Heuristic Evaluation and Teenagers as Experts

Anastasia Pasiali
University of Central Lancashire
Preston PR1 2HE UK
apasiali@uclan.ac.uk

CHI'13, April 27-May 2, 2013, Paris, France, Europe

Abstract
Heuristic evaluation can be described as a usability engineering method for quick, cheap and easy evaluation of a user interface. This method attempts to detect the problems of a system in terms of usability. Most commonly this kind of evaluation is carried out by an expert. However, this paper proposes a different perspective and asks teenagers to become experts and carry out a simplified form of heuristic evaluation. This evaluation was carried out by teenagers on software designed for children and adults. The aim is to define the difficulties with this method adapted by teenagers and to foresee problems of the system.

Author Keywords
Children, evaluation, heuristics

Introduction
Heuristic evaluation is a method used to recognize possible usability problems of a user interface. This method is described as a comparison between a set of recognized usability principles (the heuristics) and a user interface in order to find the problematic aspects of it and to improve it during the follow up stages of the design [6].

This method of evaluation was originally proposed by Nielsen and Molich 1990 [7], who created a set of usability guidelines to test software. This kind of evaluation is carried out by an expert (or a set of experts) that is not a domain expert or a user or even a

designer but is expert in the use of this method. The method can be equally applied to a final product or even to a prototype.

Compared to the traditional user testing, heuristic evaluation is more subjective because it is based on the evaluators' skills. The main advantage of this method is its cost-effectiveness [1]. It appears to be the simplest and cheapest evaluation method because the evaluator does not need laboratory equipment to carry out the method.

More guidelines than usability

Similar to the usability heuristics developed by Nielsen and Molich, a set of ten heuristics for video game design had developed by Pinelle et al [1]. Moreover, Desurvive et al [2] developed a set of heuristics for playability. There are also other sets of guidelines that test respectively 'fun', instructional design' and 'game playability'. However, the usability guidelines designed by Nielsen and Molich are the ones that have undergone extensive testing and used by many experts.

Problems with heuristic evaluation and children products

Conducting an evaluation method for products for children seems to have some challenging aspects. Although this method has been used frequently by experts in order to test user interfaces for adults, it appears that when it is used for products made for children that the expert should take other aspects into account [5]. Adults and children are two different users with different requirements.

The experts evaluating products for children should have an experience in working with children, since children seem to be more resilient to issues that adults consider possible problem. Macfarlane et al [3] report that problems that might be spotted as quite severe by adults could be totally ignored by the children even if the latter are not able to comprehend the meaning of the wording; it appears to be a minor issue for them. It seems that children succeed to understand without instructions.

Heuristic evaluation and children

A work carried out by Macfarlane et al [3] report that children aged 7-8 are able to distinguish between concepts in a user interface. Although the children are not experts and do not have enough experience in evaluating user interfaces in terms of usability, they have a better understanding of the potential problems a young user may come across when interacting with the system. The work in this paper proposes that teenagers share the skill of becoming experts and are able to evaluate educational software.

The experimental work

A web based French language tutorial, freely available on the BBC website designed for use by children and adults was the product that was given to teenagers to evaluate (http://www.bbc.co.uk/languages/french/lj). The product was evaluated for both usability and fun. Nielsen's guidelines for usability (Table 1) and Malone's guidelines for fun (Table 2) were used to evaluate the system.

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

Match between system and the real world
The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order
User control and freedom
Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
Consistency and standards
Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
Error prevention
Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
Recognition rather than recall
Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
Flexibility and efficiency of use
Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
Aesthetic and minimalist design
Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
Help users recognize, diagnose, and recover from errors
Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
Help and documentation
Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Table 1 – usability heuristics [6]

Challenge
Not too easy, not too hard. Goals should be clearly identifiable. The user should be uncertain about achieving the goal (not too easy – not too hard)

Personal performance feedback should be frequent, clear, constructive and encouraging.
Fantasy
Evoke mental images of objects/situations not present. Appeal to imagination and encourage identification with characters or contexts.
Curiosity
Sensory curiosity supported by variable auditory and visual effects. Cognitive curiosity supported by surprises, paradoxes, humour, dealing with topics that already interest the user.
Control
The environment should be responsive. The user controls the outcome of the system.

Table 2 – fun guidelines [4]

In order for the teenagers to be able to avoid misunderstanding of the meaning of the usability guidelines a rephrasing was considered a necessity. The first step was to simplify the language and to remove the 'usability jargon'. One of the heuristics was split into two for clarity. The revised guidelines for usability are listed in Table 3.

Additionally, the language was simplified for the fun guidelines as well; see Table 4. Some of the fun heuristics were split into two to make the evaluator more confident when carrying out the evaluation.

The software keeps the user informed about what is going on
The software includes the users' language rather than words, phrases or concepts that are not familiar to the user.
Information appears in a natural and reasonable order
Users are able to exit pages and undo mistakes (for example to go back or to start again)
The system is consistent. Users do not have to wonder whether different words, situations or actions mean the same thing.
The software is designed to help users avoid making errors
The user does not have to remember how to use the software. Information required is visible or easily accessed
The software allows frequent users to use shortcuts and adjust settings to suit
The design is simple and there is no irrelevant information
Errors messages are explained in plain language, clearly indicate the problem and recommend a solution
The software provides appropriate help and documentation which is ease to access and relevant to the users' needs

Table 3 – revised usability heuristics

The goals of the software are clear to the user
The tasks are not too easy and not too hard
The user gets frequent, clear feedback that encourages him/her to carry on
The system allows the user to use his/her imagination and identify with the characters and contexts
The software includes sound and visual effects.
The system includes surprises, humour and interesting things for the user
The user feels in control of the system.

Table 4 – revised fun heuristics

The experiment took place at a local co-educational high school during the time of their Information Technology class. The data was collected by 15 teenagers as the first part of a bigger study. The children that were asked to be experts aged 13-14. The experiment was carried out with the present of their teacher. Before the teenagers started to carry out the evaluation method a brief description of the method and an explanation of the concepts of usability and fun were told to them.

The two sets of guidelines were allocated alternatively around the room to avoid collusion. Half of the children carried out a heuristic evaluation for usability and the other half carried out a heuristic evaluation for fun.

The results from this experiment are rather promising. The teenagers seemed enthusiastic to carry out the method from the beginning and contributed all. During the experiment all the children carried out the evaluation with comfort, only 2 pupils needed some extra help with the meaning of some of the wording included in the guidelines. The explanations were given mainly by their IT teacher and sometimes by the researcher. The children spend extra time to explore the software first and then spend 20-25 minutes to carry out the evaluation.

References

[1] Pinelle et all cited in Alsumait, A. Ali-Osaimi, A. 'Usability Heuristics Evaluatuon for Child E-learning applications.' Journal of Software, VOl 5. NO 6. June 2010

[2] Desurvire, H., Caplan, M., and Toth, J. A. (2004). Using heuristics to evaluate the playability of games. In CHI'04 Extended Abstracts on Human Factors in Computing Systems (Vienna, Austria, April 24 – 29 2004). CHI '04. ACM Press, New York, NY, 1509-1512.

[3] MacFarlane, S., Sim, G., and Horton, M. (2005) Assessing Usability and Fun in Educational Software, in Proceedings of Interaction Design and Children 2005, Boulder CO, USA, ACM Press.

[4] Malone, T.W. Heuristics for designing enjoyable user interfaces: Lessons from computer games. In John C. Thomas and M. L. Schneider (Editors), Human Factors in Computing Systems, Norwood, NJ: Ablex Publishing Corporation, 1982.

[5] Markopoulos, P. Read C.J., Mcfarlane, S., Hoysniemy, J. (2008). 'Evaluating Children's Interactive products' [online]. Morgan Kaufman. Available from: http://my.safaribooksonline.com/book//978012374110 [accessed 30 January 2013]

[6] Nielsen, J. (1994). Heuristic evaluation. In Nielsen, J., and Mack, R.L. (Eds.), Usability Inspection Methods, John Wiley & Sons, New York, NY.

[7] Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. Paper presented at the ACM CHI'90 Conference on Human Factors in Computing Systems, Seattle, WA.