BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **7**, No 3

Sofia • 2007

Applications

E-Learning Environment for Support of Secondary School Education

Todorka Glushkova

University of Plovdiv, Bulgaria

Abstract: *The paper presents a model for the creation of an e-learning environment for secondary school education.*

Keywords: *E-Learning, DeLC, SCORM, user model, domain model, pedagogical model, educational portal.*

1. Introduction

The application of Information and Communication Technologies in a classroom training process allows us to compare two basic stages – computer based training and e-Learning [17]. These two concepts are similar but there are some differences between them. Computer based training (CBT) is an attempt to automate education, replace an educator, and develop self-paced learning. It is place, time and content-predetermined learning [20]. This kind of teaching is a new method in the traditional form of classroom learning. E-learning has its origins in CBT[23]. The main focus of e-Learning is not only to educate without barriers of time and distance, but to adjust to the user's goals and needs. For the realization of this kind of teaching we are obligated to develop appropriate learning resources and mobile services.

Our team from the e-Commerce Laboratory of the University of Plovdiv has developed a conceptual model of the entire system for e-Learning in secondary schools and created a prototype model of the system based on the SCORM (Sharable content object reference model) standard [29] and the architectural framework of the DeLC model [16]. We have created a prototype version of the SCORM-based educational portal in the secondary school – Brezovo, and carried out experiments during the actual training process.

The main goal of the project is to create a conceptual model for the creation of SCORM-based e-Learning system in secondary schools. To reach this goal, it is necessary to develop an information model of an e-Learning portal that includes: models of informational and technological infrastructure; domain models of different school subjects; models for the creation, usage and sharing of e-Learning materials by adaptation of the SCORM standard; a pedagogical model; an user's model [26], etc.

The aim of the paper is to present some aspects for modeling and methods for realization of the system.

2. Architectural model of the e-Learning portal

The portal is a WEB-site that is connected to different groups of users, according to their goals and needs. It supports the processes of analysis, synthesis, delivery of information and services based on users' personalization. To create an architectural model of the educational portal we investigated a set of schools' portals from different countries. There are different kinds of portals [27, 10]. Our team decided to adapt the model of the corporate educational portal of the Delphi group as it developed the architecture framework of the portal (Fig. 1) Our portal architecture framework includes nine different layers. Some of them contain sub-layers or are grouped in larger logical groups [18].

- ***Presentation layer*** – it is the portal architecture component that supports the user's "single point of access". The presentation layer provides the individualized professional desktop. We could characterize our portal presentation requirement in some areas as: color schemes – because colors are an important aspect for e-Learning in secondary school; layout – the portal could permit users to change the layout of the user's screen and to optimize their personal settings; dynamic – this means that the portal creates dynamic pages based on the profile of the user requesting the information.

- ***Personalization layer*** – this layer is responsible for providing information filtered specifically for the user's working style and content preferences. The request analyzer is a sub-layer of the client part, which has to analyze the user's incoming requests and, in relation to the current domain, user and pedagogical instances (profiles), can select the suitable e-Service(s). It is connected with the user model. For realization of the model we use overlay approach and hierarchy of stereotypes. Every user, according to his profile characteristics and results from the initial tests, becomes a member of a stereotype group and gets access to appropriate e-Lessons and e-Services. For example, the student from 5th class who has 95% success of the initial test becomes a member of the stereotype "Users" ->"Students" -> "5th class" ->"Expert". The next screenshots (Fig. 2) present the main desktops of the guests, students – 1st class, teachers and school administration staff.

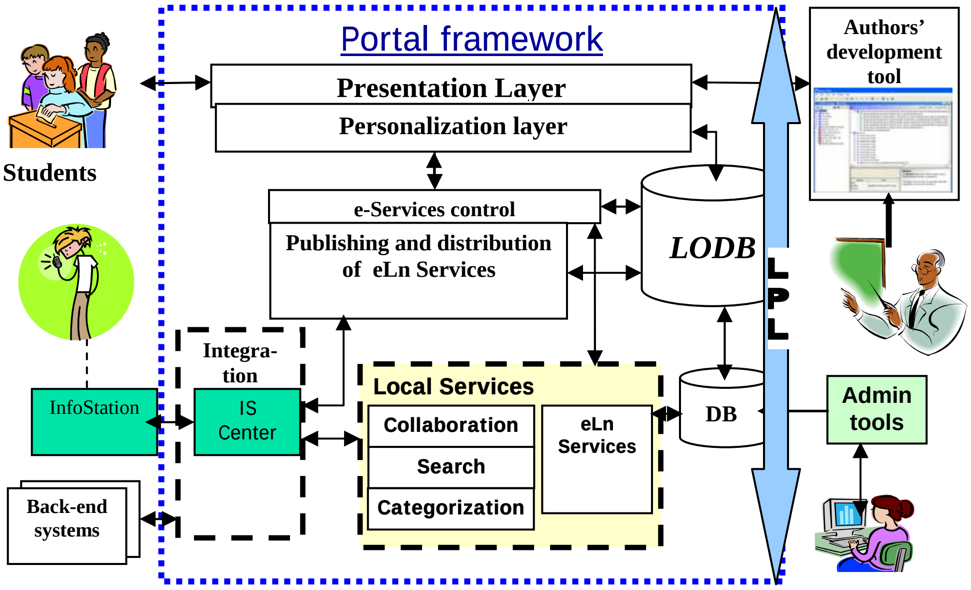

Fig. 1. Architecture framework of the educational portal

Fig. 2. Portal screenshots for different stereotype groups of users

Every user from a stereotype group can see and use resources and services from parents' stereotypes of the hierarchy.

• ***E-Services control layer*** – this layer is critical for our portal, because it is service-based. This is the component that enables the portal users to initiate and participate in online e-Business processes. We could look at the processes as: integrated – when they are supported by infrastructure external to the portal itself and are integrated into the portal services; inherent – when they are supported by infrastructure internal to the portal; communication – to support processes across wide area networks and manage multiple communication protocols; role and stereotype-based – support the processes, based on the common roles of different kinds of users (students, teachers, administrators, administration staff, etc.); e-Learning linked – support the links to remote educational portals from the e-Learning cluster. These links are very important for the exchange of e-Learning services and resources between the portals and the educational net.

The presentation of the different models and their relationships is achieved with the help of ontology. They are connected with the Local Ontology Data Base (LODB) that supplies a set of intelligent services to different groups of users.

- ***Integration layer*** – the e-Service model of DeLC includes three types of services, according to their location – local, remote, and back-end. This component supports the use of all kind of services. It is responsible for the selected e-service(s) run-time management and control and ensures appropriate Gateway Management for all kinds of services – Local Gateway, Remote Gateway and Back-end Gateway. The Info Station Center (ISCenter) is accessible in this layer. The IS Center is connected with a set of Info Stations, which provide opportunities for the use of mobile e-Services [4]. We develop different scenarios for mobile access to e-learning portal, but the results will be published additionally.

The next three layers are united into the Local services. All of them are connected to the Local DB and Local Ontology DB [21]. The services placed here are connected with the e-Learning process and they are built according to the DeLC services concept model [16]. Distributed organization of the services allows the removal or inclusion of different kinds of services avoiding the effect of these actions on the other ones. We developed special administrative tools to support the services.

- ***Collaboration layer*** – it represents an asynchronous (forums, off-line consults) and synchronous (on-line consults, chat-rooms) communication between different stereotype groups. Collaboration should focus and support different levels of communication – student-student, teacher-student, teacher-teacher, teacher-parent, etc. The next screenshot (Fig. 3) presents the actual realization of the communication teacher-teacher in the portal.

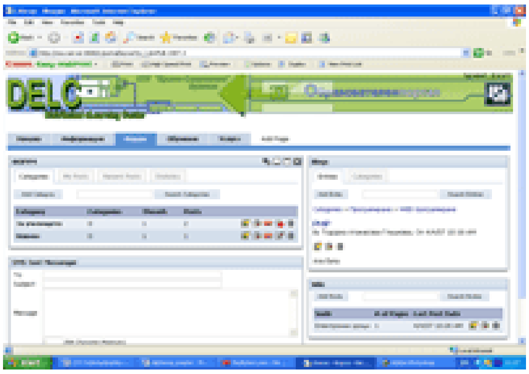

Fig. 3. Communication between teachers

- ***Search layer*** – it provides tools for identifying and accessing specific information available within or via the educational portal. The collections of information, which are connected with e-Learning, are usually quite large. There are different possibilities for implementation of a search, such as: text box search, parametric search, search based on the choices or behaviours of other users (collaborative search), natural language context-based search, and so on. Another possibility is the use of a case-based reasoning model [12, 1]. It leads the user through a series of steps to determine the most appropriate choices and helps him with recommendations for search refinement. The algorithm uses CRN-net and communicates with Web Services using XML-transactions via WSDL, UDDI and SOAP.
- ***Categorization layer*** – the educational portal supports different groups of users and provides the possibility for categorization of information according to their profiles and preferences. We could use a combination of automatic, manual, dynamic and modifiable categorizations. One of the methods for categorization of the information is the application of the rough set theory [14].
- ***Loop process layer*** – this layer differs from the other architectural elements in that it is not connected with a specific aspect of information management. It enables the educational portal to adjust easily to changes in the organizational work and information environment. The layer takes advantage of e-Services such as: authoring and posting, community, transactions, process involvement, preferences, and so on.

The architecture given is independent on the integrated packages within it - they could be easily added or removed. This is ensured by the distributed structure and the fact, that each educational portal is built as an independent unit for electronic training.

This distributed structure of the portal framework ensures:

- Realization and interaction between the main models: the user model, the domain model, the pedagogical model, the service model, etc.
- Possibilities for the creation of a SCORM RTE and management of the training process by an appropriate LMS that allows publishing, registration and removal of e-Learning packages; sharing and use of learning resources for different kinds of students according to their profiles and stereotypes; support of the submission of meta-information from SCOs, lessons, packages and authors into the on-line content repository, etc.

Realization of the infrastructure model of DeLC by Remove services ensures an appropriate interface (remote gateway) for sharing and usage of training resources and services from other e-Learning portals into the regional or national learning nets. The advantages of this net-based infrastructure model are connected with its open structure. We can create new virtual structures (e-Learning clusters.) and transform or remove current ones. These actions don't disturb the work of the other portals in the system because they are independent e-Learning items with a closed cycle of teaching. The given architectural framework makes the system independent from integrated e-Learning packages – they can be easily added or removed.

3. Parameterization of templates and models

The development process and creation of e-Lessons for students in Bulgarian schools requires keeping a lot of didactics and methodology rules that are drawn from traditional lesson structures and defined educational and cognitive aims. This is a reason why the main step of the development process is the structuring of the e-Lesson according to cognitive levels of the Bloom taxonomy [3, 31]. These definite educational aims give us possibilities for appropriate selection, structure and connection between SCOs in the lessons. The lesson structure and sequencing of its behaviours will be various for different kinds of lessons – for new knowledge, exercises, testing, etc. [5, 6]. Traditional computer-based systems allow students to pass linearly from one part of a lesson e-Content to another, between lessons or modules. The e-Learning standard SCORM, with its Sequencing and Navigation Model (S&N), allows passage between e-Learning resources, based on a set of rules, connected with didactic aims, the context, the knowledge and behaviours of the students. The LMS controls the process of training, according to the sequencing of rules, created by the author of the e-Content.

The e-Lesson is the main educational unit for projection and the cross point for all the models. For example, the lesson “The Solar System”, using the concept structure of the domain “Physics and Astronomy”, is created by ten informational SCOs, and is managed by the SCORM LMS of the educational portal. According to didactic purposes, the lesson will be used for independent training of students from 11th class. The Content Structure Diagram determines the sequencing of steps for teaching new knowledge and testing at the end of the lesson. The user model includes personal information about the student, his individual profile, the user’s learning session, his WEB Browser (IE), hardware platform (Home Desktop PC, mobile device), software platform (Windows XP, portal framework [28], SCORM RTE, WEB server…), etc. The learning portal offers suitable services according to the user profile of the student such as “Test system”, “Virtual excursion into the cosmos”, “Question game”, possibilities to get additional information from the “Virtual School Library” and more (Fig. 4).

There are different guides for the creation of standard-based e-Content. They describe the sequence of steps for development of new or transformation of e-Lessons. *The Best Practice Guide for Content Developers* of Carnegie Mellon University (BPG) [24] (BPG) supports the latest version of SCORM standard. It suggests methods for structuring of the e-Content; planning and standardization in the e-Lesson development process; determination of learning objects-SCOs; structure and creation of tests; sequencing and navigation of learning activities. The guide describes the basic package from ten templates and five models that allow us to create different kinds of lessons according to didactics aims and tasks. The Bloom's Taxonomy of Cognitive Objectives includes six levels: Knowledge, Comprehension, Application, Analysis, Synthesis and Evaluation. Table 1 presents the correlation between kinds of lessons, didactic cognitive objectives and main BPG- templates.

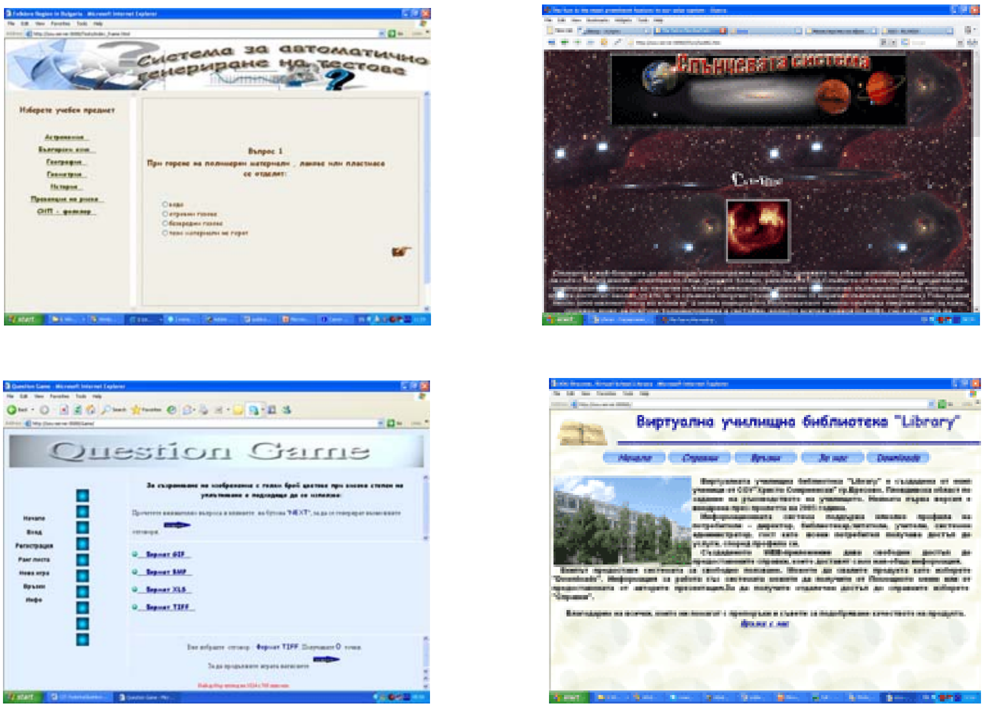
Система за автоматично генериране на тестове
Слънчевата система
Question Game
Виртуална училищна библиотека "Library"

Fig. 4. E-Learning services

Table 1

BPG Temp-lates	Description	Kind of lessons			
		New know-ledge (know-ledge, compre-hension)	Exercise and review (application, analysis)	Genera-lization (analysis, synthesis)	Test (application, evaluation)
1	**“Single SCO”** – single SCO with one Asset. It is a main template. The template can be used as a part of different models	yes			
2	“**SCO with Assets**” – single SCO with many Assets. We can use it for creation of learning resources with a single structure, containing some Assets – pictures, graphics, music, etc. – and a final test. The LMS doesn’t use the results from the test but the system stores them for future analysis		yes	yes	

Table 1 (continued)

3	“**The Black Box**” – the common structure from one SCO with a set of links and many Assets. In the LMS this structure is called “The Black Box”. We can use this template to present new knowledge when the learner must achieve a perfect test score. This template presents a typical CBT lesson	yes			
4	“**Multiple SCOs with Assets**” – many SCOs and Assets	yes	yes	yes	
5	“**Remediation Using Objectives**” – for new knowledge with testing. This template allows a learner to get additional information from informational SCOs when he makes a mistake on a SCO-test. The LMS manages this process with the Objective mechanism		yes	yes	
6	“**Pre- and post-test Sequencing**” – this template allows us to use a pretest and a posttest. According to the answers of students from the pretest, OBJs are passed or failed. If the student passes the pre-test he can pass to posttest; if he makes a mistake he must see the informational SCOs. LMS doesn’t manage results from the posttest		yes		
7	“**Pre- and posttest Sequencing – 2**” – a variant of teaching with two tests – before and after the training process. The informational SCOs are grouped in a separate Aggregation. The learner must give answers to the test-items from the pretest. If he makes a mistake he must see the respective informational SCOs from the Aggregation. Finally, he must make a posttest. The LMS manages the results from pre- and posttests by OBJs		yes	yes	
8	“**Remediation Using Objectives – 2**” – allows the LMS to control the access of the learner to the final test. He can’t pass to it if he hasn’t completed the learning process in the Aggregation. The LMS manages the results from the tests by OBJs	yes	yes		yes

Table 1 (continued)

9	"**Basic Three-Way Branching**" – we can use the intra-SCO rules that are similar of the traditional CBT lessons. According to the choice of the learner, OBJ set a value from [–1;1] and the LMS branches out the learning process according to this value. We could make a parameterization of the template according to the number of choices			yes	yes
10	"**Pre- and posttest Sequencing with New Content for Remediation**" – usage in the final stage of the learning process for control of the student's knowledge. According to the behavior of the learner and the result from the pretest, he can pass directly to the final test or see additional information and take the final test again	yes	yes	yes	yes

Our goal is to parameterize the basic templates and models. This process allows us to create a series of different templates and models according to didactic aims. For example, the content structure diagram of basic template 5 is visualized by Fig. 2. This template allows the student to get additional information from informational SCOs when he makes a mistake on a SCO3-test. The LMS manages this process with the Objective mechanism. We can parameterize this template using the next parameters:

– ***Number_of_SCOs*** – (default value is 3);

– ***Has_test*** – (default value "Yes"). When Has_test="yes", the Questions= (Number_of_SCOs – 1)= number of Objective variables;

– ***(OBJ_n, min_value_n),*** (*n* < Number_of_SCOs);

– ***(SCO-n, pattern_num)***, *n* < Number_of_SCOs and pattern_num<5. Default value (SCO1,pattern_2(1,no)); (SCO2,pattern_2(3, no));

– ***Set (SCO_Number_of_SCOs(Asset_k); OBJ_k),*** *k* < Number_of_SCOs;

– ***Read (SCO_k, OBJ_k).***

The teacher will determine the values of the parameters and the system will generate the most appropriate structure of the e-Lesson. If any values aren't selected, the system will post default values of parameters. The parameterization of the sequencing and navigation rules for template 5 according to behaviours of every student is presented in Table 2.

Table 2

Behaviours	Scorm Function
1. To complete RA the student must complete Posttest in the last SCO- (Number_of_SCOs)	RA: Rollup: All: satisfied, completed All SCO-k: isRolledUp=false, k< Number_of_SCOs SCO- (Number_of_SCOs): isRolledUp=true

2. The student must complete SCO-(k-1) before SCO-k, k< Number_of_SCOs	RA: Flow=true;
	Choice = false

3. To complete SCO- Number_of_SCOs the learner must successfully pass every OBJs	No SCORM Function
4. If the student makes a mistake in OBJ-k from SCO-(Number_of_SCOs) – passes directly to SCO-k	SCO- (Number_of_SCOs): set OBJ-k SCO-k: skip if OBJ-k passed
5. The student has two attempts for every SCOs	All SCOs : Attempt Limit=2
6. If the student makes a test in SCO(Num_of_SCOs) after the second attempt – the training is failed	No SCORM Function

For example, we can post the next values of the parameters:

- Number_of_SCOs = 10;
- Has_test = ”yes”;
- (OBJ_n;0.75), n < 10;
- (SCO_n, pattern_2(0,no)), n < 10;
- Set (SCO_10(Asset_n), Obj_n), n < 10;
- Read (SCO_n, Obj_n), n < 10.

The content structure diagram of the parameterized template 5 is visualized in Fig. 5.

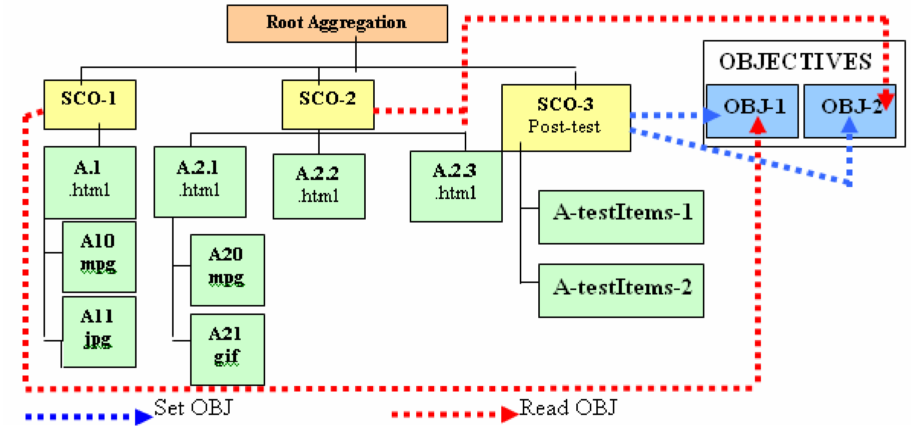

Fig. 5. Content structure diagram of the main template 5

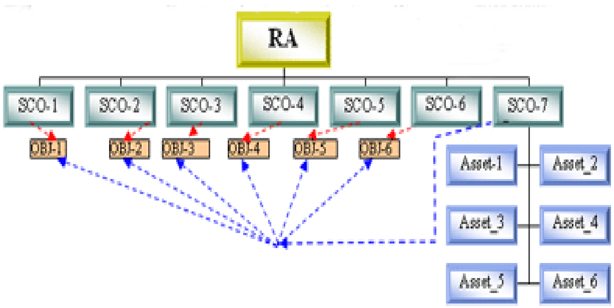

Fig. 6. Content structure diagram that a system generates for certain values of the parameters

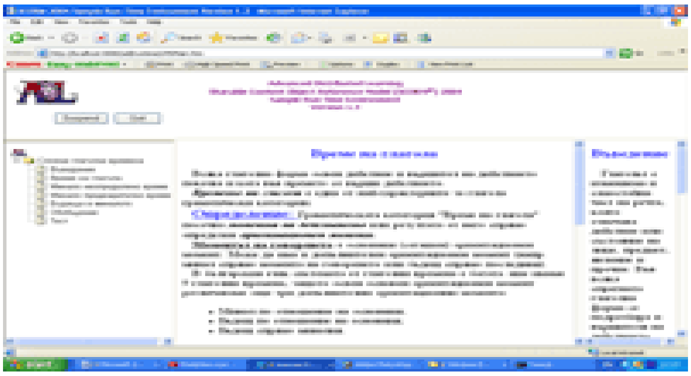

Fig. 7. Lesson based on the parameterized template 5 and the diagram from Fig. 5

The result is a set of lessons for generalization of knowledge with 6 informational objects-SCOs and final test in SCO-7 with six questions (Fig. 4). The answers of the questions are connected with respective objectives and SCOs. If the student makes a mistake, the LMS present respective SCO again.

The parameterization of the basic templates from the BPG allows us to create different variants of e-Learning resources. The change of the abstract level in the templates and models allows for the generation of different levels of e-Content: lessons, modules, courses, packages. Communication between authors and the system for determination of the values of parameters allows the system to dynamically generate the concrete model of e-Content.

4. Authoring tools and teacher's workplace

The BPG describes the participants of the development team in two main groups. The first of them includes authors of e-Learning content, the second – programmers and creators of the system. The specialists in the first group need to have only basic knowledge of the standard, but the participants in the second group should know the technical direction of SCORM. Our goal is to create appropriate, convenient tools for use by specialists from the first group – instructional designers and teachers. We can describe their roles, rights and specific characteristics as follows:

- ***Instructional Designers (ID)*** – they analyze and create content structures of the lesson after consultation with teachers. IDs could create basic learning resources (SCOs, Assets) and structure e-Lessons according to the content structure diagram, sequencing and navigation rules. They test and package SCORM-based e-Content. We created a specific environment and tools in our SCORM editor SELBO, which support the creation of SCOs, Assets, structuring of the organizational tree, creation of content structure diagrams and rules according to the behaviour of learners, to package e-learning content and creation of an XML manifest-file.
- ***Teachers*** – they are pedagogical specialists, who create e-Lessons according to specific preliminary definite didactic aims and tasks. One and the same SCOs and Assets could be used in the structure of different kinds of lessons – for different classes, students, educational subjects, forms of teaching, etc. The teacher is a specialist in some learning domain – literature, biology, chemistry, history, etc. It is not necessary to know the technical details of SCORM or computer science. Therefore, it is necessary to work with specific workplace of the SCORM editor SELBO. After the determination of didactic and cognitive aims, the development environment offers more appropriate templates and models for structuring of the e-Lesson. This process will be realized as a dialog between the teacher and his personal agent, who will suggest optimal solutions according to definite requirements [22, 19, 8]. The choice of an e-content model and the connection of the organization's three nodes with concrete learning resources (SCOs, Assets) can be realized as a step-by-step process, managed by the agent.

Our team created a model for the development of e-Learning content that includes the following steps:

- **Step 1.** Create a domain model. We structure ontology of different educational subjects. The main building elements correspond with informational items- SCOs, Assets.
- **Step 2.** Specify SCOs and Assets. The development authoring tool SELBO allows us to create new and support existing, multi-use SCOs from the on-line Content Repository. To search, choose and access more appropriate SCOs, we use Meta information stored in the database.
- **Step 3-A.** To choose more appropriate templates and models which ensure application of preliminary didactic aims and tasks in the training process. The author of an e-Learning content defines values of parameters in a dialog with his personal agent and the system dynamically generates an instance of a concrete template or model. This parameterized template includes the entire structure of the SCORM- based lesson, according to the requirements of the main SCORM models- CAM, S&N, RTE.
- **Step 3-B.** If the author of an e-Content does not use parameterize templates and models, he should create a specific structure of the lesson and define sequencing and navigations rules [15].
- **Step 4.** Create Meta data for SCOs, e-Lessons, e-Courses, e-Modules etc., which includes multi-profile information for characteristics of e-Learning content. If the author does not define any definite values, the system uses the default ones.

- **Step 5.** Make a package from the created e-Learning content and upload it into the learning portal.

We also developed the SCORM-oriented editor SELBO with two separate layers:

- Editor, which ensures a comfortable environment to create, edit, update, support, and store SCOs, to structure new and transform existing lessons. SELBO includes a set of tools for realization of the basic SCORM models (CAM, S&N, RTE) and a special module for interaction with domain ontology of different educational subjects. This layer requires a high level of knowledge for the standard and it is mainly for IDs.
- Teacher's workplace with comfortable, user friendly interface that will help teachers to create e-Lessons. We could describe the interaction between teachers and the system with the UML sequencing diagram in Fig. 8.

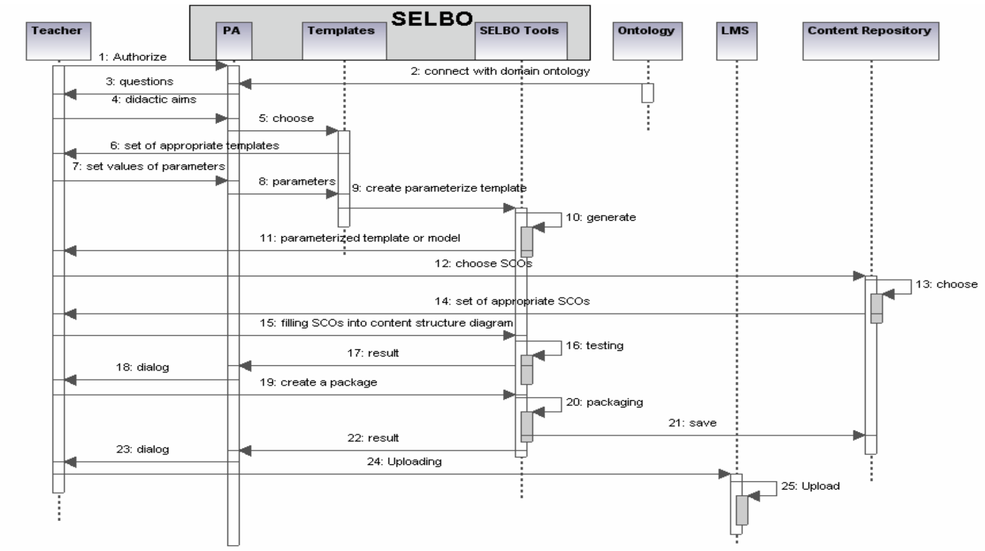

Fig. 8. UML sequence diagram

The template structure is similar to the ontology structure, because both are virtual structures, described by meta-data – characteristics, classification structures and organizations, interaction between pedagogical aims and cognitive levels of Bloom's taxonomy, etc. We would develop the next teachers' workplaces – history, Bulgarian literature, Bulgarian folklore, discrete mathematics.

5. An adaptation model of the system

The role of the adaptation model is to interpret the main models as the domain, goal, user, pedagogical and even presentation models. It can update these models and generate the appropriate presentation of e-Learning resources for every user. Typical elements of the adaptation model are condition-action rules that change variables values of the learning environment and adapt it to individual user's characteristics, knowledge background, abilities, preferences etc. We can consider three levels of the model:

- **Elementary level (EAL)** – adaptation to the static user profile information, type of the devices for access to the e-Learning portal (mobile or standard), etc. For realization of this adaptation level we use stereotypes, the CC/PP protocol, and mechanisms for realization of the mobile services that we develop with our partners from Limeric University. At this level the adaptations to pedagogical, goal and domain models are based on the stereotype user information and it results in a set of e-Lessons from typical domains, according to traditional didactics aims and pedagogical methods for this group of learners (for regular students, 3th class graduation, Math, etc.).
- **Adaptive level (AL) –** this level is connected with some mechanisms for presentation of more appropriate e-Learning materials for every student, according to his knowledge background, personal aims, interests and ambitions. This adaptation can be realized on the base of log-information for interactions between this user and LMS and the set of condition-action rules, created by the author of the e-content. Discovering of the base user knowledge is being done by a test checkout. The next step is to compare this knowledge with targets from the domain ontology and to get rules and conclude which concepts of what level must be learned. After this filtering, the system compares individual student's aims and ambitions with the pedagogical aims of the teacher and government educational standards [26]. The result is a personalized e-Lesson, which is adapted to particular characteristics of the learner (Fig. 9).
- **Dynamic adaptable level** (DAL) – the adaptation of this level is connected with the dynamic interaction between the user and the LMS. After generation of a personalized e-Lesson with appropriate learning resources, the LMS starts the individual training process as a sequence of actions that the learner must accomplish. Depending on his actions and behaviours when he interacts with the system, and based on the information stored from past e-Learning sessions, LMS dynamically adapt themselves to the changed characteristics of the environment and generates new condition-action rules for continuing of the training process.

The conceptual multilayer adaptation model of the school e-Learning system describes the interactions between different models. The student wants to learn a given topic that is connected with a set of concepts from relevant domain ontology. From a technical point of view, every concept is connected with one or more SCORM-based learning resources (SCOs), which are stored in a special on-line Content repository. The SCOs are transmitted in the relevant parameterized templates – *apart from the Pedagogical model*? By the consecutive creation of a set of rules, the free template variables are connected with the values from the previous two levels. The adaptation process results in specific actions that are managed by the LMS. According to the student's behaviors we create specific images of template rules and via the Objective mechanism of SCORM RTE, the LMS manages the level of realization of the pedagogical aims.

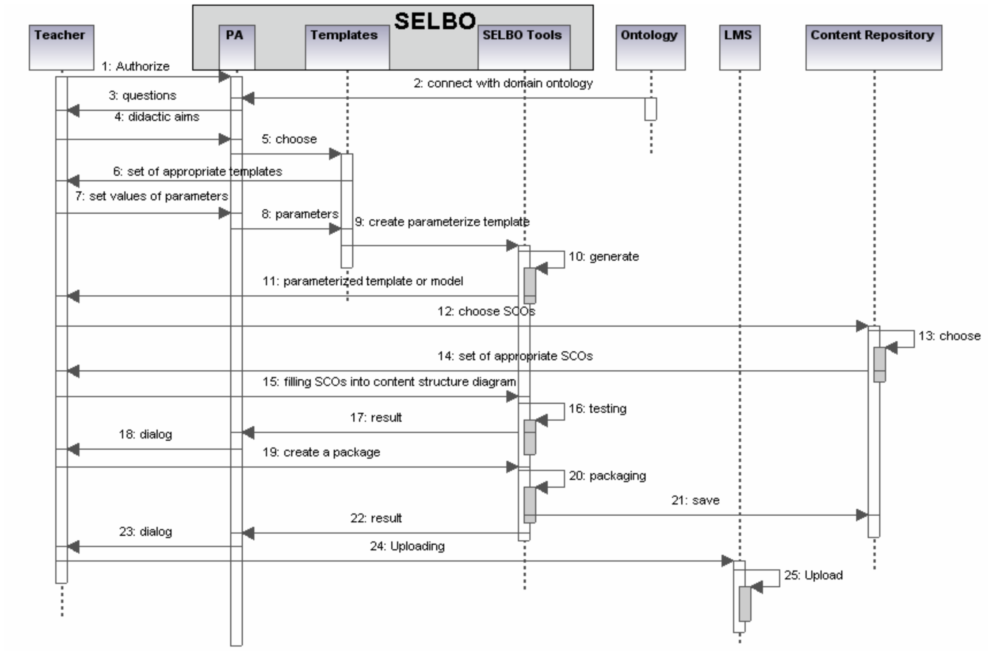

Fig. 9. Adaptation Model of the e-Learning system

6. Conclusions and future work

The report presents a conceptual model and some results for realization of the e-learning system [30]. Our future tasks are connected with the formalization of the adaptive model by UML, ITL and policies [9]. We have considered that SCORM standard allows the creation of multi-use and personalized e-Learning resources. The application of parameterized templates, scenarios and models for the design and application of SCORM-based e-Learning resources and the separation of the Goals&Tasks Model makes the Pedagogical Model more independent. The distributed model of services gives us possibilities for sharing, individualization and adaptation to the user’s preferences and requirements. The realization of the mobile services model will allow us to supply e-Learning resources and services just-in-time and from anywhere [13, 2].

Acknowledgement. This publication is partly written after the research conducted with the financial support of the Bulgarian Ministry of Education and Science under the Research project “Intelligent portal for the Secondary School in Bulgaria”, Ref. No MI-1508/2005.

References

1. Aamodt, A., E. Plaza. Case-Base Reasoning: Foundational Issues, Methodological Variations, and System Approaches. – AI Communications, **7(1)**, 1994, 39-59.
2. Brewster, M. (EDT) Dunlop, Mobile Human-Computer Interaction. – Mobile HCI 2004. Springer, ISBN-3540230866.
3. Bloom, B. S. Taxonomy of Educational Objectives. Handbook I: The Cognitive Domain. New York: David McKay Co Inc, ISBN-058232386X, 1956. 288 p.
4. Ganchev, I., M. O'Droma, S. Stoyanov, I. Popchev. Provision of Mobile Services in a DeLC. – In: Proc. of International Conference on Authomatics and Informatics, Sofia, 6-7 October 2003, 79-82.
5. Glushkova, T. Framework for e-Learning in Secondary School by Application of DeLC System. – In: 2nd International Workshop on e-Services and e-Learning, Smolian, Bulgaria, 2004, ISBN-3929757761, 1st Publ., 93-113.
6. Glushkova, T., M. Trendafilova, N. Uzunova. Application of SCORM Standard for e-Learning in the Secondary School. – In: International Scientific Conference – Informatics in the Scientific Knowledge,Varna, 2006, 128-143.
7. Glushkova, T. User Modeling of Distributed e-Learning Systems for Secondary Schools. – In: International Conference DIDMATHTECH, 2006, Komarno, Slovakia.
8. Handler, J., D. McGuiness. The DARPA Agent Markup Language.– IEEE Intelligent Systems, **15 (6)**, 2000, 72-73.
9. Janicke, H., A. Cau, F. Siewe, H. Zedan, K. Jones. A Compositional Event&Time-Based Policy Model. – In: Seventh IEEE International Workshop on Policies for Distributed Systems and Networks, 2006, ISBN-0-7695-2598-9.
10. Terra, Jose, C. Gordon. Realizing the Promise of Corporate Portals. Elsevier, 2002, ISBN-0750675934.
11. Kobsa, A. Supporting User Interfaces for All Througt User Modeling. – In: Proc. HCI International'95, Yokohama, Japan, 155-157.
12. Lenz, M., B. Bartsch-Sporl, H. Burkhard, S. Wess. Case-Based Reasoning Technology – from Foundation to Applications. – In: Lecture Notes in Artificial Intelligence 1400. Springer Verlag, 1998, p. 654.
13. O'Droma, M., I. Ganchev. Future Wireless eLearning: a Proposal for Fundamental Change to the Wireless Network Business Model and Technical Infrastructure. – In: 2nd International Workshop on e-Services and e-Learning, Smolian, Bulgaria, 2004, ISBN-3929757761, 35-47.
14. Pawlak, Z. Rought Sets: Theoretical Aspects of Reasoning about Data. Dordrecht, Netherlands, Kluwer, 1991.
15. Sarafov, M. Application of SCORM Standard for Creation of Variants e-Lessons for Bulgarian Language Training in Secondary Schools. – In: International Conference "DeLC: Solutions & Problems", Pamporovo, 2005, 19-28.
16. Stojanov, S., I. Ganchev, M. O'Droma. A Model for Integration of Services in a Distributed e-Learning Center. – In: Proc. of the 14th Annual EAEEIE Conference, Gdansk, Poland , June 2003.
17. Stoyanov, S., I. Ganchev, I. Popchev, M. O'Droma. From CBT to e-Learning, Information Technologies and Control, ISSN-1312-2622, 2005, 2-11.
18. Stojanov, S., T. Glushkova, M. Trendafilova, G. Cholakov. Adaptation of DeLC System for e-Learning in Secondary School. – In: International Scientific Conference-CompSysTech, Varna, 2005, IV-15.
19. Stojanov, S., I. Ganchev, M. O'Droma, E. Dojchev, G. Cholakov. An Approach for the Development of Agent-Oriented Distributed e-Learning Center. – In: International Scientific Conference CompSysTech, Varna, 2005, IV-13.
20. Goad, Tom. The First-Time Trainer. AMACOM Div American Mgmt Assn. 1997, ISBN-0814479421, p. 166.

21. Van Harmelen, F., Horrocks, I.. FAQs on OIL: The Ontology Inference Layer. – In: IEEE Intelligent Systems, **15 (6)**, 2000, 69-72.

22. Wooldridge, M. Agent-Based Software Engineering. – In: IEEE Proc. Of the Software Engineering, **144 (1)**, 1997, 26-37.

23. Web-Based Intelligent e-Learning Systems. Idea Group Inc (IGI). ISBN-159140729X. 2005. 388 p.

24. Best Practices Guide for Content Developers, LSAL, Carnegie Mellon University.
http://lsal.org/lsal/expertise/projects/developersguide/

25. CC/PP framework.
http://www.w3.org/TR/CCPP-struct-vocab/

26. Educational Requirements in Bulgaria.
http://www.paideiafoundation.org/norma

27. Corporative Portals.
http://www.ontoportal.org.uk/

28. Liferay Framework.
http://liferay.com/

29. SCORM 2004.
http://www.adlnet.gov

30. SCORM-Based e-Learning Portal of School-Brezovo.
http://sou-brezovo.org

31. Taxonomy of Bloom.
http://www.nwlink.com/~donclark/hrd/bloom.html