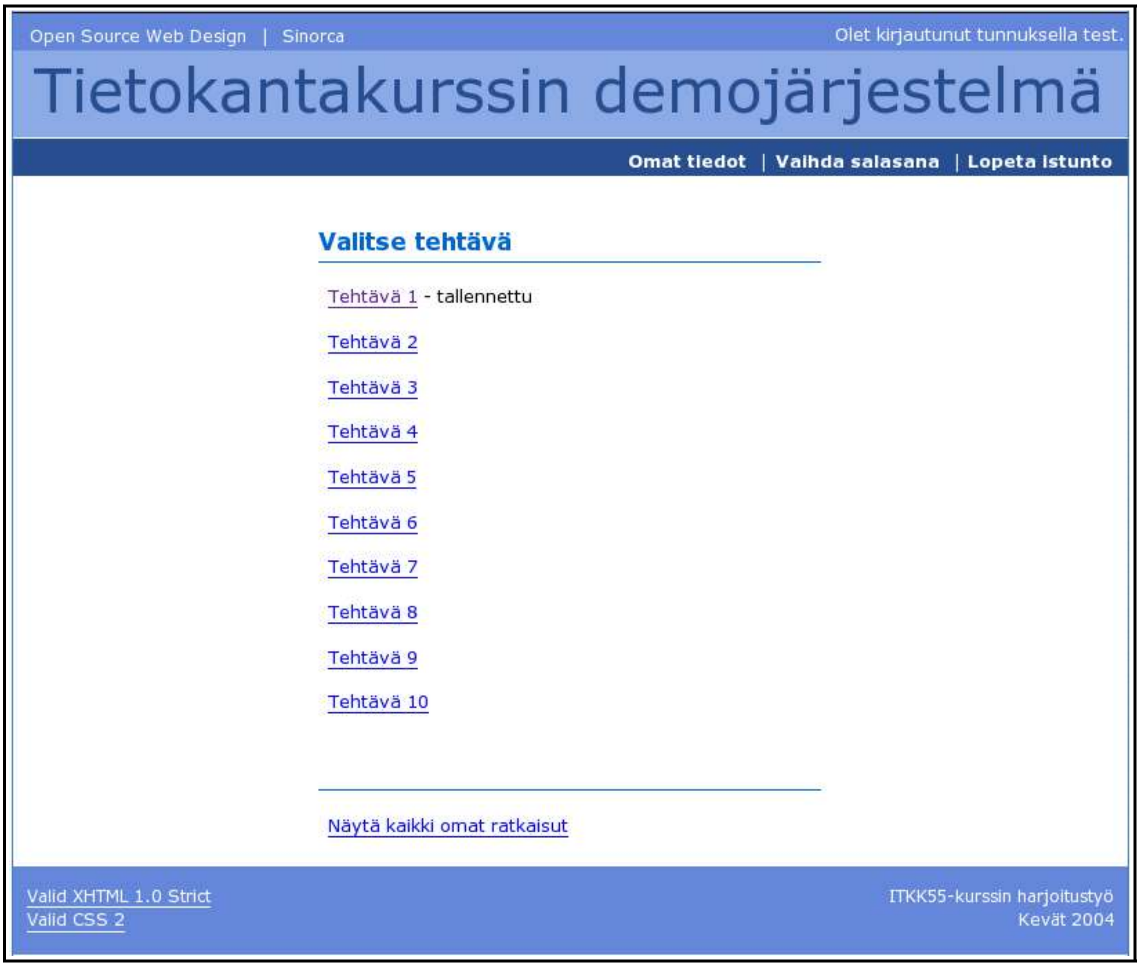
The user needs only to know the URL of the gene-interface and then he or she can start using it right away with a web-browser. The relational schema and a text-area in which the user can type in SQL-queries are presented as shown in the following figure.

The user types in the query and presses the "Kokeile kyselyä" -button. The system responds by presenting the results of the query (as shown in the following figure) or an error message if there is a syntax error in the query. Now the user can choose to modify the query or to continue with a new query. The system returns to the screen presented in the previous figure either with the old query in the text-area or a blank text-area depending on the user's choice.

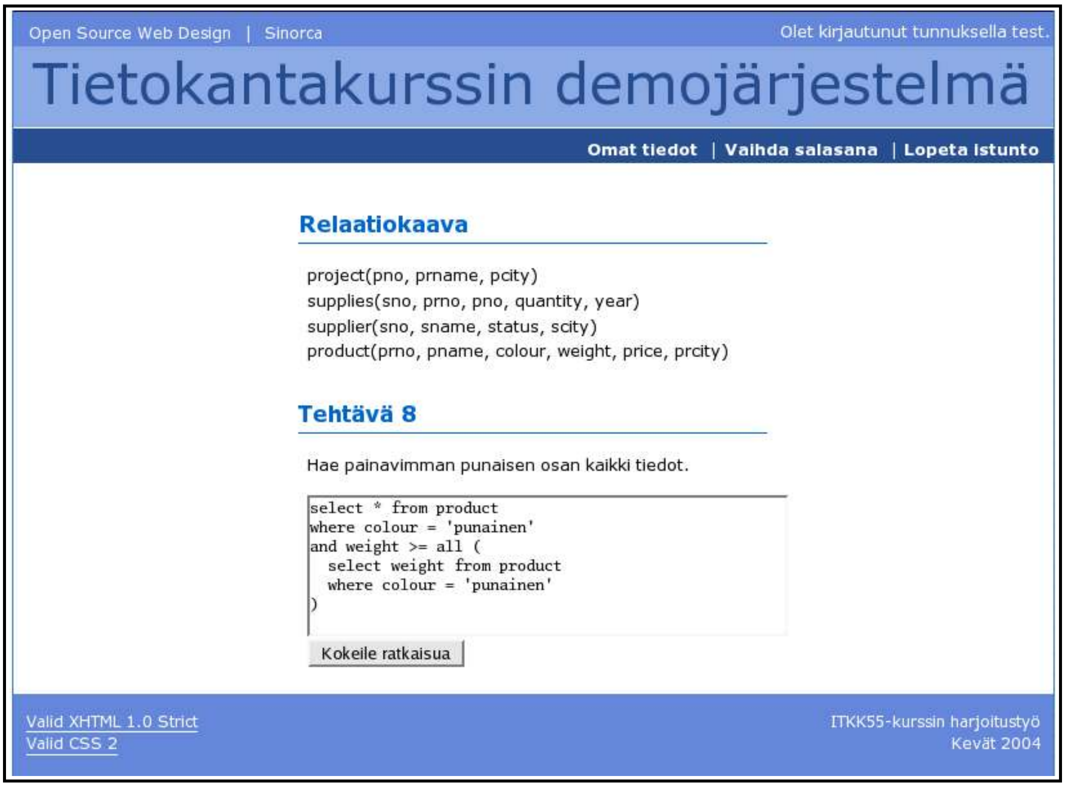
The conv-interface

Both the conv- and the visi-interface require the user to provide a username and a password before use. After this the conv-interface presents a screen in which the user chooses the demo as presented in the following figure.

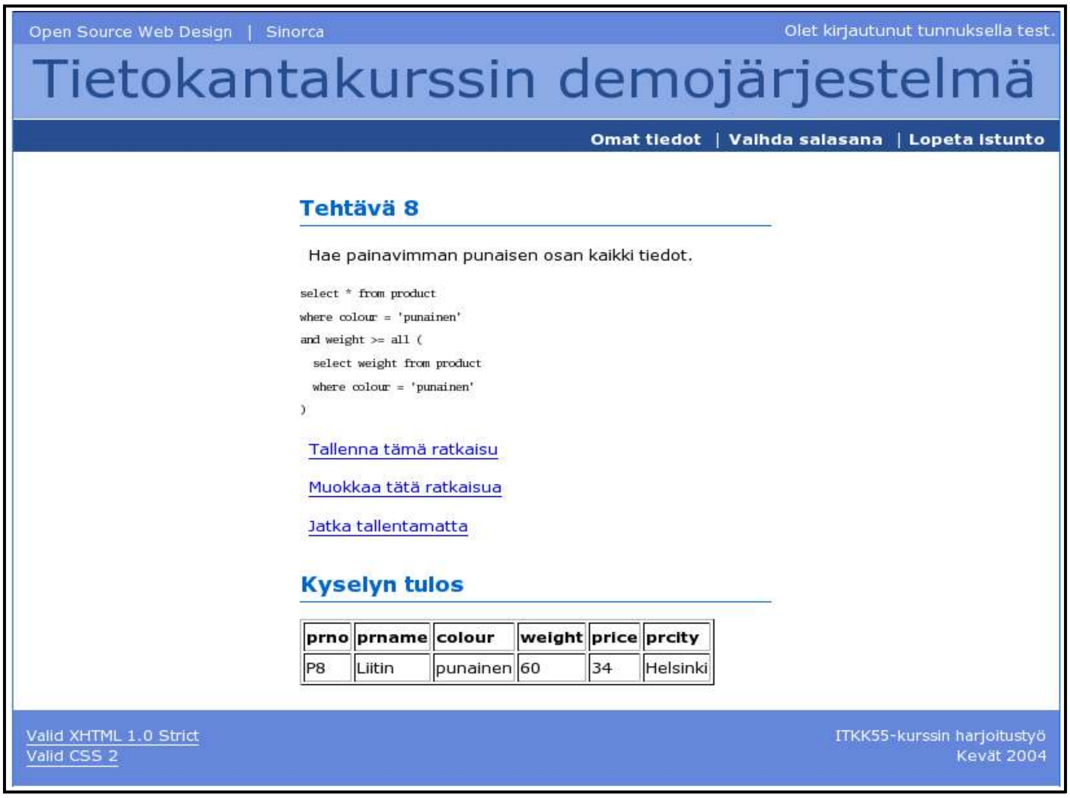
When the user has chosen a demo, the system presents a list of exercises that belong to the chosen demo as shown in the following figure. Now the user continues the conversation by choosing an exercise.



When the user has chosen an exercise, the system presents the relational schema and a text-area for the SQL-query as shown in the following figure.

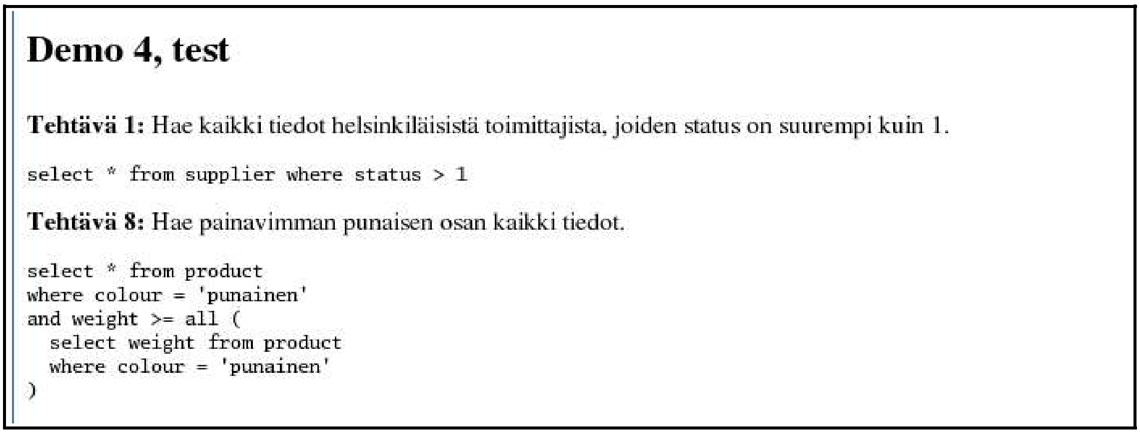


The user types in a query and continues with the "Kokeile kyselyä" -button. Now the system responds with the query-results (or possibly an error message) and asks the user whether she wants to save the query or modify it as shown in the next figure.



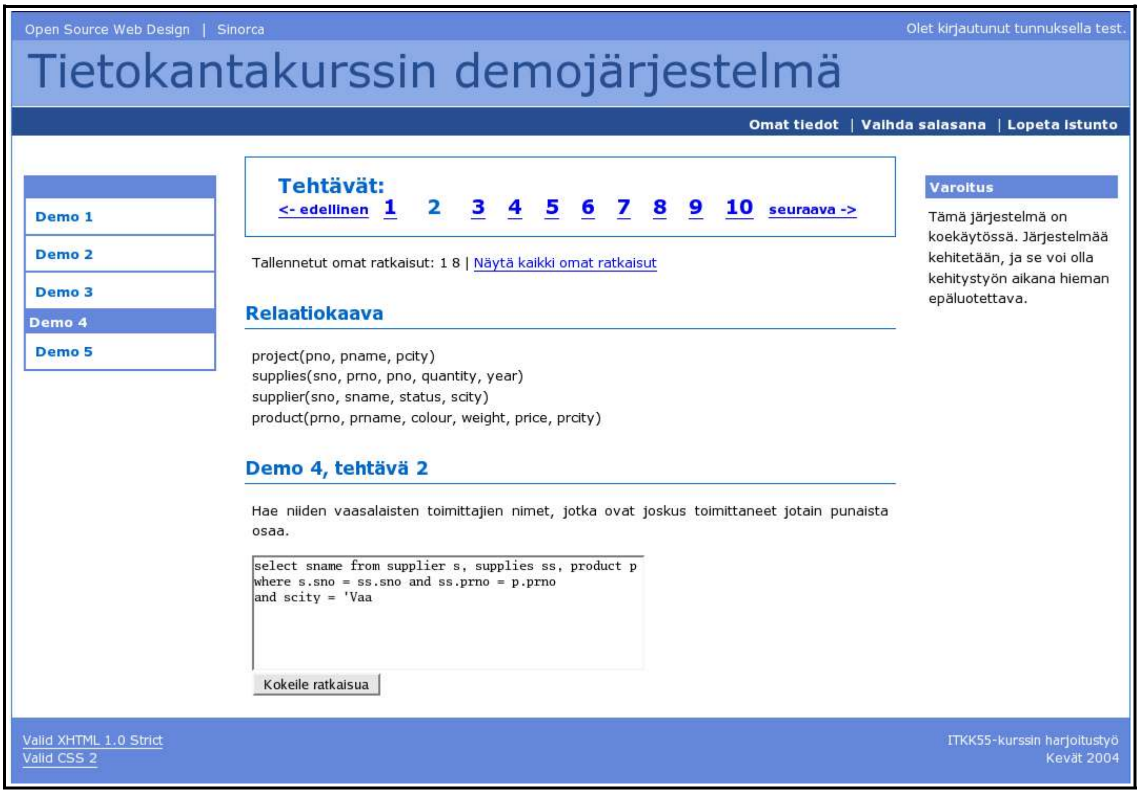
In this scenario, the user chooses to save the query because the result-row seems to make sense. After saving the query, the system starts the conversation over again from the exercises, but this time it adds at the beginning of the page a note that the query has been saved as shown in the following figure. This page also puts a label after the exercises that has already been saved. If there are any saved exercises, the system adds a link after the list of exercises to let the user view all saved queries. Most users probably want to print the solutions after finishing the exercises.

Now the user does not have time to do more exercises so she chooses to view the saved queries and the system responds with a printable page that is shown in the next figure.



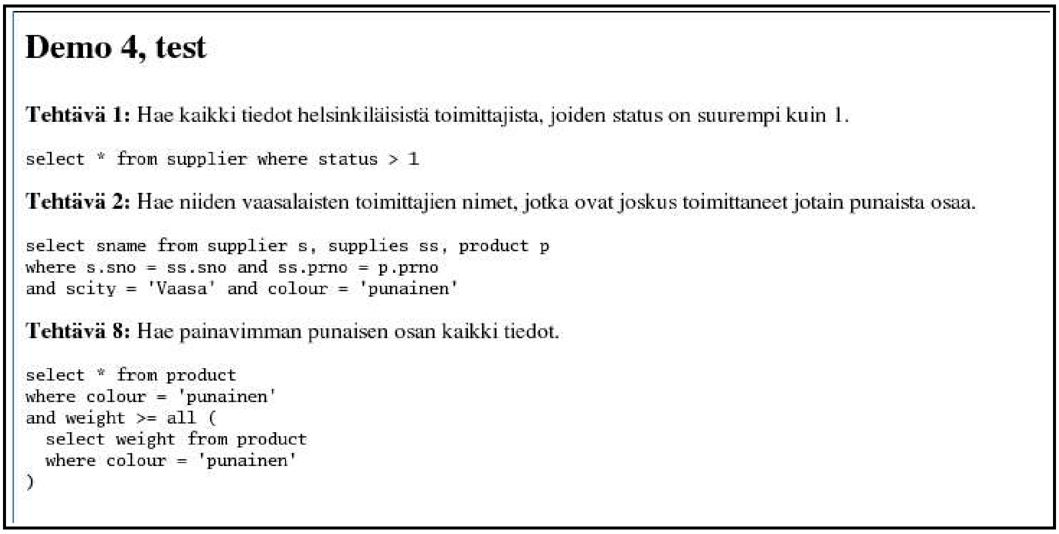
The visi-interface

The visi-interface keeps all the information and functionality visible all the time and everything is presented in the same basic screen that simply includes extra messages when needed. A basic state of this screen is shown in the following figure. This is the state that the previous user arrives at when she signs in the next time after the scenario that was presented with the conv-interface.



The demo can be changed from the navigation on the left and the exercise is chosen from the list at the top of the page. Now the user types in a query and presses the button to test it. The system stays in the same screen but it includes now a message that tells that the query has been saved and the results of the query are included at the bottom of the page as shown in the following figure.

The user now chooses to view all the saved queries.



Demo 4, test

Tehtävä 1: Hae kaikki tiedot helsinkiläisistä toimittajista, joiden status on suurempi kuin 1.

```
select * from supplier where status > 1
```

Tehtävä 2: Hae niiden vaasalaisten toimittajien nimet, jotka ovat joskus toimittaneet jotain punaista osaa.

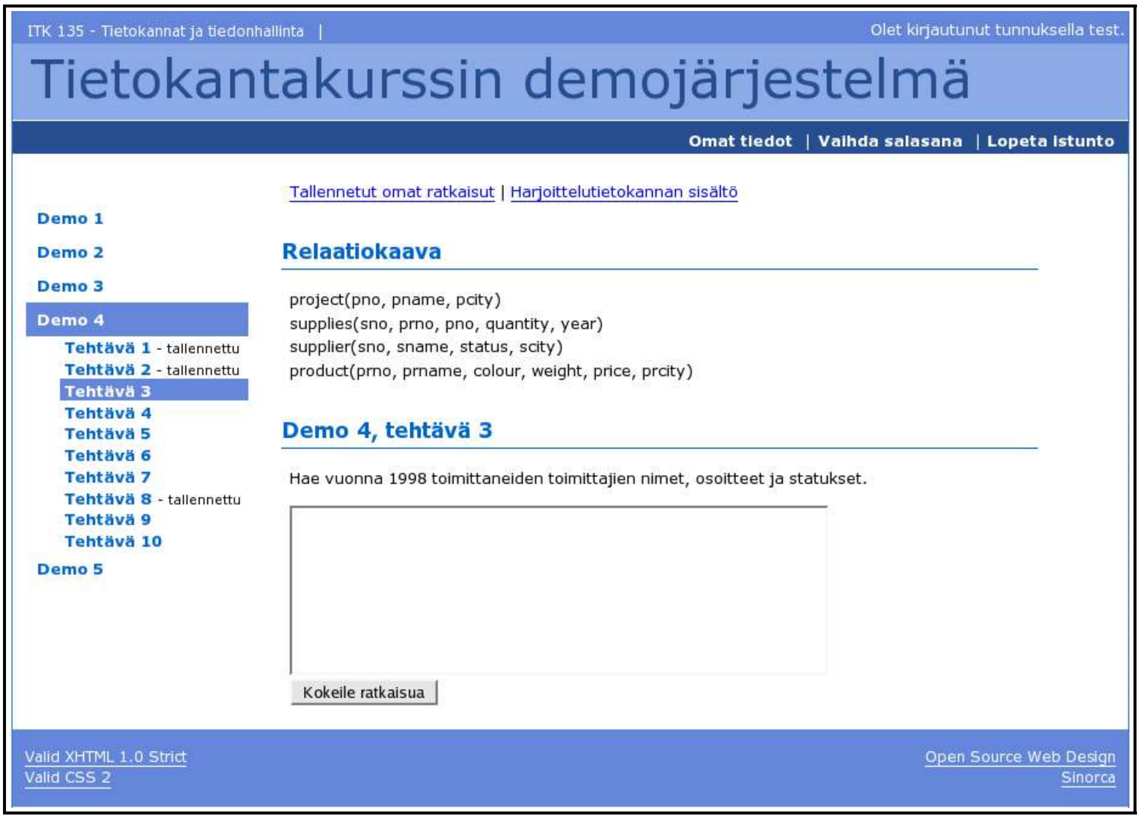
```
select sname from supplier s, supplies ss, product p
where s.sno = ss.sno and ss.prno = p.prno
and scity = 'Vaasa' and colour = 'punainen'
```

Tehtävä 8: Hae painavimman punaisen osan kaikki tiedot.

```
select * from product
where colour = 'punainen'
and weight >= all (
  select weight from product
  where colour = 'punainen'
)
```

The final version of the user-interface

The feedback from the students was also used to create a final version of the user-interface. It is shown here as the last figure of this appendix.



APPENDIX 2

USABILITY TESTS OF THE USER-INTERFACE ALTERNATIVES

The user-interface alternatives and the e-mailed instructions used in the field experiment were tested with users before the experiment to eliminate major offending usability problems. These usability tests are reported in this appendix.

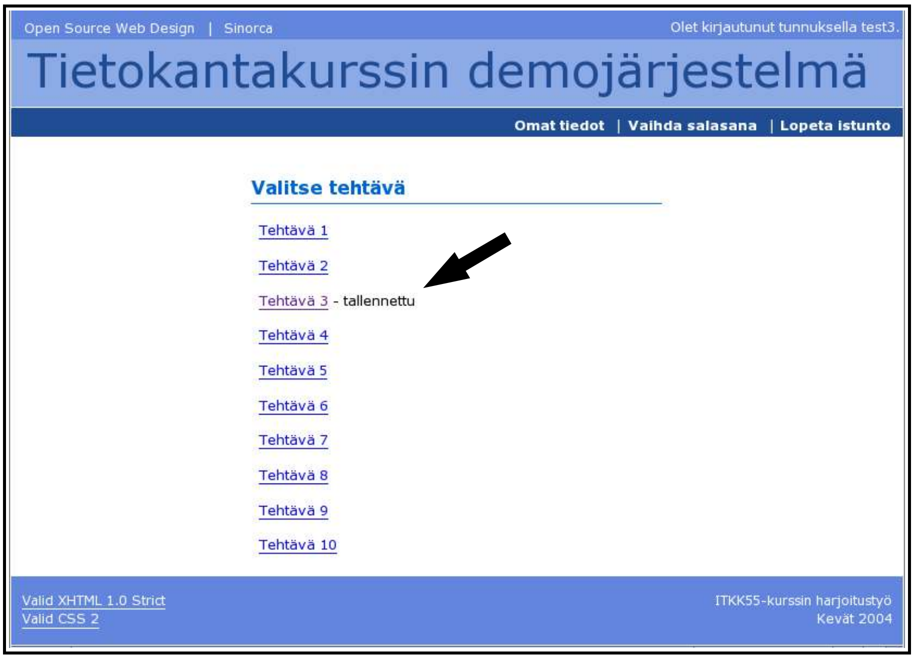
Method

The test was carried out with three participants one at a time. The participant was greeted and the procedure of the test was explained. The participant was asked to put him/herself in the place of a student who had received an e-mail message asking to participate in the experiment. A printed version of the message was handed to the participant and he/she was asked to read it and then start working with a computer. The participant was encouraged to think aloud and tell what part of the instructions he was completing and what he was doing with the user-interface. The participants were interviewed during the test if they did not talk spontaneously about what they were doing and what they thought about the user-interface. The computer screen and the conversation was recorded on video-tape during the test.

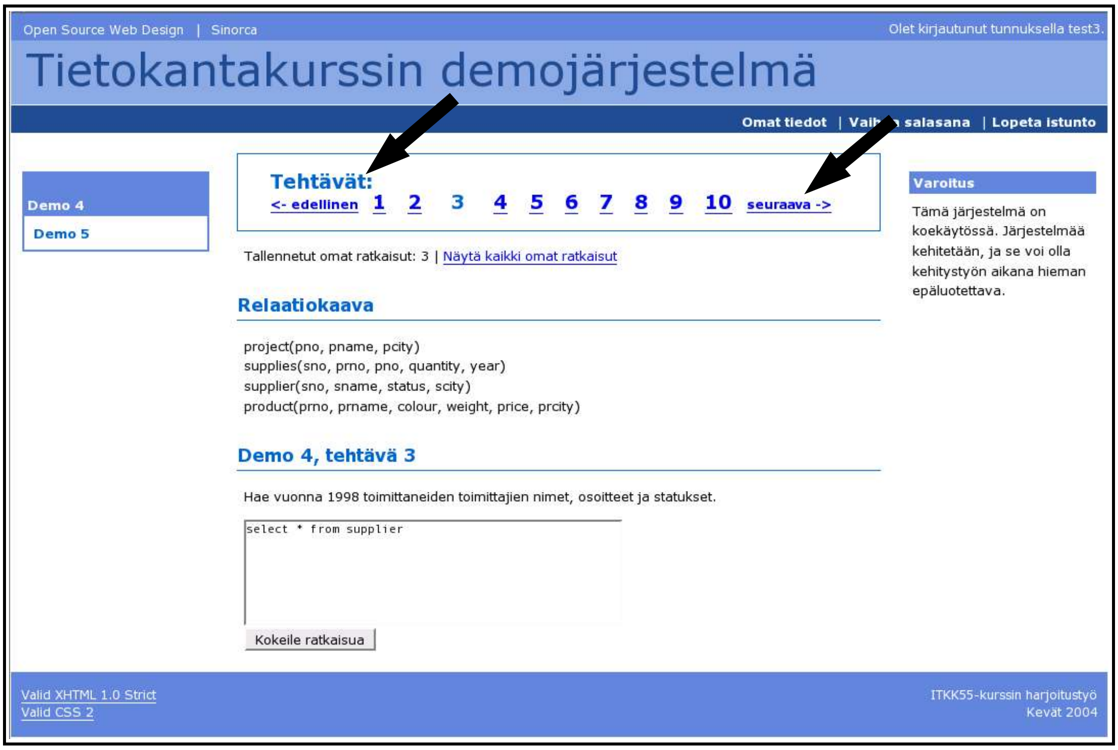
Results

The participants did not have problems with the instructions.

No errors occurred in the use when testing the conv-interface. However, one participant complained that he could not remember which exercises he had completed, and he suggested that the completed exercises could be marked in the list of exercises. Because the visi-interface also showed which exercises had been completed a label for saved exercises was added to the conv-interface to make it equal in functionality when compared to the visi-interface. The fix is marked with an arrow in the screen-shot below.



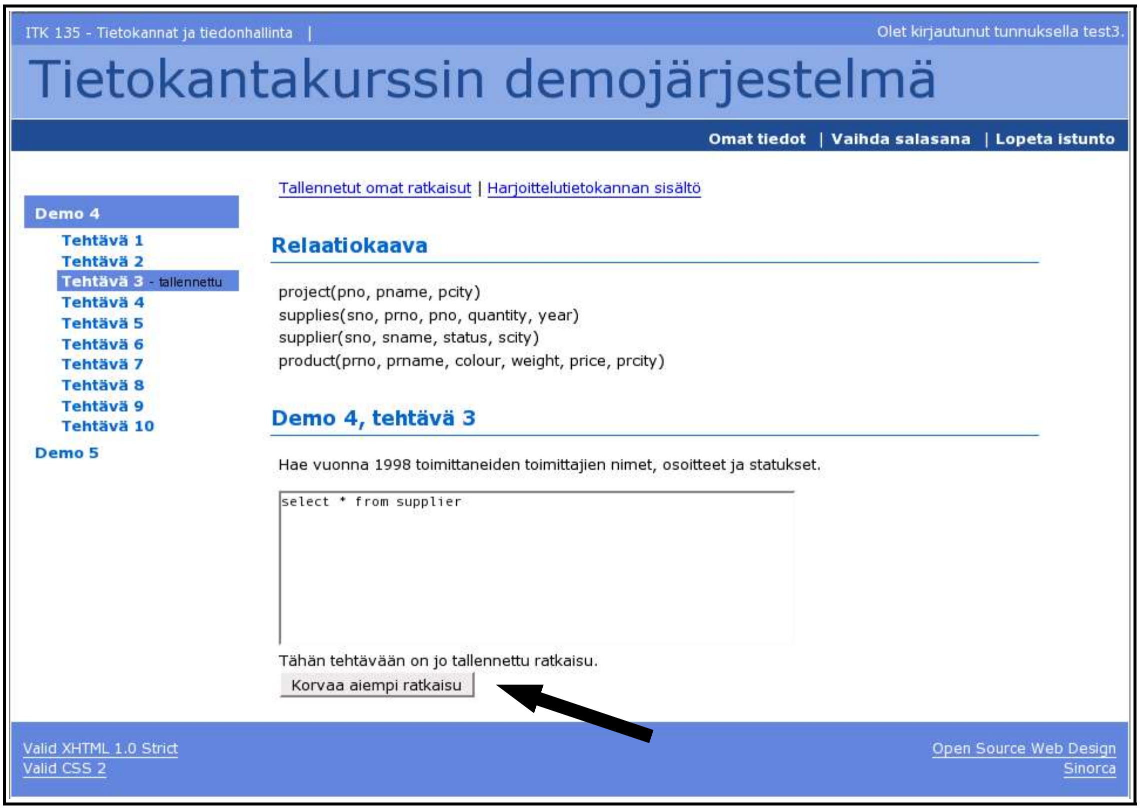
The visi-interface was notably more difficult to use. The users were not aware of which exercise they were about to save, and it was not clear that the numbers at the top represented exercises so that the user could select which exercise to work on. The labels "exercises," "previous" and "next" were added to the numbers to show that they were meant for navigation between exercises. These fixes are marked with arrows in the screen-shot below.



The comments from respondent number 18 show that these fixes were not enough to make the users aware of which exercise was active:

> Respondent #18: "*Itse en aina muistanut siirtyä seuraavaan tehtävään vaan vahingossa tein uuden tehtävän vanhan päälle. Jotenkin sitä pitäisi selkeyttää tai muuttaa, ettei niin helposti overwritetä aikaisempaa.*"
>
> Translated: I didn't always remember to move to the next exercise but I overwrote the old exercise with the new one by accident. It should somehow be clarified or changed, so that one wouldn't so easily overwrite an earlier exercise."

The user-interface development was continued after the experiment and this problem was fixed by adding a warning between the text-area and the button and changing the label of the button to "Overwrite previous query" if there was already a query saved for the active exercises. These fixes are shown with an arrow in the screen-shot below.



The gene-interface was very simple, and no usability problems were found in it during the tests.

APPENDIX 3

USABILITY TESTS OF THE QUESTIONNAIRE

The web-based questionnaire was tested with two persons who were undergoing non-military service at the university. These participants were chosen simply because they were available within the restrictions of a tight schedule. The participants were asked to imagine that they were students of the ITK135-course, who had been asked to answer the questionnaire. Both participants were asked to complete the questionnaire using different roles. The roles were:

- a student who had not done any exercises,
- a student who had done exercises but had not used the system developed for the field experiment, and
- a student who had used the system for doing exercises.

The tests revealed that respondents do not necessarily read instructions about which questions to answer. The questionnaire was fixed after the test with the first participant so that it disabled form elements that should not be answered if a previous question was answered in a specific way. The test with the second participant still showed that a disabled form element was confusing as the respondent had not read the instruction to skip the question. One question was removed from the questionnaire to make it as simple as possible for those respondents who had used the system during the course. The removed question was displayed afterward only to those who answerer that they had not used the system. This extra question was not utilized in this study. It was just for the purpose of the user-interface development.

The participants were also interviewed during the tests to check that the questionnaire items were understandable. The final version of the questionnaire used in this study is available on-line at <http://www.cc.jyu.fi/~mabengts/gradu/itk135/form1.php>.