"Graphical" Jogthrough: expert based methodology for user interface evaluation, applied in the case of an educational simulation interface

Stavros Demetriadis, Athanasios Karoulis, Andreas Pombortsis

"Graphical" Jogthrough: Expert Based Methodology for User Interface Evaluation, Applied in the Case of an Educational Simulation Interface

STAVROS DEMETRIADIS, ATHANASIOS KAROULIS, ANDREAS POMBORTSIS
Computer Science Department,
Aristotle University of Thessaloniki,
P.O. Box 888, 54006 Thessaloniki, Greece
sdemetri@csd.auth.gr

"Walkthrough" and "Jogthrough" techniques are well known expert based methodologies for the evaluation of user interface design. In this paper we describe the use of "Graphical" Jogthrough method for evaluating the interface design of the Network Simulator, an educational simulation program that enables users to virtually build a computer network, install hardware and software components, make the necessary settings and test the functionality of the network. Graphical Jogthrough is a further modification of a typical Jogthrough method, where evaluators' ratings produce evidence in the form of a graph, presenting estimated proportion of users who effectively use the interface versus the time they had to work with it in order to succeed effectiveness. We comment on the question: "What are the possible benefits and limitations of the Graphical Jogthrough method when applied in the case of educational software interface design?" We present the results of the evaluation session and concluding from our experience we argue that the method could offer designers quantitative and qualitative data for formulating a useful (though rough in some aspects) estimation about the novice-becoming-expert pace that end users might follow when working with the evaluated interface.

INTRODUCTION

The term "evaluation" generally refers to the process of "gathering data about the usability of a design or product by a specified group of users for a particular activity within a specified environment or work context" [7, p. 602]. Evaluation of user interface design is of special importance in the overall software evaluation plan, for two major reasons: first because it concerns exactly that part of the software product which enables users to communicate their instructions to the machine. Evaluation should verify that the interface design delivers to end-users a friendly, intuitive and transparent yet powerful environment for the accomplishment of their goals. Second because evaluation of user interface should be carried out at the right time; early enough to offer designers the chance of getting valuable feedback about their design ideas and possibly proceed to interface redesign, while all important interface characteristics have been designed and are included for evaluation.

Having designed and developed a first prototype of an educational simulation program that we call "Network Simulator", we wanted to obtain early evaluation data concerning the design principles that we followed. We chose to apply the Cognitive Jogthrough method [10], an expert based evaluation method intended for evaluation in the design or early development phase. The method has been reported to be a valuable information source during a system design process [1], is relatively cheap to apply and has already been applied for the evaluation of educational interfaces of a certain kind [2]. Since our main purpose was to produce evidence of interface quality that would take into account the user's gradual familiarization with the interface tasks, we proceeded to modify the experts' questionnaire used in a typical Jogthrough method by introducing a special graph for recording and presenting evaluators' ratings. We call this modified version "Graphical" Jogthrough.

In this paper we try to formulate an answer to the question: "What are the possible benefits and limitations of the Graphical Jogthrough method when applied for the evaluation of educational

software interface design as in the case of Network Simulator?". We describe the characteristic features of Network Simulator, the Graphical Jogthrough method along with the results of our session and we conclude discussing the usefulness and limitations of the method.

WHAT IS THE "NETWORK SIMULATOR"

User Interface Qualities: Intuitiveness and Transparency

Any simulation is a software medium that utilizes the interactive capabilities of the computer and delivers to learner a properly structured environment where user – system interaction becomes the means for knowledge acquisition. A simulation holds a model of some real system and learners generally are expected to discover and understand the properties of the model by giving and modifying input parameters and analyzing the output. The actions that users are allowed to perform and the experiences they are guided to have create a virtual cognitive world which is materialized not only in the internally used model of the simulated phenomenon, but also in the external component of the user interface. The user interface therefore of a simulation software becomes a highly critical element of the overall design not only as far as its perceptual characteristics are concerned but also (and more important) in relation to the cognitive functions that it supports.

There are two (among others) important user interface qualities that designers always should try to keep at high levels:
- intuitiveness (using proper and easily understandable metaphors),
- transparency (not interfering with the learning procedure)

An intuitive interface minimizes "the mismatch between the user's expectation of what an interface object should and should not do, based on their previous knowledge, and what the interface object actually does" [7, p.147]. Selecting suitable metaphors allows users to quickly become accustomed with the way of accomplishing tasks, easily remembering and effectively using interface options in order to reach their goals. A transparent interface respectively would help learners concentrate on "what to do" (accomplishment of the educational tasks) and not "how to do" it (trying to figure out how the interface functions) [9]). In a simulation program, where the majority of available interactions are not merely navigational (guiding users to browse through content) but learning oriented (offering user the ability to alter the system state and learn by observing the subsequent system response), the above mentioned characteristics are of paramount importance when quality of learning experiences is considered.

Users working in a transparent and intuitive interface are generally expected to quickly become effective users of it.

So the core idea of the usage of the graph that we are introducing, is that if we have a tool to record estimated pace of user familiarization with a specific interface then these estimations (being simultaneously evidence concerning the interface intuitiveness and transparency) could help us draw conclusions about software usability and make comparisons between various interface design approaches.

The Network Simulator Interface

Network Simulator is an educational simulation software that enables users to build a computer network, make the necessary hardware and software installations and test the network functionality.

These networks may vary from a small typical LAN (Local Area Network) to an Intranet or a WAN (Wide Area Network). The program has been designed for educational use and it may operate in either of two modes: (a) stand-alone or (b) in communication with ISTOS, an interactive learning environment. Details about ISTOS have been published elsewhere [3], but for the scope of this paper it is enough to say that ISTOS is a hypermedia educational program for instruction in the Computer Networking domain, utilizing a case based instructional approach, following primarily Cognitive Flexibility instructional prescriptions [4] [11]. Students, among other activities, have to cope with real world problem scenarios concerning the installation of a computer network under specific conditions and needs. They use Network Simulator in order to design their proposed networking solutions and they submit this solution to ISTOS for further assessment and feedback. In Fig.1 a general view of Network Simulator interface is presented. Users of the simulation manipulate a network much as word processor users manipulate a written document. All common user interface actions (such as "New", Open", "Save", "Save as…", "Rename", "Close", "Cut", "Copy, "Paste") are supported, along with others that are specially designed for the Network Simulator. Users select the "New" menu command to activate on the screen a specific topology or network architecture selecting from the various supported in the simulation. An activated topology is a properly structured grid consisted by a number of cells where the hardware components of the network may be placed. Users have to place the appropriate computer machines, communication media and other elements of the network (network cards, cables, routers, etc.) and proceed later on to install and setup the necessary software so that the initially empty grid is transformed into a fully functional computer network. Network Simulator extensively utilizes the Select – Carry – Install metaphor. Users may select the component they need from the toolbox (a thematically organized inventory), carry it to the desired cell (cursor changes to denote that an item is being carried) and install it by clicking on the empty cell (Fig. 2). When they finish configuring the network they may ask for OSI control, a network functionality control based on the OSI (Open Systems Interconnection) model. OSI control will inform users on any mismatch present in the network at any level of the OSI model, explain the reasons for this mismatch and prompt them to correct it in order to have the network function properly.

Fig. 3 presents a computer in the way that it appears after there have been installed on it a network interface card (NIC) and the Windows 95 software. Student-users may click on NIC and/or software icon and proceed to set the necessary settings for these hardware and software components (such as connecting NIC to cable and/or setting the desired networking protocols for the operational system) so that computer is fully functional in the network.

USER INTERFACE EVALUATION METHODOLOGIES

Walkthroughs and the Jogthrough Version

Interface evaluation of a software system is a procedure intended to identify and propose solutions for usability problems caused by the specific software design. A usability problem may be defined as "anything that interferes with user's ability to efficiently and effectively complete tasks" [5]. Generally speaking an evaluation can be either formative or summative. In a formative evaluation the product is evaluated before it reaches the end user and the objective is to adjust the design to the user's need according to what is already known about the end user profile. A summative evaluation on the contrary involves usage of the software by the end user and tries to identify the level of accomplishment of the targets set in the design phase. Expert based methodologies offer formative evaluation and are reported to be efficient methods that could be applied on system prototypes or even early design specifications [1] [2] [10] [12]. This is an additional advantage of the method since experts are expected to be able to understand interface functionality without necessarily seeing the product completed. Reeves [8] points out that "expert review may be the most frequently used evaluation strategy ... different types of experts can provide different perspectives on the critical aspects of your product". The main idea in expert based evaluation is to present the interface supported tasks to a group of experts who will play the role of would be users and try to identify possible deficiencies in the interface design. Designers guide expert evaluators to a walk through the tasks and offer them an appropriately structured

questionnaire to record their ratings. The original Walkthrough method [6] is a substantially slower proceeding method since the entire recording is done manually. In the Jogthrough version [10] instead a video camera is used for recording evaluators' comments and proposals, thus speeding up the pace of the session. The method is characterized as "cognitive" to denote that the focus is on the cognitive dimension of the user - interface interaction and special care should be given to understand the tasks in terms of user defined goals and not just as actions on the interface (click, drag, etc.). The words of C. Lewis (one of the Cognitive Walkthrough originators) that "the Walkthrough does not identify problems with an interface; it identifies mismatches between system affordances and user goals" [cited in 12] is a clear indication of the cognitive orientation embedded in the method. One should also have in mind that Cognitive Walkthrough is a method originally designed for the evaluation of "Walk Up and Use" interfaces. Users therefore are supposed to be basically novices or infrequent users working with relatively simpler interfaces (e.g. information kiosks). The method though has been applied to more complex interfaces [12] in an attempt to identify needed refinements and augmentations that could turn it into a useful evaluation methodology even under these circumstances. The Jogthrough version has also been implemented for educational software evaluation [1], the kind of software that users may gradually become familiar with through repetitive usage.

A typical Jogthrough session setup

Fig. 4 presents a typical setup of a Jogthrough session. Software to be evaluated is presented on a computer screen (or it could be a simulated software behavior using ordinary slides). The presenter presents the pending task and action to be evaluated. The moderator helps to keep up with the timetable and generally to maintain the pace of the session and the recorder manipulates all recording media and does the note taking. Evaluators finally record their opinions on questionnaires and discuss with the design team issues of interface design

Why "Graphical" Jogthrough?

The original Jogthrough method asks the evaluators to estimate proportion of users that understand and work efficiently with the interface only the first time they work with it. So it is intended for the kind of "walk up and use" interfaces, those that are going to be used once by perspective users (e.g. in an information kiosk). In a different case though, such as an educational application, learners are going to repeatedly use interface supported tasks in various phases of instruction. A properly structured interface being intuitive and transparent enough will enable users to quickly understand how to do things in the virtual environment and allow them to focus only on the learning aspects of their actions. On the other hand a complex or badly designed interface may still offer the means for doing things but users will take longer to understand interface functionality and to remember how to make effective use of it. The novice-becoming-expert pace will be different in the two instances. It is exactly this pace that we wanted to capture in our evaluation and proposed a slightly modified Jogthrough version in order: a) to introduce time variable in terms of user familiarization with the interface and b) to allow evaluators to record quantitatively their opinion. We introduced time variable by designing a graph based answer sheet for the evaluators and asked them to estimate proportions of successfully working users in four distinct phases of user familiarization with the interface:
a) the first time users try to perform a task action,
b) after having performed it few (2 or 3) times,
c) after enough times (4 to 6) and
d) after more than 6 times (see Fig. 5).

These numbers of repetitions were arbitrarily set, aiming simply to offer evaluators a reasonable quantitative representation of what we mean when we use the terms "few", "enough" and "more". In this way evaluators' estimates may be presented in the form of a graph denoting number of users able to

successfully perform interface tasks after some repetitions. The faster this curve reaches the high-end limit, the more intuitive and transparent the interface is evaluated to be.

CONDUCTING THE EVALUATION SESSION

The Graphical Jogthrough Evaluation Session

Our session lasted about 4 hours and in this time interval 6 tasks (analyzed into 20 actions) were evaluated. Initially we gave evaluators a small presentation of the overall interface design and allowed them to have hands-on experience working a little time with it. Would be users of the simulator were described as students who have enough working experience with the "Microsoft Windows 95 ™" environment, know the basics of a word processor and a spreadsheet, have introductory knowledge of basic programming languages such as BASIC and/or Pascal and are familiar with some popular computer games. We invited four experts to act as evaluators in the session. The general advice when selecting evaluators is to form an interdisciplinary team and try to enlist people with Cognitive Science or HCI background [12]. Our evaluators were very experienced in multimedia software programming and development and one of them in the subject matter. Two of them had important educational experience but none of them possessed any special cognitive science or HCI background. The tasks selected for evaluation were representative of the complete set of tasks available in Network Simulator, presenting to evaluators all the major interaction mechanisms available in Network Simulator. Each task was analyzed into a series of actions that the user had to perform for completing it and these actions were defined both in terms of user goals and interface actions that had to be executed (Fig. 6). So we tried to define user goals at the granularity of individual interface actions although this is reported to cause problems in certain cases [12]. The reason for doing this was the limited Cognitive Science background of our evaluators and the fact that they were unfamiliar with the method. We believed it would be helpful to see user goals and interface actions tightly bound, avoiding thus getting involved in considerations that might lead to session delay and failure of completing the evaluation in the time available.

Recording Evaluators' Ratings

Evaluators were handed a booklet that contained the forms to record their ratings. The header of each form included the task and the pending action. The form displayed the five basic questions that the evaluators had to answer. These questions were:
(After Moderator described the action)
a)   How many users will think this action is available?
b)   How many users will think this action is appropriate?
c)   How many users will know how to perform the action?
(After Moderator executed the action)
d)   Is the system response obvious? YES   NO
e)   How many users will think that the system reaction brings them closer to their goal?
When the task was finally completed evaluators were prompted by question (f) to generally comment on the evaluated task.
f)   Do you find the task useful? YES   NO
     If NO then how can it be modified to become useful? Comments / Opinions

The above questionnaire had to be answered for every action of each evaluated task. No alternative or auxiliary actions were considered for evaluation. In a typical Jogthrough session such as those described in [1] and [10], evaluators answer questions a, b, c, and e by using a simple arithmetic code: 0=very few, 1=less than half, 2=more than half, 3=nearly all, NS/NC=Doesn't Know/No answer. Their ratings define a number of potential users without considering the process of user becoming experienced with the interface after repeatedly having completed the various tasks. In our session instead each time the evaluators had to answer one of the questions a, b, c, or e, they put a check mark

in every column of the graph presented in Fig. 5, declaring the number of estimated users that could successfully understand interface functionality.

RESULTS

The graph presented in Fig. 5 was the main instrument for recording evaluators' ratings in our session. The hypothesis when using such a tool is that evaluators will be (or at least feel) capable of estimating user proportions in the four distinct phases that appear in the graph. In our case there was no comment or any other indication against this hypothesis, from the evaluators themselves. After a short introduction by the moderator, they were able to comfortably use the graph and denote their ratings by checking on the proper boxes. It soon became a completely transparent recording tool.

In Fig. 7 evaluators' ratings are presented after they have been coded (1=None, 2=Few, 3=About Half, 4=Most, 5=All) and statistically elaborated to obtain the mean scores across all the evaluated tasks. We interpret this final graph as a kind of quality "fingerprint" displaying visually an estimate of how fast the majority of the users would be effectively working with the specific interface. The four curves of the graph focus on four cognitive dimensions of the interaction between human and computer: a) "available" curve: understanding the availability of a specific interface option, b) "appropriate" curve: understanding that a certain option is the appropriate for the action in mind, c) "know to perform" curve: knowing how to interact with the interface in order to perform the specific action, and d) "closer to goal" curve: understanding that the interface response (after user's action) brings user closer to his/her goal. These curves present the expected proportions of users who could successfully understand actions as available, appropriate, getting them closer to their goal or know how to perform them versus the times they would actually have to perform the tasks in order to achieve this level of understanding.

A first observation is that a great number of users (~ 60%) is expected to effectively understand the availability, appropriateness and way to perform the actions even from the first time. Even more users (~ 90%) will understand the first time that system responses are getting them closer to their goals. The overall result is very positive for the Network Simulator since it indicates that evaluators expect the majority of users (90%) to be able to completely understand the task performance after they actually perform the tasks for a few (2 or 3) times. One can see that "Closer to Goal" curve is distinctly higher than the three others. We believe that this might be an indication of evaluators having connected strongly the user understanding of getting closer to his/her goal (and hence their ratings) to the perceptual alteration of the system state (obvious when an action was executed) and not so much to the cognitive process underlying the task (something that would demand them to closer focus on the cognitive aspects of system state alteration).

Apart from these quantitative results, what we see as equally important is that the graph structured evaluators' thought towards the process of user gradually understanding the interface functionality and effectively working with it. This perspective became the cognitive lens through which they tried to formulate their evaluation. Experts, in other words, tried to understand user-system interaction as time-extended and repeated and not only as quick "walk up and use" experience. As a result they were more willing to accept apparent task complexity provided that a) they felt that the proposed design adequately met the educational needs of the simulation and b) the design of the task was evaluated as easily understood after few times of performance.

DISCUSSION

Quantitative Description

We believe that an important aspect of the method is that it presents recorded data as a graphical representation of novice-becoming-expert pace of would be users. This form allows one to easily make concluding remarks that correlate percentages of effectively working users and effort needed to reach certain ability levels. Statements such as "about 90% of target group users are estimated to work comfortably with the software after they have completed tasks for 2 or 3 times" or "the majority of users are expected to understand availability and appropriateness of interface options even from the first time they use it", could be conclusions based on the graph results. In a case where a group of experts evaluates a series of interfaces then the resulting graphs might offer the basis for reaching a final decision.

Employing therefore the Graphical Jogthrough approach allows evaluation to include the time variable and offer more appropriate quantitative results in cases where gradual user familiarization with the interface is an important factor of the overall user – machine interactivity.

Structuring Experts' Review

Reeves [8] stresses the fact that "it is often useful to structure the expert's review so that you are assured of getting the types and depth of information you desire". This is what we have accomplished by using the graph based answer sheet. But as we mentioned previously, the graph did not only offer the chance for recording quantitative data, but also set the general framework for the evaluators to follow in the evaluation process. So their comments and proposals were based on this approach, i.e. taking into account users' repetitive usage of the interface and therefore their gradual familiarization with it.

We believe that this approach is much more appropriate when design of educational interface is concerned. Especially in the case of an educational simulation interface, one can not possibly follow the same evaluation approach as in the case of a "walk up and use" interface, exactly because the interface tasks that are necessary for the integration of the simulation, may be initially experienced by users as complex but after a few times of performance as reasonably well designed and structured.

The Decisive Role of Evaluators' Expertise and Evaluated Tasks

It is clear of course that the curves of the graph are the product of the interaction between evaluator's expertise and tasks evaluated. More experienced evaluators examining a broader set of available tasks might locate interface deficiencies and result into a graph with slower ascending curves related to those presented in a graph by less experienced evaluators who would examine only a small and possibly not representative portion of the interface tasks. Evaluators' specific expertise may lead them to focus on characteristics they are more familiarized with. As Preece et al. point out "experts are often renowned for their strong views and preferences, in other words, biases. They may concentrate on certain features and virtually ignore others" [7, p. 675]. It is a common advice [7] [12] that care should be taken so that evaluators' team should be as interdisciplinary as possible and that the evaluated tasks constitute indeed a representative group of the overall available tasks. In our case it was clear from the evaluators' observations that multimedia experts focused more on perceptual and organizational characteristics of interface (e.g. appearance and functional characteristics of buttons) while the content expert addressed more subjects related to the simulated characteristics of the domain. Similarly, lack of educational interface design experience might mislead evaluators to propose tentative redesign approaches, something that might also affect their ratings. We came across such a case when the content expert proposed that selection of a specific network interface card might well be unavailable when user is working with a network type where this card is not used, so that interface could support users' decisions by limiting the available options only to the accepted ones. This suggestion was discussed and rejected since it was in contradiction with our pedagogical approach to allow users make mistakes that in the real world would be possible for novices to do.

So although our experience is limited to only one evaluation session we support the view that the resulting curves are certainly biased both by evaluators' expertise and tasks presented and that in order for the graph to be useful for reviewers, it should be accompanied by a description of evaluators' background and evaluated system tasks. This description would set the frame for reviewers for better understanding the kind of the interacting factors that produced the specific graph.

Points of Concern

We believe that one major point of criticism against the approach we described is that experts are capable of producing only rough estimations about the users proportions that the graph asks for and this kind of information does not convey the details that designers need in order to better understand shortcomings of their interface design. Moreover different interface designs could not possibly produce distinctly different curves unless someone compares the curves produced by an exceptionally good and a poor interface. Resourcing to our experience from the session we conducted, we comment this line of thought as follows: certainly the graph asks only for rough estimations since it is based on the questionnaire used in original Jogthrouh method, which too allows for the recording of rough estimations. Nobody, we believe, is expecting that evaluators in their role play could give more precise quantitative measures. But as already underlined, we see as equally important the fact that the graph frames evaluators' point of view towards the process of user getting gradually familiar with the interface. This perspective reflected into evaluators' comments, proposals and ratings is what designers gain (apart from the final graph) by employing the Graphical Jogthrough approach. As far as the second part of the argument is concerned our opinion is that these rough estimations could classify evaluated interfaces into one of three general categories: "successful design" (curves climb fast in 90 – 100 percent area), "some redesign suggested" (curves exhibit a more or less 45 angle slope), and "poor design" (curves remain in the low user percentage area). In certain cases this classification might be enough for decisions to be taken (accept or revise an interface design). It is a matter of further research and application of the method to produce evidence for supporting (or rejecting) the above expressed opinion.

One might of course ask "how do we possibly know that evaluators (whatever their expertise might be) are capable of estimating user proportions that the graph asks for?" Well, obviously we do not. This is only a hypothesis. But what we could do is to cross-examine results of expert session with those produced in an empirical evaluation session by a sample of end users. If the results of the two sessions conclude then this would be a strong indication that Graphical Jogthrough can really produce trustworthy evidence.

CONCLUSIONS

Concluding from our experience we argue that the Graphical Jogthrough method, in spite of the limitations discussed, can offer useful quantitative and qualitative feedback to designers because the introduced graph guides evaluators' review to focus (as intended) on the process of user gradually getting familiar with the interface tasks. It is also advisable that designers should try to understand in depth the interaction between the two main factors of the session (evaluators' expertise and selected tasks), and present a description of them along with the produced results.

Further research could verify the results of such a session by comparing them to the results produced by end users themselves and could also examine other possible forms of the graph for enabling more reliable and useful data recording.

References

1. Aedo, I., Catenazzi, N., Diaz, P. (1996). The Evaluation of a Hypermedia Learning Environment: The CESAR Experience. Journal of Educational Multimedia and Hupermedia, (1996) 5 (1), 49-72.
2. Catenazzi, N., Aedo, I., Sommaruga, L., (1997). The Evaluation of Electronic Book Guidelines from two Practical Experiences. Journal of Educational Multimedia and Hupermedia (1997), 6 (1), 91-114.
3. Demetriadis, S., Pombortsis, A., Bleris, G., Valassiades, O., (1997). Design Issues for Hypermedia Educational Environments: The Case of "ISTOS". Proceedings of ED-MEDIA 97 & ED-TELECOM 97. Calgary, Canada, 1997. Association for the Advancement of Computing in Education (AACE), Charlottesville, VA., p. 1441 – 1445.
4. Jacobson, M.J., Maouri, C., Mishra, P., Kolar, C. (1996). Learning with Hypertext Learning Environments : Theory, Design, and Research. Journal of Educational Multimedia and Hupermedia, 5 (3/4), 239-281.
5. Karat, C., Campbel, R., Fiegel, T. (1992) Comparison of Emperical Testing and Walkthrough Methods in User Interface Evaluation. Proceedings of ACM CHI '92. Monterey, California, May 3-7, 1992, p. 397-404.
6. Lewis, C., Polson, P., Wharton, C., Rieman, J. (1990). Testing a Wlakthrough Methodology for Theory-Based Design of Walk-Up-and-Use Interfaces. Proceedings of ACM CHI '90. Seattle, Washington, April 1-5, 1990, p. 235-242.
7. Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T. (1994). *Human – Computer Interaction.* Addison-Wesley, 1994.
8. Reeves, T.C. (1993). Evaluating technology-based learning. In Piskurich, G.M., (Ed.), *The ASTD handbook of instructional technology* (pp. 15.1 – 15.32) New York: McGraw-Hill.
9. Roth, W., Chair, L. (1997). Phenomenology, Cognition, and the Design of Interactive Learning Environments. Proceedings of ED-MEDIA 97 & ED-TELECOM 97. Calgary, Canada, 1997. Association for the Advancement of Computing in Education (AACE), Charlottesville, VA., p. 1101-1107.
10. Rowley, D.E., Rhoades, D.G. (1992). The Cognitive Jogthrouth: A Fast-Paced User Interface Evaluation Procedure. Proceedings of ACM CHI '92. Monterey, California, May 3-7, 1992, p. 389-395.
11. Spiro, R.J. & Jehng, J. (1990). Cognitive Flexibility and Hypertext: Theory and Technology for the Nonlinear and Multidimensional Traversal of Complex Subject Matter. In D. Nix & R. Spiro (Eds.), Cognition, Education, Multimedia. Hillsdale, NJ: Erlbaum.
12. Wharton, C., Bradford, J., Jeffries, R., Franzke, M. (1992). Applying Cognitive Walkthroughs to More Complex Interfaces: Experiences, Issues and Recommendations. Proceedings of ACM CHI '92. Monterey, California, May 3-7, 1992, p. 381-388.

Author Note