

Applications

A Scenario-Based Approach for the Creation of a Virtual Environment for Secondary School Instruction

*Stanimir Stoyanov*¹, *Veselina Valkanova*², *Ivan Popchev*³, *Ivan Minov*¹

¹ *University of Plovdiv, 4000 Plovdiv*

E-mails: stani@uni-plovdiv.bg van_min@mail.bg

² *“St. Sofronij Vrachanski” Secondary School, 4000 Plovdiv*

E-mail: veselinaviva9@gmail.com

³ *Institute of Information Technologies, 1113 Sofia*

E-mail: i.popchev@iit.bas

Abstract: *In this paper we present a scenario-based approach for designing an eLearning system, specifically targeted towards the secondary school education. The key concept underlying our approach is focused on the application of three models – domain model, user model and pedagogical model – which influence the learning process. They have been developed on the basis of the analysis of the existing lessons used for the Mathematics subject. We also present the agent-oriented environment VESSI (Virtual Environment for Secondary School Instruction) to show the practical applicability of the proposed approach.*

Keywords: *e-Learning, software architectures, intelligent agents, pedagogical model, user model, domain model.*

1. Introduction

Over the past few years the use of Information and Communication Technologies (ICT) in education has become an area of ever growing research and development interest as well as a topical application area. As a result a number of strategies, specifications, standards and technologies for implementation of e-Learning supporting tools are available. The different terms and notions used in the specialized bibliography sources, however, are often confusing and do not fully

express the essence of the problems and the complexity of the tasks that have to be solved by creating the e-learning systems. Moreover, the traditional approaches and technological infrastructure, focused mainly on electronic communications, don't give adequate opportunities for supporting of didactical- and methodological-oriented models in modern any-where, any-time personalized electronic education applications.

On the other hand, being familiar with available theoretical models is not enough to face the real-world challenges when putting them into practice. Choices of pedagogy and technology are complex. If students are offered regular teaching and learning sources (books, lectures, face-to-face seminars) together with theory-driven e-Learning scenarios that do not correspond to their real needs, it is obvious that they are going to prefer the well-known means for learning and disregard the new ones. For development of effective e-Learning applications new scenario-based approaches incorporating real-world practical experiences are much more significant.

Our ongoing work is mainly targeted at the development of adequate ICT-based tools for the effective support of the learning process. We look also for new approaches, models and architectures that are compliant with the up-to-date requirements and the specific of the contemporary education. In meeting these challenges we aim to carry out our research in a different way in the sense that we focus rather on the development of practice-based approaches.

In this paper we present a stepwise approach for the creation of an e-Learning system for secondary school education. The aim of the approach is the creation of a Virtual Environment for Secondary School Instruction (VESSI) supporting personalized electronic and distance education in the secondary school. In respect to providing needed flexibility, adaptability and collaboration the environment is developed as a multi-agent system. The education process is managed and supervised by the help of pedagogical agents. Similar concepts are used in US River City [1] and Singapore SRC [2] projects.

On principle the approach could be applied for implementation of any e-Learning system. However the pedagogical agents, which are the most important component of the resulting system architecture, present a particular pedagogical model. In the case of the secondary school we need powerful tools supporting and guiding the pupils during the self-dependent usage of an e-Learning system. Therefore suitable components have to be developed in the architecture.

The rest of the paper is organized as follows: in Section 2 a brief description of our approach is presented; in Section 3 development phases are discussed in more detail; the VESSI architecture is presented in Section 4, and Section 5 concludes the paper.

2. Our approach

The presented approach, used for development of VESSI, is a practical application of the more general approach described in [3]. The concept model, depicted in Fig. 1, is the basis of our research approach to elaborate a suitable e-Learning

infrastructure. In general two main processes in each automated environment supporting e-Learning have to be supported – the creation (or generation) and interpretation of electronic content. All information needed to serve the processes is presented in three models – domain model, student model, and pedagogical model (includes the educator model as well) [4].

To achieve the necessary flexibility, adaptability and collaboration we have to support explicit components in the software architecture [5].

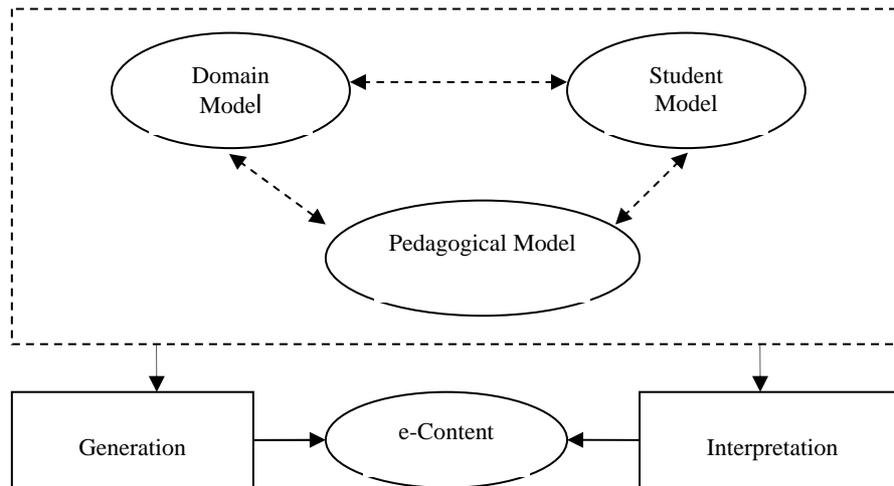


Fig.1. Concept model for electronic education support

The proposed approach comprises four main steps:

- *General teaching scenario* – this phase aims at the creation of a general scenario by analyzing the real-world teaching examples. The scenario is presented as a structured record where the knowledge is clearly identified and classified in one of the three models. This is the first step of our effort to define an appropriate abstract structure.
- *Abstract lesson model* – in this phase the scenario is transformed in a conceptual model which presents a higher abstraction and formalization. In the model the following elements are distinguished:
 - Subjects and their roles in the scenario realization;
 - Learning resources and their types – the resources are considered as basic building blocks;
 - Relationships among the resources and their types.
- *Architecture components* – the components of the target software architecture and their functionality are specified on the basis of the developed abstract model in this phase.
- *Architecture assessment* – in this phase the applicability and adequacy of the proposed architecture is going to be assessed. For this purpose the development of a prototype, performing the scenario, is envisaged. The prototype will be tested at a selected secondary school.

3. Development phases

In this section the preliminary two phases for the development of VESSI are discussed where the architecture (the third phase) itself is presented in Section 3. Due to the specifics of the architecture assessment (forth phase) this topic is out of context of the paper.

3.1. First step: General teaching scenario

Different lesson types have been conducted in the secondary school – for example “new knowledge”, “exercise”, “summary”. Each lesson type has different structure. In this step, by analyzing real education scenarios and paper textbooks, we aim to extract and identify common building elements and the type of specific elements. Furthermore the building elements will be subsequently described in terms of formalized notations starting (in this step) by using selected key words. In the next step the textual description is transformed into a set of UML diagrams. Thereby, by means of software tools the building elements can be used for construction of electronic content. We assume that the lessons are built according to the theory of Bloom [6]. In the next figures, description of lesson type „new knowledge” is shown.

A scenario comprises two parts – head and body, presented as a sequence of steps. The head part (Fig. 2) contains information about the general description of the lesson (meta-information). The terms, used in this description, are referred to the subject model (in this case “Mathematics”) and to the pedagogy model.

<p>Header</p> <p>Identifier: “New Knowledge Lesson” Type: “Standard Lesson” Duration: 45 min</p> <p>Discipline: “Mathematics” Theme: “Geometrical Figures” Subject: “Circumference” Class: “6th”</p> <p>Pedagogical goal: “New Knowledge Acquisition”</p>
--

Fig. 2. Scenario header

The lesson itself is presented as a sequence of steps (Fig. 3), each identified as a single pedagogical goal. Basing on the subject model different types of elements could be identified in each step (definitions, explanations, tasks, tests, examples and so on). The main characteristics of the scenario are:

- identification of the corresponding subjects – for the particular activities in the scenario the corresponding subjects (teachers and students), who perform them, are identified;

- differentiation between new and already acquired knowledge – in the subject model the differentiation between new knowledge (it should be learned) and the already acquired one (it is only referred to) is made;
- personalization – in particular parts of the scenario individualized approach (personalization of the content) could be applied.

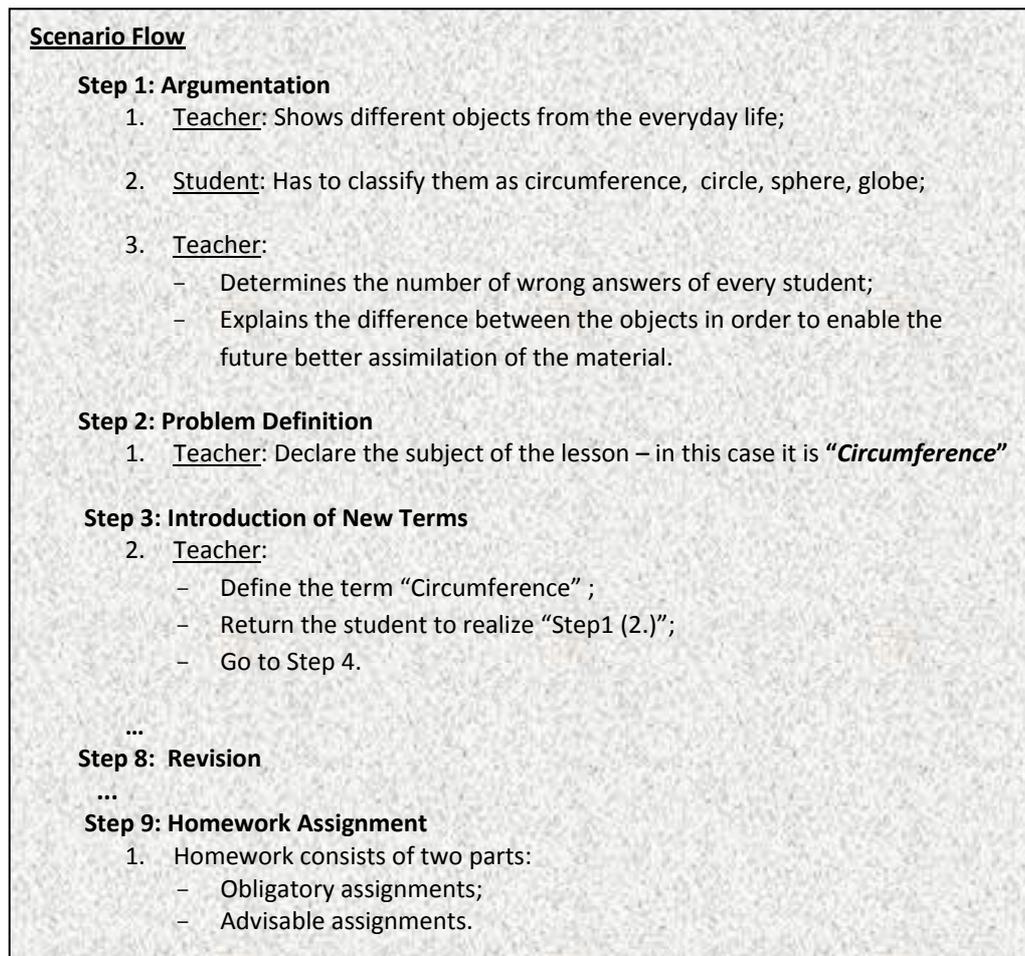


Fig. 3. Scenario flow

On the basis of the scenario analysis some essential conclusions could be made, which will be used in the next approach steps:

- The entire information needed to accomplish an education scenario could be correlated to one of the three models – domain model, student model and pedagogical model.
- Paper textbooks contain obvious knowledge to a particular subject selected in respect to pedagogical rules which aren't manifested in the content.
- Knowledge presentation in the paper textbooks is conformed to the students' age as well. Further differentiation of students is not possible.

- Two main subjects (student and teacher) described in the scenarios are autonomous and interact by means of shared data structures (messages).

Second step: Lesson abstract model

In this step, the general teaching scenario is transformed in a more abstract model presented by means of UML diagrams [7]. Some selected diagrams for “new knowledge” scenario are shown. The diagrams represent different aspects of the scenario. The most general one present the functionality of the scenario (Fig. 4), where the separate use-cases represent the pedagogical sub-goals, comprising the scenario. Fig. 5 depicts the activity diagram of one of the tests in the lesson and the sequencing diagram of the step “Introduction of new terms” is given in Fig. 6.

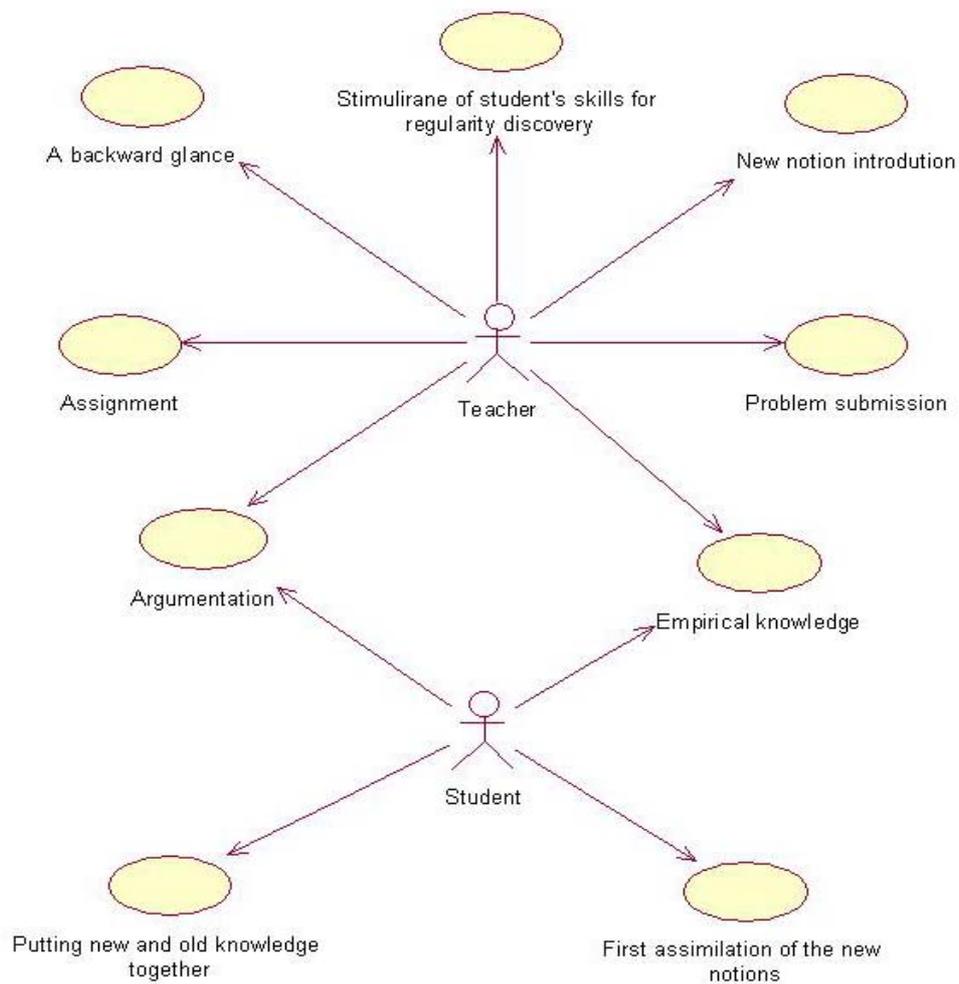


Fig. 4. “New knowledge” use-case diagram

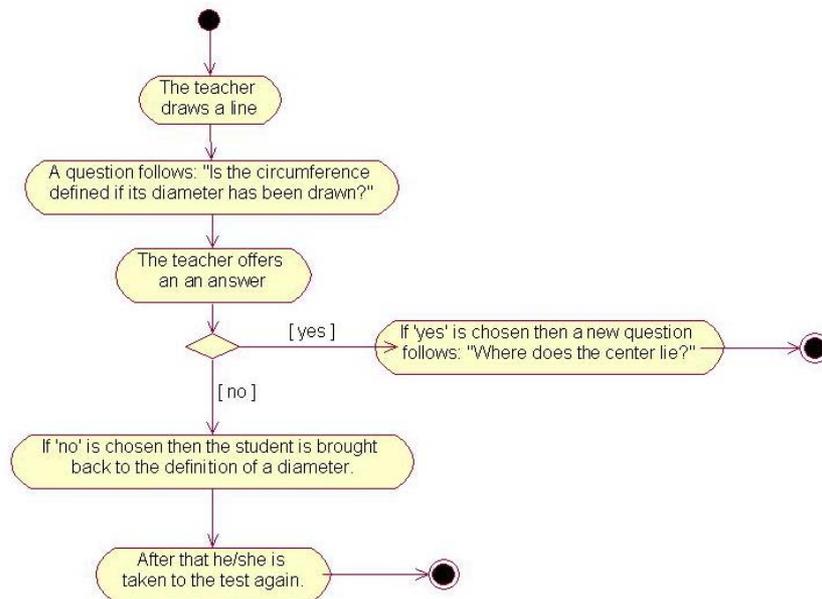


Fig. 5. "Test" activity diagram

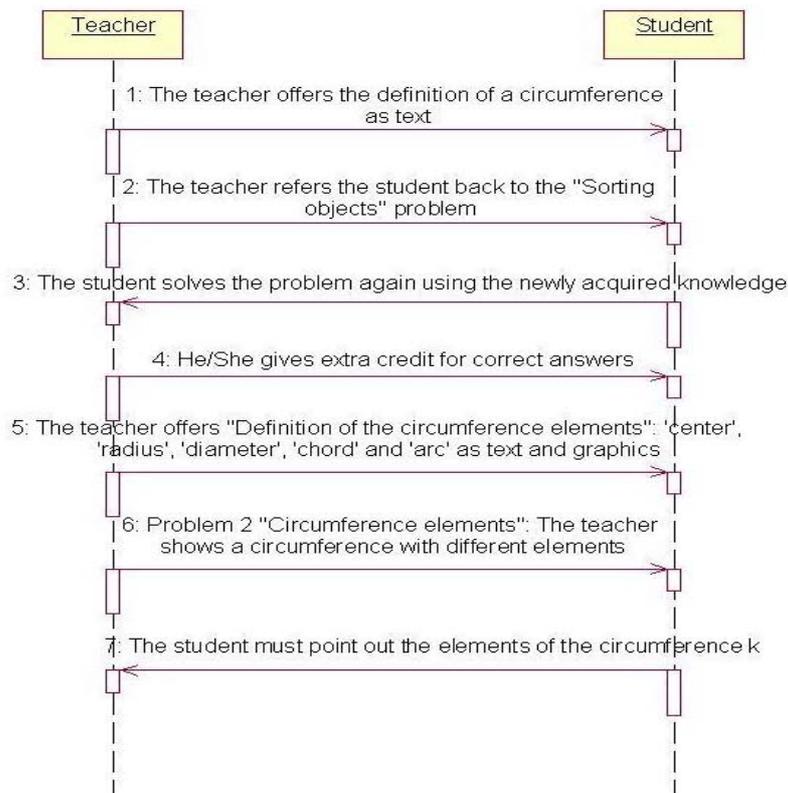


Fig. 6. "Introduction of new terms" sequencing diagram

4. VESSI architecture

In conformity with the presented approach in this section we introduce an architecture supporting e-Learning applications, called VESSI (Virtual Environment for Secondary School Instruction). The different artifacts and aspects presented in UML diagrams (users groups, roles, cases, activities, and functionality) could be combined in separated logical building blocks of the emerging architecture. In addition it is developed in compliance with the agent paradigm. Our motivation for choosing this approach is related to the fact that the subjects, participating in the educational process (students and teachers) are autonomous, could take initiative, i.e. are proactive and interact with each other by means of well structured and formalized environment (electronic learning content) and therefore could be easily represented and realized as agents. The architecture provides also a set of electronic training services. In the virtual environment VESSI the following types of agents could operate (Fig. 7):

- *Personal student Assistants (PAs)* – help every student during the work with the environment;

- *Intelligent Editors (IEs)* – context-aware editors enable the teachers in the development of electronic learning content. The easy-to-use graphical user interface (front-end module) facilitates them additionally and the intelligent agent (back-end module) enables them in the work with the corresponding repository; these could interact, if necessary, with the agents of the other editors. This way the feature “collaboration” of the architecture is provided. The following three editors are envisaged to operate in the environment:

- *Subject Editor* – facilitates the teachers in the preparation of the electronic lessons, depending on the studied subject;

- *Pedagogy Editor* – facilitates the definition and assignment of different pedagogical goals as educational patterns in compliance with the e-Learning standard SCORM [8];

- *User Profile Editor* – facilitates the creation and actualization of individual user profiles.

- *Instructor* – an agent that plays an essential role as a teacher when a student works self-dependently with the system. This agent could interact also with the personal assistants of the students;

- *System agents* – support the server resources of the system. An essential class of system agents is the *configuration agents*. In the system, different *stereotypes* characterizing different user groups are supported. The configuration agents use them to generate the corresponding personal assistant (carries out the further interaction with the system) for every student who takes advantage of the system for the first time (his affiliation to a particular user group is taken into consideration). A similar decision is proposed in [9]. Other system agents, the so called *SMILES agents*, can gather statistical information concerning the conduction of the educational process which could be used to improve the quality of the learning material.

The environment, where the VESSI agents operate, is divided in two parts – *structured* and *unstructured*. The structured part contains mainly the developed electronic lessons which are compliant with the SCORM 2004 standard [10]. The students could take advantage of the electronic lessons by means of the SCORM Engine which is integrated in the architecture as an electronic service.

Agent Environment

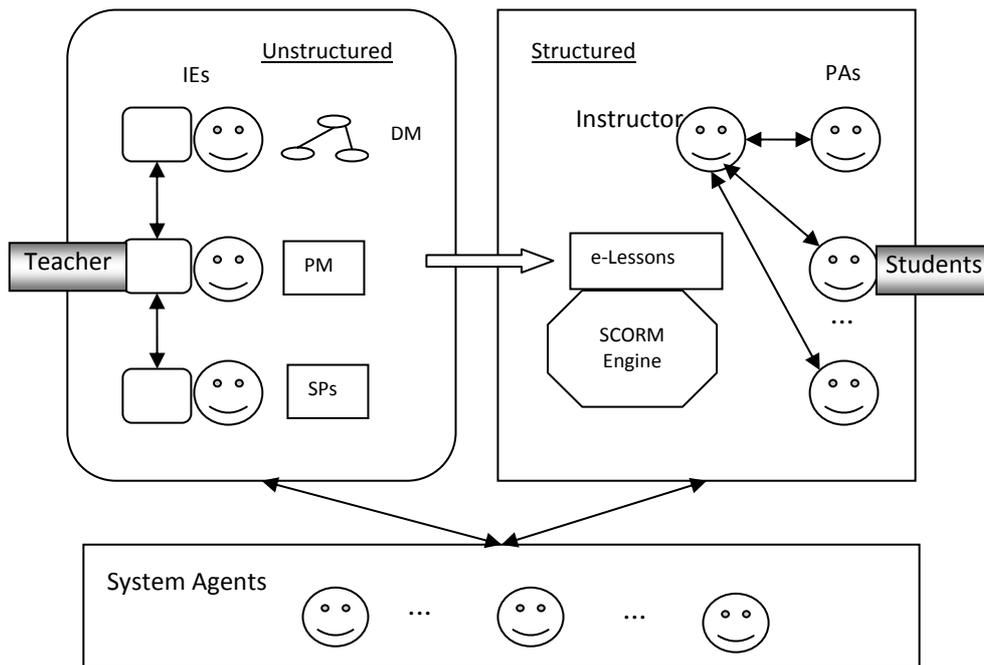


Fig. 7. VESSI architecture

The unstructured part is the component offered to the teachers. It comprises the intelligent editors and could be considered as a development environment or “work place” for them. We envisage the development of the workplace for the teachers in Mathematics by adapting the development environment Selbo 2 [11]. Selbo 2 is a development environment for creating SCORM compatible electronic content. The environment uses intelligent editors (combination of component and agent) to manipulate learning content and aid content developer during content creation. Ontologies provide developers with predefined resources covering specific domain that can be used directly in the content. Selbo 2 also utilizes education templates that define pedagogical goals and agents to govern them. Furthermore, the environment employs schemes for adapting itself to its user and for collaborating with the learning management system (LMS).

The following elements build up the unstructured environment: *Domain Model (DM)*, *Pedagogical Model (PM)* and *Student Model (SM)*. The Domain Model is presented as a hierarchy of ontologies, where:

- *Top-Level-Ontology* – presents a basic classification of the fundamental concepts in the particular domain (the subject, in our case – “Mathematics”);
- *Middle-Level-Ontology* – describes the separate themes in the subject;
- *Low-Level-Ontology* – represents particular themes (concepts).

In the first version of the architecture the information from the pedagogical model is represented formally as educational patterns which are compliant with the SCORM standard. The user model is represented as a set of student profiles.

5. Conclusion

In this paper we presented our approach for the creation of e-Learning systems for the secondary schools. It is based on the assumption that the practice in teaching and knowledge about the specific features of the educational process could be effectively combined with theoretical models in order to develop user-friendly and intuitive e-Learning systems. The proposed approach, as a consequence, envisages the use of training scenarios, driven from actual teacher practice.

We consider carrying out the development process in a stepwise manner, where the most significant phase is the VESSI architecture design. The division of the environment into structured and unstructured components is of essential importance for the realization of the proposed architecture – this way the two components of the architecture could be developed independently.

The proposed stepwise, agent-oriented approach and the chosen up-to-date technologies for its implementation is, in our opinion, a sound precondition for realizing an easy to use system to support students in secondary schools. Moreover, our considerable experience both in educating and software engineering of e-Learning systems will significantly contribute to the successful realization of the system.

VESSI architecture is developing as an educational portal by help of the open-source framework LifeRay [12]. ADL SCORM RTE [13] interpreter of electronic content is going to be integrated in the portal. For the realization of the agents we decided to use the development environment JADE [14].

Acknowledgment: This publication has emanated partly from the research conducted with the financial support of the Bulgarian Ministry of Education and Science under Research Project Ref. No H-1403/2004.

References

1. US River City Project.
<http://muve.gse.harvard.edu/rivercityproject/index.html>
2. Virtual Singapore Project.
<http://www.ntu.edu.sg>
3. Stoyanov, S., I. Ganchev, I. Popchev, M. O’Droma. An Approach for the Development of InfoStation-Based e-Learning Architectures. – *Compt. Rend. Acad. Bulg. Sci.*, **61**, 2008, No 9, 1189-1198 (in print).

4. Stoyanov, S., I. Ganchev, I. Popchev, M. O'Droma. From CBT to e-Learning. – J. Information Technologies and Control, 2005, Year III, No. 4, ISSN 1312-2622, 2-10.
5. Grosz, B. J. Collaborative Systems. AI Magazine 17, Summer 1996, No 2, 67-85.
6. Bloom, B. S. Taxonomy of Educational Objectives. Handbook I. The Cognitive Domain. New York, David McKay Co. Inc., 1956.
7. Unified Modeling Language.
<http://www.uml.org> (to date)
8. Learning System Architecture Lab (LSAL) at Carnegie Mellon University, SCORM Best Practices Guide for Content Developers.
www.lsal.cmu.edu/lsal/expertise/projects/developersguide/ (to date)
9. Costantini, S., L. Mostarda, A. Tocchio, P. Tsintza. Dalica: Agent-Based Ambient Intelligence for Vultural-Heritage Scenarios. – IEEE Intelligent Systems, March/April 2008, 34-41.
10. SCORM 2004.
<http://www.adlnet.gov/Scorm> (to date)
11. Stoyanov, S., D. Mitev, I. Minov. E-Learning Development Environment for Software Engineering Selbo 2. – In: 1st International Workshop on Data Management in Virtual Engineering (DMVE'08) September 1-5, 2008, Turin, Italy, 100-104.
12. Life Ray – Enterprise Open Source Portal.
www.liferay.com/ (to date)
13. SCORM RTE.
<http://adlcommunity.net/mod/resource/> (to date)
14. JADE – Java Agent DEvelopment framework.
<http://jade.csel.it> (to date)