A Usability Assistant for the Heuristic Evaluation of Interactive Systems

Costin Pribeanu

National Institute for Research and Development in Informatics – ICI Bucharest,

8-10, Averescu Avenue, Bucharest 1, Romania

e-mail: pribeanu@ici.ro

Abstract: The increasing demands for usable interactive systems in the context of limited project budgets bring in front the need for faster and cheaper evaluation methods. Heuristic evaluation is a kind of inspection method that proved to be cost effective. Typically, the method involves a small number of evaluators that are testing the interactive system against a set of usability principles called heuristics. A way to increase the efficiency of usability evaluation methods is to provide evaluators with software tools able to assist in documenting and recording of usability problems. This paper presents a software assistant for usability evaluation which provides with various facilities to conduct heuristic evaluation: definition of the tasks set, specification of heuristics used, and documenting of usability problems. In order to support the specific requirements of a target application domain a set of usability guidelines could be specified that are detailing the heuristic set. These guidelines could be consulted during usability problem identification and specification. This way, a broader range of evaluator preferences and requirements could be accommodated.

Keywords: usability, heuristic evaluation, usability evaluation assistant, software tools, tools for working with guidelines

Costin Pribeanu is currently a Senior Researcher at the National Institute for Research and Development in Informatics – ICI Bucharest. He received his PhD degree in Economic Informatics in 1997 from the Academy of Economic Studies in Bucharest, Romania. His research interests include usability evaluation and design of interactive systems, with a focus on task analysis, task-based design of user interfaces, design patterns, and guidelines-based evaluation. He is author of several scientific papers on user interface design and evaluation. He is an editorial board member for Springer HCI Book Series and served as program committee member / reviewer for several major conferences in this domain. Costin Pribeanu served as Chair of RoCHI (ACM SIGCHI Romania) from April 2001 to June 2009.

1. Introduction

The increasing demand for usable interactive systems in the context of a limited project budget and strict deadlines creates an extra pressure for evaluators and designers. This reveals the need for faster and cheaper evaluation methods.

Depending on the purpose and the moment when it is done, usability evaluation could be formative or summative (Scriven, 1991). Formative usability evaluation is performed in an iterative development cycle and aims at finding and fixing usability problems as early as possible (Teofanos and Quesenbery, 2005). The sooner these problems are identified, the less costly the effort to fix them is.

Formative usability evaluation can be carried on by conducting an expert-based usability inspection and / or by conducting user testing with a small number of users. In this last case, the evaluation is said to be user-centered, as opposite to expert-based formative evaluation.

Heuristic evaluation is a kind of inspection method which typically involves a small number of evaluators that are testing the interactive system against a set of usability principles called heuristics. This method proved to be cost effective and is widely used by the usability practitioners' community (76% according to UPA Survey, 2005).

Heuristic evaluation provides with two kinds of measure: quantitative (number of usability problems per severity level) and qualitative (detailed descriptions of individual usability problems).

The quality of usability problem description is critical for the usefulness of a usability report. On the other hand, there is a lot of work to be done in order to properly describe each usability problem. A way to increase the efficiency of any evaluation method is to provide evaluators with suitable tools able to assist them during the evaluation process. As shown by Hvannberg et al. (2007), not only these problem registration tools are improving the immediate management of usability problems but they are also supporting a structured usability problem reporting.

This paper presents a software tool for usability evaluation which provides several facilities to conduct a heuristic evaluation:

definition of the tasks set, specification of the heuristics used and structured description of usability problems. In order to better support evaluators' expertise, a set of usability guidelines that are detailing the heuristic set could be specified. This facility is also useful to fulfill specific requirements of a target application domain.

The rest of this paper is organized as follows: Related work in usability research is briefly presented in the next section with a focus on usability problem extraction, matching and reporting. The software tool is presented in Section 3. The paper ends with conclusion and future work in Section 4.

2. Related Work

2.1 Usability evaluation methods

The ISO/IEC standard 9126 (2001) defined usability as the capability of a software system to be understood, learned, used, and liked by the user when used under specified conditions.

How to measure and improve the usability of interactive systems is a key research concern in HCI research. There are several approaches to usability evaluation and, consequently many usability evaluation methods. There is also a continuous debate on the effectiveness, efficiency and usefulness of individual usability evaluation methods (Cockton & Woolrych, 2001; Hartson et al., 2001; Hornbaek, 2006; Law et al., 2007).

All evaluation methods aim to find usability problems. A usability problem was defined by Nielsen (1993) as any aspect of the user interface which might create difficulties to the user with respect to an important usability indicator (ease to understand, learn, and use, subjective user satisfaction).

According to their impact, usability problems could be classified on three levels of severity:

– Severe: the user is not able to accomplish the task goal or the task ends with a significant loss of data or time.

– Moderate: problems with an important impact on task execution but the user is able to find a solution.

– Minor: problems that are irritating the user but they don't have an important impact on the user's task.

Formative evaluation methods are grouped in two broad categories: inspection methods and user testing. Inspection methods are based on testing the user interface by a small number of experts. Evaluators could have expertise in usability, application domain or both (the best case). As such, the usability problems reported are anticipated problems (not real).

There are several types of inspection methods: heuristic evaluation (most often used), guideline based evaluation and cognitive walkthrough. Heuristic evaluation was proposed by Molich & Nielsen (1990) and it is based on a small set of widely accepted usability principles called heuristics.

User testing methods are based on testing the user interface with representative users. The user behavior is observed and recorded. Usually, users are required to "think aloud" during testing in order to better understand the difficulties in using the interactive system. Then the evaluators are analyzing the thing aloud protocols and extract the usability problems. In this case, usability problems are real, not anticipated.

It is a good practice to validate the results of heuristic evaluation by carrying on a user testing (Law & Hvannberg, 2002). This way the validity of the heuristic evaluation could be measured in terms of "false positives", i.e. anticipated by the heuristic evaluation but not confirmed by user testing and "false negatives", i.e. found by user testing and missed by heuristic evaluation.

In order to compare results, experts and users should evaluate the same thing, i.e. perform the same tasks set with the user interface.

2.2 Usability problem reporting

Usability problem reporting is an issue of interest in current usability research that is related to various aspects such as: how to document a usability problem, usability evaluation workflow, problem matching, report format, and downstream utility.

There are several approaches on documenting a usability problem. Hvannberg and Law (2003) proposed a comprehensive description template based on a classification of usability

problems (CUP). Later on, Vilbergsdottir et al. (2006) extended the template by targeting the need of an organization to improve the usability of a system under development. The augmented format consists of nine Pre-CUP attributes and 4 Post-CUP attributes.

Cockton & Woolrych (2001) proposed a more compact format having four main attributes:

- a short description of the problem (headline)

- likely difficulties for the user

- specific context where the problem occurred (the location in the interface)

- possible causes of the problem (what is wrong in design)

The trend is to use a structured format to document a usability problem in order to better support the activities in the evaluation workflow. Typically, the management of usability problems requires the carrying out of the following activities:

Problem extraction and recording performed individually by each evaluator. This work results in list of usability problems for each task.

Individual problem filtering. This work results in a unique list of usability problems found by an evaluator.

Collaborative problem filtering in order to produce a unique list of usability problems.

The first step is the immediate management of usability problem (Hvannberg et al, 2007). It is a time consuming activity which could benefit from a software tool.

The last two steps require an analysis that is known as problem matching which is a difficult and important activity (Hornbaek & Frokjaer, 2008). Matching means to determine if usability problems are similar or not and is the basis for removing the duplicates. As such, it is critical for the reliability and validity of the evaluation results.

There are several techniques for problem matching and Hornbaek & Frokjaer (2008) shows that different techniques produce different results in terms of problem grouping and unique problems found. On another hand, a structured usability problem template could help evaluators to better compare individual usability problems.

2.3 Software tools

Several software tools to assist evaluators during the evaluation process exists. However, most of them are targeted to guidelines based evaluation (Vanderdonckt & Farenc, 2000).

In a previous work we presented a web tool for working with guidelines that are targeting e-commerce applications (Barbu & Pribeanu, 2009).

Reporting of usability problems using a software tool has several advantages: better explanation of usability problems, easier entry and modification of usability problem attributes (important given the iterative evaluation process).

Based on a comparison between paper vs. tool recording of usability problems, Hvannberg et al. (2007) concluded that using a tool may increase the effectiveness and efficiency of usability evaluation. However they also mentioned that the evaluators using the tool for the first time complained about switching back and forth between the target application and the tool.

3. The Usability Assistant

3.1 Approach and implementation

In a previous work we presented a software tool for the management of usability problem that was integrated with an existing tool for working with guidelines (Pribeanu 2009). The heuristics set was built by adapting the ergonomic criteria elaborated by Bastien & Scapin (1993).

The idea of integrating a tool for working with guidelines with a tool for usability problem management relies on a common function: management of ergonomic criteria which, in our approach, are linked both to guidelines and usability problems.

Then we tried to use it in order to record the results of a heuristic evaluation targeting an augmented reality based interactive system (Iordache & Pribeanu, 2009). Although we used an updated set of ergonomic criteria (Bach & Scapin, 2003), we realized that the tool is not suitable given the lack of specific usability guidelines.

On another hand, the heuristics are available in a different dialog unit (a feature inherited from the tool for working with guidelines) which makes it less fast and straightforward to associate a usability problem and a heuristic.

Therefore we developed an improved version of the usability assistant whose purpose is twofold:

− Assist evaluators during the evaluation preparation and immediate management of usability problems.

− Provide an additional guidance with a set of usability guidelines that are detailing a given heuristic.

In order to prepare the evaluation, evaluators have to specify the set of tasks to be performed with the target interactive system. Also, they could prepare the set of heuristics to be used. Although the tool is providing with one, each evaluator could replace it with a custom set of heuristics.

The software tool was designed in a task-based approach to user interface design (Pribeanu & Vanderdonckt, 2002). The task model for the management of usability problems is presented in Figure 1 using the Concur Task Tree (CCT) notation developed by Paterno et al. (1997).

The task model has a layered structure that is reflecting the relationships between the main entities in the domain model: evaluation, task set, and usability problems. In this respect, the user interface design is based on both the task and domain models.
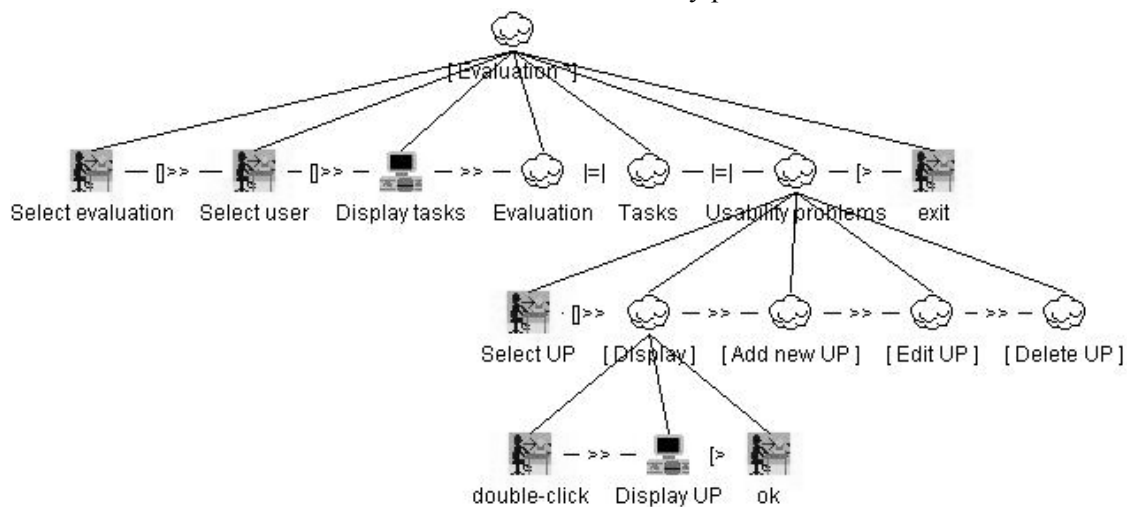
This version is intended for PC platforms under Windows XP operating software. The prototype is implemented in Microsoft Access. The code is written in Access Basic.

3.2 Managing the evaluation

The dialog unit for evaluation management is presented in Figure 2. The general layout is reflecting the master-detail relationship between the main entities in the domain model: evaluation, tasks and usability problems.
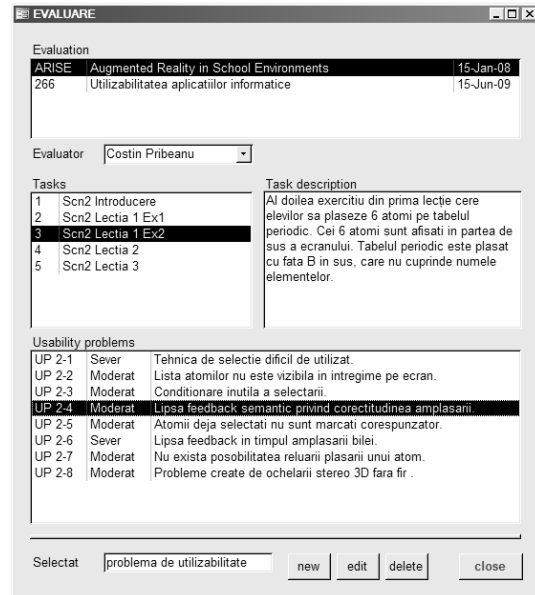

Figure 2. Management of evaluations

The command buttons at the bottom part of the form apply for the currently selected entity. For example, in Figure 2 the fourth usability problem in the list is selected. Hence pressing the "new" button a new usability problem could be introduced.

Figure 1. Task model for the management of usability problems

Firstly, the evaluator has to select an evaluation. A double click on the currently selected item displays the evaluation details. Then the evaluator selects her or his name from the drop-down list bellow.

The corresponding task set to be evaluated is displayed. For each task the task number and name are shown in the list box. A click on a task in the list displays the task description. The evaluator could also add a new task or edit / delete and existing one. These are the functions that enable an evaluator to create an evaluation task set.

Management of heuristics and usability guidelines is performed in separate dialog units. In Figure 3 the dialog unit for heuristics management is presented.
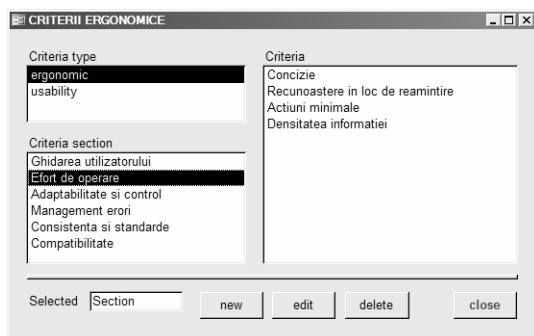

Figure 3. Management of heuristics

We are currently using a set of 24 heuristics (ergonomic criteria) which was built by merging and adapting the ergonomic criteria of Bastien & Scapin (1993) and the ten heuristics of Jacob Nielsen (1994). The list box in the left part of the screen is displaying the titles (headlines) of heuristics.

The set is structured in six groups: user guidance, workload, adaptability and control, error management, consistency and standards, and compatibility.

The dialog unit has a similar structure with the previously presented one, that is mirroring the relationship between criteria, criteria groups and heuristics.

3.3 Editing a usability problem

In Figure 4 is presented the dialog unit for editing a usability problem. The usability problem description follows the guidelines of Capra (2007).

The interactors (list boxes) are placed following the natural ordering of the

evaluation task. The structure of the presentation has two parts:

Usability problem description (left part) in a structured format adapted after Cockton & Woolrych (2001).

Titles of the heuristics used and the definition of the currently selected heuristic (right part).

When a usability problem is displayed with a double click on its title as explained in the previous section, only the left part of the dialog unit is shown. The right part is available during editing in order to better assist evaluators in the problem extraction process.
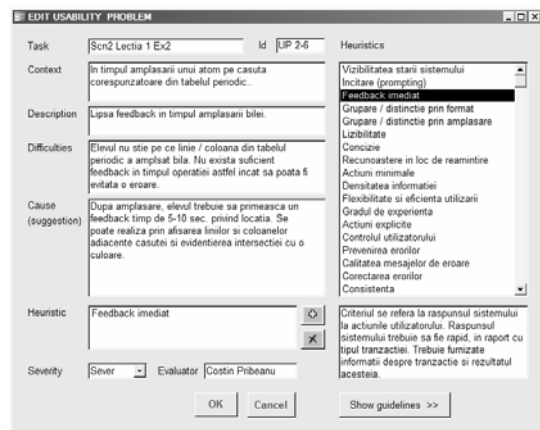

Figure 4. Editing a usability problem

The task name is only displayed by the system in the upper left side.

The evaluator has to type in a meaningful usability problem identifier. In our evaluation approach, this is done by prefixing each UP with the task number. We believe that giving meaningful identifiers for tasks and usability problems helps the further processing of individual problem lists.

Then the evaluator is specifying the context of use where the usability problem occurred. The best is to specify it as a step in the evaluation task that is describing what the user wanted to do.

The UP description is a concise sentence (title or headline) which is further detailed in the list box bellow by explaining the difficulties anticipated to be encountered by the user.

Then the evaluator has to describe the cause (what is wrong) and suggestions for designers. We believe that causes and suggestions should be recorded together given the fact

that they are pointing to a design solution that seems to be unusable.

The list of heuristics is displayed on the right side of the screen. When a heuristics is selected (click on the title) its definition is displayed in the bottom right side of the dialog unit. This way the evaluator could better document the usability problem. More over, less experienced evaluators could use this facility to better understand usability heuristics.

The selected heuristic is associated with the usability problem by pressing the "+" button. More than one heuristic could be selected. The evaluator could also delete an association made by mistake or after individual or collaborative filtering.

Finally, the evaluator selects the appropriate level of severity from the drop down box.

3.4 Consulting usability guidelines

An additional facility of the usability assistant is the consultation of usability guidelines that are linked to a heuristic.

Currently, there are hundreds of usability guidelines available in different public or private collections. Most of them are targeting the design and evaluation of e-commerce applications, e-learning applications or public administration web sites.

The process of guidelines collecting and organizing is on-going and many projects require a custom set of guidelines addressing specific requirements.

Based on our previous experience in usability evaluation of an educational system based on augmented reality, it is difficult to use only a set of heuristics (Iordache & Pribeanu, 2009). Although we employed a more specialized set of ergonomic criteria adapted to virtual environments by Bach & Scapin (2003), it was difficult to document usability problems that are related to specific interaction techniques. Moreover, during the evaluation several guidelines that are specific to this kind of platforms were proposed.

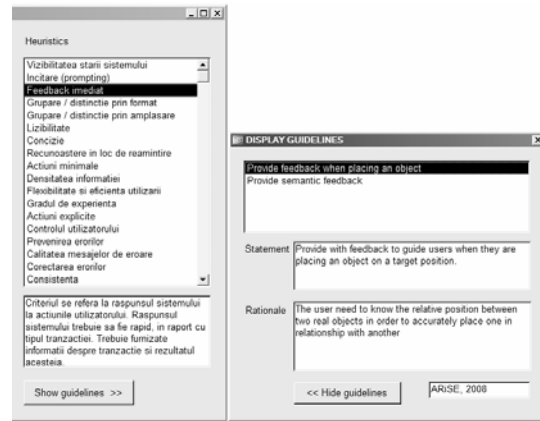The dialog unit in Figure 5 shows the guidelines that are associated to a given heuristics.

Figure 5. Displaying a guideline set

This dialog unit is opened by pressing the "show guidelines" button placed under the heuristic definition in Figure 3. When the evaluator selects a guideline its statement, rationale and reference is displayed bellow. The user returns to usability problem editing by pressing the "hide guidelines" button.

This feature provides with an additional assistance for evaluators and is good for better understanding and learning of usability heuristics. As such, it is useful both for novice and experienced evaluators that are less familiar with an application domain.

4. Conclusion and Future Work

In this paper we presented an improved version of a usability assistant which provides various facilities to conduct heuristic evaluation: definition of the tasks set, specification of heuristics used, and documenting of usability problems.

This tool has been developed based on previous experience with heuristic evaluation and guidelines-based evaluation. As such it integrates features from two different kinds of tools.

By enabling the specification of usability guidelines is possible to accommodate both individual preferences related to a preferred set of heuristics and specific requirements of a target interactive system.

The integration of two kinds of inspection method is beneficial for evaluators from at least three points of view. First, the usability assistant is more versatile since it is not confined to a single evaluation method. Second, the rich functionality provides with

more assistance during the evaluation process. Third, the tool provides a better understanding of usability knowledge along the relationship between ergonomic criteria, heuristics and guidelines. As such it is useful for training novice evaluators.

We intend to further develop this tool by adding functions to support the problem matching. In this respect, the next step is to support the individual filtering of usability problems.

Acknowledgement

REFERENCES

1. BACH, C., D. L. SCAPIN, **Adaptation of Ergonomic Criteria to Human-Virtual Environments Interactions**. Proceedings of Interact'03. IOS Press, 2003, pp. 880-883.

2. BARBU, D. C., C. PRIBEANU, **A Tool for Working with Guidelines for e-commerce Web Sites**. Proceedings of IE 2009 International Conference, Bucharest, pp. 169-174.

3. BASTIEN, J. M. C., D. L. SCAPIN, **Evaluating a User Interface with Ergonomic Criteria** INRIA Report, Roquencourt, 1993.

4. CAPRA, M., **Comparing Usability Problem Identification and Description by Practitioners and Students**. Human Factors and Ergonomics Society Annual Meeting Proceedings, Computer Systems (5), 2007, pp. 474-478.

5. COCKTON, G., A. WOOLRYCH, **Understanding Inspection Methods: Lessons from an Assessment of Heuristic Evaluation**. Blandford, A., Vanderdonckt, J., Gray, P.D. (Eds.), Proceedings of People and Computers XV. Springer-Verlag, 2001, pp. 171–182.

6. HARTSON, H. R., T. S. ANDRE, R. C. WILLIGES, **Criteria for Evaluating Usability Evaluation Methods**. International Journal of Human-Computer Interaction 13, 2001, pp. 373–410.

7. HORNBAEK, K., **Current Practice in Measuring Usability: Challenges to Usability Studies and Research**. International Journal of Human Computer Studies, 64, 2006, pp. 79-102.

8. HORNBAEK, K., E. FROKJAER, **Comparison of Techniques for Matching of Usability Problem Descriptions**. Interacting with Computers 20, 2008, pp. 505-514.

9. HVANNBERG, E. T., E. L.-C. LAW, **Classification of Usability Problems (CUP) Scheme**, Proceedings of Interact, 2003.

10. HVANNBERG, E. T., E. L.-C. LAW, M. C. LARUSDOTTIR, **Heuristic Evaluation: Comparing Ways of Finding an Reporting Usability Problems**, Interacting with Computers 19, 2007, pp. 255-240.

11. IORDACHE, D. D., C. PRIBEANU, **Comparison of Quantitative and Qualitative Data from a Formative Usability Evaluation of an Augmented Reality Learning Scenario**, Informatics Economic Journal, 13(3), 2009, pp. 67-74.

12. ISO 9126-1:2001 Software Engineering - Software product quality. Part 1: Quality Model.

13. LAW, L-C., HVANNBERG, E. T., **Complementarities and Convergence of Heuristic Evaluation and Usability Test: A Case Study of UNIVERSAL Brokerage Platform**. Proceedings NordiCHI Conference ACM 2002, pp. 71-79.

14. LAW, E. C., M. C. LÁRUSDÓTTIR, M. NORGAARD, (Eds), **Downstream Utility 2007: The Good, the Bad, and the Utterly Useless Usability Evaluation Feedback**, IRIT Press - Toulouse France, November 6th, 2007.

15. MOLICH, R., J. NIELSEN, **Improving a Human-computer Dialogue**, Communications of the ACM 33(3), 1990, pp. 338-348.

16. NIELSEN, J., **Usability Engineering**. Academic Press, New York, 1993.

17. NIELSEN, J., **Heuristic Evaluation**. Nielsen, J., and Mack, R.L. (Eds.), Usability Inspection Methods, John Wiley & Sons, New York, 1994.

18. PATERNÒ, F., C. MANCINI, S. MENICONI, **ConcurTaskTree: a Diagrammatic Notation for Specifying Task Models**. Proceedings of IFIP TC 13 International Conference on Human-Computer Interaction (Syndey, June 1997). Chapman & Hall, London 1997, pp. 362–369.

19. PRIBEANU, C., C. MARIAGE, J. VANDERDONCKT, **A Corpus of Design Guidelines for Electronic Commerce Web Sites**, M.J. Smith, G. Salvendy, D. Harris & R.J. Koubek (eds.) Proceedings of HCI International'2001 (New Orleans, 5-10 Aug 2001), Lawrence Erlbaum Associates, 2001, pp. 1195-1199.

20. PRIBEANU, C., J. VANDERDONCKT, **A Methodological Approach to Task-based Design of User Interfaces**. Studies in Informatics and Control, 11(2), 2002, pp. 145-158.

21. PRIBEANU, C., **O abordare bazată pe sarcină în proiectarea unui asistent software de evaluare a utilizabilității**. Revista Română de Interacțiune Om-Calculator, 2(1), 2009, pp. 31-44.

22. SCRIVEN, M., **Evaluation Thesaurus**. 4th ed. Newbury Park, CA: Sage Publications, 1991.

23. Theofanos, M., W. Quesenbery,. **Towards the Design of Effective Formative Test Reports**. Journal of Usability Studies, 1(1), 2005, pp. 27-45.

24. USABILITY PROFESSIONALS ASSOCIATION, **UPA 2005 Survey**.

25. VANDERDONCKT, J., C. FARENC, (eds.), **Tools for Working with Guidelines**, Springer, London, 2000.

26. VILBERSGTOTTIR, S.G., LAW, L-C., HVANNBERG, E. T., **Classification of Usability Problems (CUP) Scheme: Augmentation and Exploitation**, Proceedings of NordiCHI 2006, pp. 281-290.