A methodology for developing audio-based interactive environments for learners with visual disabilities

1 author:

Jaime Sánchez
University of Chile

ED-MEDIA 2004

World Conference on Educational Multimedia, Hypermedia & Telecommunications

Edited by
Lorenzo Cantoni
Catherine McLoughlin

Proceedings of ED-MEDIA 2004

WORLD CONFERENCE ON EDUCATIONAL MULTIMEDIA, HYPERMEDIA & TELECOMMUNICATIONS

June 21-26, 2004; Lugano, Switzerland

AACE
Association for the Advancement of Computing in Education
A CONFERENCE OF AACE
www.aace.org

Abstracts

Increasing interest in math,science, and other technology related careers in African American and Latino students(male and female)

Fidel Salinas, Eastern Illinois University, USA

Recent data noted sad, though not surprising statistics regarding demographic data relative to careers in math,science, and technology. Less than 0.01% of these jobs are held by African American or Latino males and/or females.The purpose of this investigation was to explore current career information provided AfricanAmerican and Latino students and to develop teaching strategies and instructional materials in the early grades in math and science that will sustain student interest and possibly lead to career choice in the ever growing demand for qualified men and women in math,science,and technology.

Towards a New Generation of Research Regarding the Influence of Multimedia on Learning

Haido Samaras, Thanasis Giouvanakis, Despina Bousiou, Konstantinos Tarabanis, University of Macedonia, Greece

Multimedia learning resources ranging from Cd-ROM titles to Internet applications have proliferated in the last three decades. At the same time empirical research has tried to provide evidence in favor of or against the effectiveness of educational multimedia. Although the results have not been decisive, two generations of multimedia research have contributed to establishing factors that influence effective multimedia design. We discuss central issues that the literature has dealt with and distinguish significant theories, models and principles that multimedia design has been grounded in. We also summarize the conclusions of the existing research, specifying key theoretical issues and research directions of each one of them. Suggestions are made regarding future trends of multimedia design and learning research.

Reusable Web Services For Building Web Based Learning Applications

Demetrios Sampson, University of Piraeus, Greece

The reusability of software components and services in different educational contexts and across different platforms is a challenging goal in the development of web based learning applications. Until now, systems have been developed from scratch without allowing the addition of third party services in order to extend their functionality. Although many approaches have been proposed, usually they do not support novel web standards that have been recently adopted. In this paper a flexible framework that provides a generic architecture for on demand service delivery for structuring flexible web based educational systems in different contexts by assembling existing components is presented with the objective to fully take advantage of the current development in Web Services.

How learning object relationships affect learning object contracts: commitments and implications of aggregation

Salvador Sanchez, Pontifical University of Salamanca, Madrid campus, Spain; Miguel A. Sicilia, University of Alcala de Henares, Spain

Learning object design by contract is a proposal for formalization of learning object metadata in order to enhance the design of Web-based educational contents by augmenting their reusability. It basically consists of a formal notation that allows stating, in the form of declarations called contracts, the conditions under which a learning object can be used and the outcomesve" that might be expected from its use. Contracts become a very beneficial instrument when new educational resources are automatically generated from combining existing learning objects. However, automatic composition needs to take into account learning object relationships as they involve some commitments that affect contracts. We find particularly valuable to examine the commitments that aggregation, the most common relationship between learning objects, imposes to learning object contracts. To reach our goal, current metadata information on relations is reviewed by looking for analogies with relationships in OOP.

A methodology for developing audio-based interactive environments for learners with visual disabilities

Jaime Sánchez, Nelson Baloian, Héctor Flores, University of Chile, Chile

Educational software has been criticized for not using explicit models to generalize and replicate good practices. Actually almost every educational program has a model, but most of them remain implicit. In this paper we propose a methodology for developing educational software for children with visual disabilities. Multimedia software for these children has some particularities reflected on our model with emphasis on process modeling including learner evaluation and feedback. The model emerges from research on developing educational software for children with visual disabilities and studies concerning the design of educational software for sighted learners. The model was validated by special education teachers and software designers trying the model with five software products based on model heuristics. We illustrate how the model can be used for both designing and developing software as well as to evaluate current software for improvement purposes.

Visualization of Curricula using Multi-Dimensional Scaling

Joseph Sant, Sheridan Institute for Technology and Advanced Learning, Canada

Abstract: Curricula in many institutions of higher learning are represented either textually or graphically as linear sequences of courses to be taken within a program. The use of Two-Dimensional maps generated using Multi-Dimensional Scaling and Information Retrieval techniques is demonstrated as an alternative to more traditional techniques of representing curriculum. The preliminary implementation of a Curriculum Visualization System and the advantages and special problems of using this approach are discussed referencing real-world examples. The benefits of using this approach to compare course offerings within and between institutions and comparisons with human analysis of curriculum are also discussed.

Instructional Design and Implementation of Interactive

Imke Sassen, Björn Paschilk, Stefan Voß, Torsten Reiners, Institute of Information Systems, University of Hamburg, Germany

The usage of interactive learning methods can improve higher education by building modelling and problem solving skills. In this paper we focus on our experience in developing and implementing complex learning tools for teaching and learning in the field of Operations

A methodology for developing audio-based interactive environments for learners with visual disabilities

Sánchez, J. Baloian, N. Flores, H.
Department of Computer Science, Universidad de Chile
Blanco Encalada 2120, Santiago, Chile
{nbaloian, jsanchez, hflores}@dcc.uchile.cl

Abstract. Educational software has been criticized for not using explicit models to generalize and replicate good practices. Actually almost every educational program has a model, but most of them remain implicit. In this paper we propose a methodology for developing educational software for children with visual disabilities. Multimedia software for these children has some particularities reflected on our model with emphasis on process modeling including learner evaluation and feedback. The model emerges from research on developing educational software for children with visual disabilities and studies concerning the design of educational software for sighted learners. The model was validated by special education teachers and software designers trying the model with five software products based on model heuristics. We illustrate how the model can be used for both designing and developing software as well as to evaluate current software for improvement purposes.

1. Introduction

There is no doubt that an underlined principle for designing educational software is that they should make extensive use of multimedia capacities designed in modern computers. This necessarily implies that users with any type of sensory disabilities are probably unable to access to this software because they make intensive use of graphical user interfaces. There has been diverse software developed for disabled people with interfaces relying on those sensorial channels available to the user, for example, using haptic interfaces devices for input and sounds for output. Not many of them have been developed for educational purposes.

Educational software for people with visual disabilities usually lacks of critical interface elements which are commonly present in software for sighted children. Most software does not include things such as an explicit model knowledge and skills learners should construct when using the software, an explicit learner model, and the implementation of appropriate feedback to improve the learners' performance. To many authors designers of educational software for children with disabilities conceive the software with interaction restrictions in their minds, fixing the interaction modes from the very beginning. Educational software for learners with visual disabilities should be designed without taking into account in the beginning the users' disabilities. They should considered important aspects such as a model representing an existing or imagined real interacting world, a model for representing the knowledge learners have to learn, and learner's model. Only when it comes to the point of mapping the inputs and outputs of the models into an interface, the learner capabilities and disabilities should be taken into consideration to map these variables on proper devices.

Since educational software development process depends on people, tools, and methodologies involved, and considering that we have not a clear methodology to carry out this process for children with visual disabilities, the results mainly depends on the skills of the involved people. This can cause many drawbacks typical in a handcrafted process. Software engineering uses methodologies to help to reduce the craftsmanship level of software development by using the best methodological practices.

There have been some proposals for methodologies to develop educational software (Alessi & Trollip, 2001; Soares, 2001; Dillenbourg & Self, 1992) and courseware (Baloian et al., 2001). We considered some elements of these methodologies and adapt them to propose a new methodology for developing educational software for children with visual disabilities. The aim of this methodology is to assist developers in considering critical components for educational software design.

Diverse software has been developed to allow interaction with virtual environments by users with visual disabilities. They are based on the presentation of graphic information by text-to-speech translation that reads Web pages displayed through browser and three-dimensional spaces of navigation environments with sounds that can get close, far or move to mentally represent the space (Mereu & Kaznan, 1996) and to develop cognitive skills (Savidis et al., 1996). This can be seen in Morley et al. (1998), where blind people develop a special way of navigating through a known environment and represent spatial structure with cognitive difficultness. The system was developed to be used with different output devices such as concept keyboard, tablets, switches, and tactile interfaces (Lange, 1999), and with Force feedback (Ressler & Antonishek, 2001).

The HOMER UIMS was produced by Savidis and Stephanidis (1995, 1996) developing dual interfaces to integrate blind and sighted learners. HOMER integrates visual and non visual interaction with objects and their relationships. The browser BrookesTalk reproduces a Web page by using synthesized voice with words, sentences, paragraphs, and offering different points of view of the page to simulate scanning (Zajicek et al., 1998).

A game for audio concentration is presented by Roth and Petrucci (2000). It consists of pairing different levels of geometric figures, basic, and derives. To represent geometric figures graphically they constructed a bi-dimensional sound space. This concept allows graphic representation such as icons to be represented by the perception of moving sounds in the spatial plane. Blattner and Brewster introduced "earcons" as non verbal audio messages to provide information to users about computer objects, operation, and interactions (Blattner et al., 1998; Brewster, 1998). Each dimension corresponds to a musical instrument and the points of the plot correspond to pairs of frequency in a scale. The horizontal movements from left to right are equivalent to a frequency variation of the first instrument and the vertical movement to frequency variations of the second one.

For people with visual disabilities the software tends to transform graphic information to a haptic format or audio. AudioDoom (Lumbreras & Sánchez, 1999; Sánchez, 2001a) allows blind children to explore and interact with virtual worlds by using spatial sound. The game was based on the traditional Doom game where the player move through corridors discovering the environment and solving problems simulated with objects and entities that inhabit a virtual world. VirtualAurea (Sánchez, 2002) was developed after it was proved that sound-based virtual environments can help to develop tempo-spatial cognitive structures of blind children. VirtualAurea is a spatial sound tool editor that can be used by parents and teachers to design a wide variety of spatial maps such as the inner structure of a school, classrooms, corridors, and diverse structures of a house. Users can integrate different sounds by associating them to objects and entities in a story.

3. Process modeling

The methodology we propose is based on the following hypotheses:

- The knowledge and skills a learner has to develop with the aid of the software are measurable and can be represented. This implies that a learner's model can be constructed
- The software represents an interactive environment (real or imagined) for learning. This means that the software allows the construction of knowledge by the learner.

The "workflow" for developing educational software for learners with visual disabilities proposed by this methodology is depicted in the Figure 1. To explain this methodology we use AudioDoom (already introduced in chapter 2). Normally, a software development process starts with the definition of software requirements. In this case, the requirements are represented by the learning goal that in AudioDoom is the ability to create a mental model of the surrounding environment. According to the learning goal, an appropriate scenario should be conceived to allow learners to develop a skill or knowledge. In AudioDoom the idea was having the learner to discover a labyrinth full of sound emitting objects and entities. The next step is modeling the environment. At the same time, and based on artificial intelligence strategies, the learner's knowledge should be modeled (Baloian et al., 2002). Developing and describing a model has its own process (Zeigler, 1976). The result of this step is a formal model description in paper for both the learning environment and the learner's knowledge. A computer program has to implement this. At this point, it is important to consider if it is plausible to develop an (or use an already existing) editor for generating different environments, such as editors for constructing different labyrinths, instead of having a single environment "hardwired" represented by the program. After this process, model input and output variables are clearly identified. Then we need to map or project them on input/output devices suitable for children with visual disabilities. Our experience in doing this process is summarized in the chapter 5 (guidelines) of this paper.

As we see in Figure 1, cognitive goals will not only influence the definition of learning environments but also the generation of metrics for evaluating knowledge acquisition by the learner. This will be discussed in more detail in the following chapter. Then learners should explore the environment as a way of testing. Test results should be evaluated through usability methods and determine the effectiveness/impact to help learners achieve the cognitive goals. The evaluation may cause a revision of the real world representation, the model, and the interface. Revisions of the real world and its modeling can be mostly caused by the failure of the software's effectiveness in supporting learners to achieve the cognitive goals: the environment does not provide adequate learning activities and the model does not implement them properly. Revisions of the interface (projection of the input/output values on adequate devices) can be caused by usability drawbacks in the software.

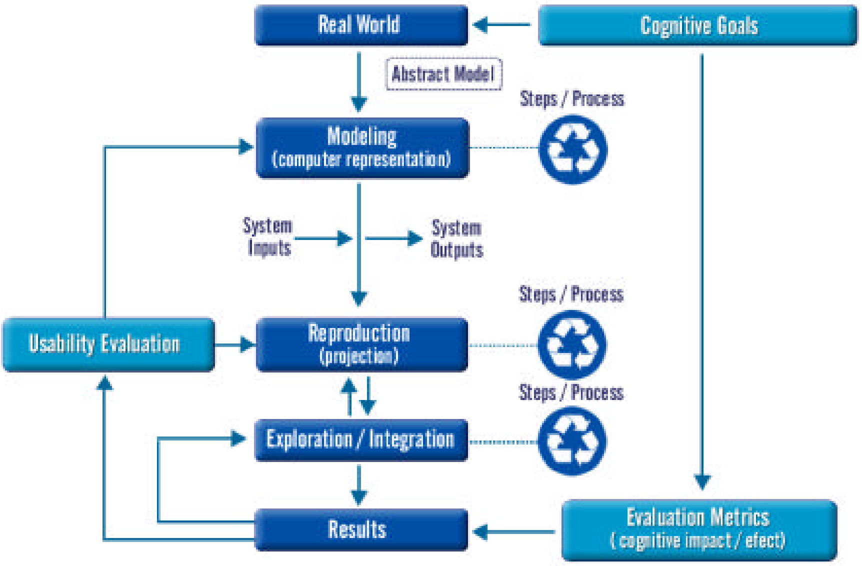

Figure 1. The workflow for developing educational software for learners with disabilities

4. Modeling the resulting architecture

Figure 2 represents the architecture of the resulting software. The model architecture contains components and formal modeling.

4.1 Model components

Metaphor of the real world (Model). According to the cognitive skills to be developed real world metaphors are designed as well as the activities that learners have do to attain the cognitive goals by considering their interest and motivation.

Editor. Tools to construct an internal model based on 2D/3D graphic representations or auditory representations. For learners with visual disabilities models can be generated by drag and drop actions on icons and images from a gallery. The equivalent representation of entities is in a text form or phonetic transcription. Children with visual disabilities choose internal world objects through sensitive tablets or haptic devices.

Computer representation of the real system. This component corresponds to the computer representation of the problem or the real world metaphor, it is the knowledge modeling. Here are functions, parameters, and variables of the system state describing the situation of the represented world and how the transition from one state to other will be made by considering the interaction of the learner with the software and reflected on the entry variables

Strategies. This component provides the strategies to be used to model the state of the learner's knowledge. They are taken from the field of artificial intelligence applied to intelligent tutoring. Some of them are the Overlay model (Kass, 1989) that treats the learner's knowledge as a subset of an expert knowledge. The Differential model (Clancey, 1987) extends the previous model by dividing the learner's knowledge in two categories: knowledge that the learner should know and knowledge that is not expected to be known by learners. The Perturbation model (Kass, 1989) supposes learners should posses a potentially different knowledge in quantity with respect to an expert. The model can represent the knowledge and beliefs of learners beyond the ranks of the expert's model.

Learner model. This component represents what the system thinks about the state of the student's learning in a certain point. It contains knowledge and skill representations the learner should construct, the variables of the learner's state representing the level of learning in a certain moment, and the rules about how to upgrade this information given the interaction with the system and reflected in the change from one real world model state to another. Thus the learner model is given by making inference of the individual knowledge by analyzing the performance (Dillenbourg & Self, 1992).

Evaluation. This component defines the difference between the knowledge model represented in the software and the knowledge model of the learner generated by the strategies. Thus an error measure is produced and projected to the interface as student's feedback.
System projection. This is the main component to certify that the software can be fully assimilated by children with visual disabilities. It is in charge of projecting most interactions, state variables, and feedbacks from and to the software. Below we present the attributes to develop software for children with visual disabilities adequately.

4.2 Formal Modeling

To formally define the model we describe the following functions:

1) $f(s_{t_i}, s_e) \rightarrow (s_{t_{i+1}}, s_s)$

s_t : system state variables in the moment t

s_e : system input variables

s_s : output variables

2) $g(a_t, s_{t+1}, s_t) \rightarrow a_{t+1}$ a_t: state variables of the learner's model in the instant t

This function represents a change in variables describing the state of the learner in the learning process as a result of the last interaction.

3) $E(a_t, a) -> e_t$

a : represents the knowledge or skill to be attained by the learner. This information is internally represented in the software.

e_t : represents the system evaluation of the learner's state of learning.

This function compares the actual learning level of learners represented by state variables of the student including the ideal level to be attained and generating an evaluation measure or error. This should serve to give a feedback to the learner. Feedback and output variables should reflect on values projection of output devices. To project the system elements on the interface we model the following function:

4) $P(s_s, e_t) \rightarrow I \quad and$
$P^{-1}(I) \rightarrow s_e$ I : Interface of the system, output devices

There should be a reverse process. To map the system entrances in values for the input variables of the system. P depends on the input/output devices that can be used with children with visual disabilities. This together with the specification of cognitive goals to be attained is the most differentiating aspect for developing software for learners with and without visual disabilities. To isolate this part from the rest of the design can center the attention of designers on the aspects to be considered to develop educational software for people with disabilities. Thus we avoid the design to be limited from the beginning due to the fact that the interaction with the user is somehow limited.

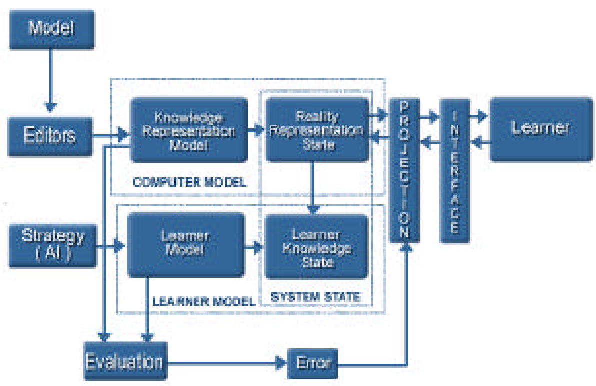

Figure 2. Architecture of the resulting software

5. Model Evaluation

We did test our model with three special education teachers and two software developers. They evaluated the model by analyzing five products to check how well they meet the methodology proposed above. To do this we designed a Likert type scale based on model heuristics. From the model proposed above we defined four major heuristics for evaluation purposes: metaphor, learning, interaction, and interface. Metaphor included adequacy to the mode of learning, how well it represents the model, and how well it defines different interaction environments (editors). Learning includes how the software represents what learners have to learn, evaluates learning adequately, and provides feedback to the learner. Interaction includes if input/output devices are adequate, if users can orient themselves and know what to do and where to go on their own. Interface included font and typography, colors, buttons, icons, audio cues, and feedback used.

Educational Software					
Heuristic	Audio Battle-Ship	Audio Memorice	Virtual Aurea	Theo and Seth	CantaLetras
1. Metaphor	4,8	4,8	4,2	4,5	4,4
2. Learning	3,9	4,4	3,8	4,5	4,8
3. Interaction	4,0	4,3	3,8	3,9	2,9
4. Interface	2,8	4,7	1,9	4,0	2,9

Table 1. Model evaluation results

The results of the model evaluation are presented in Table 1. Possible answers were from "do not meet the heuristic" (1) to "highly meet the heuristic" (5). Average resulting scores were from 3.4 (VirtualAurea) to 4.6 (AudioMemorice), evidencing that most software analyzed meet the minimum standards posed by the model.

From the results displayed we can make four initial conclusions. First, we validated the model by evidencing that using heuristics is a clear methodology for model analysis in educational software. Second, all products considered the heuristics in different degrees. Third, metaphor and learning are the heuristics that best meet the standards of our model. Fourth, interaction and interface were the least attained heuristics. Then our model was initially validated with existing educational software through walkthrough techniques used by teachers and software developers.

6. Discussion and further work

We present a methodology for developing educational software for children with visual disabilities. The model is the result of a need for models to develop, replicate, evaluate, and improve educational software for this population. The model proposed here is a process model with the resulting architecture including ways of evaluating and giving feedback to learners, as well as to set qualitative differences for children with and without visual disabilities. We describe formally and operationally the model and propose some guidelines to design educational software for children with visual disabilities by discussing main generic attributes to include in this software.

The model was tested for viability in educational software design. Interesting results came out when teachers and software developers went through existing educational software for blind children by using some heuristics drawn from the model. Most software did meet the heuristics in different degree. Interactivity and interfaces were the least ranked, meaning that these heuristics need to be carefully considered when design software for children with visual disabilities. Now we need first to improve our heuristics and evaluation instruments, and then apply them to different learning contexts for children with visual disabilities. The next step will be to design and develop software for children with visual disabilities by following step by step the methodology proposed here. Finally, we expect to contribute to the field with an explicit and functional model that can be generalized and replicated to help to improve the learning of children with visual disabilities.

Acknowledgements

This report was funded by the Chilean National Fund of Science and Technology, Fondecyt, Project 1030158.

References

Alessi, S. & Trollip, S. (2001). *Multimedia for Learning, Methods and Development*. Boston: Allyn and Bacon.

Baloian, N., Ochoa, S. & Fuller, D. (2001). A model for component-based courseware development. *Proceedings of the KSI Conference SEKE'01*. Buenos Aires, Argentina, Jun. 13-15. KSI Press. USA pp. 363-370.

Baloian, N., Luther, W., Sánchez, J. (2002). Modeling Educational Software for People with Disabilities: Theory and Practice. *Proceedings of the ACM ASSETS 2002 Conference*.

Blattner, M., Sumikawa, D. & Greenberg, R. (1998). Earcons and icons: Their structure and common design principles. *Human-Computer Interaction*, 4, (1) (1998) 11-44.

Brewster, S.A (1998). Using earcons to improve the usability of a graphics package. *Proceedings of HCI'98, People and Computers XIII*, Sept. 1-4(1998) Sheffield Hallam University, Sheffield, UK

Clancey, W.J. (1987). Methodology for Building an Intelligent Tutoring System. En Kearsley, G., *Artificial Intelligence and Instruction - Applications and Methods*, pp. 193-227, 1987.

Dillenbourg, P. & Self, J. (1992). A Framework for Learner Modeling. *Interactive Learning Environments*, 2(2), pp. 111-137,1992.

Kass, R.(1989). Student Modeling in Intelligent Tutoring Systems –Implications for User Modeling. In A. Kobsa & W. Wahlster, Eds. *User Models in Dialog Systems*, pp. 386-410, 1989.

Lange, Max O. (1999). Tactile Graphics – as easy as that. *CSUN's 1999 Conference Proceedings*. March 15-20, Los Angeles, USA.

Lumbreras, M., & Sánchez, J. (1999). Interactive 3D Sound Hyperstories for Blind Children. *Proceedings of the ACM CHI '99*, Pittsburg PA, USA (1999), 318-325.

Mereu, S.& Kazman, R. (1996). Audio enhanced interfaces for visually impaired users. *Proceedings of the ACM CHI 96*, 72-78

Morley, S., Petrie, H., O'Neill A.-M. & McNally, P. (1998). The use of non-speech sounds in a hypermedia interface for blind users, in Edwards, A.D.N., Arato, A., and Zagler, W.L.(Eds.)*'Computers and Assistive Technology'. Proceedings of ICCHP'98. XV*, IFIP World Computer Congress, 205-214.

Ressler, S. & Antonishek, B. (2001). Integrating Active Tangible Devices with a Synthetic Environment for Collaborative Engineering. *Proceedings of the 2001 Web3D Symposium*. Paderborn, Germany, February 19-22, 93-100.

Roth, P., Petrucci, L., Assimacopoulos, A. & Pun, Th. (2000). Concentration Game, an Audio Adaptation for the blind. *CSUN 2000 Conference Proceedings*, Los Angeles, USA. http://www.csun.edu/cod/conf2000/proceedings/gensess_proceedings.html .

Sánchez, J. (2001a). Interactive virtual acoustic environments for blind children. *Proceedings of ACM CHI '2001*, Seattle, Washington, April 2-5, 2001, 23-25.

Sánchez, J. (2001b). *Visible Learning, Invisible Technology*. Santiago: Dolmen Editions.

Sánchez, J. (2002). VirtualAurea: Perception through spatialized sound. *Proceedings of the 4th International Conference on Disability, Virtual Reality and Associated Technologies*, pp. 67-73. September 18-20, 2002,Veszprém, Hungary.

Savidis, A. & Sthphanidis, C. (1995). Developing dual user interfaces for integrating blind and sighted users: HOMER UIMS, *ACM CHI'95 Conference Proceedings*, Denver, CO (1995) 106-113.

Savidis, A., Sthphanidis, C., Korte, A., Crispie, K. & Fellbaum, K.(1996). A generic direct-manipulation 3D-auditory environment for hierarchical navigation in non-visual interaction. *Proceedings of the ACM ASSETS 96*, 117-123.

Soares, M.P.S. (2001). Modeling educational software for teaching quality control with emphasis on quality evaluation software, MAQS. *Proceedings of the International Symposium on Computing in Education*, Aguas Calientes, México.

Zajicek M., Powell C. & Reeves C. (1998). A web navigation tool for the blind. *Proceedings of the ACM ASSETS'98, 3rd ACM/SIFCAPH Conference. on Assistive Technologies*, Los Angeles, USA (1998) 204-206

Zeigler, B. P. (1976). *Theory of Modeling and Simulation*, Wiley, New York, USA, 1976.