Using and integrating Discount Usability Engineering in the life cycle of a health care Web application

Fabrício Ferracioli
Universidade Estadual de Londrina
Rodovia Celso Garcia Cid - PR 445 km 380 - Campus Universitário
Londrina, Brazil
fabricioferracioli@gmail.com

Maria Angélica de Oliveira Camargo-Brunetto
Universidade Estadual de Londrina
Rodovia Celso Garcia Cid - PR 445 km 380 - Campus Universitário
Londrina, Brazil
angelica@uel.br

ABSTRACT
Usability is an important characteristic in any interactive system, but sometimes is neglected by some software development teams because of the lack of knowledge of these about usability techniques. They think that the methods are hard to learn, execute and integrate on the software development life cycle. A kind of application that suffer with usability problems is the health care software. Our team work with a health care software developed by people from computer science and physiotherapy, without people related to usability. After various functions developed, the users continuously claim for easier ways to use the software, and we detect some learnability problems. At this point we decided to focus in usability, using the methods of Discount Usability Engineering. This work presents the methodology used during the heuristic evaluation and usability tests. The results show that is possible to reach a satisfactory amount of problems to correct, great feedback from users, similar results between heuristic evaluation and usability tests with users and that even people with few knowledge about usability can learn and conduct tests like ones of the Usability Engineering. Additionally, these methods can be integrated in our software life cycle, what will avoid future re-work.

Categories and Subject Descriptors
J.3 [**Computer Applications**]: Life and Medical Sciences—*health*; D.2.4 [**Software Engineering**]: Software/Program Verification

General Terms
Design, Human Factors

Keywords
Usability, Discount Usability, Health Care Web Applications

1. INTRODUCTION

In our university, there is a health care unit responsible for the physical rehabilitation of patients with chronic respiratory diseases. Chronic diseases in general require continuous attention to control the occurrence of crisis. Aiming to help health care professionals (mainly in cardiac and respiratory rehabilitation) to accompany their patients individually as well as to study groups of patients, according to specific features, it was designed, developed and implemented the software SacarWeb (Software for Assessment of Cardiac and Respiratory System for Web) [5]. A team of Health Informatics of our university works continuously in this software. Cardiac and respiratory assessment can be made through several types of exams, that have specific material and methods for both, execution techniques and analysis of results (interpretation). Despite of a lot of effort in producing improvements, the software have presented some difficulties of use. Aiming to reduce this difficulties, we have begun to investigate about software usability.

Intuitively the the development's team of SacarWeb begun to apply some fast usability tests with people from the team and to make some design decisions based on what they think that is good about usability. Later, the team evolved to a better approach, trying to base their design solutions in the literature or in rational data about similar systems. That's the phase that the team is encountered today. Unsatisfied with design bad results, generic and poor data, the team decided to evolve to a better approach doing tests with real users and with well defined methodology. The evolution of care about usability from the team is like the one described in [22].

Usability is known for its importance and results that can be reached. It is important because applications designed with usability in mind can achieve better satisfaction of their users, improved performance and better effectiveness [17]. Some studies show that the investment return rate is high [21, 23], the benefits for the users considering ease of learning and the reached objectives are also very high. In the majority of cases it is better to invest in improving usability that in training people to use software [21, 12].

Despite of its importance, usability is known by its adoption difficulties in teams consisting only of software developers or people with deep knowledge in Computer Science [10]. Besides, it is not so common to include this discipline in the undergraduate Computer Science Curriculum [7]. Sometimes the professionals simply do not give the necessary importance to usability, thinking only in working software. Even considering those that think important to adopt usability techniques during the design, it is supposed that such techniques are difficult to learn and use, once they require high knowledge and experience [20].

It seems that this is one of reasons of poor usability in several areas, resulting in low rates of software acceptance and high rates of software rejection. One of these areas is health care applications [2, 15].

Traditional approaches to evaluate usability are well known by their good results. Normally they give rich information about the use of the software and compliance with known patterns. These approaches have high time consuming, because of the amount of data generated by methods to analysis and require more knowledge to be performed, once the methodology of such methods is highly rigorous. One approach example is the full Usability Engineering life cycle, consisting of eleven general stages, where various options of methods are available, including traditional ones. Such approaches can discourage the adoption of usability techniques in teams without or few knowledge about them [20].

There are alternative approaches to traditional methods to evaluate software usability with similar quality of results indicated for people that know the importance of software usability, but do not have enough resources, or the necessary knowledge to adopt traditional methods [20]. One of these approaches is known by Discount Usability [19], that uses only the most important techniques of the full Usability Engineering.

Similar situation is found in the development team of the health care Web application, called SacarWeb [5], at our university. The design and development team consist of people related to computing and physiotherapy, but no one with deep knowledge of usability. Only common sense was used during the interface development and no one study or usability inspection was done in all the application life cycle, because of the lack of knowledge of this area in the members of the team.

After the development of several improvements related to software functionality [8, 9], some difficulties of learning how to use the software remained. Learnability is an important metric in the addressed Web application, SacarWeb, considering high rotativity of users, once most of them are students of physiotherapy. At this point, the team decided to verify the software usability using alternative techniques, like the ones in Discount Usability, aiming to improve the software usability.

Based on this, the objective of this work is to indetify a set of usability problems in SacarWeb in a way to prioritize the future improvements in the application through Discount Usability Engineering.

The remaining of this paper is organized in this way: in section 2 it is presented the main concepts of Discount Usability Engineering; in section 3 it is described the target software of the usability evaluation; in section 4 it is presented the methodology of applying the Discount usability Engineering. In section 5 the results are shown and discussed. Finally conclusion and future work are described in the section 6.

2. DISCOUNT USABILITY ENGINEERING

The Usability Engineering defines quantitatively the usability goals of an application in the beggining of its life cycle. The results are taken throughout the product life cycle [10]. The full process has eleven stages, but an usability effort that does not include all the stages can be successful. This is the aim of the Discount Usability Engineering [20].

Discount Usability is an alternative to traditional Usability Engineering approaches that use only methods that require specific knowledge of usability, complex environments for the analysis and large experience. The purpose of such approaches is to identify usability problems even if the tests are made by people without experience or specific knowledge about usability. Because of this, there is a better chance to be used in practical design situations [20, 19].

The Discount Usability Engineering is based in four methods:

- User and task observation
- Scenarios
- Simplified think aloud
- Heuristic Evaluation

The four techniques are based in both inspection and test techniques. Inspection techniques do not use final users during the tests, test techniques use them. Better results are reached mixing these two kinds of techniques [11].

User evaluation is the basic principle and first step of any Usability Engineering method, and it is maintained in the Discount Usability. The aim of this task is to know the user, or at least to have a feeling on how the product will be used. The users observation must address their individual characteristics, the tasks performed, the reasons to make a task and the users' evolution. This can be achieved in many ways, like visiting the users' workplace, since the observation does not interfere in users' workflow. Because of it, the observer should keep quiet and let the users work as they normally do without interference.

A prototype is a working model built to develop and test design ideas that can be used to examine content, aesthetics and interaction techniques from the perspective of designers, clients and users. Usability professionals test prototypes as they perform typical tasks of the intended use of the product, once prototypes are a good way to find usability problems in early stages of the design process. Prototypes more similar to the final product are called high fidelity, while those less similar are low fidelity [27]. Scenarios are a kind of low level prototype that cut down complexity of implementation. They reduce the level of functionality and the number of features. They simulate only some parts of the application, but for that the user needs to follow a previously planned path [20].

In the simplified think aloud method one user at a time uses the system for a given set of tasks while is asked to think out loud. By verbalizing their thoughts, users allow an observer to determine not just what they are doing, but also why they are doing it. This is hard to obtain in other methods, and can be a rich source of information about the interface [11], pinpointing interface elements that cause misunderstandings. The main differences between traditional and simplified think aloud is the presence of psychologists, user interface experts and videotapes on the traditional method. To analyse these sources of information takes much time and can intimidate developers, for example. The simplified think aloud use only notes taken by the observer about what the user says and does while using the system [20]. A general protocol to perform this test is given by [3].

Heuristic evaluation is a systematic method for the usability inspection of one interface with aim to find problems in its design and it is considered the most common informal method [11]. The heuristics are collections of few usability guidelines that are good to be attended in an application. With them, even non experts can find many usability problems. The heuristic evaluation involves a small set of evaluators using the software and examining the interface, judging its compliance with recognized usability principles found in the heuristics. Its conducted verifying the interface and trying to formulate an opinion about what's good and bad about it. The findings can be completed with the results of the simplified think aloud [24, 18, 20].

As Discount Usability is an approach that aims to optimize the use of resources, it is easy to apply and can achieve results with cost benefit better than traditional methods. This is just because the emphasis is in early and fast interactions with frequent usability entrance.

3. SACARWEB

SacarWeb (Software for Cardiac and Respiratory Evaluation on the Web) is a software used by a physiotherapy research group at Universidade Estadual de Londrina (UEL) responsible for rehabilitation of patients with chronic respiratory diseases [5]. These diseases, in general, need to be monitored continuously to control the crisis occurrence. With the aim to help the professionals to follow their patients individually and study groups of patients according to specific characteristics the SacarWeb was created. Using the software, students, researchers and teachers can teach and learn concepts, keeping a database with patients identity, results of clinical exams, and using it to observe the evolution of patients as well as to extract group of patients with features of interest.

By means the module Patient Assessment, which has the core system functionalities, it is possible to perform data collecting and analysis of patient exams. Each assessment consist of a set of exams. The exams available are grouped by category: Questionnaires to assess quality of life: Chronic Respiratory Disease Questionnaire (CRQ) and St. George's Respiratory Questionnaire Paul Jones (SGRQ) [13]; Tests of pulmonary function: The system provides the use of different equations of normal values, including Brazilian standard [6], American standard due to Knudson [14] and European standard by the Official Statement of the European Respiratory Society [16]; Exams to test exercise capacity: Shuttle Walk Test [25], the 6 Minutes Walking Test [26].

Various iterations for development of new functionalities occurred in the software life cycle, but no one tested the usability. Some of these functionalities were to improve the software use, others to improve the performance [1], [8], but the results were not the expected. Often the users claim for better consistency of the application, like the standardization of some application behaviors and screens, and to make the information about patients or other secondary functionalities more easier to find and use.

4. METHODOLOGY

Only most commom tasks used in SacarWeb were analyzed, because these are the most used in the software. The practical results inspecting the common tasks appears faster than analyzing the full software [20]. These tasks are:

- To insert patient's identity;
- To insert patient's exams results (six different exams);
- To verify patient's evaluation through various evaluations;
- To apply a filter to extract group of patients satisfying certain conditions.

In Heuristic Evaluation, these tasks are analyzed by suitable evaluators that interacts and evaluate the interface and in Usability Evaluation, these tasks are analyzed by evaluators too, but through observation of the users interaction with software. The data inserted in the software were collected from real evaluations with patients, and the printed forms were the same used by the clinical evaluators.

The aims of the usability test was:

- to improve the learning rate;
- to decrease the occurence of user's error.

These objectives were defined, considering that the most of the user are students of physiotherapy, with different grades of knowledge of the specific domain, and also there is a high rotating of users. It is important that the evaluation has clear objectives in order that the evaluator has the correct focus on users and interface during the evaluation [20].

Both categories of users during the evaluations: real users and heuristic evaluators use a similar computer environment, and it was established the following standard for the evaluations:

- Mouse and keyboard compatible device;

- Browser Mozilla Firefox version 3.6.x, with javascript and cookies enabled;
- 1024x768 minimum screen resolution;

In order to avoid external interference during evaluation, it was recommended the following: to minimize the presence or moviment of other people in the room, no ringing phones, do not use other programs or operating system.

By default, Discount Usability recommends the use of scenarios to evaluate the software. In our case, the software its in advanced development stage. Because its not the aim of this work to compare an alternative design, but to evaluate the actual solution, we decided to use the high fidelity prototypes instead of scenarios. The similarity with the final interface is the major characteristic of this prototype kind. By the use of the same methods, interaction techniques and look of the final product, high fidelity prototypes offer more realist interactions. Despite of this, any fidelity level is capable to identify usability problems, and there is no difference in the amount of problems found by the different levels [4, 27]. The decision in this work was made mainly to speed up the evaluation.

4.1 Heuristic Evaluation

The heuristic evaluation was made in SacarWeb by two evaluators, where only one is part of the actual development team. This one has the best knowledge about the application domain among the evaluators, but all the evaluators have few or no experience with heuristic evaluation. They followed the ten heuristics made by Jakob Nielsen [20], that can be applied in the majority of systems. In order to do the evaluation, one application guide was given to evaluators. This guide has information about the evaluation and system's knowledge domain. These informations include how to operate in specific tasks, how to read correctly the printed forms, which tasks have to be done, the tasks order and so on. One software maintainer was responsible for assisting evaluators during the heuristic evaluation, providing information about the knowledge domain or how to operate the application. This kind of information was given only if needed and after the evaluator have exposed the usability problem in question.

For a better understanding of the tasks to be evaluated in the software, heuristic evaluators were asked to perform an interface evaluation with high quantities of data to insert into the system. Each evaluator has introduced data of two patients with three complete sets of exams, in such way that each task was performed at least two times. During and after the evaluation the evaluators were asked to compare the user interface with the heuristics, and filling their findings in an online standardized document with the heuristics and tasks provided, in order to facilitate the application of the method of heuristic evaluation. For each task, the evaluator should only to identify the associated heuristic with the problem detected, to describe it and also to justify why such problem violates the heuristic.

After all the evaluations have been completed, each document with the findings of the evaluators was revised by the same SacarWeb maintainer that helped the evaluators, who compiled one document with all the usability problems found. This document was sent again to all evaluators for a severity measurement, that will guide the development of new solutions. The severity measurement was made using the proportion proposed by Jakob Nielsen [20], where:

- 0 - its not a usability problem at all
- 1 - cosmetic problem only, need not be fixed unless extra time is available in project
- 2 - minor usability problem, fixing this should be given low priority
- 3 - major usability problem, important to fix, so should be high priority
- 4 - usability catastrophe, imperative to fix this before product can be released

If the evaluators need to access the software to remember some interface problem and then rating the severity, they can use the same login and password used during the evaluation.

4.2 Usability Evaluation

The interface evaluation with users was conducted by one person of the SacarWeb development team, using the simplified thinking aloud method. In the tests, the evaluator has taken notes of all the user actions that provided some usability insight and his own considerations based in personal perception. Parallely, the user testing the interface was incentived to share his thoughts and feelings while using the software. During the evaluation all the dialog was recorded to assist in the evaluation. After the user test, the evaluator wrote a document with his perceptions and usability problems found, based in the notes taken, audio recorded and his usability knowledge.

A pilot evaluation was made before the real evaluations with two not representative users. This evaluation was made to calibrate the real test. One change due to this pilot was the data to be considered for learnability and initial performance. Initially, the evaluation analyses the two interactions with exams sets inserted by the user. But the gain of efficiency from the first to the second interaction with interface is much huge, interfering in the point that we want to measure. It was expected that the same behavior occured in the evaluation with real users. The pilot evaluation gave us a notion of time in what users can perform the request actions, what is important, to give an expected time to the evaluation for the real users during the tests.

The real evaluation was made with four potential users of the application, with no previous contact with the software, and basic knowledge related to application domain. The real users were divided by specialty into the system, testing only the exams they know. The number of users for testing is enough, once it is possible to find the majority of interface problems [20]. The tests were conducted only with the evaluator and user, at a room in the laboratory where it is common to use the SacarWeb.

5. RESULTS AND DISCUSSION

The Heuristic evaluations have identified many problems. The two evaluators were capable to evaluate the interface, based on the provided material. The amount of problems found varied with the knowledge of the domain, where the greater the knowledge about the system's knowledge domain, better the results. The interval of problems found were [32-78], with 113 problems found, where 4 (3.67%) of them are common among all evaluators, which resulted in 109 distinct problems found. This is an expected result, because a heuristic evaluation is better when several people conduct it, once more problems can be found [24]. Another factor that can influence the results are the major field of evaluator's knowledge. In this work, there were evaluators from Computer Science and Interface Design. The two kinds can found the same problems, but with different analysis and workarounds. Also, the problems related to specific knowledge domain are important, because they have great validity once they are based in the evaluator's knowledge about some specific topic that he deeply knows. The generated report was a useful resource to development team, because they can suggest solutions for future development and improvement of the system, based in well designed principles. The severity measurement can guide the development, where the problems with greater score will have priority in their solutions against others. Because of these results, it is suggested that the heuristic evaluation can be easily integrated in the SacarWeb's development cycle, and it is expected that when applied during the project the results can be better and the changes in the final software reduced.

During the tests with the real users of the application we found 109 problems. One important finding was that the general places of the system have common problems with different users. Some of them are related to problems with some concepts and terminologies used during the patients evaluation and into the system in different ways. Generally, the problems found aren't major problems, that interfere in the system learning, but some of them can severely slow the user's actions. Despite of these general findings, one major problem was found with one of the system's function, where 75% of the users could not complete the task. Another major finding was the lack of consistency in different parts of the system. Users created some bad and nocive habits for them during the system learning that difficulted the task completion, mostly because of unexpected system answers or behaviors, that were repeated through the entire system. In various places some behaviors were not necessary, but the users felt more comfortable when doing it, even when it slow down the process. One interest finding was the similarity of problems between the pilot and the real evaluation, but in some cases the real users have much better performance than the pilot users. It is suggested that this happened because of the knowledge of the domain that real users have. The performance gains were observed with real users, and also in the pilot evaluation, confirming the importance of the calibration with a pilot study. In the exams, the specific part of the system, we found that the user's knowledge of domain was important, and helped him to evaluate correctly the interface and to input the data, differing from the general users of the pilot evaluation. Despite some problems, and one task that is difficult to learn, the users felt comfortable using the system and they think that it can be useful in their daily tasks, speeding up and easing some process manually done.

Other important finding was the similarity of problems encountered in the heuristic evaluation and in the interface evaluation with users. 22 of the interface evaluations problems' (25.29%) were found in the heuristic evaluation too, but some serious problems were found only in one of the evaluations, showing the importance of the two kinds used in this work. The total of serious problems found in the interface evaluation were 3 (3.45% of total). These problems were the ones that seriously impacted the users in the completion of tasks, terminology or concepts misunderstood that impacted in the comprehension of system or items that seriously slow down the users or estimulate them to redo a finished task. Therefore, the total amount of distinct problems found were 196, the 109 problems encountered in the heuristc evaluation plus the 87 distinct problems of the usability evaluation with real users.

About the two objectives of the usability tests, it is suggested that learning can be much improved after the interface redesign considering serious problems found. As they are related to core concepts for the users, that actualy are difficult to understand, a better interface approach can solve this. With this, it is expected that the amount of users' error decrease, because the software will be easy to use. Another two main problems are consistency and information needed to be remembered by users through some screens. These two points are related to heuristics [20], and a better care of these points in the some parts of the system can improve the user experience through the entire software.

The development team can use the findings of the tests to guide their next development iterations, having some clearly points with major problems to attack, and the severity measurement and users feedback can guide the development. The maintainer that conducted the usability evaluations can guide the problems corrections, because of his knowledge about them, users and system. With this, it is suggested that is possible to integrate the Discount Usability in SacarWeb's life cycle and we have a set of usability problems to correct in their next development iteration.

A related study reported that evaluators, with similar knowledge of Computer Science and usability, as that observed in evaluation of Sacarweb, can differ from a good and a bad design after learn some usability concepts [19]. As expected by [20], people with few knowledge about usability can find usability problems with heuristics and conduct usability inspections. In our case, evaluators with more knowledge, or about the inspections methods, or about the system tested, obtained better results.

6. CONCLUSION AND FUTURE WORK

This work showed the use of techniques proposed by Discount Usability Engineering by the development's team of SacarWeb. Despite the lack of knowledge about usability, the group has learned and used the techniques, applying the knowledge by means of the heuristic evaluation and usability tests. The results of these two tests converged, and are a great source for the future development of SacarWeb. By means of the simplified thinking aloud the users gave us rich feedback about the interface and concepts of the software. All the evaluations were done with people from team, which suggests that these methods can be integrated in the software life cycle.

With the great amount of problems found, for future work we suggest a development of a second design based on the findings of this work and a second usability test, comparing the results that will be found with the ones of this work. A good way to do this task and to have a great impact in the system is to follow the priority given to the problems found after the heuristic evaluation. Another suggestion is the integration of the Discount Usability techniques in this second design phase, with early testing and more clear usability goals defined. With the current objectives reached, it is expected to increase users' satisfaction and better memorability rates in the second design. Another suggestion is the real integration and care with usability through a well accepted and documented approach, like the Discount Usability techniques, in the development cycle of the application, moving our maturity stage on usability from level 3 to 4 [22].

7. ACKNOWLEDGEMENT

We thanks to CNPq for grants, Prof. Dr. Antonio Fernando Brunetto (in memoriam), the expertise in respiratory rehabilitation at Physiotherapy Department (UEL) , Prof. Dr. Fabio Pitta, Nidia Hernandez and Thaís Sant'Anna (from Physiotherapy Department - UEL), by organizing the material of patients, indicating people to participate in the evaluations and by give in the usability test's local, the students of Physiotherapy for the colaboration in the user evaluation and Carlos Henrique Ferraz, for the colaboration in heuristic evaluation.

8. REFERENCES

[1] T. A. Amorim, M. A. de Oliveira Camargo-Brunetto, D. dos Santos Kaster, and F. Ferracioli. Remodelagem do software sacar-web usando técnicas de engenharia reversa e reengeharia de software. *Anais do CBIS 2006*, 2006.

[2] D. Badenoch and A. Tomlin. How electronic communication is changing health care: Usability is main barrier to effective electronic information systems. *BMJ*, 328(7455):1564+, June 2004.

[3] T. Boren and J. Ramey. Thinking aloud: reconciling theory and practice. *IEEE Transactions on Professional Communication*, 43(3):261–278, sep 2000.

[4] A. Coyette, S. Kieffer, and J. Vanderdonckt. Multi-fidelity prototyping of user interfaces. In C. Baranauskas, P. Palanque, J. Abascal, and S. Barbosa, editors, *Human-Computer Interaction – INTERACT 2007*, volume 4662 of *Lecture Notes in Computer Science*, chapter 16, pages 150–164. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2007.

[5] M. A. de Oliveira Camargo-Brunetto and A. R. Rossi. Uma aplicação web para análise de exames cardiorrespiratórios. In *Simpósio de Qualidade de Software - Workshop de Informática Médica: Anais do Workshop de Informática Médica*, June 2005.

[6] S. B. de Pneumologia e Tisiologia. *I Consenso Brasileiro sobre Espirometria*, 3 edition, 1996.

[7] S. Douglas, M. Tremaine, L. Leventhal, C. E. Wills, and B. Manaris. Incorporating human-computer interaction into the undergraduate computer science curriculum. *SIGCSE Bull.*, 34:211–212, feb 2002.

[8] F. Ferracioli and M. A. de Oliveira Camargo-Brunetto. A utilização das recomendações do w3c no sacarweb. *Anais do XVI EAIC*, 2007.

[9] F. Ferracioli and M. A. de Oliveira Camargo-Brunetto. Internacionalização de aplicações web: uma ferramenta para acelerar a tarefa de traduçãoo em um ambiente de aplicaçãoo médica. *Journal of Health Informatics*, 2(2):29–37, apr 2010.

[10] X. Ferre, N. Jurista, H. Windl, and L. Constantine. Usability basics for software developers. *IEEE Software*, 18(1):22–29, Jan 2001.

[11] A. Holzinger. Usability engineering methods for software developers. *Commun. ACM*, 48(1):71–74, January 2005.

[12] J. A. Jacko and A. Sears. 1 edition, sep.

[13] P. Jones, F. Quirk, and C. Baveystock. The st george's respiratory questionnaire. *Respiratory Medicine*, 85:25–31, September 1991.

[14] R. J. Knudson, R. C. Slatin, M. D. Lebowitz, and B. Burrows. The maximal expiratory flow-volume curve. normal standards, variability, and effects of age. *The American review of respiratory disease*, 113(5):587–600, may 1976.

[15] A. W. Kushniruk, V. L. Patel, and J. J. Cimino. Usability testing in medical informatics: cognitive approaches to evaluation of information systems and user interfaces. *Proceedings : a conference of the American Medical Informatics Association / ... AMIA Annual Fall Symposium. AMIA Fall Symposium*, pages 218–222, 1997.

[16] M. R. Miller, J. Hankinson, V. Brusasco, F. Burgos, R. Casaburi, A. Coates, R. Crapo, P. Enright, C. P. M. van der Grinten, P. Gustafsson, R. Jensen, D. C. Johnson, N. MacIntyre, R. McKay, D. Navajas, O. F. Pedersen, R. Pellegrino, G. Viegi, and J. Wanger. Standardisation of spirometry. *European Respiratory Journal*, 25(2):319–338, August 2005.

[17] M. Morandini. *Ergo-Monitor: Monitoramento da usabilidade em ambiente Web por meio de análise de arquivos de Log.* PhD thesis, Universidade Federal de Santa Catarina, November 2003.

[18] J. Nielsen. Finding usability problems through heuristic evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '92, pages 373–380, New York, NY, USA, 1992. ACM.

[19] J. Nielsen. Guerrilla hci: Using discount usability engineering to penetrate the intimidation barrier. Technical report, Nielsen Norman Group, 1994.

[20] J. Nielsen. *Usability Engineering.* AP Professional, 1998.

[21] J. Nielsen. Return on investment for usability. Technical report, Nielsen Norman Group, 2003.

[22] J. Nielsen. Corporate usability maturity: Stages 1-4. Technical report, Nielsen Norman Group, 2006.

[23] J. Nielsen. Usability roi declining, but still strong. Technical report, Nielsen Norman Group, 2008.

[24] J. Nielsen and R. Molich. Heuristic evaluation of user

interfaces. In *CHI '90: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 249–256, New York, NY, USA, 1990. ACM.

[25] G. E. Payne and J. D. Skehan. Shuttle walking test: a new approach for evaluating patients with pacemakers. *Heart*, 75(4):414–418, apr 1996.

[26] T. Troosters, R. Gosselink, and M. Decramer. Six minute walking distance in healthy elderly subjects. *European Respiratory Journal*, 14(2):270–274, 1999.

[27] M. Walker, L. Takayama, and J. A. Landay. High-fidelity or low-fidelity, paper or computer choosing attributes when testing web prototypes. *Human Factors and Ergonomics Society Annual Meeting Proceedings*, pages 661–665, 2002.