◆ Application Paper

A Comprehensive Review of Methods and Techniques to Evaluate Usability of Interactive IT Products

Lim, Chee-Hwan*

Abstract

Usability is playing a more important role in interactive products or systems (e. g., information technology products, computer software) development, and it has become a primary factor in determining the acceptability and consequent success of the products. This study investigates some different techniques and methods to evaluate the usability of the interactive products or systems. Various evaluation methods have been tried ranging from formal to informal and empirical techniques. The classification of the usability evaluation methods(UEMs) and comparisons between these evaluation methods are presented. Some issues raised by the UEMs are also discussed. In addition, problems of selecting appropriate usability evaluation methods are discussed.

1. Introduction

Usability is now widely recognized as important product quality alongside technical aspects such as functionality, reliability, performance, maintenance, etc. In the same way that engineers are developing objective measurement techniques to assess the achievement of these technical qualities, so human factors or user interface(UI) researchers have striven to understand what methods can be taken to evaluate the usability of an interactive product or system. Many methods for evaluating the usability of interfaces have been used and developed in the fields of human-computer interaction or software ergonomics. In this study, we propose a review of some different techniques and methods to evaluate the usability of interactive products or systems, and discuss issues raised by these evaluation methods.

2. Literature Survey

Several authors have considered the problems of efficiently evaluating the usability of the interactive products or systems. Gould and Lewis (1985) discussed four critical factors

* Dept. of Management Information System, Seowon University

that need to be considered in developing and evaluating a computer system for end users. These factors include early focus on users, interactive design, empirical measurement, and iterative design. Rubinstein and Hersh (1984) described a top-down design approach to software interfaces, which includes collecting information, producing a design, building a prototype, evaluating the system, and delivering the system. Rouse (1984) proposes both a top-down analytical evaluation of design objectives as well as a bottom-up empirical evaluation of the resulting design configuration.

Nielsen (1993a) departs from the traditional life cycle in his Usability Engineering Life Cycle, in which the emphasis is on completing as much work as possible before design begins. Another life cycle model which emphasizes product evaluation throughout the whole project is the Star life cycle developed by Hix and Hartson (1993). They hypothesized that UI development naturally occurs in alternating waves between traditional top-down analysis (analytic), and bottom-up development (synthetic). Chignell and Valdez (1991) recommend small and frequent iterations using rapid prototyping techniques. Shneiderman (1992, pp. 498-499) presented a more elaborate, eight-stage life cycle for interactive systems development.

In order to evaluate the usability of the interactive products in early design and development phases, many other researchers have considered predictive models. Tullis investigated predicting user performance for static alphanumeric displays (Helander, 1988). Perlman developed an axiomatic model of information presentation (Sears, 1993). His model allows designers to specify relationships between the information to be presented and rules for how to show these relationships.

Norman (1986) formulated a theory, "The Seven Stages of Action," to explain how people do things. Card, Moran and Newell (1983) developed GOMS (Goals-Operators-Methods-Selection rules) and Keystroke-Level models that are attempts to describe certain kinds of operator behavior in certain situations. Lohse (1991) developed a model to predict the amount of time necessary to extract information from a graph. This work focused on where users must look and the cognitive tasks required at each step. Sears (1993) developed a metric, Layout Appropriateness(LA), which can be used to evaluate the layout of widgets in an interface. A LA-optimal layout can be compared to proposed layouts or it can be used as a starting point when designing a new interface.

Riley and O'Malley (1984) developed planning nets to describe the hierarchical goal structure of user activity (Shackel and Richardson, 1991). Kieras and Polson (1985) described the user with a set of production rules derived from a GOMS analysis of activity, and described a product in terms of general transition networks(GTN). Grey, John and Atwood (1992) developed CPM-GOMS (Critical Path Methods-GOMS) which attempts to adapt the GOMS technique to handle complex tasks. In addition, there are different formalisms for constructing user models: MAD(Scapin and Pierret-Golbreich, 1990), TAG (Payne and Green, 1986), CLG (Moran, 1981), and various kinds of grammars BNF (Reisner, 1981), etc.

On the other hand, various methods which rely on rules of thumb and knowledge of evaluators have been considered in the literature. Smith and Mosier (1986), with 944 guidelines, Brown (1988), with 302 guidelines, and Mayhew (1992), with 288 guidelines, presented extensive sets of usability guidelines. Nielsen and his colleagues (Nielsen and Molich, 1990; Nielsen, 1992, 1993a, 1994) developed a usability inspection method known as heuristic evaluation. Polson, Lewis, Rieman, and Wharton (1992) presented cognitive walkthrough method that focuses on evaluating a design for ease of learning, particularly by exploration.

Bias (1991) proposed pluralistic walkthrough as a technique to utilize different kinds of expertise. Wixon et al. (1994) described a standards inspection where a single UI expert systematically demonstrates the different aspects of the interface under evaluation to a design team. Wixon et al. (1994) also described the use of a consistency inspection held at Digital Equipment Corporation to evaluate and improve the interface consistency across different components of an individual application.

When considering the usability evaluation of interactive products or systems, the consensus appears to that at some stage in the development life cycle, user-based evaluation should be undertaken (Gould and Lewis, 1985; Hix and Hartson, 1993; Nielsen, 1993a, Rubin; 1994; Helander et al.; 1997). Whiteside, Bennett, and Holtzblatt (1988) provided a good overview of usability engineering. Nielsen (1993a) stated that usability testing is the most fundamental usability method because it provides direct information about how people use products, and their problems with the interface being tested. Nielsen and Landauer (1993) analyzed usability problems described in 11 published projects and concluded that the maximum cost/benefit ratio for a medium-large software project would have been obtained through the use of four evaluators.

Rubin (1994) highlighted that there are four main types of usability tests: exploratory, assessment, validation, and comparison. Lindgaard (1994) indicated the importance of a pilot study. Lea (1988) identified several possible rating scales that can be used in post-experiment questionnaires. Dix et al. (1993) outlined that interviews are particularly useful in eliciting information about user attitudes, impressions, and preferences. Hoiem and Sullivan (1994) reviewed the design and use of integrated data collection and analysis tools including both program event capture and video logging. Smilowitz, Darnell, and Benson (1994) presented an evaluation of testing carried out in a usability laboratory and in actual work environments. Cross-comparisons have been made between evaluator-based approaches and user-based methods (Desurvire et al., 1991, 1992; Jeffries et al., 1991; Karat et al., 1992; Nielsen and Landauer, 1993; Nielsen and Mack, 1994). Several authors analyzed development projects for the costs and benefits of usability engineering (Brown, 1991; Gould et al., 1991; Karat, 1990, 1992; Mantei and Teorey, 1988).

The MUSiC performance measurement method was developed by the European MUSiC (Metrics for Usability Standards in Computing) project to provide a valid and reliable means of specifying and measuring usability (Macleod et al., 1996). The method gives useful feedback on how to improve the usability of the design and also includes tools and techniques for measuring user performance and satisfaction. Wixon and Wilson (1997) described usability engineering framework for product design and evaluation.

Henninger (2000) developed a methodology and an associated technology to create context-specific usability guidelines. Han et al. (2000) presented a systematic approach to enhancing the objective and subjective usability of consumer electronic products. Multiple linear regression techniques were used to model the relationship between the usability and design elements. The models help the designers not only identify important design variables but also predict the level of usability of a specific product. Andre et al. (2001)

developed User Action Framework, a structured knowledge base of usability concepts and issues, as a framework on which to build a broad suite of usability engineering support tools. The User Action Framework helps to guide the development of each tool and to integrate the set of tools in the practitioner's working environment.

3. Purposes of Usability Evaluation

There are three different purposes which can be identified in usability evaluation of the interactive product, irrespective of the type of interface, the hardware or software, the use or not of human subjects in the evaluation, the method used, and the type of data gathered. These purposes are: 1) the diagnosis of usability problems of the product, 2) the evaluation of the capabilities of the product, and 3) the evaluation of two or more products, or of design options.

The first is concerned with evaluation of a product still being designed to see which aspects of the UI work and which cause usability problems. The issue of concern is how to mitigate the effects of usability difficulties, or better still, how to ensure that usability problems never arise. The second involves incorporating usability goals into product specifications together with associated user performance requirements that the design must meet. It is concerned with the evaluation of a more or less finished product or interface to see whether usability goals have been achieved. This evaluation can qualify as usability goal-oriented approach. The third is concerned with the relative evaluation of two or more products, or of two or more options for the design of an interface to measure which alternative is the most usable. This evaluation objective is to compare several products or design options, e. g., to find out whether A is easier to use than B, or to find out which of A, B, and C is easiest to use. This purpose requires some sort of systematic experimental plan.

In the early stages of the design process usability evaluation is used to sort out alternative user interface designs and then to identify the preferred design. Later on, evaluation is performed to determine whether the design meets its requirements. Before an evaluation is performed it is important to know what the purpose of the evaluation is. There are several methods which have proved to be useful in supporting usability evaluation.

4. Classification of Usability Evaluation Methods

When considering the usability evaluation of interactive products, the consensus appears to be that at some stage in the development life cycle, user-based evaluation should be undertaken (Gould and Lewis, 1985; Nielsen, 1993a; Whiteside, Bennett, and Holzblatt, 1988; Wilson and Corlett, 1990). Some work has been undertaken towards the development and refinement of evaluator-based iterative usability evaluation methods (Nielsen and Mack, 1994; Nielsen and Molich, 1990; Rieman, Davies, Hair, Esemplare, Polson, and Lewis, 1991; Shackel and Richardson, 1991).

Usability evaluation methods fall into three broad categories: predictive models, usability inspection methods, usability testing. Table 1 shows the extent of correspondence between the evaluation methods and the three purposes of usability evaluation.

Table 1. The extent of correspondence between methods and purposes in usability evaluation (H: high, M: medium, L: low)

Purposes / Methods	Diagnosis of usability problems	Evaluation of capabilities (usability goals)	Comparative evaluation
Predictive models	L	M	L
Usability inspection	H	L	M
Usability testing	H	H	H

4.1 Predictive models

The formal methods include models and formulas used to predict usability. As Shackel and Richardson (1991) highlight, models can be used to predict user performance instead of empirical measurements. Predictive models can be constructed without the need for prototypes or users to test them. This allows them to be constructed early in the development life cycle and at a low cost.

Predictive models can be used in two different ways by designers: 1) to generate preliminary design ideas; 2) as tools for evaluating preliminary designs by predicting user performance and satisfaction. Predictive models can be classified under two headings: task analytic models, and display models.

Task analytical models focus on the translation of goals into actions which can be seen on the left side of Normans cycle of HCI activities (Gugerty, 1993). The display models, however, concentrate on the users perception and interpretation of the display which is related to the right side of Normans model. As Gugerty highlights, these models offer the potential of improving early designs and replacing some of the early phases of the usability testing, thus reducing the cost of interface design.

Examples of task analytical models include: *CLG*(Command Language Grammar), *GOMS*(Goals, Operators, Methods and Selection rules), *CCT* (Cognitive Complexity Theory), *PS*(Production Systems), *TAG*(Task Action Grammar), *GTN*(Generalized Transition Networks). Predictions obtained from formal models could be confirmed in a number of studies. However, it should be noted that only small interaction tasks have been studies so far in highly controlled experiments. In order to complement these formal methods, one will certainly need experimental work with realistic tasks and a higher ecological validity.

Display models are useful methods for assessing the suitability of an interface without requiring empirical testing in the lab. Display models can either be task-independent such as Tullis' Model (Tullis, 1986), or can be task-sensitive like Layout Appropriateness (Sears, 1993). Display models are useful methods for assessing the display suitability of an interface. However, display models only measure very specific UI attributes. Tullis's model evaluates an interface based on search times. LA evaluates on the basis of distance between commonly used widgets in the interface.

4.2 Usability inspection methods

Usability inspection is the generic name for a set of methods that are all based on having evaluators inspect or examine usability-related aspects of an UI. (Nielsen and Mack, 1994). Typically, usability inspection is aimed at finding usability problems in an existing UI design, though some methods also address issues like the severity of the usability problems and the overall usability of an entire design.

Usability inspection methods have been adopted rapidly by industry because they are cheap to use, do not require any special equipment, and can be conducted early in the development life cycle. Usability inspection is the generic term for several methods, including at least the following eight: heuristic evaluation, cognitive walkthroughs, pluralistic walkthroughs, guideline reviews, feature inspection, formal usability inspections, consistency inspection, standards inspection.

Although each method has a different emphasis, they all share the goal of detecting as many usability problems as possible before lab-based testing begins. However, although these methods have the advantage that they do not require any test subjects, Desurvire (1994) found that the heuristic evaluation with UI experts only found 44% of the usability problems identified in the lab (Nielsen and Mack, 1994, p 184). Early evaluation methods in general provide an initial rough filter which can be used to detect problems early in the design when they are easier to correct.

4.3 Usability testing

In usability testing, representative users work on typical tasks using a product or system and evaluators use results to see how an UI supports the users to do their tasks. As shown in Table 1, usability testing is the most fundamental usability method because it provides direct information about how people use computers, and their problems with the interface being tested. However, several contexts of testing can dramatically alter experimental results. It is therefore necessary to work out the exact purpose of the test and select an appropriate testing methodology, and then conduct pilot studies to remove as many experimental errors as possible before any serious tests are conducted.

Usability testing differs in a goal of evaluation and in a type of data collected. The goal of the evaluation can either be: 1) to explore different design proposals, 2) to determine how well low-level interface detail has been implemented, 3) to check that an interface meets pre-defined usability standards, or 4) to compare several different competing products or systems. After determining the goal of the evaluation a small pilot study should be conducted to see if the experiment generates the type and amount of data required. The type of data collected can either be quantitative or qualitative. Quantitative data is useful for identifying problems, such as user performance is too slow, whereas qualitative data is better at providing clues as to why these problems occur (Hix and Hartson, 1993).

Many articles have been written on methods for usability testing; articles by Whiteside et al. (1988), by Dumas and Redish (1994), and by Rubin (1994) provide useful introductions. Some of the more common usability testing methods are as follows: thinking aloud protocols, constructive interaction (co-discovery learning), coaching method (question-asking protocols), shadowing method, performance measurement, retrospective testing (video recording), automatic computer logging (software logging), direct observation, structured questionnaires.

As a general rule, usability testing early in the design of a product is especially cost-effective because results can be used to refine product design before the product itself has been built. Because the product itself is not available early in the design process, prototypes of the product UI are tested for usability. Prototypes may take different forms at different stages, they may be simple paper-pencil drawings or screen designs rapidly created using a graphics editor. At later stages, they may be simulations of a portion of the UI necessary for completing some particular task.

Usability engineering is user testing that is more formal in the sense that UI specialists or developers set explicit, quantitative performance goals known as metrics (Nielsen, 1993a). Usability engineering is a widely accepted approach to evaluating and improving a product's UI. A wide variety of usability methods and many different measures are available for use in the evaluation process.

5. Comparison and Selection of Usability Evaluation Methods

Various usability evaluation methods have been tried ranging from formal to informal and to empirical methods. Several comparisons of usability methodologies have recently appeared in the literature. Comparisons of usability methods are, however, sometimes difficult to interpret. Each method is still only loosely defined and overlap across methods is common. Under the current state of the art, formal methods are very difficult to apply and do not scale up well to handle larger UIs. According to Nielsen and Molich (1990), practical constraints dictate that often expert-based, heuristic evaluations are the only feasible option for most organizations.

Jefferies et al. (1991) compared the relative effectiveness of four evaluation methods at uncovering usability problems in a software application: applying guidelines, heuristic evaluation, cognitive walkthroughs, and usability test. The results of this study show clearly the strengths and weaknesses of these methods. Usability testing finds global problems very well but is poor at uncovering local problems. Heuristic evaluation, on the other hand, finds many specific, local problems. It would appear, therefore, that heuristic evaluation and usability testing nicely complement each other.

While using both heuristic evaluation and usability testing may prove complementary, Bailey, Allan, and Raiello (1992) have shown that changing many of the local problems that are uncovered with heuristic evaluation did not improve the usability of software UI (Dumas and Redish, 1994). Desurvire et al. (1992) compared usability testing with heuristic evaluation and cognitive walkthroughs. The usability test found more than twice as many problems as the heuristic evaluation, which, in turn, found more problems than the cognitive walkthrough.

Karat et al. (1992) found that in their tests: empirical testing, individual walkthroughs, and team walkthroughs, a third of the problems identified were common to all methods. Karat et al. (1992) continued by suggesting that although there was significant overlap of uncovered problems, these methods are in fact complimentary. As Jeffries and Desurvire (1992) note, the best evaluation of an UI comes from applying multiple evaluation methods. Nielsen (1993a) states that ideally usability engineering methods should be applied throughout the development life cycle.

As a result of the comparison, usability engineering has shown some successes in addressing usability in the development cycle but it requires too much expertise and takes too much time. It is expensive to adopt this method. Usability inspection methods such as heuristic evaluation and usability walkthrough, are less formal techniques which require experts' opinions. These are fast and cheap but subject to bias and not representative of the real world.

Table 2 shows advantages and disadvantages of some typical usability evaluation methods. As you can see, the methods are intended to supplement each other, since their advantages and disadvantages can partly make up for each other. It is therefore highly recommended not to rely on a single usability method to the exclusion of others. With such a large array of methods to choose from, how should usability evaluators select the

Table 2. Advantages and disadvantages of some typical evaluation methods

Adv. of task analytic models	Disadv. of task analytic models
· can limit designs to psychologically plausible alternatives; · can be used to decide between two alternative interface designs; · can estimate performance times for specified tasks; · can indicate "hot spots" where interface errors or delays can be expected; · can contribute to the development of training programs.	· assume error-free interaction; · most are time consuming; · most require specialist psychological knowledge
Adv. of heuristic evaluation	**Disadv. of heuristic evaluation**
· quick to perform; · relatively inexpensive; · evaluators can be used to suggest possible solutions; · uncover lots of potential usability defects.	· several evaluations must usually be performed for maximum benefit; · ideally done by experts; · evaluators can have biased views; · can be difficult to distinguish between trivial and serious problems.
Adv. of thinking aloud protocols	**Disadv. of thinking aloud protocols**
· requires little training so designers can perform evaluations; · provides qualitative data about the users' cognitive processes; · data collection is fast.	· is unnatural to most people; · analyzing results can be difficult and time consuming; · value depends on analyst's skill and knowledge; · may not provide accurate descriptions of what is happening.

Table 2. Continued

Adv. of cognitive walkthroughs	Disadv. of cognitive walkthroughs
· allows detailed usability insight early in the project; · allows evaluation of competing solutions before development begins; · forces the evaluators to consider the background knowledge and environment of the intended users; · provides a method of mediation between the requirements and the development sides of the design team.	· only a small sub-set of tasks can realistically be evaluated; · requires knowledge of cognitive science terms, concepts, and skills; · task selection can be problematic; · can lead to suboptimal or erroneous solutions; · does not provide a global view of the interface.
Adv. of video recording	**Disadv. of video recording**
· powerful method for persuading developers that usability problems do exist; · useful as a backup when events become too quick to record in real-time; · can be shown to users (retrospective think-aloud protocols) or to independent observers.	· data analysis is time consuming; · technique is highly intrusive; · difficult to synchronize video sources with other sources such as audio and software logging data.
Adv. of software logging	**Disadv. of software logging**
· completely unobtrusive to subjects; · data is highly accurate (e.g. time stamps); · cheap (except in terms of disk storage); · useful for recording longitudinal data about user performance.	· generates large quantities of data; · data events recorded are low-level; · sometimes difficult to work out user actions and motives from recorded events.

most appropriate method for a specific project? There is no clear answer on which usability method is the best. The choice of appropriate evaluation methods involves a comparative analysis based on multiple dimensions. Consider some of the dimensions that can influence the choice of evaluation methods.

· What is the purpose of the evaluation?
· Who will be doing the evaluation?
· What information to collect?
· How much resource?

Table 3 shows relative cost of usability evaluation methods. In Table 3, to determine the cost of conducting a usability evaluation method, the following were considered: personnel required: number of users, usability experts, human factors specialists, and software developers; time required for data collection and analysis; need for coordination that is, whether the method requires the participants to be present together.

Table 3. Relative cost of the usability evaluation methods

Low Cost		
• Heuristic evaluation		
Medium Cost		
• Cognitive walkthroughs • Direct observation • Formal usability inspections	• Feature inspection • Standards inspection	• Guideline reviews • Questionnaires
High Cost		
• Constructive interaction • Performance measurement • Pluralistic walkthroughs	• Coaching method • Retrospective testing • Thinking aloud protocols	• Shadowing method

Consequently, the choice of appropriate evaluation methods involves a comparative analysis, based on several factors: the development stage of the product; time required to conduct the evaluation; cost of evaluation (equipment, personnel and time); importance of accuracy; number of users required; number of evaluators required; level of evaluator expertise required; number, type and severity of problems identified.

6. Conclusions

One of several goals of human factors is to improve usability for real users in the real world, to make sure that the interface between human and interactive product serves the human while working under the products constraints. The approaches to improving usability in the development process are to perform usability testing with a product in the usability lab, to review a product by UI or human factors experts, and to make recommendations for redesign based on the results of evaluation. To obtain this goal, useful and precise evaluation methods are required.

Various usability evaluation methods have been tried ranging from formal to informal and to empirical methods. Usability evaluation methods are usually classified into predictive models, usability inspection methods, and usability testing. These methods have their advantages and disadvantages. Thus, the choice of appropriate evaluation methods involves a comparative analysis, based on the development stage of the product, time required to conduct the evaluation, the importance of accuracy, costs, etc.

The usability of the interactive product is affected not only by the features of the product itself, but also by many other factors. These include the characteristics of the users, the tasks they are carrying out, and the organizational and physical environment in which the product is used. Therefore, awareness of these contextual factors is important throughout the evaluation process.

References

[1] Andre, T., Hartson, H., Belz, S. and McCreary, F., The user action framework: a reliable foundation for usability engineering supports tools, *International Journal of Human-Computer Studies*, Vol. 54, pp. 107-136, 2001.

[2] Bias, R., Walkthroughs: Efficient collaborative testing , *IEEE Software*, Vol. 8, pp. 94-95, 1991.

[3] Brown, J. S., Research that reinvents the corporation , *Harvard Business Review*, pp. 102-111, 1991.

[4] Card, S. K., Moran, J. P., and Newell, A. *The Psychology of Human Computer Interaction*, Erlbaum., 1993.

[5] Chignell, M. H. and Valdez, J. F., Truncated experiments and field experiments in usability engineering , *Proceedings of the IEEE International Conference on Systems*, Man and Cybernetics, pp. 1349-1353, 1991.

[6] Desurvire, H., Lawrence, D., and Atwood, M., Empiricism versus judgment: Comparing user interface evaluation methods on a new telephone-based interface , *ACM SIGCHI Bulletin*, 23(4), pp. 58-59, 1991.

[7] Desurvire, H. W., Kondziela, J. M., and Atwood, M. E., What is gained and lost when using evaluation methods other than empirical testing , *In People and Computers VII*, Cambridge Univ. Press., pp. 89-102, 1992.

[8] Dix, A., Finlay, J., Abowd, G., and Beale, R., *Human-Computer Interaction*, Prentice Hall International (UK) Ltd., 1993.

[9] Dumas, J. S. and Redish, J. C., *A Practical Guide to Usability Testing*, Ablex Publishing Co., 1994.

[10] Gould, J. D., Boies, S. J., and Lewis, C., Making usable, useful productive-enhancing computer applications , *Communications of the ACM*, 34(1), pp. 75-85, 1991.

[11] Gould, J. D., and Lewis, C., Designing for usability: Key principles and what designers think , *Communications of the ACM*, Vol. 28, pp. 300-311, 1985.

[12] Grey, W. D., John, B. E., and Atwood, M. E., The precis of project Ernestine, or, an overview of a validation of COMS, Proceedings of ACM Conference on Human Factors in Computing Systems, ACM Press, pp. 307-312, 1992.

[13] Gugerty, L., The use of analytical models in human-computer interface design, *International Journal of Man-Machine Studies*, Vol. 38, pp.625-660, 1993.

[14] Han, S., Yun, M., Kim, K. and Kwahk, J., Evaluation of product usability: development and validation of usability dimensions and design elements based on empirical models, *International Journal of Industrial Ergonomics*, Vol. 26, pp. 477-488, 2000.

[15] Helander, M., *Handbook of Human-Computer Interaction*, Elsevier, pp. 755-927, 1988.

[16] Helander, M., Landauer, T., and Prabhu, P., *Handbook of Human-Computer Interaction*, Second, Completely Revised Edition, Elsevier, 1997.

[17] Henninger, S., A methodology and tools for applying context-specific usability guidelines to interface design, *Interacting with computers*, Vol 12., pp. 225-243, 2000.

[18] Hix, D. and Hartson, H. R., *Developing User Interfaces: Ensuring Usability Through Product and Process*, John wiley & Sons, Inc., 1993.

[19] Hoiem, D., and Sullivan, K., Designing and using integrated data collection and

analysis tools: Challenges and considerations , *Behaviour and Information Technology*, 13(1-2), pp. 160-170, 1994.

[20] Jeffries, R., Miller, J., Wharton, C. and Uyeda, K., User interface evaluation in the real world: A comparison of four techniques , *Proceedings of CHI 91 Conference on Human factors in Computing Systems*, ACM Press, pp. 119-124, 1991.

[21] Karat, C., Campbell, R. and Fiegel, T., Comparison of empirical testing and walkthrough methods in user interface evaluation , *Proceedings of CHI 92 Conference on Human factors in Computing Systems*, ACM Press, pp. 397-404, 1992.

[22] Karat, C. M., Cost-justifying human factors support on software development projects , *Human Factors Society Bulletin*, 35(11), pp. 1-4, 1992.

[23] Kieras, D. E. and Polson, P. G., An approach to the formal analysis of user complexity , *International Journal of Man-Machine Studies*, 22, pp. 365-394, 1985.

[24] Lea, M., Evaluating user interface designs , *User Interface Design for Computer Systems*, Ed. T. Rubin, Ellis Horwood Ltd., pp. 134-167, 1988.

[25] Lindgaard, G., *Usability Testing and System Evaluation*, Chapman & Hall, 1994.

[26] Lohse, G. L., A cognitive model for the perception and understanding of graphs, *CHI '91 Conference Proceedings*, LA, ACM Press, pp. 131-138., 1991.

[27] Macleod, M., Bowden, R., and Bevan, N., The MUSiC Performance Measurement Method, *HCI '96*, Tutorial 14, *Measuring Usability-MUSiC Methods*, London, 1996

[28] Mantei, M. M. and Teorey, T. J., Cost/benefit analysis for incorporating human factors in the software lifecycle , *Communications of the ACM*, Vol. 31, pp. 428-439, 1988.

[29] Mayhew, D. J., *Principles and Guidelines in Software User Interface Design*, Prentice-Hall, Inc., 1992.

[30] Moran, T. P., The command language grammar: a representation for the user interface of interactive computer systems , *International Journal of Man-Machine Studies*, 15, pp. 3-50, 1981.

[31] Nielsen, J., Finding usability problems through heuristic evaluation , *Proceedings of CHI 92 Conference on Human factors in Computing Systems*, ACM Press, pp. 373-380, 1992.

[32] Nielsen, J., *Usability Engineering*, Academic Press, London, 1993a.

[33] Nielsen, J., Noncommand user interfaces , *Communications of the ACM*, Vol. 36, No. 4, pp. 83-99, 1993b.

[34] Nielsen, J., Usability laboratories , *Behaviour & Information Technology*, Vol. 13, pp. 3-8, 1994.

[35] Nielsen, J. and Landauer, T. K., A mathematical model of the finding of usability problems , *Proceedings of ACM CHI93*, pp. 206-221, 1993.

[36] Nielsen, J. and Mack, R. L., *Usability Inspection Methods*, John Wiley & Sons, Inc., 1994.

[37] Nielsen, J. and Molich, R., Heuristic evaluation of user interfaces , *Proceedings of ACM CHI90*, 249-256, 1990.

[38] Norman, D., Cognitive engineering, (Ed.) Norman, D. and Draper, S., *User Centered System Design*, Lawrence Erlbaum Associates, pp. 31-61, 1986.

[39] Payne, S. J. and Green, T. G. R., Task-action grammars: a model of the mental representation of task languages , *Human-Computer Interaction*, 2, pp. 93-133, 1986.

[40] Polson, P., Lewis, C., Rieman, J., and Wharton, C., Cognitive walkthroughs: A method for theory-based evaluation of user interfaces , *International Journal of Man-Machine Studies*,

36, pp. 741-773, 1992.
[41] Reisner, P., Formal grammar and human factors design of an interactive graphics system , *IEEE Transactions on Software Engineering,* 7(2), pp. 229-240, 1981.
[42] Rieman, J., Davies, S., Hair, D. C., Esemplare, M., Polson, P., and Lewis C., An automated cognitive walkthrough , *Proceedings of CHI 91 Conference on Human Factors in Computing Systems,* ACM Press, pp. 427-428, 1991.
[43] Rouse, W. B., Design and evaluation of computer-based decision support systems , In Salvendy, G., Ed., *Human-computer interaction,* Amsterdam, Elsevier, 1984.
[44] Rubenstein, R. and Hersh, H. M., *The human factor: designing computer systems for people,* Burlington, MA, Digital Press., 1984.
[45] Rubin, J., *Handbook of Usability Testing: How to plan, design, and conduct effective tests,* John Wiley & Sons, Inc., 1994.
[46] Scapin, D. L. and Pierret-Golbreich, C., Towards a method for task description: MAD , *Work with Display Units,* Elsevier, pp.371-380, 1990.
[47] Sears, A., Layout appropriateness: A metric for evaluating user interface widget layout , *IEEE Transactions on Software Engineering,* Vol. 19, No. 7, pp. 707-719, 1993.
[48] Shackel, B., and Richardson, S., *Human Factors for Informatics Usability,* Cambridge Univ. Press, 1991.
[49] Shneiderman, B., *Designing the User Interface,* 2nd Ed., Addison-Wesley, pp. 471-500, 1992.
[50] Simon Wilkinson, S., *Applicability of Software Usability Evaluation Techniques,* University of Abertay, Dundee, 1994.
[51] Smilowiz, E., Darnell, M., and Benson, A., Are we overlooking some usability testing methods? A comparison of lab, beta, and forum tests , *Behaviour and Information Technology,* 13(1-2), pp. 183-190, 1994.
[52] Smith, S. L., and Mosier, J. N., *Guidelines for Designing User Interface Software,* Report No. MTR-10090, Esd-TR-86-278, The Mitree Co., Beford, MA., 1986.
[53] Thomas, J. C. and Kellogg, W. A., Minimizing ecological gaps in interface design , *IEEE Software,* January, pp. 78-86, 1989.
[54] Tullis, T. S., A system for evaluating screen formats , *Proceedings of the Human Factors Society 30th Annual Meeting,* pp 1216-1220, 1986.
[55] Whiteside, J., Bennett, J. and Holzblatt, K., Usability engineering: our experience and evolution , In: M. Helander (Eds.), *Handbook of Human-Computer Interaction,* Elsevier, pp. 791-817, 1988.
[56] Wilson, J. R. and Corlett, E. N., *Evaluation of human work - a practical ergonomics methodology,* Taylor & Francis Ltd, pp. 218-247, 1990.
[57] Wixon, D., Jones, S., Tse, L., and Casaday, G., Inspections and design reviews: framework, history, and reflection , *Usability Inspection Methods,* Ed. J. Nielsen and R. L. Mack, John Wiley & Sons, Inc., 1994.
[58] Wixon, D. and Wilson, C., The usability engineering framework for product design and evaluation, In: M. Helander (Second Eds.), *Handbook of Human-Computer Interaction,* Elsevier, pp. 653-688, 1997.