Teaching-Learning Environment Tool to Promote Individualized Student Assistance

Rafael Santos[1], Bruno Nogueira Luz[1,2], Valéria Farinazzo Martins[3], Diego Colombo Dias[1], and Marcelo de Paiva Guimarães[1,4](✉)

[1] Programa de Mestrado em Ciência da Computação, FACCAMP, Campo Limpo Paulista, Brasil
rafael@renovaci.com, {bnogueira.luz,diegocolombo.dias,marcelodepaiva}@gmail.com
[2] Instituto Federal de São Paulo, São Paulo, Brasil
[3] Faculdade de Computação e Informática, Universidade Presbiteriana Mackenzie, São Paulo, Brasil
valfarinazzo@gmail.com
[4] Universidade Aberta do Brasil, UNIFESP, São Paulo, Brasil

Abstract. To develop an effective teaching-learning process for a group of students respecting their individual learning pace is a challenging task for teachers. To assist students individually, it is necessary to identify each student's difficulty and take appropriate teaching action. This paper presents an assisted learning tool based on the web that monitors and reports the student's learning behavior for the teacher. This tool, called eTutor, also performs preconfigured actions (i.e., displays a video or text) according to the current state of student learning. We tested this tool in two different topics for two groups of students. The evaluation showed that this tool promotes student assistance, helping the teachers to be closer to their students.

1 Introduction

Learning is a fundamental and challenging activity for human beings, providing satisfaction for all, especially when students take ownership of knowledge and apply it in different contexts [1]. Each student has a different learning pace [2, 3], so persistence is necessary to create situations that allow the students to achieve mastery of a practice or understanding of a subject [4]. Generally, students who actively participate in classes (i.e., exposing doubts, trying to complete exercises, not being absent, and not giving up) reach the learning goal at some point. Teachers can maximize the students' ability to learn by assisting them directly, regardless of their physical proximity (either through distance learning or classroom teaching). This is one of the assumptions emphasized by authors who discuss the student's zone of proximal development [5, 6].

Due to the disparity of the learning pace of each student, it is essential that teachers be aware of students' learning behavior (i.e., when students are using a learning web tool, it is important to know whether they are interacting with the environment). Teachers can carry out several actions in order to enhance learning when they know

O. Gervasi et al. (Eds.): ICCSA 2015, Part I, LNCS 9155, pp. 143–155, 2015.
DOI: 10.1007/978-3-319-21404-7_11

the students' learning behavior. For example, they can resolve questions, adapting their strategies to the context, taking individual questions and sharing with everyone; most student questions are common to all. However, monitoring the students' learning behavior is a challenging task for teachers, because it is influenced by several variables, such as the number of students per class and the differences in the learning pace of each one [2, 7].

This paper presents the assisted learning tool, eTutor, which aims to promote individual assistance to students, whether reporting their learning behavior to the teacher in real time or not (either in distance or classroom learning). This web-based tool also helps the development of learning situations, tailoring content to the students (i.e., offering a video or text). The eTutor tool was designed to allow the teacher to provide private assistance to each student during a course with small groups of students (10–30). We maintain that students who work with private assistance can learn more quickly than those in a typical classroom. Reiser, Anderson, and Farrell [8] show that students with private human tutoring could learn approximately four times more than students attending traditional classroom lectures.

Technological tools can monitor students' learning and offer some help to them, for instance, providing texts, multimedia, and simulations. Several intelligent tutoring systems have been proposed [9, 10, 11, 12, 13, 14, 15, 16], which are computer applications to assist human teaching. Commonly, these tools do not aim to provide details about students' learning behavior to teachers; they try to adapt the material content to each student. In our system, information about the students' learning behavior is collected, interpreted, and delivered to the teacher. This can help not only to highlight problems during the teaching-learning process but can also hint at complementary content. Information analyzed by eTutor allows teachers to be closer to their students. This approach allows the teacher to act at the moment when a student experiences difficulty in working on an exercise. It also helps the student to become confident.

The eTutor tool has a Student Modeling Module that uses fuzzy logic [17, 18, 19] to trigger actions according to the actual student's learning behavior and the status of the Interactive Learning Objects (ILOs). ILOs are a set of items created, having a clear educational purpose and including the following features: reusability (can be adapted), interoperability (can be supported by any system), accessibility (can easily be stored and retrieved), manageability (can be updated over time) and interactivity (can provide the items' status and automatically generate actions). These objects are an extension of the Sharable Content Object Reference Model (SCORM) [20] of a traditional learning object. The eTutor tool provides a web interface to create, manage, and distribute ILOs among the students.

The paper is organized as follows: the introduction explains the motivation and idea behind eTutor; section 2 presents eTutor's architecture and the details of its implementation; section 3 shows the methodology used to test our tool; Section 4 shows tests performed with this tool, as well as the results obtained; and finally, the implications of this work are discussed in section 5.

2 The eTutor Architecture

The purpose of eTutor is to bring teachers closer to their students, allowing each student to follow his or her own learning pace. Figure 1 depicts the architecture overview of eTutor:

- Teacher Web Interface, which allows teachers to create and/or modify ILOs and teach distance learning or present courses;
- Student Web Interface, which allows students to attend the classes;
- Student Modeling Module, which uses the following, to monitor the student's learning behavior:
  - Interactive Learning Objects (ILOs), which consist of educational content that reports its current status and engages in preconfigured actions;
  - Fuzzy logic, which analyzes the ILO status to trigger preconfigured actions (for example, by sending an alert to the teacher and/or suggesting a video to the student);
- Web technologies, which are the back-end solution implemented.

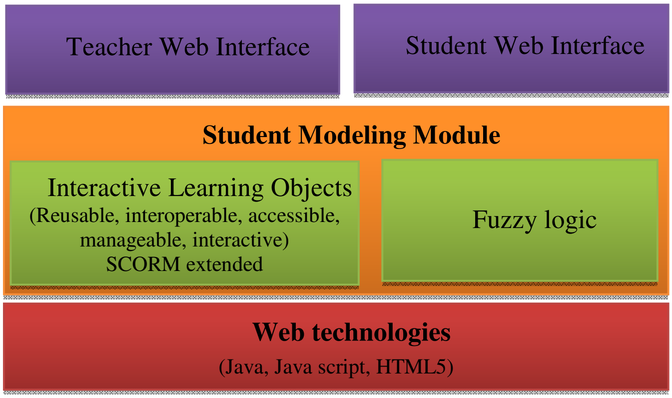

Fig. 1. eTutor - Architecture overview

2.1 Web Technologies

Recent advances in web technologies have allowed sophisticated applications to be written. Major browsers currently support HTML 5 [21], which makes it the de facto markup language for developing complex web applications. This made it the chosen language in which to develop the eTutor. Currently, it is possible to run complex applications on the web and even on mobile devices, high quality image and video and which are highly responsive, with immediate feedback. With HTML 5 and hardware improvement (portable and desktop devices), a whole range of new applications can be created.

All ILOs and eTutor information (i.e., information about students, teachers, and courses) is stored in an internet server and is accessed through an HTTP REST protocol. Both the website and the applications (student and teacher interface) use this same protocol, keeping the back end simple. The response is sent in the JavaScript Object Notation (JSON) standard, which is a way to store information in an organized way, completely language independent, and in a lightweight data-interchange format.

2.2 Student Modeling Module

This module reflects the actions of the teacher and student interface, generated from the analysis of the students' learning behavior. The two independent components work together to monitor the environment.

2.2.1 Interactive Learning Objects

The advancement of the Internet has driven several mechanisms capable of providing learning and knowledge through sets of didactic educational materials that are reusable and shareable, such as the Learning Objects [22]. According to L'Allier [23], a Learning Object is defined as "the smallest independent instructional structure that contains a target, i.e. a learning activity," which consists of a digital component, the basis of a course, unit, or lesson that can be reused to create other unique instructional structures. A Learning Object's main features are as follows: 1) flexibility; it is designed to be reused; 2) personalization; customization of content allows Learning Objects' recombination; 3) interoperability; it allows the definition of design specifications' development and presentation, based on organizational needs; and 4) it increases the significance of knowledge; through reuse many times and in different situations, the contents of Learning Objects are consolidated naturally as timed steps.

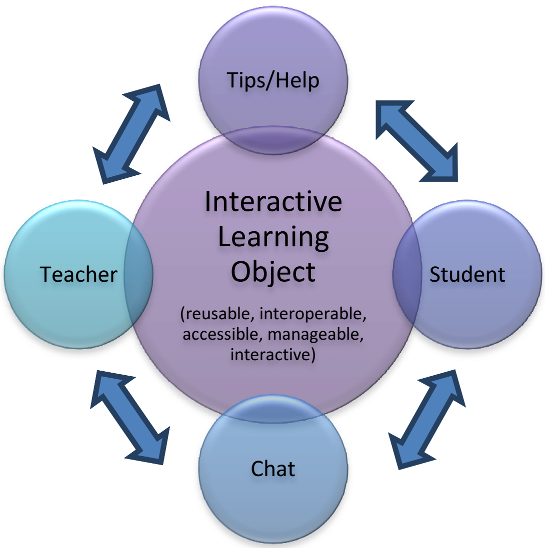

Fig. 2. Operation of an Interactive Learning Object

A Learning Object in this paper is also considered as a unit able to report its current status, which allows a learning tool to trigger preconfigured actions. For our purpose, we expanded the SCORM standard, adding 26 new elements (e.g., period with no interaction, level of difficulty, and help: text, video, and audio). Thus, we propose the Interactive Learning Objects (ILOs) that are focused on interaction. The compatibility between SCORM and ILOs was kept to allow the export/import of objects along with the tools that support them. Kemezinski et al. [24] have already described a methodology for building interactive learning objects; however, they did not provide details about the construction of these objects or whether they follow a standard such as the SCORM [20]. Figure 2 depicts the concept of an ILO. In eTutor, this object is preconfigured by the teacher with some parameters, for example, when the teacher will be alerted that the student is not interacting with the object or when tips such as texts and videos will be offered to the student. The teacher can inspect the student ILO and contact the student at any time.

The created ILOs and their current status are stored on a server. When a class starts, the teacher can select an ILO and send a copy of it to the student's computer, or the students can access the previous ILO.

2.2.2 Fuzzy Logic

The Student Modeling Module uses fuzzy logic to trigger actions, according to the ILO status, that were preconfigured by the teacher. For example, if the student does not add any information within five minutes in a descriptive exercise, then the eTutor offers some tips and how-to videos; if the situation persists, the teacher is alerted. This module is similar to the adaptive model described by Seters, Ossevoort, Tramper, and Goedhart [25], which integrates previous information (such as downtime and tips already used) in order to select appropriate learning content to be presented to the student or which even carries out actions such as alerting the teacher.

This module is based on input and output variables, which represent the final outcome of the student's performance (and the intervention, or not, in the learning process). The following input variable values were used to trigger actions:

- Period of time with no interaction: period during which an ILO goes without receiving interaction from the student;
- Number of characters entered for descriptive answers: percentage of characters that the user typed in terms of the average number of characters per exercise. The number of characters may be low, medium, or high;
- Requests for assistance: number of requests already made by the student, which can be low, medium, or high;
- Exercise difficulty: can be easy, intermediate, or difficult.

Considering these input variables (Table 1), eTutor determines the level of assistance to be offered to the students (Table 2):

- Level 0: no assistance is required;
- Level 1: audio, text, or video are offered;
- Level 2: a similar exercise commented on by the teacher is offered;

- Level 3: the tutor (teacher-designated person who provides help in learning) is alerted about the current status of the student's learning behavior;
- Level 4: the teacher is required to contact the student directly.

There are also penalty factors associated with the levels of assistance used. These penalties are optionally configured by the teacher. Each penalty is associated with a value to be deducted from the grade. The pertinence function used was the trapezoidal, depicted in Equation 1. Tables 1 and 2 present the values for the degree of pertinence, where *a*, *b*, *c,* and *d* are the edges of the trapezoid. The degree of pertinence is a real value in the interval [0,1]. This pertinence function is fundamental to enable the use of fuzzy logic [17, 18, 19].

Table 1. Fuzzy input variables

Input variables		
Variable	**Linguistic terms**	**Relevance degree (range)**
Period of time with no interaction	small	[0,60]
	medium	[58,180]
	high	[170,300]
Number of characters entered	low	[0,10%]
	medium	[9%,30%]
	high	[28%]
Amount of assistance	low	[0,1]
	medium	[1,3]
	high	[2,5]
Exercise difficulty	low	[0,5]
	medium	[5,7.5]
	high	[7.10]

Table 2. Fuzzy output variables

Output variables		
Variable	**Linguistic terms**	**Degree of relevance (range)**
Level of aid	Level 0 – no assistance	[0,1]
	Level 1 – audio, video, and text	[1,2]
	Level 2 – review exercise	[2,3]
	Level 3 – tutor assistance	[3,4]
	Level 4 – teacher assistance	[4,5]

$$trapmf\ (x; a, b, c, d) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right)$$

Equation 1. Trapezoidal pertinence function

2.3 Teacher and Student Web Interface

Each ILO consists of some attributes such as name, internet links, and multimedia objects (audio, video, and text). When an ILO does not receive interactions, eTutor alerts the teacher and offers assistance to the student. Figure 3 depicts the main student interface that allows the students to answer a descriptive question.

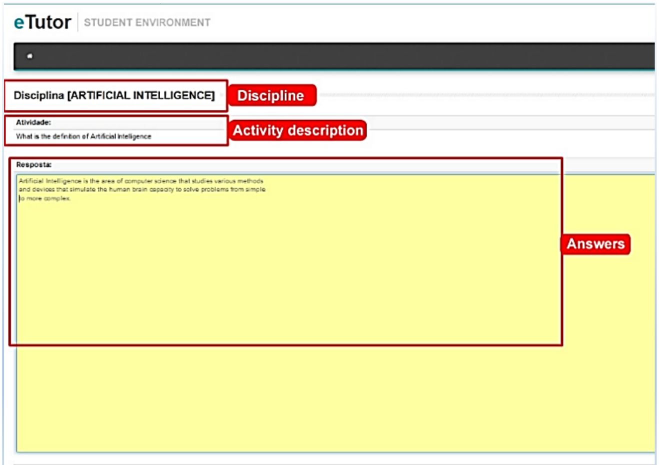

Fig. 3. eTutor: Main student interface

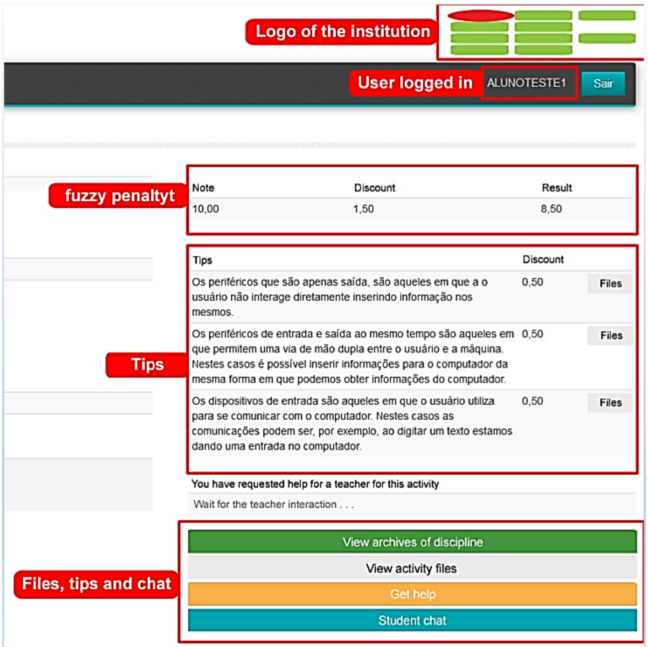

Fig. 4. eTutor: Current status of a student ILO

In order to discourage students from asking for tips when they don't need them, eTutor keeps score and deducts points (optional) for each help request. Figure 4 depicts the current status of a student ILO with the penalties applied, tips requested, and other information, for instance, attached files and chat history.

The teacher uses a specific interface (Figure 5) to monitor the students. In this interface, the teacher can request a copy of a student ILO, which allows verification of the current status. If an intervention is needed, the teacher can contact the student. The teacher interface also has a color system that is associated with the state of student learning.

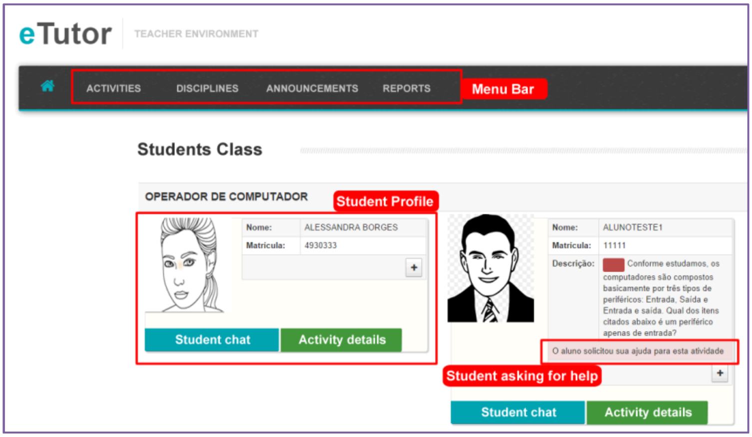

Fig. 5. Teacher's dashboard monitor

One eTutor challenge is to detect the ILO state and trigger the action associated with it. The behavior parameters of each object are set by the teacher at the time of the creation of an ILO. The teacher can also configure the ILO to determine which task the student should do next, linking to the next ILO. This allows the students to start a new activity at their own pace.

While a student attempts to complete an activity that is part of an ILO, the following interactions can occur:

- The teacher can copy the student ILO to his or her interface, which allows inspection of the status. If the teacher decides to intercede, he/she can contact the student (face-to-face in the case of classroom teaching, or via chat/voice, in the case of distance learning);
- The teacher can receive an alert message, notifying him or her that a student is not interacting with an ILO because the student may be facing some difficulty;
- The eTutor can offer some help to the student. The teacher may have preconfigured the ILO, for example, to offer three tips (text content messages) while the activity is being worked on;

- The teacher can share an ILO instance with all students, which allows the start of a discussion about an activity. This can be a new ILO instance or a previous one from a student;
- The students can share their own instances among the other students, promoting collaborative activity, although the teacher should approve;
- Students can request a video or chat session with the teacher/tutor, which may or may not be private. The system manages the requests using a FIFO.

3 Methodology

The eTutor tool was validated in two topics: "Introduction to Information Security" and "Introduction to Hardware and Computation." Both classes were given in different courses and universities. The first course (Class A) had five students enrolled, while the second had twenty (Class B). Before the tests started, the teachers and students were registered on eTutor. Each teacher received training in using eTutor and creating ILOs, including how to follow up the activities. Additionally, we showed the interaction options available to the students.

The teachers were free to create their ILOs as needed to meet their educational goals. In class A, the teacher created one ILO with five questions. In group B, the teacher created one ILO with six questions. Each ILO had at least three tips that were available to students, which were preconfigured to be presented automatically according to input data defined by the teacher and shown by the Student Modeling Module. At the end of the course, both groups answered a questionnaire to analyze the eTutor as an assisted learning tool and with respect to its usability.

4 Results and Discussion

The analysis of the results was performed per class. However, the discussion involves both.

4.1 Introduction to Information Security Class

One teacher and five students participated in the "Introduction to Information Security" class. Answers to the questions given to them after the class were tabulated and are presented in Tables 3 and 4. In all cases, the students received automatic tips from the eTutor environment according to their learning behavior and the preconfiguration done in the ILOs. Only one student felt that the tips had little influence on the outcome; others regarded as great/excellent the interaction provided by the ILOs. All of them had already used other virtual learning environments such as Moodle [26].

Table 3. Student and teacher behavior

Question	Yes	No	I did not request
1- Have you ever used any virtual learning environment?	100%	0%	Does not apply
2- Did you feel accompanied by the professor during the activity?	80%	20%	Does not apply
3- Did you request tips during the activities?	100%	0%	Does not apply
4- When needed, did the teacher offer some assistance?	60%	20%	20%

All students requested tips, and 80% of them judged these tips as excellent. Only 20% of them did not request direct assistance form the teacher during the activities; the other 80% judged the assistance as satisfactory. In total, as shown in Table 4, 100% of the students considered the eTutor tool and the ILOs as Excellent (60%) and Great (40%).

Table 4. eTutor evaluation

Question	Regular	Great	Excellent
5- How do you classify the interaction offered in the activities?	0%	80%	20%
6- How do you classify the eTutor?	0%	40%	60%

4.2 Introduction to Hardware and Computation Class

Three tips were configured for each question of the ILO created. They were to be offered after a period of one minute without student interaction and could also be requested at any time. When the activity was completed, an evaluation questionnaire was given. Tables 5 and 6 present the results.

Table 5. Student and teacher behavior

Question	Yes	No	I did not request
1- Have you ever used any virtual learning environment?	15%	85%	Does not apply
2- Did you feel accompanied by professor during the activity?	85%	15%	Does not apply
3- Did you request tips during the activities?	55%	45%	Does not apply
4- When needed, did the teacher offer some assistance?	65%	5%	30%

Table 6. eTutor evaluation

Question	Regular	Great	Excellent
5- How do you classify the interaction offered in the activities?	25%	30%	45%
6- How do you classify the eTutor?	15%	30%	55%

Considering the results from both classes, the proposal presented in this work had a positive rating of 85% (Great and Excellent). Similarly, the interaction achieved with ILO received a positive approval of 75%. It is noteworthy that only 15% of students had used other Learning Objects in other virtual learning environments. Students who already knew other learning environments reported that using the ILO in eTutor made them feel accompanied by the teacher, and they evaluated the possibilities of interaction as excellent. Of the students, 67% were assisted by the teacher when they requested it, while 33% did not request any assistance. Students who felt accompanied (85%) considered the evaluation of the interaction model offered as great or excellent (83%).

4.3 Discussion

Combining both tests applied to the classes, it is possible to trace a single scenario with respect to eTutor. The results were positive. Although there were significant differences when comparing questions 1 and 3 in both classes, the questions that really aimed to measure the student assistance were 2 and 4, which showed positive results. The direct teaching assistance occurred, on average, in 80% of the activities. Figure 6 depicts the final results.

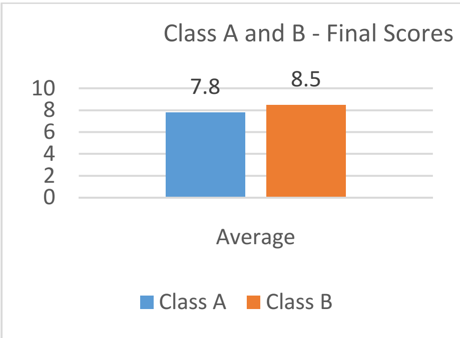

Fig. 6. Final scores, Class A and B

5 Conclusions

This paper presented a tool to assist students based on their learning behavior. This web environment aims to alert the teacher when the students are facing difficulties. It also offers preconfigured help (video, text, and audio). As result, we expected to bring the teachers closer to their students, helping the students to become confident and allowing each one to follow his or her own learning pace, regardless of physical proximity (i.e., whether in distance learning or classroom teaching).

This environment was based on ILOs, which are learning units able to store content as well as their status. We projected the ILOs to be reusable, interoperable, accessible, manageable, and interactive. These objects are an expansion of the SCORM standard to which we added 26 new elements. These new elements allowed the eTutor to report the ILO status to the teacher. The ILOs were created and managed by the teacher, who distributed and monitored them using the teacher interface.

We also presented a case study that used the eTutor in the teaching of two topics, "Introduction to Information Security" and "Introduction to Hardware and Computation." From the observation results, we can conclude that the students felt assisted by the teachers.

As future work, we intend to improve the teacher interface to support other options for multimedia content (i.e., PowerPoint, Flash, augmented reality) for the ILOs. We also intend to investigate how to deal with eTutor usability issues and test this tool with more teachers and groups of students.

References

1. Svinicki, M., McKeachie, W.J.: McKeachie's Teaching Tips: Strategies, Research, and Theory for College and University Teachers. Cengage Learning; 13 edn, 416 pages (2010)
2. Freire, P.: Pedagogia Da Autonomia: Saberes Necessários à Prática Educativa. Paz e Terra, São Paulo (1996)
3. Zabala, A.: A Prática Educativa: Como Ensinar. Artmed, Porto Alegre (1998)
4. Wells, G.: Dialogic Inquiry: Towards a Sociocultural Practice and Theory of Education. Cambridge University Press, Cambridge (1999)
5. Vygotsky, L.S.:. Thinking and speech (n. Minick, Trans.), In: Rieber, R.W., Carton, A.S. (eds.) The collected works of L. S. Vygostky: vol. 1. Problems of general psycology, pp. 239–285. Plenum Press, New York (1987)
6. Vygotsky, L.S.: The problem of age (M. Hall, Trans.). In: Rieber, R.W. (ed.) The collected works of L.S. Vygotsky: (Vol. 5 Child psycology), pp. 187–205. Plenum Press, New York (1998)
7. Guimarães, M.P., Martins, V.F., Dias, D. C.: Uso de Lógica Fuzzy no Auxílio ao Acompanhamento Automático de Alunos utilizando um Ambiente de Aprendizagem. XXIV Simpósio Brasileiro de Informática na Educação (SBIE), doi:10.5753/CBIE.SBIE.2013.707, (2013)
8. Reiser, B.J., Anderson, J.R., Farrel, R.G.: Dynamic student modeling in an intelligent tutoring for lisp programming. In: Proceedings of Ninth International Joint Conference on Artificial Intelligence, pp. 8–14. Morgan Kaufman, Los Altos, CA (1985)
9. VanLehn's, K.: The Behavior of. Tutoring Systems. International Journal of Artificial Intelligence in Education **16**(3), 267–270 (2006)
10. Yang, F.: The ideology of intelligent tutoring systems. ACM Inroads. **1**(4), 63–65 (2010)
11. Chakraborty, S., Roy, D., Basu, A.: Development of Knowledge Based Intelligent Tutoring System, Chapter 5. TMRF e-Book Advanced Knowledge Based Systems: Model, Applications & Research(Eds. Sajja & Akerkar) **1**, 74–100 (2010)
12. Chen, H., Yu, C., Chang, C.: E-Homebook System: A web-based interactive education interface. Computers & Education. **49**(2), 160–175 (2007)
13. Huanga, Y., Lina, Y., Chengb, S.: An adaptive testing system for supporting versatile educational assessment. Computers & Education **52**(1), 53–67 (2009)
14. Chen, C., Chen, M.: Mobile formative assessment tool based on data mining techniques for supporting web-based learning. Computers & Education **52**(1), 256–273 (2009)

15. Yang, Z., Liu, W.: Research and development of web-based virtual online classroom. Computers & Education **48**(2), 171–184 (2007)
16. Chen, C.: Intelligent web-based learning system with personalized learning path guidance. Computers & Education **51**(2), 787–814 (2008)
17. Zadeh, L.A.: Fuzzy sets. Information and Control **8**(3), 338–353 (1965). doi:10.1016/s0019-9958(65)90241-x
18. Bellman, R.E., Zadeh, L.A.: Decision-making in a fuzzy environment. Management science **17**(4), B-141 (1970)
19. Ross, T.J.: Fuzzy Logic with Engineering Applications, 3rd edn. Wiley; 3 edition (March 1, 2010). 606 pages (2010)
20. SCORM. Content Aggregation Model – CAM. 4th Edition SCORM 200 (2009)
21. Sarris, S.: HTML5 Unleashed. Sams Publishing, 1st edn (July 26, 2013).432 pages (2013)
22. Downes, S.: Learning Objects: Resources for Distance Education Worldwide. International Review of Research in Open and Distance Learning **2**(1) (2001)
23. L'Allier, J.J.: A Frame of Reference: NETg's Map to Its Products. Their Structures and Core Beliefs (1997)
24. Kemezinski, A, Costa, I.A., Wehrmeister, M.A., Hounsell, M.S., Vahldck, A.: Metodologia para Construção de Objetos de Aprendizagem Interativos", Anais do 23º Simpósio Brasileiro de Informática na Educação (SBIE) (2012). ISSN 2316-6533
25. Seters, J.R.V., Ossevoort, M.A., Tramper, J., Goedhart, M.J.: The influence of student characteristics on the use of adaptive e-learning material. Computers & Education. **58**, 942–952 (2012)
26. Moodle (2015). https://moodle.org/. Accessed on 20 February 2015