

# A Learning Social Network with Recognition of Learning Styles Using Neural Networks

Ramón Zatarain-Cabada<sup>1</sup>, M.L. Barrón-Estrada<sup>1</sup>, Viridiana Ponce Angulo<sup>1</sup>, Adán José García<sup>1</sup>, and Carlos A. Reyes García<sup>2</sup>

<sup>1</sup> Instituto Tecnológico de Culiacán, Juan de Dios Bátiz s/n, Col. Guadalupe, Culiacán Sinaloa, 80220, México

<sup>2</sup> Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)  
Luis Enrique Erro No. 1, Sta. Ma. Tonanzintla, Puebla, 72840, México  
{rzatarain, lbarron, vponce, rcabada}@itculiacan.edu.mx,  
kargaxxi@inaoep.mx

**Abstract.** The implementation of an adaptive learning social network to be used as an authoring tool, is presented in this paper. With this tool, adaptive courses, intelligent tutoring systems and lessons can be created, displayed and shared in collaborative and mobile environments by communities of instructors and learners. The Felder-Silverman model is followed to tailor courses to the student's learning style. Self Organizing Maps (SOM) are applied to identify the student's learning style. The introduction of a social learning network to create, view and manage adaptive intelligent tutoring systems, and a novel method to identify the student's learning style, are the contributions of this paper.

**Keywords:** Adaptive mobile learning, Social learning networks, Authoring tools, Learning Styles, SOM.

## 1 Introduction

New technologies as Web 2.0 applications have appeared during the last recent years. These novel technologies besides the retrieval of information allow to implement and to hold its control. Moreover, users can perform *harnessing Collective Intelligence [1]* by adding more value to the information as they make use of it. Several well known Web 2.0 applications are, **YouTUBE**, **Wikipedia**, **Yahoo! Answers**, and **Digg**. As a matter of fact, the distinguishing technologies of Web 2.0 are Social Network Sites. A study identifying social learning networks and analyzing seven e-learning 2.0 systems (EctoLearning, Edu 2.0, eLearningCommunity 2.0, *LearnHub*, *LectureShare*, *Nfomedia*, *Tutorom (Eduslide)*), was presented by Ivanova in [2].

According to Murray et al [3] by 2003 over twenty authoring tools for Intelligent Tutoring Systems had been developed. Authoring tools are classified according to the type of tutoring system they help to produce; for example the author tool SIMQUEST [4] produces “simulation-based learning” systems, IRIS [5] creates “multiple knowledge types” systems, and InterBook [6] generates Intelligent/adaptive Hypermedia.

A common denominator in all the named tools is that they are designed only for authors without taking in count the learner.

With the idea of giving learners a more protagonist role, many efforts have been done recently to capture the way students learn in order to model their learning style [7, 8, 9, 10]. The ILSQ questionnaire is used to calculate learning styles with the support of Bayesian Networks, Linear Temporal Logic, or Neuro-Fuzzy Systems. Neither of these reported systems are authoring tools.

On the other side, some authoring tools have been developed with the capability of implementing mobile applications; among them are MyLearning [11], Test Editor [12], or mediaBoard [13]. In these systems, PocketPC orientation, quiz editing or game-based learning are some of the approaches taken. All of them are author tools without the capability to be adapted to the student learning style, and cannot be interchanged among different operating system platforms.

Trying to overcome the particular limitations inherent to each kind of the described tools, we have developed an adaptive learning social network tool named Zamná. The tool is implemented with the regular capabilities of social networks, like new users register, profiles and communities' creation, etc, and with the capabilities of intelligent learning, dynamically adapting, and visualization in mobile devices. A new methodology to identify learning styles, based on Self-Organizing Maps is also introduced. This process is executed whenever a tutoring system is displayed, either in our system or in a mobile device. In general, our system is integrate with components coming from different domains like; Social Learning Networks, Mobile Learning, Adaptive or Web-based Systems, Artificial Neural Networks, and Intelligent Tutoring Systems.

This paper is organized as follows: Section 2 is devoted to explain the Zamná Architecture. In Section 3 we describe the Predictive Engine, and the neural network training and testing. In Section 4 some resulting products are shown. And in Section 5 we present our Conclusions and propose some Future work.

## 2 Learning Styles and Zamná Architecture

A Learning Style model categorizes both the ways in which students learn, and how teachers teach students. There are several proposals on how to approach this problem; to implement Zamná we followed the Felder-Silverman model. The Felder-Silverman model was proposed by Richard Felder and Linda Silverman [15] and includes four dimensions or categories. The four included dimensions are perception (sensory/intuitive), processing (active/reflexive), submission of entry (visual/verbal), and understanding (sequential/global). With Zamná the learning material can be built and adapted to the identified learning style of each learner.

In Figure 1 we describe the architecture of the Web 2.0 tool Zamná. As can be noticed from the diagram, users have access to Zamná through any browser. For any access the user workspace will compose several components, which are; a news section, a user profile, an inbox part, a section of courses, communities, documents,

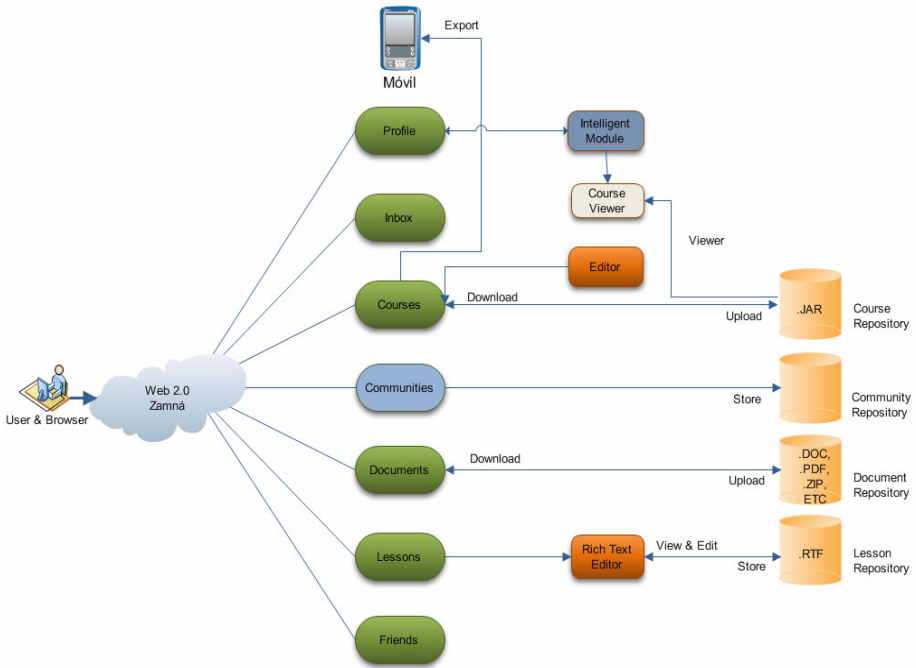


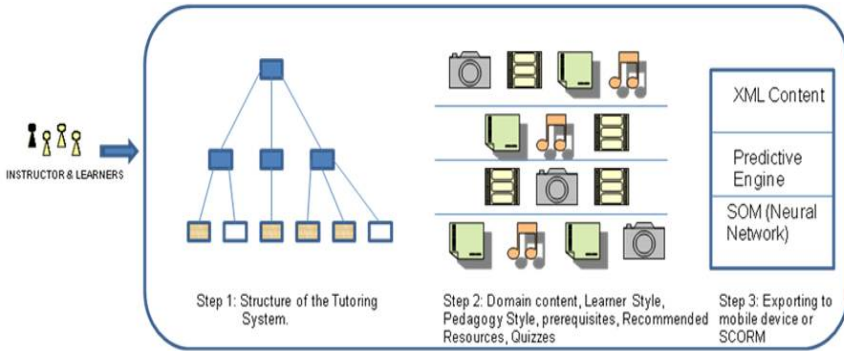
Fig. 1. Zamná Architecture

lessons and friends. To compose the component *profile* Zamná makes use of the intelligent module to identify the user's learning style. The intelligent module makes use of a course viewer for displaying the contents of a course according to the learning style identified by the intelligent module. The course materials are stored in data bases or repositories from where they are downloaded to be exported and studied on any mobile device. Another component is *Communities* which is composed of small sets of networks, each of which is a community focused on a particular area of knowledge for specific purposes. Te communities are also stored in an assigned repository. The same description holds for the component *Lessons*.

## 2.1 How to Build a Course or ITSs

To implement an adaptive intelligent system, three steps have to be followed, as shown in Figure 2. Step1. A tree structure of the adaptive or intelligent tutoring system should be designed by the main instructor(s). On the tree structure, the designer inserts examinations in the form of quizzes (multiple selection and choice). Quizzes are important elements which provide adapting capabilities to the produced tutorials.

Step 2. The tree structure is filled up with the corresponding domain contents, and other learning resources. During the creation, the instructor or teacher compose the



**Fig. 2.** Building a Tutoring System

tutoring system by the introduction of learning objects like text fragments, images, audio/video clips, and by defining learning styles, prerequisites, tags and quizzes. Further on, more learning resources can be included by the same learners, who can also recommend resources they find regularly on the web. Step 3. This step is for saving/exporting packages containing: the learning resources or contents (an XML file), a predictive engine for navigation purposes, and a SOM Neural Network for the classification of learning styles.

### 3 The Predictive Engine

The predictive engine is designed to dynamically identify the student’s learning style every time a tutoring system is run. After the corresponding learning style is identified, and in combination with the student profile, the course contents are selected by an interpreter in the form of learning objects. At any time, the learning style can be adapted as a result of the evaluation applied to the student.

#### 3.1 Identifying Learning Styles

Based on the nature of our problem, for the selection of the learning classifier we needed one with; unsupervised learning abilities, good proved performance, and fast training potential. One model with such capabilities is the Kohonen’s Self-Organizing Maps SOM, which is the one we choose. The identification of the student’s learning styles is a pedagogy discipline, and we have thought that SOM will do the role of an always available pedagogue performing such a task.

#### 3.2 SOM Configuration and the Input Layer

The input data are provided as input signals, which are part of the training data space. The signals are vectors with three components; two are vectors as well, and the other is a scalar value. A description of signal X is described by Eq. 3.1, where the vector  $d_{FS}$  represents the student's learning style, which is identified by the application of the

questionnaire titled *Learning Styles Inventory* proposed by Felder-Soloman [14]. Vector,  $d_c$ , is the learning style used to previously design the learning material which will be studied by the student. And component  $p$  stands for the student's performance, having a learning style  $d_{FS}$  and studying a course designed with the configuration for learning styles  $d_c$ . Stated in another way,  $p$  is taken as the grade of the student in a course offered in a learning style  $d_c$  where the student has a learning style orientation  $d_{FS}$ . Values  $d_{FS}$ ,  $d_c$  and  $p$  were obtained from 47 students in three different courses, previously to the network training.

$$X = [d_{FS} \ d_c \ p] \quad (3.1)$$

A detailed description of vectors  $d_{FS}$  and  $d_c$  is given by Eq. 3.2. As can be seen, both vectors have three elements, each of which represents a dimension of the learning styles identified by Felder-Silverman. The value of each element represents the student's proclivity toward a specific learning style. For the processing of these two vectors plus the  $p$  value, the SOM's topology was implemented as follows; there are 7 nodes in the input layer, while the Kohonen layer is formed by 1600 neurons which are organized in a lattice composed by 40x40 neurons.

$$d_c = d_{FS} = [c_1 \ c_2 \ c_3] \quad (3.2)$$

The SOM designed is then trained to identify learning styles of students. In Eq. 3.3, the structure of the output vector as obtained from the neural network after training or testing is described. The output from the SOM neural network represents the student's learning style.

$$D = [d_c \ p] \quad (3.3)$$

For our experiments the programming language used for the neural network implementation was Java.

### 3.3 A Framework for Training and Testing the SOM

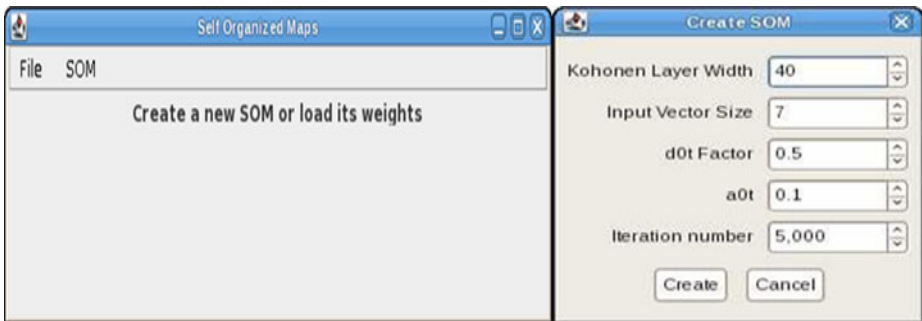
In order to be able to export the trained SOM to mobile devices or web-based learning systems supported by an interpreter to provide intelligent tutoring, a framework allowing the creation, training and testing/validating self-organizing maps is implemented. The framework and the interpreter were developed in a standard and free platform that is Java. From the beginning of this research, another important goal was to have a framework to allow us to implement universal neural networks to be adapted to recognize diverse learning styles from different learning models or theories. As a first effort we are here testing the model proposed by Felder-Silverman, and we are currently working with the multiple intelligences model developed by Gardner.

The experimental training procedure is performed in two stages. The first stage is devoted to the collection of teaching materials. To start testing the system we collected material for three high school courses with the help of high school teachers. These courses were *basic computing* (history, parts of a computer, computer

technology and resources for exchanging information with the PC, etc.), *wind energy* (wind energy story, basic constitution of a wind turbine, wind turbine types, etc.) and *photography* (Introduction, purpose, the film compartment, the shutter, etc.). The teaching material to be added to the intelligent tutorial system was prepared in eight different versions, by following the Felder-Silverman theory. Each version corresponds to each learning style described by the theory. They are the result of the combination of visual-verbal, sensitive-intuitive and sequential-global dimensions.

At this step, we did not consider dimension processing, mainly due to the limitations associated to the nature of mobile and electronic learning. For our study, one version of the course was randomly given to each selected student. Then, they had to study the provided materials during a period of approximately 40 minutes. After the study period the students had to answer an examination, on the studied subject, composed of 20 multiple choice questions. For the same course all the selected students were evaluated with the same assessment. From the results obtained we calculated the performance  $p$  of each student. To identify and register initial learning styles the test *Questionnaire Learning Styles Inventory* was also applied to each student. The initial learning style of any student can change, with the use or, while the student is using an intelligent tutoring system (through the mobile). The expected change is to the student’s actual learning style which should be identified by the ITS through the learning process. For the initial testing we obtained data from 47 randomly selected high school students. The second stage is for the SOM training with the input data obtained in the previous step (student tests). The number of iterations for training the neural network, heuristically selected, was set to 5000.

Two other values, heuristically determined also, were; initial learning rate, with a value of 0.1 and the size of the initial neighborhood, with a fixed value of 20.



**Fig. 3.** Initial and Dialog Window in the Framework

The 40x40 neurons in the *Kohonen layer* of the SOM are shown as a small colored rectangle in the left side of Figure 4. While on the right side of the same figure, the eight zones identified, at the end of training, in the Kohonen layer and corresponding to the eight combinations of learning styles are shown.

After the SOM has been trained we apply a validation process, for which we use 7 parameter vectors, where 4 parameters are used to identify learning styles (three  $d_c$  items to define the style of learning of the material provided to the student, and  $p$  which represents the student performance with that teaching material). Another 3 parameters will contain the outcome response from the neural network. Table one shows some values for the input vectors with different parameters, along with expected values for each vector, used during training, and obtained results, which are the actual output of the network for each input vector.

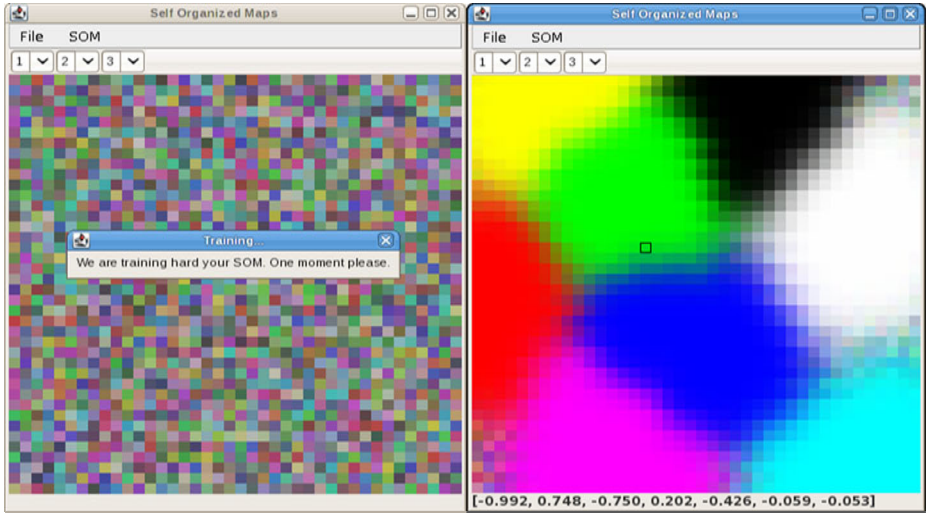


Fig. 4. The Kohonen Layer before and after Training

Table 1. Test Results in the training process of the neural network

INPUT VECTORS			EXPECTED RESULTS				OBTAINED RESULTS		
-1	+1	-1	+0.6	-.8181	-.2727	-.0909	-.6315	-.1447	-.3863
-1	-1	-1	+0.8	-1.000	+.0909	+.2727	-.4896	-.1071	-.0176
-1	+1	+1	+0.4	-.4545	-.0909	-.2727	-.0919	+.1137	-.4328
+1	+1	-1	+0.4	-.8181	+.8181	+.2727	-.3933	-.0844	-.4320
+1	+1	+1	+0.3	+.0909	-.0909	-.4545	-.2765	-.0365	-.4807
+1	-1	-1	+0.3	-.8181	-.0909	-.0909	-.7461	+.0429	+.2300
-1	-1	+1	+0.2	-.4545	+.0909	-.2727	-.7392	-.1687	-.3382
-1	+1	+1	+0.4	-.4545	-.0909	-.2727	-.0919	+.1137	-.4328
+1	+1	+1	+0.3	-.0909	+.2727	+.0909	-.3956	-.4314	+.4301
+1	+1	-1	+0.5	-.2727	+.2727	-.6363	-.5336	+.1870	-.3271
+1	-1	+1	0.0	-.2727	+.4545	-.2727	-.3730	+.0481	+.2212
-1	+1	+1	+0.8	+.0909	+.0909	-.0909	-.3133	-.0021	-.3617

### 3.4 Results Analysis

Each of the three elements in vector  $d_c$ , which is part of the input vector, represents one of the scales of the Felder-Silverman model. These scales in the input vector are codified as follows: the first element corresponds to the scale *Visual/Verbal*, the second to the scale *Sensitive/Intuitive* and the third to the scale *Sequential/Global*. For example, from the first input vector shown in Table 1, we obtain a learning style configuration corresponding to an intelligence of the type *Visual-Intuitive-Sequential* (-1 +1 -1). In the output vectors the first element is Visual/Verbal scale, the second one is Sensitive/Intuitive scale and the third one is Sequential/Global scale. Similarly, when selecting the first vector of the set of expected results, (-0.8181 - 0.2727 -0.0909), it is possible to infer the learning style that the neural network reports as a result (Visual-Sensitive-Sequential). From the results obtained it is possible to carry out an analysis at different levels. For example, if we analyze the hard numbers of the results, which indicate the resulting size for each scale, it can be observed that 16.66% of the results obtained are consistent with the expected results on all scales, and 66.66% in at least two of the three scales. The same values can be interpreted in the domain of the Felder and Soloman [14] learning styles classification. In this case, the numerical values are substituted by linguistic labels associated to the preferences in each dimension (*strong* with a difference between -1 to -.666, *weak* with a difference between -.666 to -.333 and *almost nonexistent* with a difference between -.333 to 0).

According to the results in Table 1, we can notice that 83.3% of the output results are consistent with the expected ones. The distance with a perfect 100 % score is due to considering that no matter which dimension of the scale the student has a preference for, if it is balanced, the preference might vary from one or another dimension. Thus a student with a balanced degree of preference for the visual dimension of the scale Visual/Verbal, as time goes on he can easily switch to a balanced preference for the other dimension (Verbal) of the same scale. Felder and Solomon described the meaning of the adjectives strong, moderate and balanced, related to the degree of student preference for one dimension of each scale.

## 4 SOM and Intelligent Tutoring Systems

The SOM network was tested with the production of some courses to be displayed on cell phones or on the social network Zamná. Figure 5 presents several pictures of a small tutoring system for the topic *Eolic Energy* displayed in a cell phone (first three pictures) and a *compiler course* displayed in the Zamná Social Network Site (fourth picture). Two of the mobile phones display a sample of the learning material for a course in computer networks and the third shows a trace of the students' learning styles (in three and 7 stages) along the course.



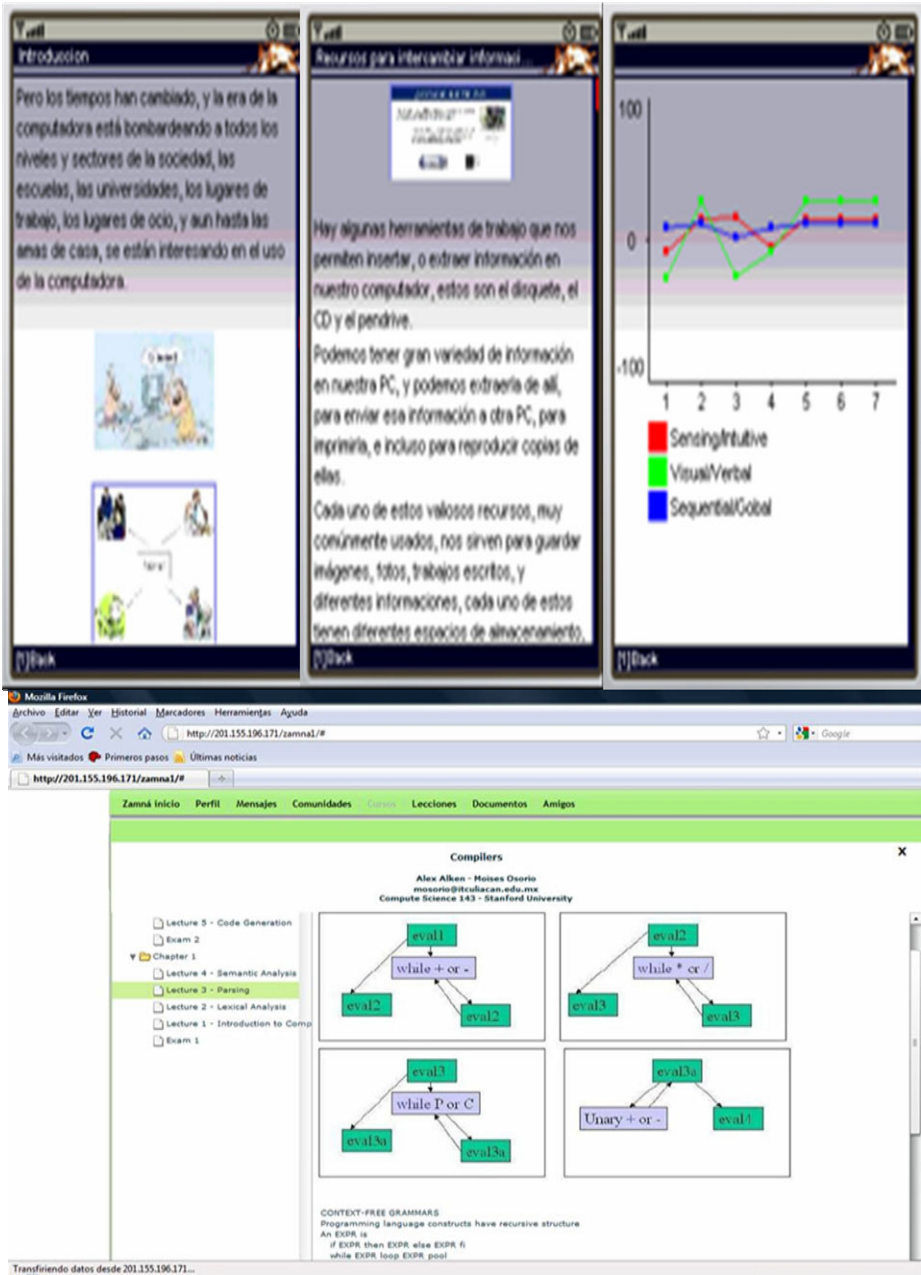


Fig. 5. Intelligent Courses for Computer Networks and Compilers

## 5 Conclusions and Future Work

The results from the performed evaluations of our designed tool show that the current version behaves as expected, showing a strong potential as an ITS authoring/learning tool. At present, our system continues being tested, evaluated and improved. We plan to test the site with groups of students from diverse levels and different areas of study. In addition, we want to test the courses created for mobile devices by following their impact and results of studying actual courses. In this case we plan to first analyze the forms to provide the learning material that is most suitable to be managed in these mobile devices. The IP address of Zamná site is <http://201.155.196.171/zamna/>.

**Acknowledgments.** The work described in this paper is fully supported by a grant from the DGEST (Dirección General de Educación Superior Tecnológica) in México.

## References

1. O'Reilly, T.: What is Web 2.0, <http://oreilly.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
2. Ivanova, M.: Knowledge Building and Competence Development in eLearning 2.0 Systems. In: I-KNOW'08, Graz, Austria, September 3-5, pp. 84–91 (2008)
3. Murray, T., Blessing, S., Ainsworth, S.: *Authoring Tools for Advanced Technology Learning Environments*. Kluwer Academic Publishers, Dordrecht (2003)
4. Jong, T., de Limbach, R., Gellevij, M., Kuyper, M., Pieters, J., Joolingen, W.R.: Cognitive tools to support the instructional design of simulation-based discovery learning environment: the SIMQUEST authoring system. In: Plomp, T., van den Akker, J., Nieveen, N., Gustafson, K. (eds.), pp. 215–224. Kluwer Academic Publishers, The Netherlands (1999)
5. Arruarte, A., Fernández, I., Ferrero, B., Greer, J.: The IRIS Shell: How to build ITSs from Pedagogical and Design Requisites. *International Journal of Artificial Intelligence in Education* 8, 341–381 (1997)
6. Brusilovsky, P., Schwarz, E.: Web-based education for all: A tool for developing adaptive courseware. *Computer Networks and ISDN Systems* 30(1-7), 291–300 (1998)
7. Carmona, C., Castillo, G., Millán, E.: Designing a Bayesian Network for Modeling Student's Learning Styles. In: Díaz, P., Kinshuk, Aedo, I., Mora, E. (eds.) *ICALT 2008*, pp. 346–350. IEEE Computer Society, Los Alamitos (2008)
8. Graf, S., Kinshuk, Liu, T.: Identifying Learning Styles in Learning Management Systems by Using Indications from Students' behavior. In: Díaz, P., Kinshuk, Aedo, I., Mora, E. (eds.) *ICALT 2008*, pp. 482–486. IEEE Computer Society, Los Alamitos (2008)
9. Limongelli, C., Sciarrone, F., Vaste, J.: LS-PLAN: An Effective Combination of Dynamic Courseware Generation and Learning Styles in Web-based Education. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) *AH 2008. LNCS*, vol. 5149, pp. 133–142. Springer, Heidelberg (2008)
10. Zatarain-Cabada, R., Barrón-Estrada, M.L., Sandoval, G., Osorio, M., Urías, E., Reyes-García, C.A.: Authoring Neuro-fuzzy Tutoring Systems for M and E-Learning. In: Aguirre, A.H., Borja, R.M., Reyes-García, C.A. (eds.) *MICAI 2008. LNCS (LNAI)*, vol. 5317, pp. 789–796. Springer, Heidelberg (2008)

11. Attewell, J.: Mobile technologies and learning: A technology update and mlearning project summary. Learning and Skills Development, <http://www.m-learning.org/reports.shtml>
12. Romero, C., Ventura, S., Hervás, C., De Bra, P.: An Authoring Tool for Building Both Mobile Adaptable Tests and Web-Based Adaptive or Classic Tests. In: Wade, V.P., Ashman, H., Smyth, B. (eds.) AH 2006. LNCS, vol. 4018, pp. 203–212. Springer, Heidelberg (2006)
13. Attewell, J.: From Research and Development to Mobile Learning: Tools for Education and Training Providers and their Learners, <http://www.mlearn.org.za/CD/papers/Attewell.pdf>
14. Felder, R.M., Solomon, B.A.: Index of Learning Styles Questionnaire, <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>
15. Felder, R.M., Silverman, L.K.: Learning and Teaching Styles in Engineering Education. *Engineering Education* 78, 674–681 (1988)