USING EXPERT SYSTEMS TECHNOLOGY
FOR STUDENT EVALUATION
IN A WEB BASED EDUCATIONAL SYSTEM

Ioannis Hatzilygeroudis, Panagiotis Chountis, Christos Giannoulis and Constantinos Koutsojannis
University of Patras, Dept of Computer Engin. & Informatics
26500 Patras, Hellas (Greece)
&
Research Academic Computer Technology Institute
P.O. Box 1122, Patras, Hellas (Greece)
{ihatz/giannoul/ckoutsog}@ceid.upatars.gr

ABSTRACT
In this paper, we present a web-based intelligent education system to help students and tutors in the context of an AI course. We concentrate on the knowledge management and student evaluation aspects of the system. Knowledge management mainly refers to test questions construction and management. Student evaluation refers to the evaluation of the knowledge level of a student with regards to taught concepts. A rule-based expert system helps in student evaluation. A number of statistics provided by the system give valuable information to the tutor.

KEY WORDS
Web-based education, student evaluation, expert systems, knowledge management.

1. Introduction

Recently, there has been a large research activity on web-based intelligent educational systems (WBIESs) [4]. WBIESs use Artificial Intelligence (AI) Techniques in order to adapt mainly to student needs for self-study. Their basic structure comes from Intelligent Tutoring Systems (ITSs) [1] [6]. On the other hand, e-learning environments provide facilities mainly for helping course generation and management and refer to both the tutors and the students. Adding facilities (intelligent or not) for tutors in WBIESs make them a kind of intelligent e-learning systems (IELSs) [2], [3], [6]. An important function of such systems is student evaluation, which provides results for adapting to both the students and the tutor. There are many techniques used for student evaluation. However, a few of them use an expert system approach [3].

To help the students and the tutors in our Department, we constructed an intelligent e-learning system to assist learning and teaching in the context of the course of Artificial Intelligence (course 451). The system provides means to the tutors for constructing questions and tests in a structured way. Also, with the help of an expert system, evaluates the students and provide useful statistics to the tutor. These are extensions to [5].

2. System Architecture

The architecture of the system is depicted in Figure 1. The system consists of five units: the Identification Unit (IU), the Student Model (SM), the Tutoring Unit (TU), the Evaluation Unit (EU) and the Virtual Agent (VA). Through IU the student initially subscribes to the system. During subscription some personal settings are saved. After subscription the student can, at any time, enter the system through the IU.
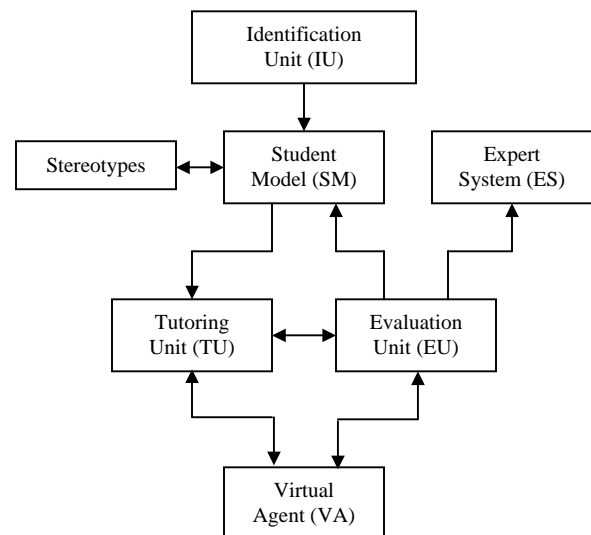

Figure 1. System Architecture

SM contains all the information about students, like their preferences, interests, knowledge level, etc. We use stereotypes to set up an initial profile for each student, so

the system can use that profile for adaptation. The choice of the stereotype is achieved by using a questionnaire that gets information about student's knowledge, preferences and goals. This information is saved to the SM, which can later be updated by acquiring new information.

TU is responsible for the teaching process. A student can select a learning goal from the domain knowledge tree (see next section). The corresponding material is then presented to the student. The material consists of theory and examples. The student is also able to select a test to evaluate him/herself about how well he/she has learned the involved concepts so far. After the student has taken a test, the system stores the statistical results concerning each concept, which the student has been examined at, and, if the results are not satisfactory, it advices him to study again the corresponding concepts and suggests some new tests that contain questions only about the problematic concepts.

The main goal of EU is to evaluate student's progress due to his/her interaction with the system. This evaluation is achieved through testing. From the testing results the tutor is able to evaluate and watch each student's progress. The tutor is also able to see some statistical results over the concepts and sections the student has been examined at. The system can also provide statistical results about all students that have examined at a specific test. The ES contributes in deciding upon the knowledge level of a student.

VA has as main goal to motivate the students and improve system's effectiveness. In our case, VA is an avatar (called Merlin) that provides information to the students in a natural and recreating way. To achieve this, VA gets information from the SM and communicates with the student in the same way that a human tutor would do. It does not contain any reasoning mechanism though.
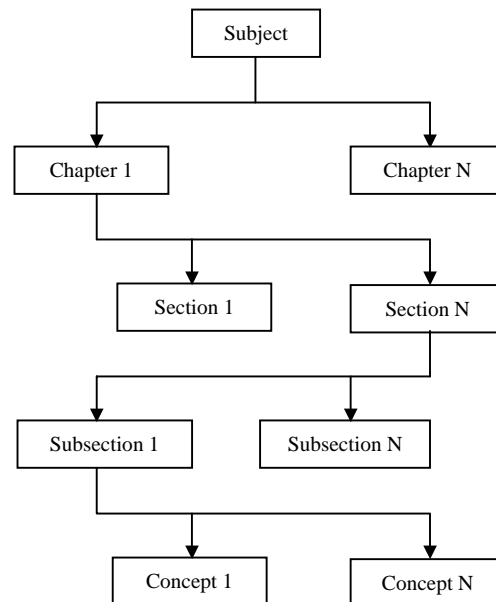
3. Domain Knowledge

The domain knowledge of the system, at the moment, concerns predicate logic as a knowledge representation language. Domain knowledge is structured in a tree-like way (see Fig. 2). The root of the tree is the above subject. The subject is divided in sections and the sections into sub-sections. Each sub-section deals with a number of concepts, which are the leaves of the tree. Subsections may have common concepts.

The domain knowledge tree, described above, as far as the subsections level, is displayed in the navigation area of the user interface (at the left side of the screen). From that tree the student can choose a *learning goal* (subsection). Each subsection corresponds to a *learning page*, which is a php page. That is, only subsections correspond to displayable material. The learning page of the selected subsection is currently presented in the content area (which resides at the centre and the right part of the screen). Each learning page deals with a number of *concepts*. More specifically, it contains an *ordered list* of concepts. Each concept is linked to the corresponding

concept page. Concept pages constitute the real *teaching material*.

The teaching material, apart from concept pages, however, includes all the available *questions*, which are stored in a database (not displayed in Fig. 1) and are used for the creation of the *tests*. Each learning page is associated with a test. Each test consists of a number of multiple choice type questions that examine the concepts of the associated learning page.


Figure 2: Domain Knowledge Tree

4. Learning Process

The learning method (implicitly followed) is based on the traditional theory-examples-exercises paradigm (although the user can follow his own method). That is, for each concept, the theory is first presented. Then, some examples are given. Finally, the student is called to make some exercises. The concepts in a learning page are presented in a simple-to-complex way. That is, the simple concepts are presented first and the complex concepts (that require the knowledge of one or more simpler concepts) are presented afterwards. This is mapped onto the ordered list.

Furthermore, the student can review a previous concept at any time. The student is also not forced to follow the system's way of teaching, but he can make his own choices for studying. For example, one can jump to complex concepts without having had a look at simpler ones. In many concept pages there are links to other concepts that are *prerequisite* to the concept of the page. So, the student, if needed, can recall the theory about the prerequisite concepts. After having looked at the recalled theory, he/she can return back to where he/she was before and go on with his/her studying. At certain moments the avatar (Merlin figure) turns up providing informative messages to the student.
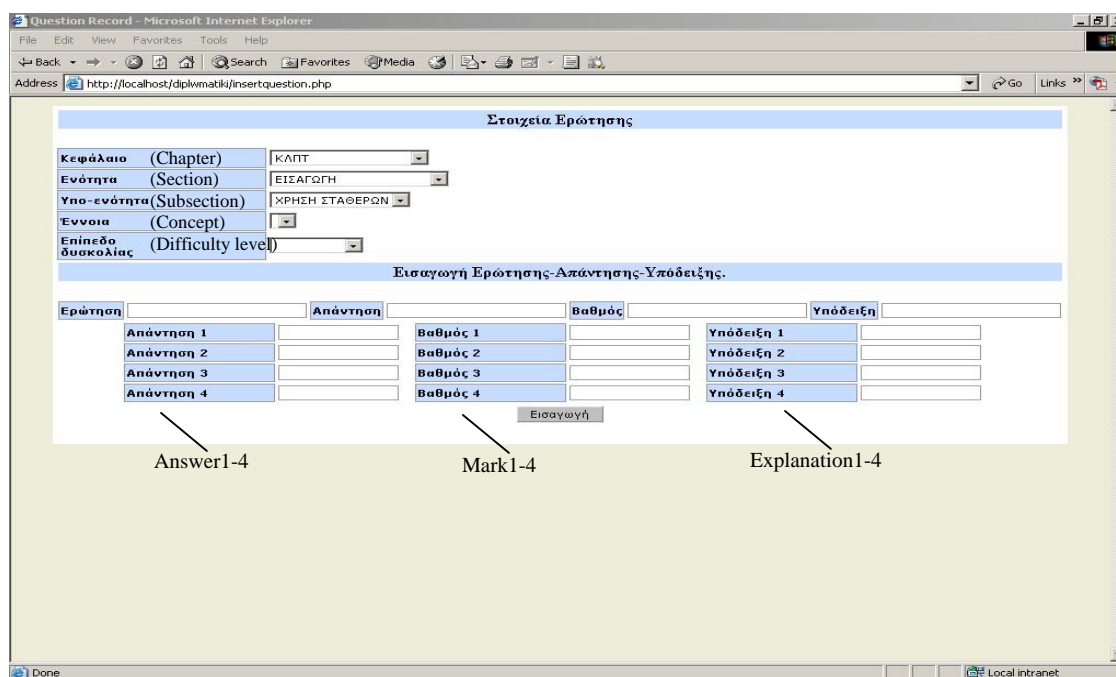
Figure 3. Screen for Question Creation

The student can check his/her knowledge level after having studied a learning page, by doing the test specified at the end that page. Any time a student has finished the study of a particular learning page, he/she can take an appropriate test. The questions of the test are presented one at a time. The student can answer each question independently and in his own sequence. In case the student gives a wrong answer to a question, he/she can try again. Before that, he/she may either see the hint (see next section) or the correct answer. Doing this, will affect the mark that the student gets on that question.

5. Domain Knowledge Management

The system offers to the tutor the capability of domain knowledge management. The tutor can deal with management of the domain tree and the teaching material (learning pages, concept pages, questions and tests).

Management of the domain tree concerns the insertion, deletion or change of a sub-tree of the domain tree. The tutor can insert a new sub-tree at any level of the tree (chapter-sections-subsections-concepts, section-subsections-concepts or subsection-concepts) or delete an existing sub-tree at any level or modify one. In other words, the tutor can change the domain knowledge tree at any time according to his/her needs.

The most important part of teaching material management is the management of the questions. A question is considered as a structured object that consists of the following parts:

(a) the name of the question
(b) the body of the question,

(c) the concept it concerns (as well as its corresponding path in the domain tree),
(d) its difficulty level (one of easy, medium and difficult),
(e) its possible answers (1 to 4) each followed by a mark and an explanation and
(f) a hint.

A mark represents the student's expected mastery degree upon the concept, if the corresponding answer is selected. An explanation provides information about why the corresponding answer is wrong or right. The hint is a kind of clue to help a student to answer the question correctly.

The tutor is able to insert, delete or modify all the parts contained in the question. To insert a new question the tutor has to select the corresponding form (see Fig. 3). In that form the tutor has to fill in all the above mentioned parts of the question. Every question inserted is referred to one of the existing concepts. The tutor has to know the chapter, the section and the subsection that concept belongs to. Two individual questions that are referred to the same concept may have different difficulty levels.

To delete a question, the tutor has to select the corresponding question and then click a delete button. Modification of a question requires first the selection of the question. Then the tutor can update any part of the question. For example, the tutor can add a new possible answer or remove one or modify existing ones. Also, the tutor can see all the questions and all information concerning them.

Another facility that is available to the tutor is the management of tests. All available tests can be shown in a list. The content of each test can be displayed by selecting the test's name. In case that the tutor wants to delete one

or more tests, he/she can check them in the list and then click the delete button. To create a new test, the tutor has first to give a name for this test. Then, through an appropriate form, he/she selects the chapter, the section, the subsection and the concept as well as the difficulty level and number of the questions that are going to be inserted in the test. Questions that satisfy the above criteria are randomly selected and then inserted in the test. The tutor can add as many questions as he/she wishes, following the same procedure.

In the above form some boxes are filled in an automated way. Every time the tutor selects a chapter, the sections contained in that chapter are shown. Also, for each selected section, its subsections are automatically recalled and shown.

Finally, new concept and learning pages can be added to or deleted form the system. Also, the tutor can modify existing pages.

6. Student Evaluation

In web-based educational systems, one of the most important functions is student evaluation. Student evaluation refers to the evaluation of the knowledge level of a student after having dealt with a learning page. In other words, how well a student has learnt the concepts of a learning page. This is achieved by processing the results of the test offered at the end of a learning page. Student evaluation is important for both the student and the tutor.
A student can be evaluated at two levels: (a) the concept-level and (b) the topic-level. The concept-level evaluation deals with the level of understanding of the concepts of a learning page test, whereas the topic-level evaluation deals with the level of understanding of the topic of a learning page, i.e. the test as a whole.

The knowledge level of a student, as far as both a concept and a topic are concerned, is classified in one of the following five categories: (a) excellent (86-100), (b) very good (71-85), (c) good (51-70), (d) average (31-50) and (e) low (0-30). Within the parentheses are the corresponding ranges of the marks to be achieved. E.g. if a student gets an overall mark of 56 points for the questions related to concept 'quantifiers', then the knowledge level of the student for 'quantifiers' is 'GOOD'.

The knowledge level of a student for a concept (CL) is calculated via the following formulas:

$$CL = \sum_{i=1}^{3} Qm_i * qw_i \qquad (1)$$

$$Qm_i = \frac{\sum_{j=1}^{n} Qm_j^i}{n} \qquad (2)$$

where i represents the difficulty level (1→ easy, 2→ medium, 3 → difficult), Qm_i is the average mark of the answers to questions of the same difficulty level related to the concept and qw_i is the question weight. The mark for each question answer (Qm_j^i) is given by the rules in Table 2. Question weight is related to the difficulty level of a question and the composition of the set of questions used for testing the concept. (See Table 1 for the values of question weights in all possible cases. For example, in case that there are no difficult questions used, $qw_1 = 40/100=0.4$ and $qw_2 = 60/100=0.6$).

Table 1. Question weight distribution

Easy (E)	Medium (M)	Difficult (D)	Question (s) weight(s) (/100)
1	1	1	20 (E), 30 (M), 50 (D)
1	1	0	40 (E), 60 (M)
1	0	1	30 (E), 70 (D)
0	1	1	40 (M), 60 (D)
1	0	0	70 (E)
0	1	0	80 (M)
0	0	1	100 (D)

The knowledge level of a student for a topic (subsection) is calculated from the following formula:

$$TL = \frac{\sum_{i=1}^{n} m_i}{n}$$

where m_i is the mark that the student gets from the answer of question i of the corresponding learning page test. So, TL is the average mark of all the questions. The mark that a student gets given the answer to a question is also based on the rules presented in Table 2.

Table 2. Rules for answer marking

Difficulty Level	No of tries	Use of Hint	Answer Status	Mark
easy	1	0	Correct	85
easy	1	1	Correct	70
easy	1	0	Wrong	30
easy	1	1	Wrong	0
easy	2	0	Correct	50
easy	2	1	Correct	30
easy	2	0	Wrong	30
easy	2	1	Wrong	0
easy	>2			0
medium	1	0	Correct	92
medium	1	1	Correct	77
medium	1	0	Wrong	40
medium	1	1	Wrong	15
medium	2	0	Correct	60
medium	2	1	Correct	40
medium	2	0	Wrong	40
medium	2	1	Wrong	15
medium	>2			15
difficult	1	0	Correct	100
difficult	1	1	Correct	85
difficult	1	0	Wrong	50
difficult	1	1	Wrong	30
difficult	2	0	Correct	70
difficult	2	1	Correct	50
difficult	2	0	Wrong	50
difficult	2	1	Wrong	30
difficult	>2			30

The rules of Table 2 are implemented via the expert system (ES) (see Fig. 4). ES is a rule-based expert system implemented in Jess, which is an expert system shell. For example, the following rule implements one of the rows of Table 2.

```
(defrule rule1
    ?level2 ← (expert (level easy) (number_try 1)
              (use_hint n) (result CORRECT))
    =>
    (retract ?level2)
    (print-level-message 85))
```

Finally, a student the knowledge level of a student concerning a concept and a topic (subsection) is classified in one of the five above categories (excellent, very good, good, average, low) based on the values of CL and TL respectively.
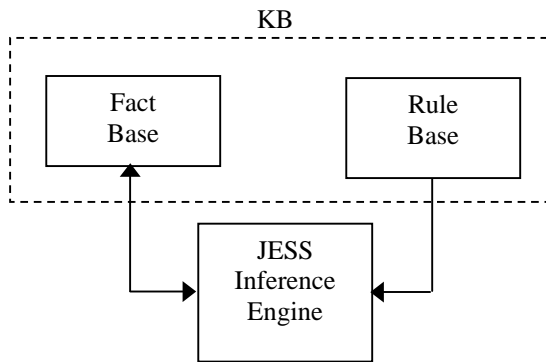

Figure 4: The structure of the ES

7. Helping the Tutor

The tutor, apart from using the knowledge management facilities of the system to update domain knowledge, is provided with another means that facilitates his/her work. The system can present some statistics about students, which are the following:
1. The percentage of the correct answers to questions per difficulty level per concept per student.
2. The percentage of the correct answers to questions per difficulty level per test per student.
3. The percentage of the correct answers to questions per difficulty level per concept for all students.
4. The percentage of the correct answers to questions per difficulty level per test for all students.

Based on the above statistics, the tutor can evaluate not only students, but also the questions of the test and indirectly the teaching process. So, the system provides to the tutor a useful tool to check the quality of each question and each test that he/she has designed for the evaluation of students.

8. Implementation issues

The system is web-based, so it works only through the WWW and cannot be downloaded and function on a student's computer. It is implemented in php. By using php, we've added a dynamic functionality to the system, which gives it the ability to interact with the student and the tutor. Usage of php can cause no problems to the communication with the database. Our database was built with Mysql.

The VA (avatar), one of the interesting parts of the system, is implemented in VBScript. To use the avatar, we first have to call its object from the Windows Controller. Then we have to connect to that object and load it. We are then able to call its functions, show it on the screen and make it interact with the user.

Another important implementation issue was the use of an ES for student evaluation. The ES is implemented in Jess, an ES cell implemented in Java. After the student has finished a test, a fact for each individual question of the test is saved in a file. Then the expert system is called to evaluate the student. This is achieved with the help of the WScript.shell object that gives the capability to execute a program from the command line. The expert system, after having taken the facts from the file, returns the results in another file.

9. Conclusion

In this paper, we present a web-based intelligent education system to help students and tutors in the context of an AI course. We concentrate on the knowledge management and student evaluation aspects of the system. Knowledge management mainly refers to test questions construction and management. Student evaluation refers to the evaluation of the knowledge level of a student with regards to taught concepts. A rule-based expert system helps in student evaluation. A number of statistics provided by the system give valuable information to the tutor.

The system, as it is now, adapts to the student needs, but not to the tutor ones. It is only adaptable as far as the tutor is concerned, in contrast to [7]. So, a further effort is to make it adaptive to the tutor needs as well.

References

[1] Brusilovsky, P., "Methods and Techniques of Adaptive Hypermedia". In: Brusilovsky, P., Kobsa A. & Vassileva, J. (Eds.): *Adaptive Hypertext and Hypermedia*. Kluwer Academic Publishers, 1998.
[2] P. D. Christea, R. Tuduce, I. A. Savescu, C. A. Grogorin, D.-C. Tomozei, V. R. Gradinescu and C. M. Rangu, "Prototype Implementation of an Intelligent E-Learning System", Proceedings of the IASTED International Conference on Web-Based Education (WBE-04), Febr. 16-18, 2004, Innsbruck, Austria, Acta Press, 441-446.

[3] El-Khouly M. M., Far B. H. & Koono Z., "Expert tutoring system for teaching computer programming languages". *Expert Systems with Applications, 18*, 27-32, 2000.

[4] I. Hatzilygeroudis (Guest Editor), Special Issue on AI Techniques in Web-Based Educational Systems, International Journal on AI Tools (IJAIT), 13(2), 2004.

[5] I. Hatzilygeroudis, C. Giannoulis and C. Koutsojannis, "A Web-Based Education System for Predicate Logic", Proceedings of the IEEE ICALT-2004, Aug. 30-Sept. 1, 2004, Joensuu, Finland, IEEE Computer Society, 106-110.

[6] S. A. Kazi, "A Conceptual Framework for Web-Based Intelligent Learning Environments Using SCORM-2004", Proceedings of the IEEE ICALT-2004, Aug. 30-Sept. 1, 2004, Joensuu, Finland, IEEE Computer Society, 12-15.

[7] L. Lesta and K. Yacef, "An Intelligent Teaching Assistant System for Logic", Proceedings of the ITS-2002, Biarritz, France (June 2002) 119-128.