

# Ontology-Based Architecture for Providing Recommendation in Java Tutoring System

Boban Vesin<sup>#1</sup>, Aleksandra Klačnja-Milićević<sup>#2</sup>, Mirjana Ivanović<sup>\*3</sup>, Zoran Budimac<sup>\*4</sup>

<sup>#</sup>Higher School of Professional Business Studies, Novi Sad

Vladimira Perića-Valtera 4, 21000 Novi Sad, Serbia

<sup>1</sup>vesinboban@yahoo.com

<sup>2</sup>aklasnja@yahoo.com

<sup>\*</sup>Faculty of Science, Department of Mathematics and Informatics, Novi Sad

Trg D. Obradovića 4, 21000 Novi Sad, Serbia

<sup>3</sup>mira@dmi.uns.ac.rs

<sup>4</sup>zjb@dmi.uns.ac.rs

**Abstract** - The Semantic Web has opened new horizons for Internet applications in general and for e-learning in particular. Although many authors have proposed the usage of ontologies in different aspects of e-learning, such as adaptive hypermedia, personalization, and learner modeling, data reuse and system interoperability are still open issues. Only systems that completely rely on Semantic Web standards and technologies present a mean of addressing these problems. In earlier work, we presented tutoring system named Protus (PRogramming TUtoring System) that is used for learning Java programming basics. It uses principles of adaptive hypermedia and content recommendation for course personalization. Redefined architecture for Protus, supported by several ontologies is proposed in this paper. This architecture extends the use of Semantic Web concepts, where the representation of each component is made by a specific ontology, making possible a clear separation of the tutoring system components and explicit communication among them.

**Keywords** - Semantic Web, ontology, tutoring system, recommendation systems, personalisation

## I. INTRODUCTION

Semantic Web technologies seem to be a promising technological foundation for the next generation of e-learning systems [1]. *Ontology*, generally defined as a representation of a shared conceptualization of a particular domain, is a major component of the Semantic Web. The initial work on implementing ontologies as the backbone of e-learning systems is presented in [2]. Since that time, many authors have proposed the usage of ontologies in different aspects of e-learning, such as adaptive hypermedia, personalization, and learner modeling [3].

Interest in ontologies has also grown as researchers and system developers have become more interested in reusing and sharing knowledge across systems [4]. Currently, one key obstacle to sharing knowledge is that different systems use different concepts and terms for describing domains. If we could develop ontologies that might be used as the basis for multiple systems, they would share a common terminology that would facilitate sharing and reuse.

Ontologies allow specifying formally and explicitly the concepts, their properties and their relationships [5].

Furthermore, they are useful in many environments and especially in educational environments. Moreover, they permit to reuse knowledge, it is not necessary to develop ontology from scratch if another ontology is available for use in the modeling of the current domain.

In our earlier work, we presented general tutoring system named Protus (PRogramming TUtoring System) that is used and tested for learning Java programming basics [6]. System is based on principles of adaptive hypermedia and content recommendation [7], [8]. The main objective of this paper is to propose a new concept of this tutoring system that will completely rely on Semantic Web. Proposed architecture extends the use of ontology, where the representation of each component is made by a specific ontology. This way, it will be possible to promote a clear separation of concerns of the components of tutoring system, to make explicit the communication among the components, to specify the support to build the components and to emphasize the gains of the use of Semantic Web in the development of tutoring systems. This will also allow better interoperability and reusability of Protus components in the future.

The use of ontologies to model the knowledge in Protus system represents a key aspect for the integration of various information that are presented to the learner, for supporting collaboration within learning community, for improving information retrieval, and more generally, it is important for reasoning on available knowledge. In Protus, ontologies will be used to model educational domains and to build, organize and update specific learning resources (i.e. learning objects, learner profiles, learning paths, etc.) Ontology will represent the terminology of a programming domain, defining its essential knowledge. Ontologies will also be used to support semantic search, making possible to query multiple repositories and discover associations between learning objects that are not directly understandable.

Another important aspect of our approach is the use of Sharable Content Object Reference Model – SCORM [9], as a guideline format for content development, and for implementation of several ontologies.

The rest of the paper is organized as follows. In the second section appropriate related work is analyzed and discussed. In Section 3 we describe the representation of

components according to Semantic Web technologies within proposed system, and discuss use of all implemented ontologies. Section 4 concludes the paper and indicates directions of our further research.

## II. RELATED WORK

Recently a lot of researches have been focused on applying Semantic Web technologies to different aspects of e-learning [3]. Most of developed systems use ontologies only for representation of concepts, knowledge or learners' data [5], [10].

The prototype system named SMARTIES is a totally ontology-aware system which fully utilizes the qualities of ontology, computationally as well as conceptually [11]. In this moment, the ontology focuses only on the abstract design of learning contents and has not been yet related to domain knowledge or learning objects to concretize the abstract design.

The Personal Reader [12] is another important result in the e-learning field. This system also uses the Semantic Web to personalize and enrich e-learning contents. It presents a service architecture relying on RDF and ontologies to exchange information about learning resources, the domain, and learners. Architecture for personalized e-learning based on Semantic Web technologies was proposed in [13]. The authors propose usage of several ontologies for building adaptive educational hypermedia systems. Unfortunately in such systems ontology does not represent the teaching strategy's functionality of a resource.

Our research is closely related to the essences of QBL system [14]. It is web-based intelligent learning system that completely relies on Semantic Web technologies and standards. It reuses a large set of learning resources taken from the web and has been used as an online support system for lab sessions of Java programming course.

Other proposals of ontologies and their usage for several aspects of the e-learning systems, such as learner model and preferences, domain ontology, task ontology, and others, can be found in [15] - [17]. In the structure of these systems, the use of ontology focuses mainly on learning objects and their related aspects. Besides, that does not facilitate the definition and communication between the other components of the system's architecture.

Architecture for tutoring system supported by several ontologies is proposed in this paper as a way of addressing and offering acceptable solution for the mentioned problems.

## III. PROTUS ARCHITECTURE

*Protus* is a tutoring system designed to support learning processes in different courses and domains. In spite of fact that this system is designed as a general tutoring system for different programming languages, the first completely implemented and tested version was used for learning introductory Java programming course [18]. It is an interactive system with primary goal to allow learners to use teaching material prepared within appropriate introductory

programming course but also includes part for testing student's acquired knowledge.

During the last decade, increasing attention has been focused on ontologies and their use in applications related to areas such as knowledge management, intelligent information integration, education and so on. Ontology engineering, as a set of tasks related to the development of ontologies for a particular domain, offers a direction towards solving wide range of problems brought by semantic obstacles. According to that ontology engineering could be a key aspect for improvements of our already developed, in traditional manner, tutorial system. Educational ontologies such as for presenting a domain (*domain ontology*), building learner model (*learner model ontology*), presenting of activities in the system (*task ontology*), specifying pedagogical actions and behaviors (*teaching strategy ontology*), defining the semantics of message content languages (*communication ontology*) and specifying behaviors and techniques at the learner interface level (*interface ontology*) must be included in the system [1]. A repository of ontologies must be built to achieve easier knowledge sharing and reuse, more effective learner modeling and easier extension of whole system. Ontologies are structured following the SCORM (*Sharable Content Object Reference Model*) e-learning standard [9]. This ontological representation (OWL/RDF) enables not only to represent meta-data but also reasoning in order to provide the best solution for each individual learner. General ideas, of redefined ontology-based architecture for Protus have been presented in [7]. This architecture is based on experiences gained from similar web-based learning systems [19], [20], [21] and architecture for ontology-supported adaptive web-based education systems suggested in [2], [22], [23]. All of the proposed architectures are highly modular with four central components: the application module, the adaptation module, the learner model (that can communicate with each other) and domain module. In all of proposed architectures, the adaptation module is explicitly separated from the domain module, but another component is introduced in Protus as in [2] – the application module. Figure 1 depicts the general architecture of redesigned and extended version of our system.

It is important to note that the original architecture of Protus did not bring any kind of homogenous representation of components. Each one was represented by different formats, using a variety of tools. The purpose, of our current research activities, is to represent each component of the system in form of the ontology. According to that, the level of abstraction of this architecture will be higher [24].

This approach will make it easier to understand the role of each component and, consequently, to promote interoperability among the components of the architecture. The developed system is highly modular, which allows better flexibility and future replacement of various components as long as they comply with the current interface. In the rest of the section, details of construction of different kinds of ontologies employed in our system will be separately addressed.

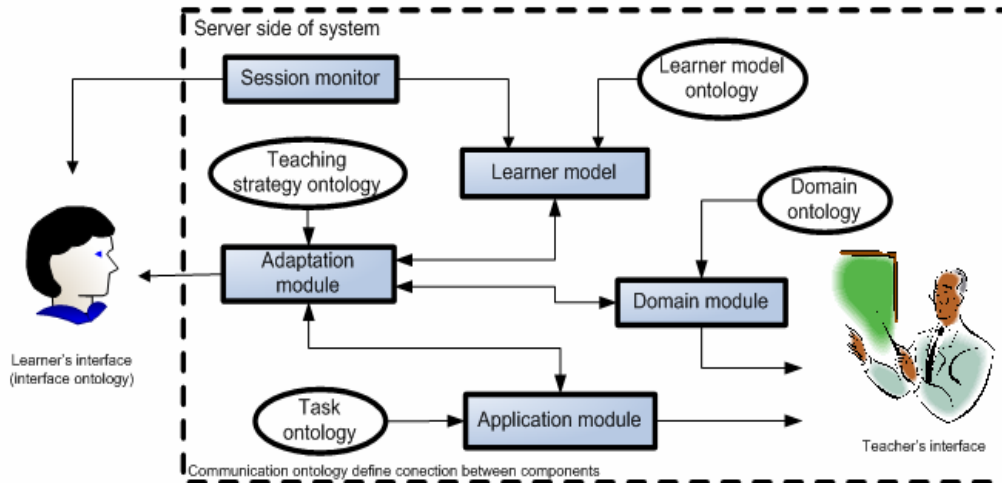


Fig. 1 Protus Architecture

### A. Domain Ontology

One of the main goals of the learning process is to understand and to acquire a body of knowledge for a given domain [25]. Domain model presents storage for all essential learning material, tutorials and tests. It describes how the content intended for learning has to be structured [26]. Often the domain model can be structured as a taxonomy of concepts, with attributes and relations connecting them with other concepts, which naturally leads to the idea of using ontologies to represent this knowledge.

An excerpt of ontology as resource topology is depicted in Figure 2. The ontology depicts resource types in the programming domain. The most general resource type is *DomainResource*. *DomainResource* has three subtypes:

*CourseMaterial*, *AdditionalMaterial* and *ExaminationMaterial*. Classes *CourseMaterial* and *AdditionalMaterial* represent the theoretical and practical explanations, respectively, that are displayed to the learners. *ExaminationMaterial* can be further specialized to *Task* and *Exam*. The *Exam* is consisted of various *Tasks* [27].

*CourseMaterial* can be further specialized into *Introduction*, *BasicInfo*, *Goals*, *Theory*, *SyntaxRule* and *ProjectAssignments*, which corresponds to the main concepts and essential elements of a programming language course.

This ontology further provides information for the *Task ontology* and the *Teaching strategy ontology* which will be explained in more details in the rest of the section.

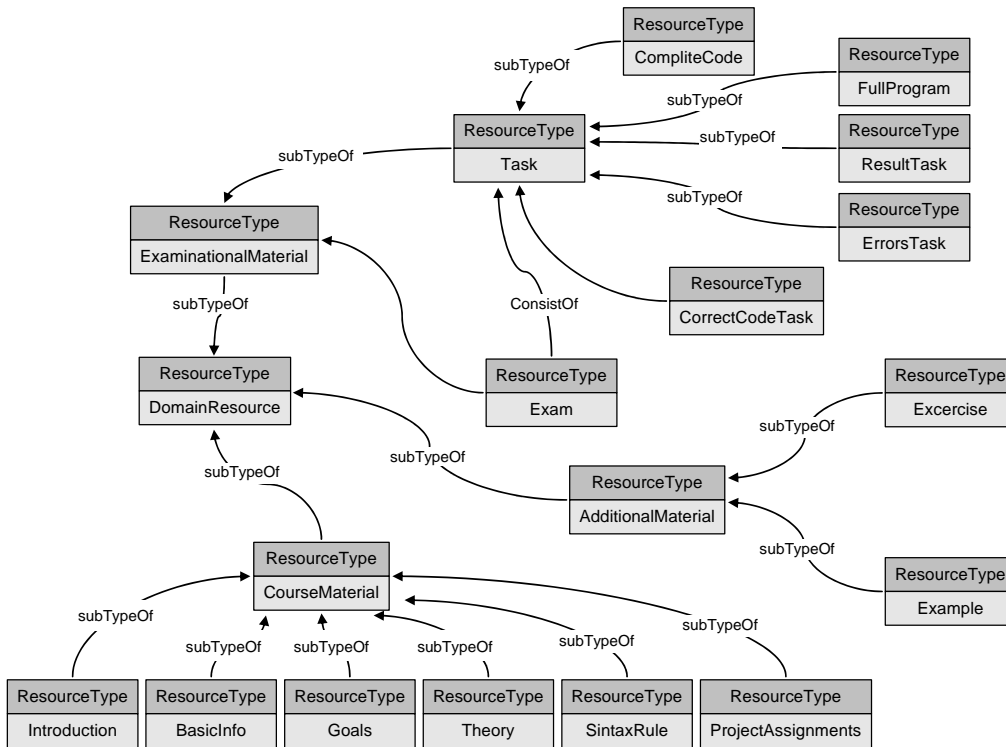


Fig. 2 An excerpt of ontology as resource topology of Protus

**B. Task Ontology**

Task ontology is a system of vocabulary for describing problem solving structure of all existing types of tasks domain independently. It complements the domain ontology in that it represents semantic features of the problem solving [1]. Task ontology specifies domain knowledge by giving roles to each object and relations between them. For instance, what role a paragraph in a textbook, can play.

Task ontology shows the role of specific concept from domain otology. For example, if concept has *fact* or *definition* role it is used to increase basic knowledge and if its role is *example*, than it is used to increase learner’s practical skills.

An excerpt of task ontology of resources in Protus is depicted in Figure 3. The ontology represents learning material grouped by the resources. The class *Concept* is used to annotate a unit of knowledge which is represented by some *Resource*.

Like in [14], concepts and resources are related by the *hasResouce* property. Concepts can be arranged by the *hasPrerequisite* property. The *hasPrerequisite* property is proposed for navigational purposes. It allows pointing out concepts that must be known before starting to study a concept, and the concepts for which it is a prerequisite. Concept will not be covered unless that the prerequisite condition is satisfied. There can be different sequence of resources that depends on navigational sequence determined for particular learner.

Concepts play certain roles in particular resource fragments. For example, some concepts represent the crucial information, while the others just represent a mean to provide

additional information or a comparison. In the proposed ontology, we represent these facts by instances of *ConceptRole* class and its two properties: *hasRole* and *supports*. For example, concepts like *BasicInfo* and *Example* have different roles. The role of the first is to represent introductory information for lesson and the role of the former is to provide additional information. On the other side, both concepts support adaptation to learner with Reflective style of learning [28]. Resource properties can be further extended by assigning a *ResourceType*. Similarly, the concept roles can be further extended by specifying their types. Concepts, their types and resources form task ontology of Protus system.

**C. Teaching Strategy Ontology**

Authoring of adaptation and personalization is actually authoring of learner models and applying different adaptation strategies and techniques to ensure efficient tailoring of the learning content to the individual learners and personalized task and navigation sequencing [15].

The major goal of learning systems is to support a given pedagogical strategy [14]. In this scope pedagogical ontologies can be associated with reasoning mechanisms and rules to enforce a given strategy. Often this strategy consists of selecting or computing a specific navigation sequences among the resources. Thus, formal semantics are required here to enable such computation.

Figure 4. shows how the adaptation is carried out by the *Teaching strategy ontology*. The decisions are drawn on the basis of the information contained in the *Condition* class (that is generated by the information about learning style and

performance of the learner) as well as teaching goals and previous behavior patterns. These conditions are composed of data coming from several other components such as *Learner model ontology*, *Task ontology* and *Domain ontology*.

Personalization presents the choice of the most appropriate learning pattern or resource that will be

recommended to the learner. This action is depending of many conditions but it implies only one decision. The decision determines what concept and resource the system is going to present for the learner.

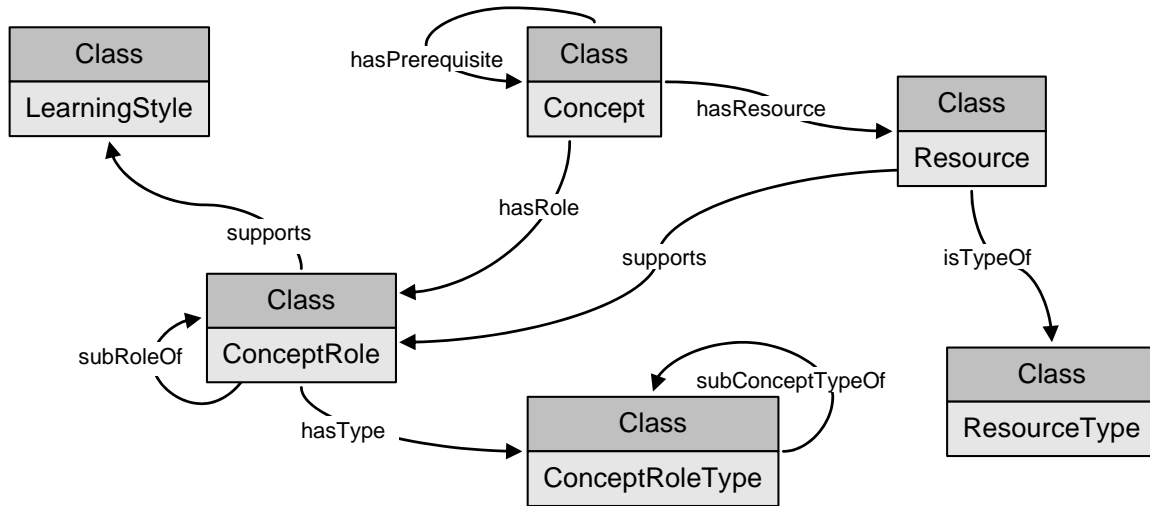


Fig. 3 An excerpt of task ontology of resources

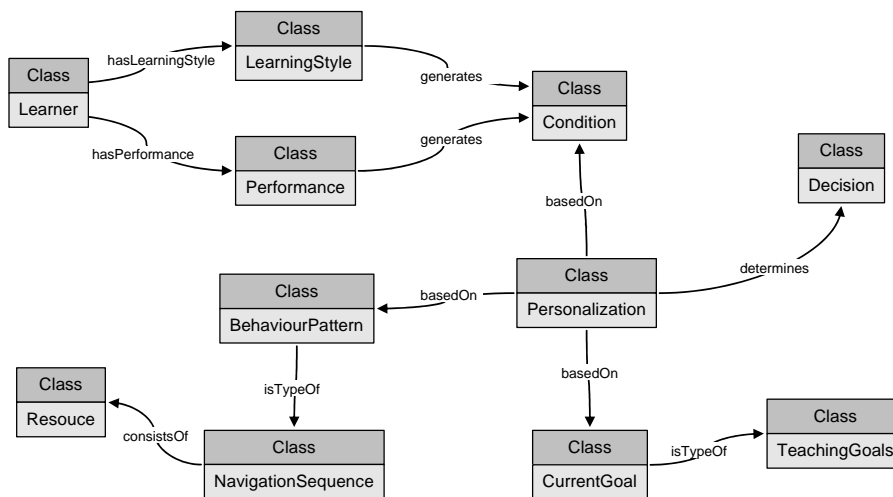


Fig. 4 Teaching Strategy ontology of Protus

*D. Communication Ontology*

Ontologies facilitate collaboration among computer systems, among people, and between computers and people. All components communicate with each other by exchanging messages [1]. Communication ontology defines the semantics of these messages and describes the vocabulary of terms used in them. Without such an ontology the terms used may not have the same meaning for all parties and confusion may occur which can lead to misunderstanding. Also, when having appropriate formal structure of the system it is possible for computers to assist in searching for needed information.

*E. Interface Ontology*

Interface ontology is the final stage of communication among the components of the architecture. Figure 5 shows the steps performed by the *Interface ontology*. It reads a decision from the *Teaching strategy ontology*, and based on that decision it creates navigation sequence of resources recommended for particular learner and generates an interface view to the learner.

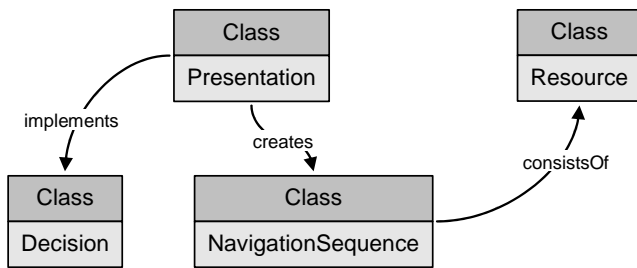


Fig. 5 Interface ontology of Protus

Interface ontology can be used to specify the content of the pages or to standardize content and query vocabulary.

#### F. Results

An ontology framework for tutoring system supported by several ontologies was proposed as a way of addressing the problems of maintenance and reuse of components. This framework extends the use of Semantic Web concepts, where the representation of each component is made by a specific ontology, making possible a clear separation of concerns of the components of tutoring system and explicit the communication among the components.

#### IV. CONCLUSION

In this paper we proposed how Semantic Web technologies and in particular ontologies can be used for building Java tutoring system. We propose a framework for such adaptive and personalized tutoring system that completely relies on Semantic Web standards and technologies. The form of several ontologies has been proposed which correspond to the components of a tutorial system. For generating presentation structures, learner interface ontology has been introduced.

From the presented survey it follows that ontologies will fundamentally change the way in which systems are constructed. Today, knowledge bases are still built with little sharing or reuse - almost each one starts from a scratch. In the future, intelligent systems developers will have libraries of ontologies at their disposals. Rather than building from scratch, they will assemble knowledge bases from components drawn from the libraries. This should greatly decrease development time while improving the robustness and reliability of the resulting knowledge bases.

The explicit conceptualization of system components in a form of ontology facilitates knowledge sharing, knowledge reuse, communication and collaboration and construction of knowledge rich and intensive systems. Ontologies are being more and more widely used for construction of systems that require an explicitly encoded knowledge. Therefore, those systems are able to interoperate better providing learners with an extended access to information resources.

The proposed architecture is highly modular, which allows better flexibility and future replacement of various components as long as they comply with the current interface. Although ontologies have a set of basic implicit reasoning mechanisms derived from the description logic which they are

typically based on (such as classification, instance checking, etc.), they need rules to make further inferences and to express relations that cannot be represented by ontological reasoning. Thus, ontologies require a rule system to derive/use further information that cannot be captured by them, and rule systems require ontologies in order to have a shared definition of the concepts and relations mentioned in the rules. For the future work we plan to present rules that will allow adding expressiveness to the representation formalism, and reasoning on the instances.

#### REFERENCES

- [1] V. Devedžić, *Semantic Web and Education*, Springer Science, New York; 2006.
- [2] R. Mizoguchi, and J. Bourdeau, Using Ontological Engineering to Overcome AI-ED Problems, *International Journal of Artificial Intelligence in Education*, 11(2), pp. 107-121, 2000.
- [3] J. Jovanović, R. Rao, D. Gašević, V. Devedžić and M. Hatala, Ontological Framework for Educational Feedback, *SWEL Workshop of Ontologies and Semantic Web Services for IES, AIED* pp. 54-64, 2007.
- [4] W. Swartout and A. Tate, Ontologies, *Intelligent Systems and their Applications, IEEE*, 14(1), pp. 18-19, 1999.
- [5] J. M. Gascueña, A. Fernández-Caballero and P. González, Domain Ontology for Personalized E-Learning in Educational Systems, *The Sixth IEEE International Conference on Advanced Learning Technologies*, pp. 456-458, 2006.
- [6] B. Vesin, M. Ivanović and Z. Budimac, Learning Management System for Programming in Java, *Annales Universitatis Scientiarum De Rolando Eötvös Nominatae, Sectio Computatorica*, vol. 31, pp. 75-92, 2009.
- [7] A. Klačnja-Miličević, B. Vesin, M. Ivanović and Z. Budimac, Integration of Recommendations into Java Tutoring System, *The 4th International Conference on Information Technology ICIT 2009 Jordan*, Paper no 154, 2009.
- [8] B. Vesin, M. Ivanović, Z. Budimac and I. Pribela, MILE - Multifunctional Integrated Learning Environment, *IADIS Multi Conference on Computer Science and Information Systems MCCSIS'2008, Amsterdam, Netherlands*, pp. 104-108, 2008.
- [9] SCORM 2004 3rd Edition - Overview, 16 November 2006, Advanced Distributed Learning. <http://www.adlnet.gov>.
- [10] C. Hee Lee, J. Hyun Seu and M.W. Evens, Building an Ontology for CIRCSIM-Tutor, *13th Midwest AI and Cognitive Science Society Conference*, Chicago, pp. 161-168, 2002.
- [11] R. Mizoguchi, Y. Hayashi and J. Bourdeau, Inside Theory-Aware and Standards-Compliant Authoring System, *SWEL Workshop of Ontologies and Semantic Web Services for IES, AIED*, pp. 1-18, 2007.
- [12] P. Dolog and W. Nejdl, SemanticWeb Technologies for the Adaptive Web, *The Adaptive Web, LNCS 4321, Springer-Verlag Berlin Heidelberg*, pp. 697-719, 2007.
- [13] N. Henze, P. Dolog and W. Hejdl, Reasoning and Ontologies for Personalized E-Learning in the Semantic Web. *Educational Technology & Society*, 7 (4), pp. 82-97, 2004.
- [14] S. Dehors and C. Faron-Zucker, QBLs: A semantic Web Based Learning System. *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pp. 795-802, 2006.
- [15] L. Aroyo and R. Mizoguchi, Process-aware Authoring of Web-based Educational Systems, *International Workshop on Semantic Web and Web-based Education SWWL*, Velden, Austria, pp. 212-221, 2003.
- [16] S. Sosnovsky, A. Mitrovic, D.H. Lee, P. Brusilovsky, M. Yudelson and V. Brusilovsky, Towards Integration of Adaptive Educational Systems: Mapping Domain Models to Ontologies, *Proceedings of 6th International Workshop on Ontologies and Semantic Web for E-Learning at ITS 2008*, Montreal, Canada, pp. 60-64, 2008.
- [17] M.C. Lee, D. Yen Ye and T.I. Wang, Java Learning Object Ontology, *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies*, pp. 538-542, 2005.
- [18] A. Klačnja-Miličević, B. Vesin, M. Ivanović and Z. Budimac, Integration of recommendations and adaptive hypermedia into Java

- tutoring system. *Computer Science and Information Systems – ComSIS* 8 (1), (in print) 2011.
- [19] C.M. Chen, Ontology-Based Concept Map for Planning a Personalized Learning Path, *British Journal of Educational Technology*, Blackwell publishing, pp. 1-31, 2008.
- [20] G. Šimić, The Multi-courses Tutoring System Design, *Computer Science and Information Systems – ComSIS* 1(1), pp. 141-155, 2004.
- [21] P.J.M. Merino and C.D. Kloos, An Architecture for Combining Semantic Web Tehniques with Intelligent Tutoring Systems, ITS, Springer-Verlag Berlin Heideberg, pp. 540-550, 2008.
- [22] P. De Bra, L. Aroyo and V. Chepegin, The next big thing: adaptive Web-based systems, *Journal of Digital Information* 5, Article No. 247, 2004.
- [23] B. Vesin and M. Ivanović, Modern Educational Tools, In Proceedings of PRIM2004, *16th Conference on Applied Mathematics*, Budva, Montenegro, pp. 293-302, 2004.
- [24] A.S. Jacinto and J.M. Parente de Oliveira, An ontology-based architecture for Intelligent Tutoring System, *Interdisciplinary Studies in Computer Science* 19(1), pp. 25-35, 2008.
- [25] L. Razmerita, T. Nabeth, A. Angehrn and C. Roda, InCA: An Intelligent Cognitive Agent-Based Framework For Adaptive And Interactive Learning, *Cognition and Exploratory Learning in Digital Age, Proceedings of the IADIS International Conference*, Lisbon, Portugal, pp. 373-382, 2004.
- [26] A. Bork, Tutorial Learning for the New Century, *Journal of Science Education and Technology*, Vol. 10, No. 1, pp. 57-71, 2001.
- [27] M. Ivanović, I. Pribela, B. Vesin and Z. Budimac, Multifunctional Environment For E-Learning Purposes, *Novi Sad Journal of Mathematics, NSJOM*, 38(2), pp. 153-170, 2008.
- [28] R. M. Felder and B.A. Soloman, Index of learning styles questionnaire. Retrieved 17 December 2009, from <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>, 1996