

Architecture for an Intelligent Tutoring System that Considers Learning Styles

María Lucila Morales-Rodríguez, José Apolinar Ramírez-Saldivar,
Arturo Hernández-Ramírez, Julia Patricia Sánchez-Solís,
and José Antonio Martínez-Flores

Instituto Tecnológico de Ciudad Madero, División de Estudios de Posgrado e Investigación,
Ciudad Madero, Tamaulipas, México

{lmoralesrdz, jpatricia.sanchez, jose.mtz}@gmail.com,
apolinar_r@yahoo.com, ahr@prodigy.net.mx

Abstract. In this paper we propose the architecture of an Intelligent Tutoring System that considers the student's learning style and the competency-based education. We also describe the processes that have been implemented so far. Our architecture presents innovations in the representation of the tutor module and in the knowledge module; the tutor module incorporates a selector agent, which will choose the content to show, considering the teaching strategies that support the student's learning style.

Keywords: Intelligent Tutoring System, learning style, teaching strategies.

1 Introduction

The idea of applying IT tools to teaching goes back to the 50's, but it wasn't until the 80's when computer-assisted teaching regained a special interest due to the techniques of Artificial Intelligence. At that time, the systems called Intelligent Tutoring Systems (ITS) arose, with the aim of developing the processes of education adapted to the different users/students [1].

An Intelligent Tutoring System (ITS) provides learning and / or customized training to students [2]. The personalized training is an argument on behalf of these systems and it is supported by the analysis realized by Bloom [3]. This analysis discusses the importance of adapting education to each student, confirming the individualized instruction as the most effective way of learning.

In general, an ITS is composed by three modules: the student module, the knowledge module and the tutor module [4]. Nowadays ITS are called Learning Management Systems (LMS) [5]. In our project, the LMS Moodle will be used.

The purposes of this paper are: a) to present the integration of an intelligent agent in the tutor module of a LMS, b) to describe the implementation of modules that have been developed.

This paper is structured as follows: section 2 describes the concepts related to intelligent tutoring systems. Section 3 deals with issues of learning styles and teaching strategies. Section 4 presents the proposed architecture. Section 5 describes the processes implemented and section 6 presents conclusions and future work.

2 Intelligent Tutoring Systems

Guardia (1993) cited by Salgueiro [6], presents a definition for the intelligent tutors: "An ITS is a system of computer-assisted instruction, which uses Artificial Intelligence techniques, mainly for representing knowledge and drives a teaching strategy; and it is able of behaving like an expert, both in the knowledge domain that it teaches (showing the student how to apply that knowledge), as in the pedagogic domain, where it is able of diagnosing the situation in which the student is and offer a solution that allows him/her progress in the learning".

Within the literature reviewed, we found that areas like Mathematics, Physics and Programming, had implemented the basic architecture of an Intelligent Tutoring System. Among the related works we can cite Butz [7], Conati [8], Graesser [9] and Melis [10].

According to Butz [7], the basic architecture of an ITS is composed by a *student module*, a *knowledge module* and a *tutor module* which is also called teaching strategies module. These modules operate interactively and communicate through a central module, which it is often called *user interface*. This architecture is shown in Figure 1. Its modules are described below.

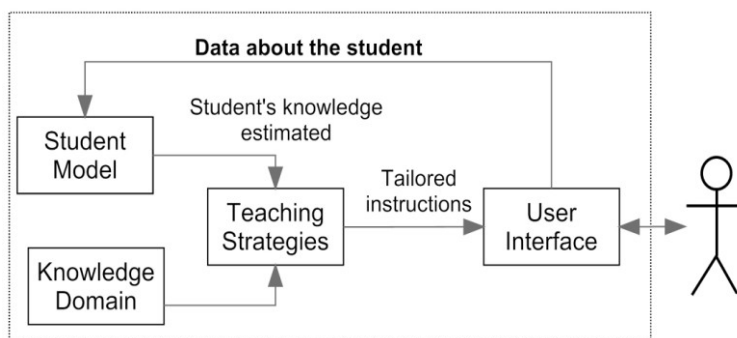


Fig. 1. Basic architecture of an ITS [7].

The student module aims to perform the student's cognitive diagnosis and the student's representation for future system feedback. Cataldi [11] proposes to incorporate learning styles in the ITS. According to her, the student module is composed by the following components:

- A database with learning styles available in the system.
- A map of the knowledge obtained initially from the domain module, which will be modified by the update of knowledge, based on the assessments made by the tutor module.

The knowledge module aims to store the dependent and independent knowledge of the scope. Basically, this module is composed by [11]:

- *Knowledge*: It refers to the content that must be loaded into the system, through the concepts, questions, exercises, problems and their relationships.
- *Didactic elements*: they are multimedia material, i.e., images, videos and sounds that help the student obtain knowledge during the teaching session.

The tutor module defines and implements a pedagogical teaching strategy, contains the objectives to be achieved and the plans to achieve them. This module selects the exercises, monitors the performance, provides assistance and selects the learning material for the student. It consists of the following sub-modules [11]:

- *Lesson Planner* that organizes the lessons' contents.
- *Profile analyzer*, which analyzes the characteristics of students, selecting the most appropriate pedagogical-teaching strategy.

The user interface specifies and provides support to the students' activities and to the methods used to perform these activities. The interface should be easy to use and attractive. Thus, the students quickly learn how to use it, and they can focus all their attention on the process of learning the subject [12].

3 Learning Styles and Teaching-Learning Strategies

In [13], the fact that each person uses their own methods or strategies to learn is called "learning style". Teaching strategies are procedures or resources used by the teaching agent to promote meaningful learning [14]. Although the strategies vary depending on what you want to learn, each person tends to develop certain preferences or global trends that define a style of learning [13].

According to [13], the notion that each person learns differently from others, must be considered to facilitate their learning, however, we must be careful not to "label" people, since the learning styles, although relatively stable, change depending on the situation and are susceptible to improvement. Moreover, Gomez [13] states that when students are taught according to their own learning style, they learn more effectively. Some of the most known and used learning styles models are:

- Felder and Silverman model.
- Kolb model.
- Neurolinguistic Programming model of Bandler and Grinder.
- Multiple Intelligences model of Gardner.
- VARK model of Neil Fleming.

Some learning styles have been considered in the implementation of tutoring systems in order to adapt their environments of teaching toward users. For example, the Richard Felder learning style model is considered in the systems presented by Hernandez [15] and Caviedes [16]. In the system proposed by Cataldi [17], in addition to the Felder model, they also considered the Multiple Intelligences model of Howard Gardner. Finally Araújo [18] and Peter [19] used the VARK model of Neil Fleming.

4 Architecture Proposed for an ITS

This paper proposes to incorporate to the classic architecture of an Intelligent Tutoring System, a process that selects the contents to show, influenced by the teaching strategies that encourage the student's learning style. These teaching strategies will be the link to select the contents of the subject. The contents should be prepared for each one of the teaching strategies for each learning style.

Figure 2 shows the proposed architecture, in which we specified the added components with a highlighted line. We also present a paused line that represents the redesigned component of the general architecture of an ITS and a dotted line that represents the components used for the Learning Management System.

Below we describe the changes made to the modules of the general architecture of the ITS.

In the *tutor module* we incorporated the teaching-learning strategies considered in the design of the themes of the subject, as well as the redefinition of teaching strategies according to student's learning style. We also incorporate 2 processes to adapt the contents to be presented: 1) that selects the topics to show to the students by linking their learning style with teaching strategies used in the creation of the topics, thus support their learning, and 2) a process for the diagnosis of competencies.

In the *knowledge module* we added a corpus that will store the competencies of the subject and some metadata to label the contents of the subject and characterize the competencies to develop.

In the interface module we incorporate a filter to show the contents chosen by the selection process of the tutor module.

The proposed architecture process is shown in Figure 3. The process begins determining the student's learning style. To achieve this, we decided to use the VARK model of Neil Fleming [20], which uses a questionnaire to identify the student predominant style. This questionnaire will be applied once, when the student initiates the session for the first time in the platform.

Once the student's learning style is determined and stored, the student's diagnose process (competencies) begins. The student's diagnose process doesn't take any action when students enter the system for the first time, because there is still no information to be processed.

The next step is to choose the items to be displayed by the selector agent, which must be designed in conformity with a teaching-learning strategy according to the student's learning style. The agent is also responsible for readjusting (increasing) the exposure time of the subjects based on the following: a) if the subject is elaborated in

the English language and the student doesn't fulfill the level required to understand the English language, and b) if the item to be displayed isn't elaborated under any teaching strategy to support the student's learning.

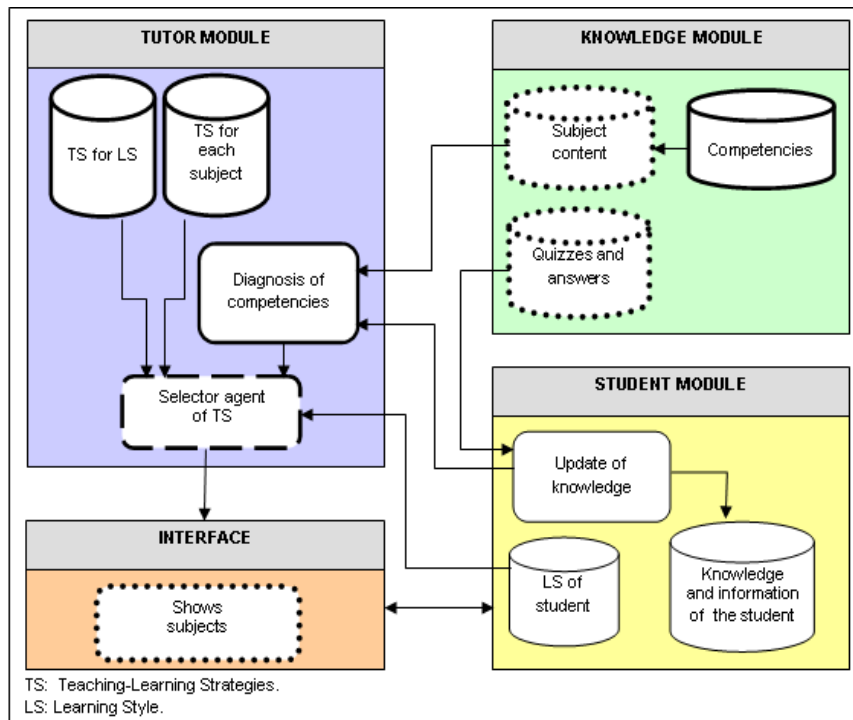


Fig. 2. Proposed architecture.

Since the subjects have been selected and personalized, these will be shown to the student through the Moodle platform. Once the student has revised a subject, its evaluation will be applied and the database that stores the student's performance will be updated with the results obtained from the evaluation.

On the other hand, when the student's diagnose process has historical information of him/her, the process will take the information of the student's performance (elapsed time of the activity, amount of attempts and assessments) and it will evaluate if the student gets the competency (Bayesian network), if so, the student will continue to the next topic and the selector agent will not execute any action. Otherwise, the selector agent will select another resource to display, elaborated with a different teaching strategy that supports the student's learning style.

5 Implemented Processes

The implementation of the VARK questionnaire is part of the process of the proposed architecture, see Figure 3. This implementation will help to determine the student's learning style. The following sections describe its development.

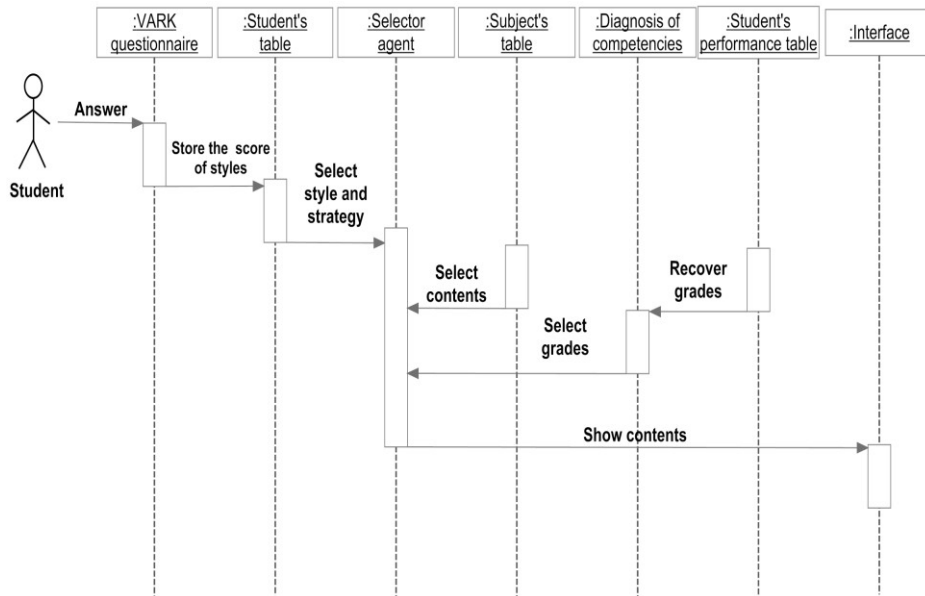


Fig 3. Process sequences of the proposed architecture for an intelligent tutor system.

5.1 VARK Model

The VARK model was developed by Neil Fleming [20] in 1987. This model is a tool to determine the preferences of individuals for information processing. The model considers four different learning styles: Visual, Aural, Read / Write and Kinesthetic.

The model consists of a questionnaire with 16 questions and each one has four answer options that correspond to each one of the learning styles. To answer the questionnaire, it is necessary that the student: choose the answer which best explains his/her preference and mark the letter(s) that represents his/her choice. The student can select more than one response to a question if one does not fit with his/her perception. The student can also leave a blank if any question isn't applied to his/her preferences, but the student must answer at least 12 questions.

Table 1 shows some of the Neil Fleming's proposals for teaching-learning strategies for each learning style.

5.2 Implementation of the VARK Questionnaire in Moodle

The integration of the VARK questionnaire in Moodle allows us to determine the student's preferred learning style. The following section describes the four activities for its implementation.

Table 1. Teaching-learning strategies for the VARK model.

<i>Learning Style</i>	<i>Teaching-Learning Strategies</i>
Visual	▪ Pictures, Videos, Posters.
Aural	▪ Discuss topics with your teachers. ▪ Explain new ideas to other people. ▪ Use a tape recorder.
Read / Write	▪ Dictionaries, Textbooks, Notes.
Kinesthetic	▪ Field tours. ▪ Applications. ▪ Trial and error.

5.2.1 Configuration in Moodle of the student's custom fields to store the questionnaire results

Moodle allows customizing fields to the student's profile, which was used for adding the fields: Visual, Aural, ReadWrite and Kinesthetic. These fields were used to store the results obtained from applying the VARK questionnaire to the student. To add these fields the following steps were done:

Step 1. Log in to the Moodle platform with an administrator's account, access the block Administration of the site and choose the options: *Users > Accounts > User profile fields*.

Step 2. Create a new custom field of text type.

Step 3. Configure the field as blocked and left empty the default value option.

This same procedure was used to add the *Actual Style* field, which will store the learning style resulting from the application of the questionnaire to the student. Also, the *ActualStrategy* field was added for storing the actual teaching strategy.

5.2.2. Presentation of the questionnaire to the student

Due to the fact that the VARK questionnaire must be applied and stored only once, it was necessary to detect in Moodle's code, the point where the student's login is validated (index.php file, located in the moodle/login folder). In this file we added a consult of the custom fields values: Visual, Aural, ReadWrite and Kinesthetic. If these values are empty this indicates that VARK questionnaire was not applied, so the web page that contains the questionnaire will be shown. The code added to the index.php file is:

```
if ($USER->Visual == "" and $USER->Aural == "" and
    $USER->ReadWrite == "" and
    $USER->Kinesthetic == "")
    {redirect($CFG->httpswwroot.'/login/p_vark_preguntas.php');}
```

Custom fields Visual, Aural, Read-Write and Kinesthetic, ActualStrategy and Actual_Style are automatically loaded for the entire student session in the global object \$USER, which gives us access to them.

When the values of the student custom fields are empty, the next line will be executed: `redirect($CFG-> httpswwroot. '/ Login / p_vark_preguntas.php')`, which redirects to the VARK questionnaire, otherwise the VARK questionnaire will not be displayed.

5.2.3. Implementation of the questions and answers of the questionnaire

To integrate the VARK questionnaire the following files were incorporated into Moodle: *p_vark_preguntas.php*, *p_vark.html* and *p_vark.php*, which were located in the folder `moodle \ login`.

The *p_vark_preguntas.php* file incorporates as part of its code, the contents of the *p_vark.html file*, which has the code to display the questions and answers of the questionnaire. Figure 4 shows the VARK questionnaire web page. The *p_vark.php* file is responsible for the evaluation of the questionnaire and it stores the results of the evaluation in the custom fields.

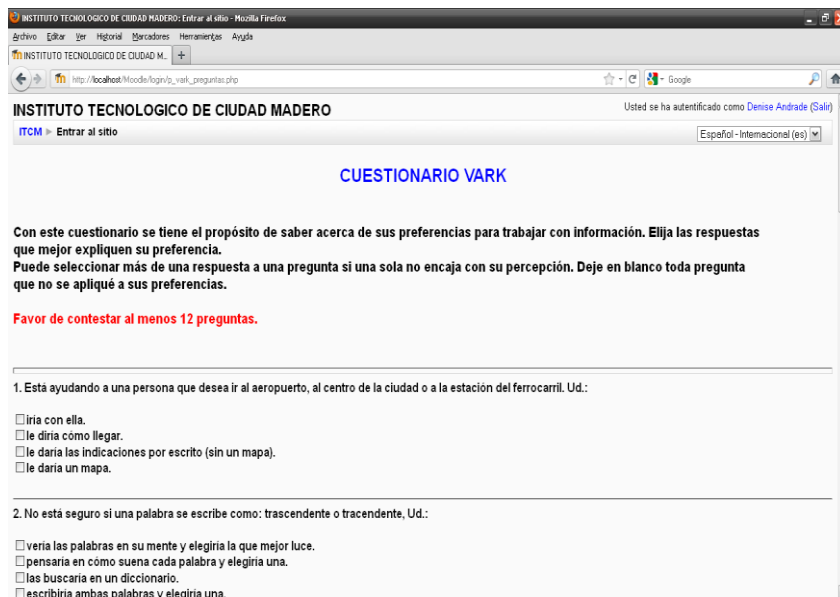


Fig. 4. Web page showing the VARK questionnaire.

5.2.4. Evaluation and storage of the results of the questionnaire

The process of obtaining the results of the VARK questionnaire considers three steps, which are detailed below:

1. To have a table of predefined answers for the model [20]. A fragment of this is shown in Table 2.

Table 2. Fragment of the table of predefined answers.

<i>Question</i>	<i>Letter a</i>	<i>Letter b</i>	<i>Letter c</i>	<i>Letter d</i>
1	K	A	R	V
2	V	A	R	K
.
.
15	K	A	R	V
16	V	A	R	K

2. Identify and mark in the table the letter(s) that corresponds to the answer(s) selected, for example, if the student chose the letters **b** and **c** in question 1, he/she must mark the letters **A** and **R** in the row in question 1, see Table 3.

Table 3. Identification of answers.

<i>Question</i>	<i>Letter a</i>	<i>Letter b</i>	<i>Letter c</i>	<i>Letter d</i>
1	K	A	R	V

3. Summarize each of the letters V, A, R, K chosen in each question. The letter that has obtained the major number of points will be the one that indicates the learning style preferred by the student. In case of tie between two or more letters, a *Multi-modal* learning style is considered.

The previous steps were implemented in the file *p_vark.php*. This file calculates the punctuation for each letter V, A, R, K and stores the information on the custom fields: Visual, Aural, ReadWrite and Kinesthetic. Also, it stores in the custom field, *Actual_Style*, the learning style with the highest score. And finally, it is responsible for selecting and storing in the custom field *ActualStrategy* one of the teaching strategies associated with the current learning style.

6 Conclusions and Future Work

In this paper we propose an extension of the classical architecture of an Intelligent Tutoring System, which incorporates a selector agent of contents. This selector agent, in its selection process, considers the teaching strategies that support the student's learning style. This extension will be implemented using the Moodle LMS.

Also we described the process of how to integrate the VARK questionnaire in Moodle to obtain the student's learning style.

The proposed architecture is currently in development. The VARK model implementation was the first step of the project, leaving the selector agent implementation to display the subject content and the competency evaluation process as future work.

References

1. Urretavizcaya, L. M.: Sistemas Inteligentes en el ámbito de la Educación. Revista Iberoamericana de Inteligencia Artificial, volumen 5, Num. 12, pp. 5-12 (2001)
2. VanLehn, K.: The Behavior of Tutoring Systems. International Journal of Artificial Intelligence in Education, 16:227-265 (2006)
3. Bloom, B. S.: The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-one Tutoring. Educational Researcher, Vol. 13, No. 6, pp. 4-16 (1984)
4. Cataldi, Z., Lage, F. J.: Sistemas Tutores Inteligentes orientados a la enseñanza para la comprensión. EDUTEC Revista electrónica de Tecnología Educativa, No. 28 <http://edutec.rediris.es/Revelec2/Revelec37/> (2009)
5. Cárdenas, P. J. L.: Sistemas de Gestión del Aprendizaje, SGA (LMS). Facultad de Educación, Universidad Autónoma de Yucatán, México.
6. Salgueiro, F. A.: Sistemas Inteligentes para el Modelado del Tutor. Tesis de grado en Ingeniería Informática, Facultad de Ingeniería, Universidad de Buenos Aires, Argentina: 196p. (2005)
7. Butz, C. J., Hua, S., Maguire, R. B.: A Web-based Bayesian Intelligent Tutoring System for Computer Programming. Department of Computer Science, University of Regina (2006)
8. Conati, C.: Intelligent Tutoring Systems: New Challenges and Directions. In Proc. of the 20th International Joint Conference on Artificial Intelligence, pp. 2-7 (2009)
9. Graesser, A. C., Person, N. K., Harter, D. and The Tutoring Research Group.: Teaching Tactics and Dialog in AutoTutor. International Journal of Artificial Intelligence in Education, vol. 12, pp. 257-279 (2001)
10. Melis, E., Siekmann, J.: Activemath: An intelligent tutoring system for mathematics. Seventh International Conference 'Artificial Intelligence and Soft Computing' (ICAISC), 3070:91-101 (2004)
11. Cataldi, Z., Lage, F. J.: Modelo de Sistemas Tutor Inteligente distribuido para educación a distancia. Laboratorio de Informática Educativa y Medios Audiovisuales, Facultad de Ingeniería, Facultad Regional Buenos Aires, Universidad Tecnológica Nacional, Argentina, (2009)
12. Millán, D. E.: Sistema bayesiano para modelado del alumno. Tesis doctoral, Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, España (2000)
13. Gómez, N. C. L., Aduna, L. A., García, P. E., Cisneros, V. A., Padilla, C. J.: Manual de Estilos de Aprendizaje. Secretaría de Educación Pública, Dirección General del Bachillerato, http://www.dgb.sep.gob.mx/informacion_academica/actividadesparaescolares/multimedia/Manual.pdf (2004)
14. Díaz, F. D., Hernández, G.: Estrategias docentes para un aprendizaje significativo. Capítulo 5, McGRAW-HILL, pp.13-28 (1999)
15. Hernández, Y., Rodríguez, G., Arroyo-Figueroa, G.: Integrating learning styles and affective behavior into an intelligent environment for learning. Research in Computing Science 51, pp. 172-180 (2010)

16. Caviedes, P. D., Medina G. V. H., García P. O.: Diseño de un sistema tutor inteligente basado en estilos cognitivos. Realizado en la Maestría de Ciencias de la Información y las Comunicaciones de la Universidad Distrital Francisco José de Caldas en Bogotá – Colombia (2009)
17. Cataldi, Z., Lage, F. J.: Modelado del Estudiante en Sistemas Tutores Inteligentes. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología, No. 5, Art. 4, (2010)
18. Araújo, G. A., Miranda, R., Almeida, Z., Rocha, F. L.: A Web-Based Intelligent Tutoring System on Teaching and Learning Electrical Project. Instituto Superior de Engenharia do Porto, Porto, Portugal (2010)
19. Peter, S. E., Bacon, E., Dastbaz, M.: Learning styles, Personalisation and Adaptable e-Learning. Fourteenth International Conference on Software (2009)
20. VARK: a guide to learning styles, <http://www.vark-learn.com/english/index.asp> (2011)