JOURNAL OF SOFTWARE ENGINEERING & INTELLIGENT SYSTEMS
ISSN 2518-8739
30 April 2020, Volume 5, Issue 1, JSEIS, CAOMEI Copyright © 2016-2020
www.jseis.org

ONTOLOGY DOMAIN MODEL FOR E-TUTORING SYSTEM

GHANIM HUSSEIN ALI AHMED [1], LÁSZLÓ KOVÁCS [2]

[1,2]*University of Miskolc, Hungary*
Email: ghanim@iit.uni-miskolc.hu [1]

ABSTRACT

OPEN ACCESS

Academic Editor: Hikmat Ullah Khan, COMSATS, Pakistan.

Published: 30 April, 2020

Funding: Authors are ssponsored by the European Union, co-financed by the European Social Fund.

Article Type: Research Article

E-tutoring system is a sophisticated, integrated software system that applies Artificial Intelligence (AI) concepts and methods to teaching and learning problems and requirements. They allow the student to search for level of knowledge and learning methods used to enhance or correct the students' knowledge. In a specified area of knowledge, they are intended to support and improve teaching and learning process while preserving the learner's uniqueness. E-tutoring system research has successfully delivered systems providing support for one-to-one tutoring. Most of these systems are actively used in real-world settings and have even contributed to changing traditional education. To improve the future research related to the enhancement of tutoring, we proposed an ontology domain model including besides the standard modules a common shared ontology database and knowledge-based background, too. The shared ontology databases can improve the quality of the domain models both in tutor and student models. In this paper, we give a survey on current E-tutoring systems, E-tutoring system technologies and proposed E-tutoring architecture. In this paper, we focus on domain module, our goal here is to let the e-tutor system prepare the material according to the students' capabilities. As a result, the proposed model can solve the problem of the current E-tutoring systems, can meet the features of reusability, standardization, open knowledge and flexibility, by using the ontology domain model we can avoid the problem of isolated database. The constructed ontology can be used in the future to control adaptive intelligent e-tutor frameworks.

Keywords: ontology; domain model; e-learning; e-tutoring; ontology database; knowledge-base;

1. INTRODUCTION

E-learning is not just concerned with providing easy access to learning resources anytime, anywhere, via a repository of learning resources, but is also concerned with supporting such features as personal definition of learning goals, synchronous and asynchronous communication, and collaboration between learners and between learners and instructors [1]. E-learning is a way to perform teaching and learning using computer and internet technologies. This means that the development of e-learning is a collaborative work involving a number of different people in different disciplines such as instructional design, and the subject material [1]. Therefore, the existence of knowledge gap among participants can significantly affect the development process of e-learning [1].

There are some new technologies like ontology to increase the efficiency level of e-learning models. Ontology can play an important aspect in semantic web technology and can influence the whole field of e-learning [2]. According to Gruber's definition, ontology is "a shared and common understanding of a domain that can be communicated between people and across application systems" [3].

Thus, ontologies represent a formal and consensual common knowledge of the domain of interest as agreed by the domain community instead of relying just on individual human experts. The opportunities arising with the availability of domain ontologies have an impact on the development of the e-learning. Our research is concerned with developing of a general ontology domain module for E-tutoring system, this domain ontology module can be used among others as a basis for developing adaptive e-learning and to evaluate the efficiency of the training processes.

The ontology can be used to control the workflow in the e-learning management systems. The domain ontology also provides a tool for formal specification and validation of e-learning to be communicated across people and applications. The paper is organized as follows: the second section shows the related work and definitions of some terminologies, the third section defines the proposed ontology domain model, and the fourth section explains our proposal on Ontology domain Model of E-tutoring system in details. We present some case studies on ontology implementation using Python in the fifth section.

Corresponding Author: Ghanim Hussein Ali Ahmed
Email Address: ghanim@iit.uni-miskolc.hu

2. LEARNING MANAGEMENT SYSTEM

Although there are many definitions of LMS in the literature. Among them the most popular is given by [4] A Learning Management System (LMS) is a software application for the administration, documentation, tracking, reporting and delivery of e-learning education courses or training programs. It allows easy management of the learning materials and tracking of user's learning progress and results [4]. LMS is typically a large system that includes many features such as: access control, personalized working spaces, assessment tools, searching facility, content creation tools, information distribution mechanisms (i.e. Calendar, News announcements), or communication and collaboration functions (i.e. email, chat, discussion forum) [4]. According to [5], Learning Management Systems (LMSs) can be defined as web-based systems that enable teachers and students to share materials, to submit and return assignments and to communicate online. LMS is the most widely used software to plan, implement and evaluate specific learning processes [6].

LMS can be defined, according to the user perspective, as a virtual environment that aims to simulate face-to-face learning environments with the use of Information Technology [7]. In LMS, the interaction happens through devices that enable communication either synchronously or asynchronously, allowing the creation of different strategies to encourage a dialogue and active participation of students. LMSs are web-based systems that enable teachers and students to share materials, to submit and return assignments and to communicate online [5]. LMS is used by audiences like students, teachers and administrators. Also, LMS can be used by anyone who is interested in conducting the online classes and who wants to store and retrieve the student's documents. There are four critical issues that govern the success or failure of LMS. They are Scalability, Compatibility, Reusability, and High Availability. Other characteristics that define an LMS are:

a) Data Management: Data including text, animations, videos, static pictures, etc. is managed in a great amount.
b) Processing Load: A significant proportion of processing load needs to be distributed from the Web server to the Web client.
c) Personalization: Different users should not have access to the same data. The data view should be customized for various users.
d) Intelligent Agents: These incorporate intelligent web agents that tailor information customized for individual users. e) Meaningful and Unlimited Tags: HTML has a limited set of pre-defined tags and all these tags cannot define the content in a human readable form.
e) Sharable Courseware Object Reference Model (SCORM) Compliancy: An LMS has to be developed in compliance with SCORM which requires that the content developed be reusable and easily shared between different LMS.

Nowadays there are many technologies available to manage the educational systems, among them ontology is the best technology can be used to manage and represent the material, because ontology can be used for knowledge representation and management and show the relationship between the concepts.

We can find many definitions of ontology in the literature. Among them the most popular is given by [8], ontology is an effective technology that enables integration of related resources, sharing the right knowledge and avoids irrelevant information. ontology is the basic description of things in the world. In information science, an ontology refers to an engineering artifact, constituted by a specific vocabulary used to describe a certain reality [9]. According to [10], ontology in computer science is a formal representation of the knowledge using a set of concepts and relationships between those concepts. Ontology is used both to reason in target domain and to validate the created semantic model. In theory, ontology is a "formal, explicit specification of a shared conceptualization [3]".

Ontology provides a shared vocabulary, which can be used to model a domain including the type of objects, related concepts and their properties and relations. In [11] the author investigates ontology languages as a classifying the models according to the applied knowledge representation formalism: enrich first-order predicate languages, frame-based approaches, and description logics. The common language for ontology modeling and reasoning is Web Ontology Language (OWL), which is a language based on description logic. OWL language recommended by the World Wide Consortium in 2004. In [8] the authors define OWL is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be exploited by computer programs, e.g., to verify the consistency of that knowledge or to make implicit knowledge explicit [8].

3. E-TUTOR SYSTEMS

E-tutor Systems have offered the vision of a fully integrated learning experience nearly 30 years ago but given millions of dollars spent and promising student learning outcomes they have not succeeded [12]. E-tutor System is a computer system designed to provide learners with immediate and personalized instruction or feedback, usually without a human teacher's intervention. Varies researchers, designers, and developers define ITSs in different ways. According to authors in [13], E-tutor Systems can be viewed as “a determination to capture in computer technology the capabilities and practices of a human instructor who is expert in the subject matter as well as one-to-one tutoring." Scientists have struggled from the earliest days of computers to build smart tutoring systems that are as successful as human tutors [14]. E-tutor systems also known as the online tutor or e-moderator - are directed to facilitate student activities [15]. In many online classrooms the e-tutor will therefore be a different person than the teacher in charge [15].

E-Tutor can be a comprehensive digital learning tool that accelerates students, teachers and principal & management collaboration and interaction anytime, anywhere, on any smart device. Proven to be effective in bridging the gaps, this creative tool employs modern technology and communication aspects in order to deliver digital learning/teaching experience. Smart Tutoring (ST) is a web-based intelligent tutoring system designed to adaptive teaching strategies, student models that are based on background knowledge and skills, and teaching approaches suiting specific skill sets and the cognitive model of instructors, leads them to retrieve their previous teaching experiences by select one or more that are more close to the current situation, and adapt them for reuse [16].

4. RELATED WORK

There are a few E-tutor systems intended and designed for educational purposes. These systems assist learners to learn rapidly and increase their self-confidence. Most of these systems designed for solving specific problem by using isolate database, the problem of these systems is the lack of reusability, flexibility, standardability, and shareability. Here we mentioned some of them. In [17] the researchers developed and designed E-tutor system named JO-Tutor for helping students to learn Java programming language. Mohanad and Samy [18] Designed a knowledge-based Intelligent Tutoring System named MDB for teaching Mongo Database by using ITSB authoring tool. The system was designed to facilitate learning Mongo Database [18]. ITSB tool contains four modules: domain model, teaching model (expert model), student model and user interfaces [18].

In [19] the authors developed an intelligent tutoring system for English grammar using ITSB tool. The system helps students to study English grammar to students and overcome the difficulties they face with ease and smoothness [19]. The authors in [20] proposing an innovative architecture for a tag-based recommender system that is adapted to the adaptive and intelligent web-based programming tutoring system called Protus (PRogramming TUtoring System), taking into consideration the learner's pedagogical aspects. And also proposing an approach that can be used to enhance RSs functionality that aims to adapt to individual learners [20]. The authors in [21] presents how to build a Java tutoring system using semantic web techniques and, in specific, ontologies. They also introduce a fresh strategy to efficient personalization highly based on Semantic web techniques performed in new version of the system, called Protus 2.0 [21].

The domain model includes a representation of the knowledge to be learned, provides input into the expert module and ultimately is used to produce detailed feedback, problem selection/generation guidance and as a basis for student model [22]. The domain model can take many forms, depending on the representation of knowledge used, the domain it represents, and the level of granularity. In accomplishing such ingredients, a large chunk of E-tutoring system development time is supposedly consumed. We need to have a standard generic template-based domain module that can simply be filled in by a subject expert and get a rapidly developed E-tutoring system ready to use [22]. In Cognitive tutors, the domain model includes low-level production rules that fully describe the expected student behavior down to the components of atomic thought, while Constraint-based systems describe the potential valid statements that a response may take [22].

4.1 Ontology models

Current ITS systems are based on some key functional components. The systems usually contain the following modules (Figure 1):

1. domain module to describe the course topics including the knowledge units and competencies
2. student module to model the behaviour of the students
3. teacher module to model the behaviour of the tutor
4. course repository to store the different study aids and course materials
5. interface module to implement intelligent, human oriented interface to the system

This structure can be observed among others in Intelligent Tutoring System Builder (ITSB) [23] or in Cognitive Tutor Authoring Tools (CTAT) [24].

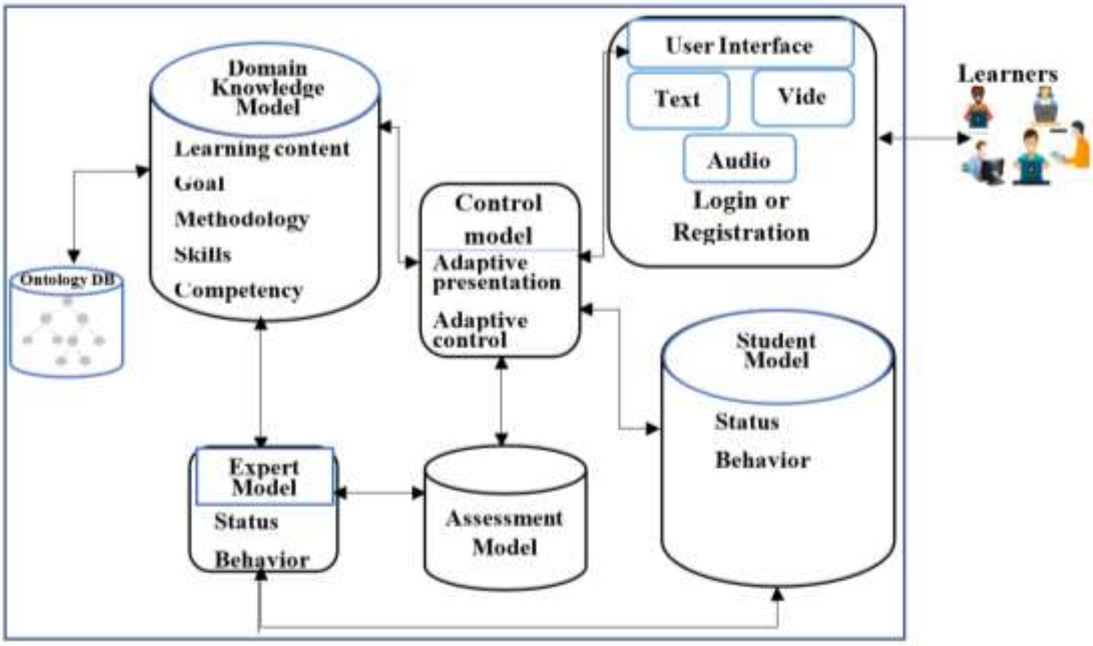


Figure. 1 Proposed e-tutor architecture model

5. THE PROPOSED ONTOLOGY MODEL

Based on the current research we suggest domain module shown in Figure 2. The benefits of this module are to prepare and update the material to students. The implementation of this module is based on a new way of integration including existing methodologies and algorithms. Figure 1 shows the proposed general model of domain module.

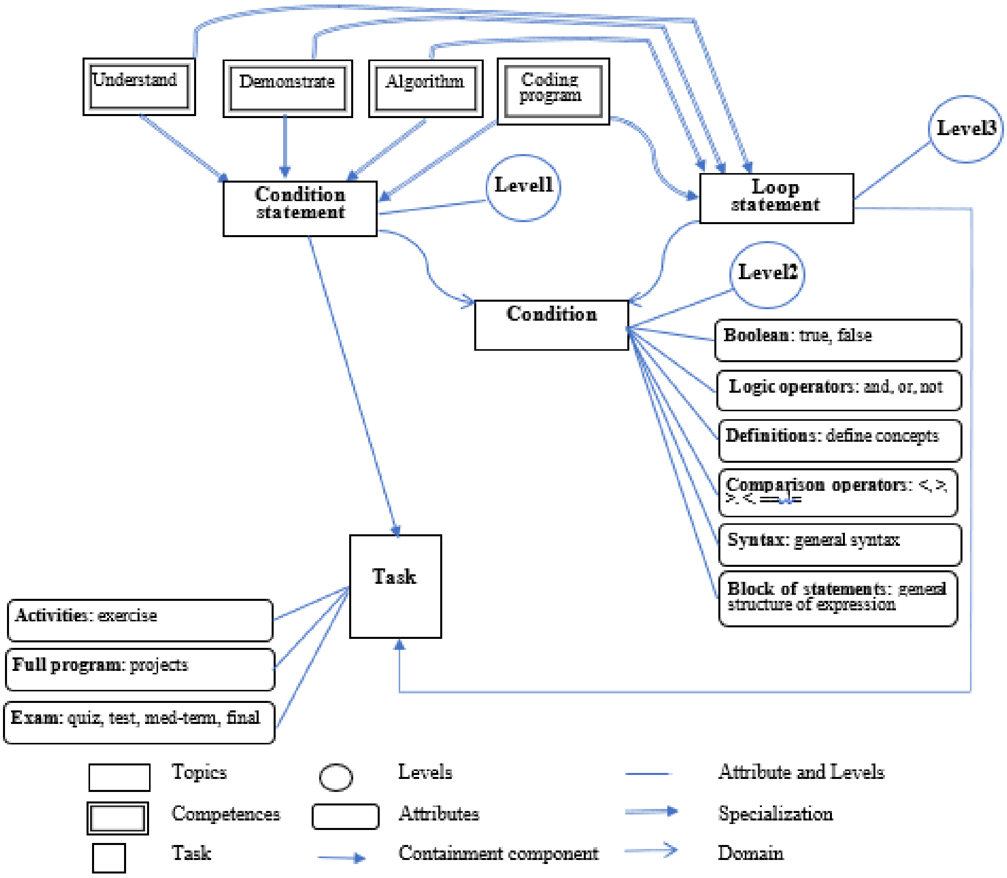


Figure. 2 Proposed general domain model

In the proposed domain model, the following terms means:

- The topic is used to define the domain knowledge or a comprehensive information unit about specific subject or course.
- The competence is used to define and demonstrate characteristics and skills that enable, and improve the efficiency of, performance of a student to understand the specific topics and got new knowledge.
- The task is used to define how can student finished an activity that needs to be accomplished within a defined period of time.
- The level is used to define how the E-tutoring system can determine the student knowledge.
- The attribute is used to define the attribute of specific topic or domain in specific domain model.
- The containment component is used to define the specific topic contain different components in the domain.
- The specialization is used to define that some topics or domains have specific characteristics
- The domain is used to define the knowledge to be learned and Contains information teaching material.

In order to improve the sharing and reusing of domain knowledge in intelligent tutoring systems, we use ontologies to represent and organize domain knowledge. According to the properties of teaching materials, we use three kinds of ontologies: subject ontology, knowledge concept ontology and learning material ontology to represent domain knowledge.

Domain knowledge (DK) is a collection of proposals that clarify all the concepts in the vocabulary to discuss or solve problems. DK is only declarative, and it does not say how students can use domain knowledge to solve a practical problem [25].

This component of Domain Knowledge (DK) contains information that the tutor is teaching, and is the most important because without it, there would be nothing to teach the student. It requires significant engineering knowledge to represent a domain so that it can be reached by other sections of the tutor. The design of the domain includes the following topics as shown in Figure 3.

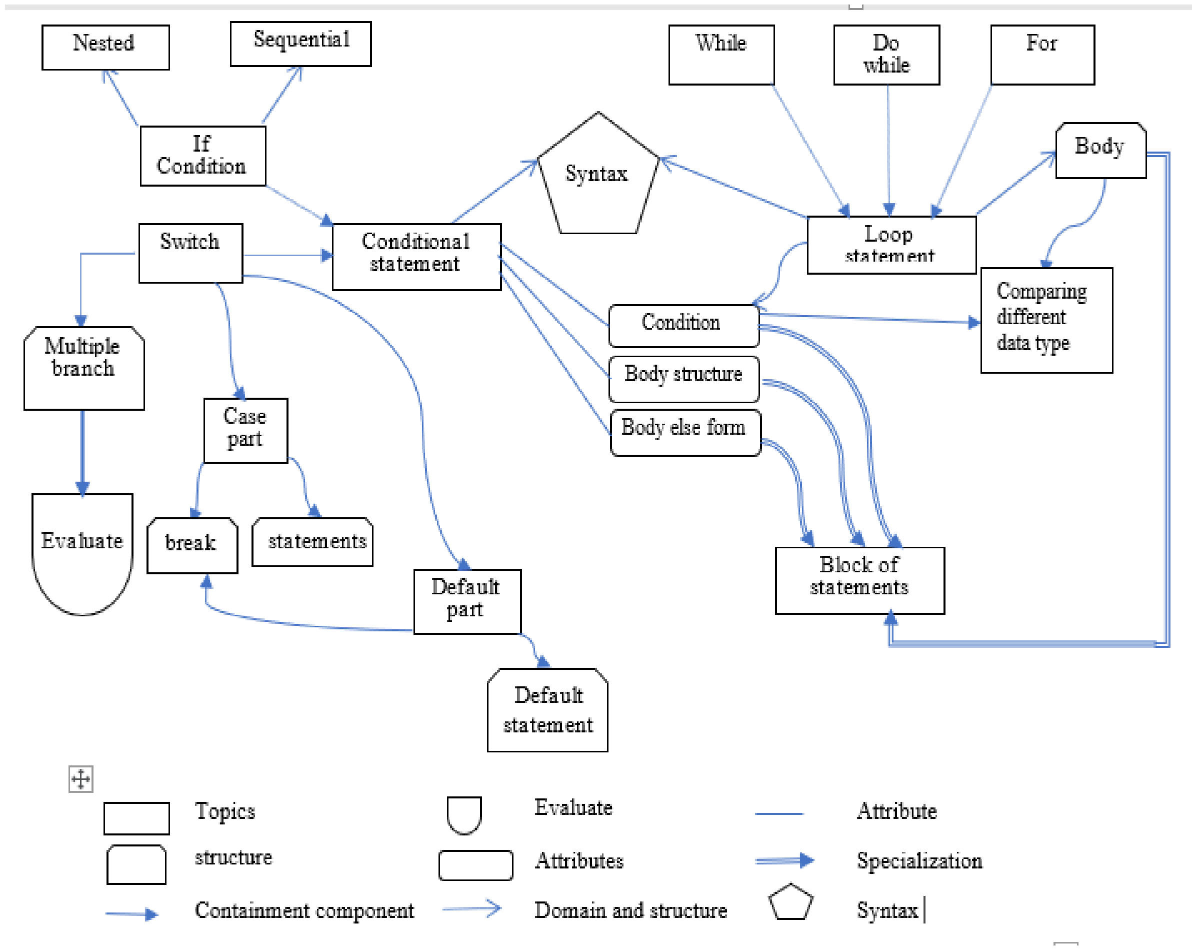


Figure. 3 Proposed domain knowledge model

In the proposed domain knowledge model, the following terms means:

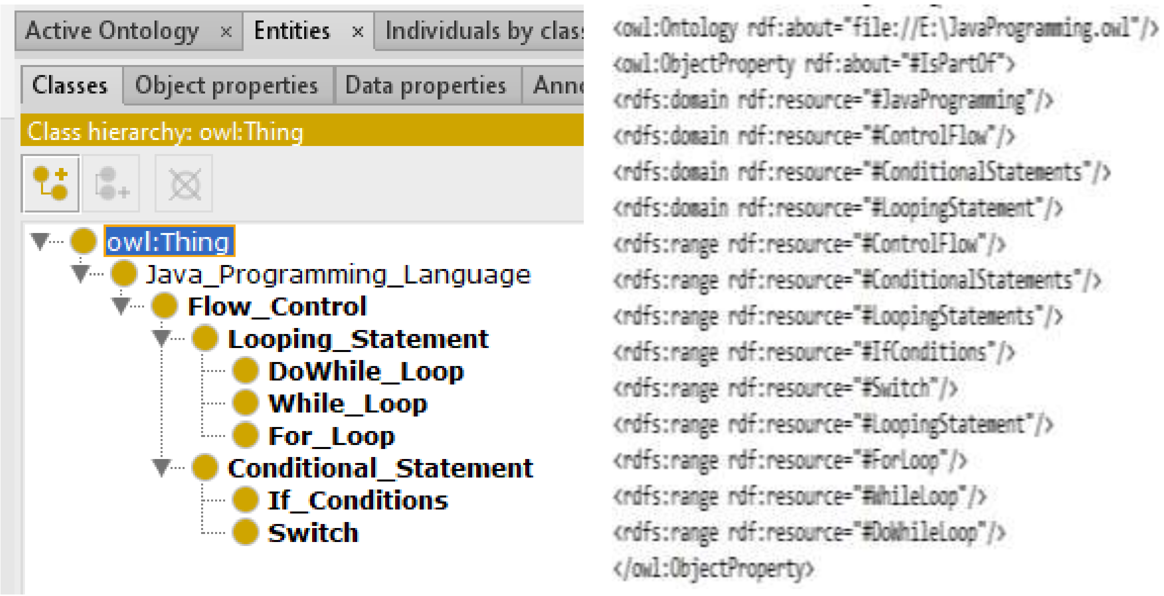
- The topic is used to present the specific domain knowledge or a comprehensive information unit about control flow in Java programming language.
- The structure is used to define the structure of conditional statements and looping statements in Java programming language.
- The containment component is used to define the specific topic contain different components in Java programming language.
- The evaluation is used to define how the system can evaluate the branch of statements in expression.
- The attribute is used to define the attribute of conditional and looping statement in Java programming language.
- The syntax is used to define the syntax of the conditional and looping statements.
- The specialization is used to define that some knowledge or domains have specific characteristics in Java programming language.

The domain is used to define the knowledge to be learned and Contains information teaching material about Java programming language.

6. DOMAIN KNOWLEDGE IMPLEMENTATION USING PYTHON

Regarding the implementation of an ontology Python is one of the most common language used. It is an interpreted object-oriented, extensible programming language [26], which provides an excellent combination of clarity and versatility in different disciplines. In information science, it offers many modules and package for management and implementing the ontology.

The knowledge base domain, considered here, is “Java Programming Language”, the ontology created consisted of subclasses of the Flow Control. Figure 4 shows the classes and the subclasses.

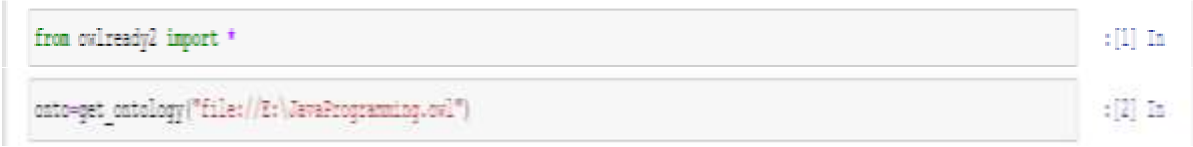


```
<owl:Ontology rdf:about="file://E:\JavaPrograming.owl"/>
<owl:ObjectProperty rdf:about="#IsPartOf">
<rdfs:domain rdf:resource="#JavaPrograming"/>
<rdfs:domain rdf:resource="#ControlFlow"/>
<rdfs:domain rdf:resource="#ConditionalStatements"/>
<rdfs:domain rdf:resource="#LoopingStatement"/>
<rdfs:range rdf:resource="#ControlFlow"/>
<rdfs:range rdf:resource="#ConditionalStatements"/>
<rdfs:range rdf:resource="#LoopingStatements"/>
<rdfs:range rdf:resource="#IfConditions"/>
<rdfs:range rdf:resource="#Switch"/>
<rdfs:range rdf:resource="#LoopingStatement"/>
<rdfs:range rdf:resource="#ForLoop"/>
<rdfs:range rdf:resource="#WhileLoop"/>
<rdfs:range rdf:resource="#DoWhileLoop"/>
</owl:ObjectProperty>
```

Figure. 4 Class hierarchies of control flow for ‘Java Programming Language’ in Protégé

There are many languages used for building ontology domain model like XML, RDF, RDF(s), OWL, and DAML+OIL, the most common used is Web Ontology Language (OWL) [27]. According to W3C [28], OWL is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be exploited by computer programs, e.g., to verify the consistency of that knowledge or to make implicit knowledge explicit

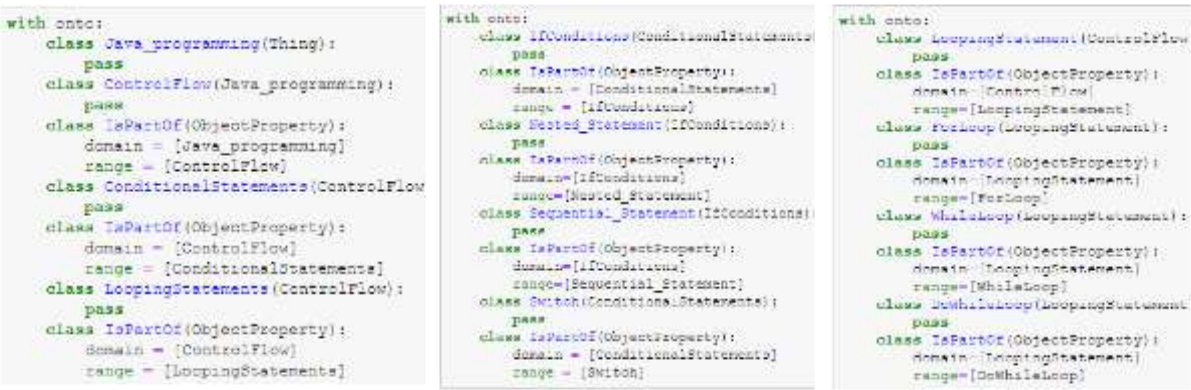
Python object model can be integrated with an OWL ontology using Python module called Owlready2. Owlready2 is used to obtain transparent access to ontologies, manipulate the classes and the individuals object properties, data properties, annotations, property domains and ranges, constrained datatypes, disjoints and class expressions (such as intersections, unions, property value restrictions etc.). Figure 5 to 7 shows a simple example, to create an ontology.



```
from owlready2 import *                                   :[1] In

onto=get_ontology("file://E:\JavaProgramming.owl")        :[2] In
```

Figure. 5 Creating ontology on Python

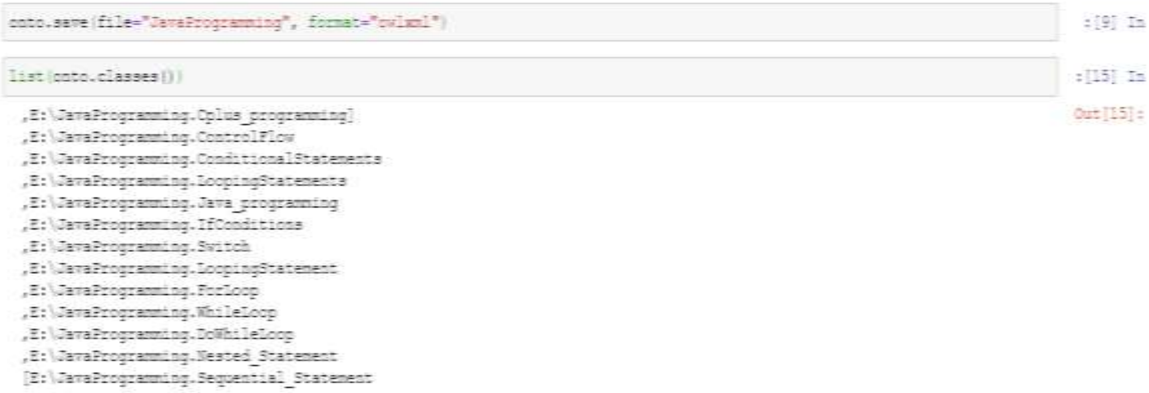


```
with onto:
    class Java_programming(Thing):
        pass
    class ControlFlow(Java_programming):
        pass
    class IsPartOf(ObjectProperty):
        domain = [Java_programming]
        range = [ControlFlow]
    class ConditionalStatements(ControlFlow
        pass
    class IsPartOf(ObjectProperty):
        domain = [ControlFlow]
        range = [ConditionalStatements]
    class LoopingStatements(ControlFlow):
        pass
    class IsPartOf(ObjectProperty):
        domain = [ControlFlow]
        range = [LoopingStatements]
```

```
with onto:
    class IfConditions(ConditionalStatements
        pass
    class IsPartOf(ObjectProperty):
        domain = [ConditionalStatements]
        range = [IfConditions]
    class Nested_Statement(IfConditions):
        pass
    class IsPartOf(ObjectProperty):
        domain=[IfConditions]
        range=[Nested_Statement]
    class Sequential_Statement(IfConditions)
        pass
    class IsPartOf(ObjectProperty):
        domain=[IfConditions]
        range=[Sequential_Statement]
    class Switch(ConditionalStatements):
        pass
    class IsPartOf(ObjectProperty):
        domain = [ConditionalStatements]
        range = [Switch]
```

```
with onto:
    class LoopingStatement(ControlFlow
        pass
    class IsPartOf(ObjectProperty):
        domain=[ControlFlow]
        range=[LoopingStatement]
    class ForLoop(LoopingStatement):
        pass
    class IsPartOf(ObjectProperty):
        domain=[LoopingStatement]
        range=[ForLoop]
    class WhileLoop(LoopingStatement):
        pass
    class IsPartOf(ObjectProperty):
        domain=[LoopingStatement]
        range=[WhileLoop]
    class DoWhileLoop(LoopingStatement
        pass
    class IsPartOf(ObjectProperty):
        domain=[LoopingStatement]
        range=[DoWhileLoop]
```

Figure. 6 Snapshot of for building classes and subclasses



```
onto.save(file="JavaProgramming", format="owlxml")        :[9] In

list(onto.classes())                                      :[15] In

  ,E:\JavaProgramming.Cplus_programming]                  Out[15]:
  ,E:\JavaProgramming.ControlFlow
  ,E:\JavaProgramming.ConditionalStatements
  ,E:\JavaProgramming.LoopingStatements
  ,E:\JavaProgramming.Java_programming
  ,E:\JavaProgramming.IfConditions
  ,E:\JavaProgramming.Switch
  ,E:\JavaProgramming.LoopingStatement
  ,E:\JavaProgramming.ForLoop
  ,E:\JavaProgramming.WhileLoop
  ,E:\JavaProgramming.DoWhileLoop
  ,E:\JavaProgramming.Nested_Statement
  [E:\JavaProgramming.Sequential_Statement
```

Figure. 7 Snapshot of for displaying classes and subclasses

The python programming language is efficient tool to combine the ontology management and machine learning functions to develop ITS systems.

7. CONCLUSION

Intelligent E-Learning and Tutoring Systems are the key technologies in supporting educational activities. In this paper, we developed an ontology domain module for e-tutor systems. Based on the common requirements we have developed an adaptive Intelligent Tutoring System architecture (ITS). The proposed architecture includes beside the standard modules a common shared database and knowledge-based background, too. Benefits of the shared database are to share common understanding of the structure of information, to reuse the data and to mix different knowledge sources. Instead of the traditional human generated behavior models machine learning can provide more flexible behavior models using the available training bases.

In this work also we give a very short survey on current research on E-tutor systems and related problems. and we argue that knowledge representation and ontologies may offer solutions to basic problems in this area. Since a well-founded system of concepts, i.e. an ontology, knowledge sharing and interoperability will significantly increase. This will make it possible to design both reusable functional components and to design E-tutor systems. Also, we introduce and discuss our approach of organizing domain knowledge in an ontology. Finally, we propose an approach how ontologies and domain knowledge model may be combined for improving both user modeling and intelligent problem-solving support.

ACKNOWLEDGEMENT

The described study was carried out as part of the EFOP-3.6.1-16-00011 “Younger and Renewing University – Innovative Knowledge City – institutional development of the University of Miskolc aiming at intelligent specialization” project implemented in the framework of the Szechenyi 2020 program. The realization of this project is supported by the European Union, co-financed by the European Social Fund.

REFERENCES

1. T. A.-E.-H. a. A. B. Tarek M. Mahmoud, "*A Framework For An E-Learning SystemBased on Semantic Web,*" IJCSE, vol. 5, no. 8, pp. 698-710, Aug 2013.
2. E. KATIS, "*Semantic Modeling of Educational Curriculum & Syllabus*," EVANGELOS KATIS, 2018.
3. T. Gruber, "*A translation approach to portable ontologies*," Knowledge Acquisition, vol. 5, no. 2, pp. 199-220, 1993.
4. R. K. Ellis, "*Field Guide to Learning Management Systems,*" 2009. [Online]. Available: http://web.csulb.edu/~arezaei/ETEC551/web/LMS_fieldguide_20091.pdf. [Accessed Thrusday February 2020].
5. S. D. T. Steven Lonn, "*Saving time or innovating practice: Investigating perceptions and uses of Learning Management Systems,"* Computers & Education, vol. 53, no. 3, pp. 686-694, 2009.I.
6. S. N. Z. N. &. A. M. Almrashdeh, "*Distance learning management system requirements from student's perspective,"* JATIT, pp. 17-27, 2011.
7. C. d. A. C. a. M. K. N. Paulo Cristiano de OliveiraCristiano Jose, "*LEARNING MANAGEMENT SYSTEMS (LMS) AND E-LEARNING MANAGEMENT: AN INTEGRATIVE REVIEW AND RESEARCH AGENDA,"* JISTEM, vol. 13, no. 2, pp. 157-180, 2016.
8. G. S. B. Sunitha Abburu, "*A Framework for Ontology Based Knowledge Management,"* IJSCE, vol. 3, no. 3, pp. 21-25, July 2013.
9. F. Fonseca, "*The Double Role of Ontologies in Information Science Research,"* Journal of the American Society for Information Science and Technology, vol. 58, no. 6, pp. 786-793, 2007.
10. D. Man, "*Ontology in Computer Science,"* DIDACTICA MATHEMATICA, vol. 31, no. 1, pp. 43-46, 2013.
11. L. Razmerita, "*An Ontology-Based Framework for Modeling User Behavior—A Case Study in Knowledge Management,"* IEEE , vol. 41, no. 4, pp. 772 - 783, 2011.
12. M. B.I., "*Intelligent Tutoring Systems: The Practical Implementation of Constraint-based Modelling,"* Ph.D. thesis, University of Canterbury, Canterbury, 2001.
13. K. B. a. X. H. Robert Sottilare, *Design Recommendations for Intelligent Tutoring Systems:*, Orlando, Florida.: U.S. Army Research Laboratory , 2015.
14. K. VanLehn, *"The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems, Educational Psychologist,"* Routledge , vol. 46, no. 4, pp. 197-221, 2011.
15. J. C. a. A. C. Annegret Goold, *"An examination of the role of the e-tutor,"* Australasian Journal of Educational Technology, vol. 26, no. 5, pp. 704-716, 2010.
16. S. Gamalel-Din, *"The Smart Tutor: Student-Centered Case-Based Adaptive Intelligent e-Tutoring,"* in International Conference on Informatics and Systems, Cairo (Vol. 17, p. 20), June 2002.
17. S. A. A. A.-M. N. D. A. M. E. &. A. M. Abu-Naser, *"An Intelligent Tutoring System for Learning Java Objects,"* IJAIA, vol. 2, no. 2, pp. 68-77, 2011.
18. M. M. H. a. S. S. A. NASER, *"Knowledge-based Intelligent Tutoring System for Teaching Mongo Database,"* EUROPEAN ACADEMIC RESEARCH, vol. 4, no. 10, pp. 8783-8794, January 2017.
19. A. A. A. S. S. A.-N. a. M. A. L. Mahmoud Abu Ghali, *"An Intelligent Tutoring System for Teaching English Grammar,"* IJAER, vol. 2, no. 2, pp. 1-6, February – 2018.
20. B. V. M. I. a. Z. B. Aleksandra Klašnja-Milićević, *"Personalisation of Programming Tutoring System Using Tag-based Recommender Systems,"* in 12th IEEE International Conference on Advanced Learning Technologies, 2012.
21. M. I. A. K.-M. a. Z. B. Boban Vesin, *"Protus 2.0: Ontology-based semantic recommendation in programming,"* Expert Systems with Applications, vol. 3, no. 9, p. 12229–12246, 2012.
22. A. M. J. S. S. A.-N. Masoud I. El Agha, *"SQL Tutor for Novice Students,"* IJAISR, vol. 2, no. 2, pp. 1-7, 2018.
23. S. S. A. Naser, *"ITSB: An Intelligent Tutoring System Authoring Tool,"* Journal of Scientific and Engineering Research, vol. 3, no. 5, pp. 63-71, 2016.
24. G. G. S. J. a. C. Z. C. Feng, *"Authoring Tools for Easy Creation of Adaptive Tutoring,"* in Spring, BHCI Capstone Project, 2018.
25. F. Wei, *"A STUDENT MODEL FOR AN INTELLIGENT TUTORING SYSTEM HELPING NOVICES LEARN OBJECT-ORIENTED DESIGN,"* Lehigh University, 2007.
26. *"What is Python? Executive Summary,"* Python, [Online]. Available: https://www.python.org/doc/essays/blurb/. [Accessed 11 11 2019].
27. L. Razmerita, *"An Ontology-Based Framewrok for Modeling User Behavior: A Case Study in Kowledge Management,"* System and Human, IEEE, vol. 41, no. 4, pp. 772-783, 2011.
28. A. R. S. H. I. H. Ian Horrocks, *"Web Ontology Language (OWL),"* W3C, 11 12 2012. [Online]. Available: https://www.w3.org/2001/sw/wiki/OWL. [Accessed 13 11 2019].

AUTHORS PROFILE