Heliyon 10 (2024) e29046

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Ontological approach for competency-based curriculum analysis

Marek Milosz [b], Aizhan Nazyrova [a,c,*], Assel Mukanova [c], Gulmira Bekmanova [a], Dmitrii Kuzin [c], Gaukhar Aimicheva [a]

[a] *Faculty of Information Technologies, L.N. Gumilyov Eurasian National University, 2 Satpayev Str., Astana, 010008, Kazakhstan*
[b] *Department of Computer Science, Lublin University of Technology, 36B Nadbystrzycka Str., 20-618, Lublin, Poland*
[c] *Higher School of Information Technology and Engineering, Astana International University, 8 Kabanbay Batyr av., Astana, 010000, Kazakhstan*

A R T I C L E I N F O

Keywords:
Ontology
Educational program
Prerequisites
SPARQL
Owlready2
Curriculum
Python
Semantic web
Protégé

A B S T R A C T

This article is dedicated to the development of a model for competencies within an educational program and its implementation through the use of semantic technologies. The model proposed by the authors is distinctive in that competencies are organized into a hierarchical data structure with arbitrary levels of nesting. Furthermore, the article presents an original solution for modelling the input requirements for studying a course, which is defined in the form of dependencies between the competencies generated by the course and the competencies of other courses. The outcome of this work is an ontological model of a competency-based curriculum, for which the authors have developed and implemented algorithms for data addition and retrieval, as well as for analyzing the consistency of the curriculum in terms of the input requirements for studying a discipline and the learning outcomes from previous periods. The findings presented in the article will prove to be valuable in the development of educational process management information systems and educational program constructors. They will also be instrumental in aligning diverse educational programs within the context of academic mobility.

1. Introduction

Currently, in higher education systems across most countries worldwide, the design of educational programs for the representation of learning outcomes predominantly adopts a "competency-based approach" [1–5]. The main idea of this approach is that the learning outcomes of an educational program are formulated as a set of attributes referred to as competencies, which must correspond to the qualifications of the graduate.

An examination of academic publications in this field enables the articulation of the following merits inherent to the competency-based approach:

- A shift in focus towards the ultimate learning outcome, prioritizing the result over the instructional process;
- Emphasis on the practical skills of graduates;
- Cross-disciplinary - the ability to develop a single competency across multiple courses;

* Corresponding author. Faculty of Information Technologies, L.N. Gumilyov Eurasian National University, 2 Satpayev str., Astana, 010008, Kazakhstan.

E-mail addresses: m.milosz@pollub.pl (M. Milosz), nazyrova_aye_1@enu.kz (A. Nazyrova), assel.mukanova@aiu.edu.kz (A. Mukanova), gulmira-r@yandex.kz (G. Bekmanova), dmitriy_kuzin@aiu.edu.kz (D. Kuzin), aimicheva79@gmail.com (G. Aimicheva).

https://doi.org/10.1016/j.heliyon.2024.e29046
Received 20 December 2023; Received in revised form 22 March 2024; Accepted 28 March 2024
Available online 4 April 2024
2405-8440/

- Unification - the possibility of comparative analysis of various educational courses and programs based on a common competency structure;
- Globalization and academic mobility - a unified competency-based approach facilitates the recognition of qualifications and educational achievements across different countries.
- In various educational systems, the structure for describing an educational program that employs a competency-based approach may vary. Nevertheless, in the majority of cases, it typically comprises a standardized set of components that pertain to the competency-based approach. For instance, the educational program at University incorporates these elements:
- Competencies classified into universal, general professional, and specialized professional;
- Learning outcomes (indicators of competency achievement), serving as a detailed elaboration of competencies;
- Disciplines that facilitate the development of competencies in students.

According to the authors, the current structure has significant limitations that hinder the full utilization of the advantages of the competency-based approach. The primary drawback of the existing educational program model lies in the insufficient level of detail in describing competencies and learning outcomes. The total number of competencies and learning outcomes tends to be in the same order as the number of disciplines within the educational program, which hinders a comprehensive exploration of the program's internal content. In an attempt to overcome this limitation, the authors of the curriculum associate multiple disciplines with a single competency, which, in turn, leads to a dilution of the educational program's content, effectively nullifying the benefits of the competency-based approach.

Another significant limitation is that the employed competency model is essentially « flat». It contains all two levels - competencies and learning outcomes. As a consequence, this results in a limited expressiveness of the competency model, making it challenging to work with the concepts of "general-specific" concerning the educational content. Furthermore, the model lacks any other mechanisms for connecting competencies, except for the link between competencies and learning outcomes.

The mentioned shortcomings of the educational program structure limit the opportunities for the analysis and synthesis of educational content. The implementation of a competency-based approach in education becomes a mere formality, only in-creasing the administrative costs of the educational program and the volume of bureaucratic documentation.

The problem of applying a competency-based approach to the design and implementation of educational programs should also be considered in the context of academic mobility. In cases of students transferring to other educational institutions, changing educational programs, or participating in student exchanges, the necessity of aligning educational programs arises. This is necessary for the recognition of completed courses and the determination of missing ones. In practice, this process is currently conducted primarily based on the course titles and credit hours, as standardized methods for describing course content are absent. The competency-based approach can help address this issue, but it requires a universally shared competency model accessible to all participants in the educational process, as well as a mechanism for its continuous updating.

To overcome the aforementioned shortcomings, the authors of the article set a re-search task to construct an informational competency model that would meet the following requirements:

- Providing any level of competency detail;
- Invariance to the education profile and subject area;
- Storing information about dependencies among competencies, reflecting the necessary sequence of their acquisition;
- Integration with the informational model of the educational program;
- The ability for collaborative use and refinement of the model by various educational organizations.

The authors believe that, an adequate tool for implementing the described competency model is ontological modelling. This choice is primarily motivated by the fact that the competency tree, as one of the main components of the developed model, has a graph structure. This aligns with the principles of building an ontology, which also represents a graph of arbitrary structure. Additionally, the emerging relationships between competencies, reflecting prerequisites, as well as connections between competencies and educational courses, further complicate the graph structure. This also supports the use of technologies for handling loosely structured information, such as ontological modelling.

With the increasing level of detailing, the competency model of an educational program will expand in volume. The number of individuals (instances of classes) in such a model can number in the thousands. To develop a high-quality and relevant model, it is advisable to combine the efforts of various educational institutions, scientific, and engineering schools. Ontological modelling technology provides such possibilities because an ontology is an open and extensible data structure. The development of an ontological model can be carried out by different independent groups of experts, each of which creates instances of concepts within its own namespace. However, this does not exclude the necessity to formalize the results of model development in the form of national and industry standards.

The ontological approach renders information not only machine-readable but also "machine-understandable" because the semantics and meaning of information are directly embedded in the ontology and are an integral part of the model. This enables the utilization of intelligent logical reasoning procedures, the effectiveness of which becomes evident when analyzing complex subject domains and loosely structured data, such as an educational program.

2. Analysis of relevant research and publications

The Organization for Economic Cooperation and Development's (OECD's) "Education 2030″ project emphasizes the development of key competencies for adapting to a rapidly changing world, including digital literacy, physical health, creativity, critical thinking, responsibility, resilience, and collaboration [6]. This approach reflects a shift from traditional educational systems to competency-based learning, which aligns with the recommendations of the ACM/IEEE-CS Association for Computing Machinery (ACM) and the IEEE Computer Society (IEEE-CS) Computing Curricula 2020 [7] and Computer Science Curricula 2023 [8], emphasizing the importance of the competence-based approach in the field of computer science.

Initiatives such as the IoC (Institute of Coding) standards [9] are aimed at bridging the gap between labor market requirements and the level of practical skills development of graduates, highlighting the importance of integrating professional fitness and identity into educational programs to improve graduates' employability [10].

The "Computing Curricula 2020″ report highlights the transition to competency development, including knowledge, skills, and attitudes, while ABET (Accreditation Board for Engineering and Technology) applies the competence-based approach in accrediting educational programs, requiring documentation of competency development in students. This alignment with the CC2020 (Computing Curricula 2020) document demonstrates a general trend towards recognizing the importance of the competence-based approach in computer science education [11–13].

Efforts by working groups at the ITiCSE (Innovation and Technology in Computer Science Education) conference and the integration of CC2020 (Computing Curricula 2020) dispositions with the SFIA (Skills Framework for the Information Age, https://sfia-online.org/en) framework indicate a desire to improve graduates' preparation for career challenges and to address the shortage of professional skills, emphasizing the importance of ongoing development of educational programs in accordance with the requirements of the information age [14–16].

The complexities of student exchange programs such as Erasmus + are often related to the mandatory use of various administrative and cartographic documents, which create significant difficulties for both students and the administrations of educational institutions. To address these issues, the "Erasmus Without Paper" project was developed [17]. The main goal of this project is to simplify the process of transferring information about students and courses, ensuring its seamless interaction between different educational institutions through a unified integrated system. This innovation implies a significant simplification of administrative procedures, making participation in the Erasmus + program more accessible and less burdensome for all interested parties [17]. While administrative procedures are simplified, the translation of credits requires manual intervention [18]. In the context of automating credit translation, significant efforts have been made to develop and implement natural language processing applications. These applications aim to automate the credit transfer process and analyze course compatibility, potentially easing administrative management procedures in educational institutions [19]. However, a problem has arisen: most developed solutions focus exclusively on the thematic consistency of courses, neglecting competencies or specific learning outcomes [19]. This means that, despite technological progress in automating processes, there is a need for additional work in integrating and analyzing competencies to ensure a comprehensive and thorough accounting of educational outcomes.

The alignment and development of competencies play a key role in the process of reverse credit translation and the development of student competencies. However, there is a problem with the lack of a common standard for defining competencies, leading to an individual approach by educational institutions to this issue [20]. Aligning competencies with labor market requirements is an important aspect that requires careful competency assessment to achieve a balance of productivity and efficiency [21]. However, competency assessment is a complex task, as it involves considering a multitude of factors, including knowledge, skills, self-assessment, behavior, and motivation [22–24]. There is an important distinction in competency assessment methods: competencies are often associated with specific skills, whereas competency reflects a person's behavior in performing tasks [25]. Various methods are used to assess these aspects, including the application of numerical indicators, conducting interviews, organizing group work, conducting seminars, and surveys [26–36]. These approaches allow for a comprehensive assessment of both skills and behavioral aspects of competency.

Adaptation and development of competencies require careful monitoring of their evolution over time. This includes the need for regular updates and the integration of current labor market data into online systems to meet changing requirements [37,38]. Various methods are used to analyze and compare competencies, among which Bloom's Taxonomy stands out. Bloom's Taxonomy allows structuring educational objectives, analyzing textual materials, and effectively comparing competencies, thereby ensuring their relevance and adaptation to the needs of the modern market [39–42].

Integration and assessment in e-learning systems include approaches to knowledge assessment, as well as the integration of various systems and methods for representing competencies [43,44]. Special attention is given to creating a tree-like structure of competencies, which allows systematizing and classifying knowledge and skills. In addition, contextual modeling is used, which aids in a more accurate understanding and interpretation of competencies in different conditions [45]. Analysis of the scope, commonality, and variability additionally helps assess the breadth and applicability of competencies, thereby ensuring their effective integration into e-learning systems.

3. Research questions

The consideration presented in the Introduction chapter, and the results of the literature and related works review allowed the formulation of the following research questions:

1) Is it possible to construct an ontological model of educational program competencies that takes into account the hierarchical embedding of competencies, and integrate this model with an ontological model of courses, learning periods, and descriptions of prerequisites?
2) Is it possible to implement logical deduction procedures for the analysis of the alignment between entry requirements and learning outcomes, while considering the hierarchical competency model?
3) Are there effective tools for manipulating data in the OWL/RDF format using the Python programming language that enable the resolution of challenges associated with the manipulation of ontological competency models?
4) In what manner can the software processing of an acyclic competency graph be structured, accounting for an unrestricted quantity of hierarchical levels contained within it?

4. Development of an ontological model of competencies

In pursuit of answers to the aforementioned research questions, the authors aimed to develop an efficient competency model that would harness the full advantages of the competency-based approach [46–51,52] in educational program design. While addressing this objective, the authors aspired to create a descriptive framework for educational program competencies that would be expressive, transparent, and amenable to computer-based processing and analysis. The structural distinctions between the developed educational program competency model and the traditional curriculum structure are illustrated in Fig. 1. The primary characteristics of the competency model proposed by the authors include:

- The collection of competencies within the educational program constitutes a flat acyclic graph (tree) in which the top level comprises three nodes corresponding to universal, general professional, and specific professional types of competencies;
- In this competency tree, nodes at lower levels intricately expand upon the content of higher-level competencies, creating a hierarchical nesting structure;
- Any node within the competency tree can be associated with a set of learner competency requirements (prerequisites) – a list of other competencies within the educational program that must be acquired before pursuing the associated competency;
- The course is designed to foster a subset of competencies, without limitations on their level or quantity;
- The course's entry requirements (prerequisites) are assembled from the prerequisites associated with the competencies developed by the course.

The presented model applies to diverse higher education programs [28,53,46–48], encompassing both bachelor's and master's degrees, and it is intended to facilitate the practical realization of the benefits associated with the competency-based approach to educational program design. The model offers the following advantages:

Due to the hierarchical nesting of competencies, the necessary level of their detailing is ensured. At the same time, the level of detailing can vary among different types of competencies and disciplines. The latter circumstance will not cause contradictions, as for data processing and analysis it is always possible to choose the level of detailing that corresponds to the task to be solved;

A single data model is employed to describe the outcomes of education and the input prerequisites, ensuring transparency and clarity in the design of the curriculum. It is evident what the sequence of courses should be to meet all the prerequisite requirements;

The prerequisites for taking a course are determined in the form of dependencies between the competencies developed by the course and other competencies. Such an approach eliminates the linkage between the curriculum and competencies, making the competency model a universal tool for the analysis and design of educational programs, shared by various educational institutions;

The strict formalization of the competency model enables computer processing and analysis of educational programs using intelligent algorithms [43,54].

The development of the ontological model of competencies for the educational program was carried out using the ontological editor

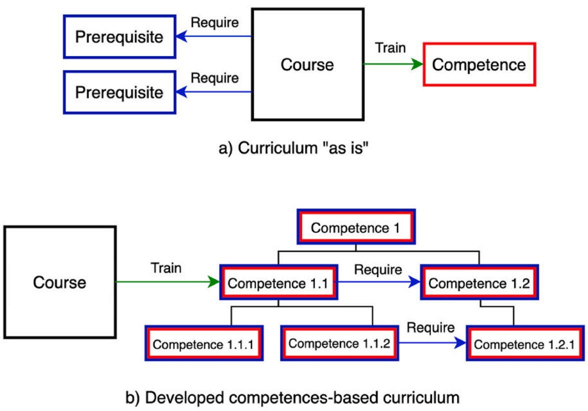

Fig. 1. Comparison of competency models.

Protégé 5.5.0 [46]. The foundation for this model was based on the educational program model developed by Refs. [46,49]. A hierarchy of classes and a hierarchy of Object Properties were constructed (see Fig. 2). The class structure follows the principle of inheritance, whereby common properties of individuals are implemented in the base class, and specific properties are implemented in the inheriting classes.

The aggregate of all competencies acquired by the learners during the educational program forms several hierarchical structures, where the elements at lower levels detail the content of the parent element. This structure of relationships, in terms of ontological modelling, is established using the 'Includes' property and its inverse, 'IsPartOf' (see Fig. 3). These properties are transitive – if Competency A is included in Competency B, and Competency B is included in Competency C, then A is also included in C. Such structures with arbitrary levels of nesting can be effectively modelled using graph models, namely ontologies.

The acquisition of a particular competency by learners may require them to possess other competencies. For example, acquiring the skill of object-oriented programming requires basic programming, procedural-oriented programming and algorithmic skills. Consequently, competencies, regardless of their level of nesting, can be linked through the requires/requiredBy object properties. This property is also transitive, meaning that the prerequisites for a given competency will encompass all chains of competencies connected through the 'requires' property.

According to the logic of the model, if a required competence has subsidiary (nested) competencies, they are also considered required, meaning they are connected through the 'requires'/'requiredBy' relationship. To implement this logic in the model, it is necessary to establish the corresponding chain of relationships 'requires' -> 'includes' = 'requires' (see Fig. 4).

In contrast to competencies, the course model is "flat," which means that courses cannot be nested and do not have any explicit relationships with each other, except for the chronological sequence of study. each course aimed at mastering a competence by the learner is linked to that competency through the 'train' property and its inverse, 'trainedBy' (as shown in Fig. 5). The possibility of these connections is not dependent on the competency's position within the hierarchy. Multiple connections are allowed for both courses and competencies.

Following the logic of the model, a course aimed at mastering a competence by a learner will also be directed towards mastering nested competencies. To implement this logic within the model, it is necessary to establish the corresponding chain of relations: 'train' -> 'includes' = 'train' (see Fig. 6).

From the perspective of time periods, an educational program possesses a typical structure. In the model, two classes are defined as subclasses of the Timeinterval class – AcademicYear and Semester, which are connected by relationships includes/isPartOf. To determine the sequence of a direct succession of intervals, the Timeinterval class, and hence its subsidiary classes, is linked by two mutually inverse properties - precede and followBy. These properties are not transitive. To establish relations in a time of a more general nature, such as which interval precedes or follows another, regardless of the "distance" between them, two superproperties – goesBefore and goesAfter – have been defined. The superproperties goesBefore and goesAfter are transitive. Consequently, any period within the chain of periods connected by the relations followBy/precede with its neighbours will be connected with other periods by the relations goesBefore and goesAfter (Fig. 7).

The concept of "Course" is linked to the semester through the functional property "studiedDuring" and its inverse property "providesStudy".

In order for the goesBefore/goesAfter relationship between AcademicYear containers to also extend to the relationship between semesters and the relationship between the AcademicYear container and the semester, a corresponding chain of relationships isPartOf - > goesBefore = goesBefore must be constructed (Fig. 8).

The complete semantic network graph of the competency model is presented in Fig. 9.

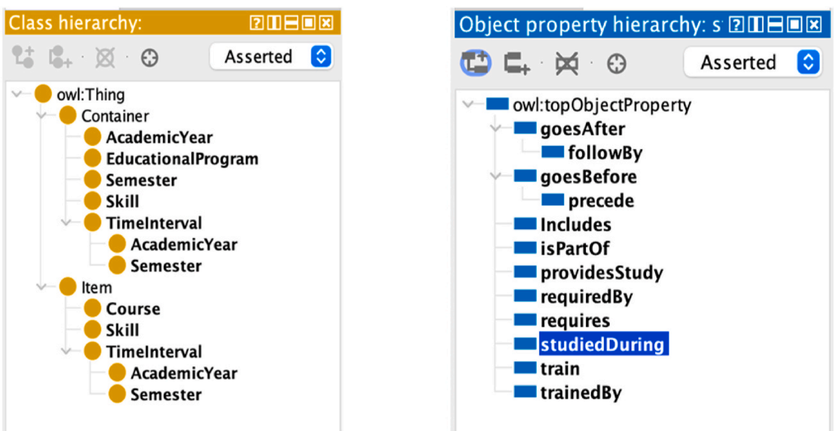

Fig. 2. Hierarchy of classes and properties in the ontological model [46].

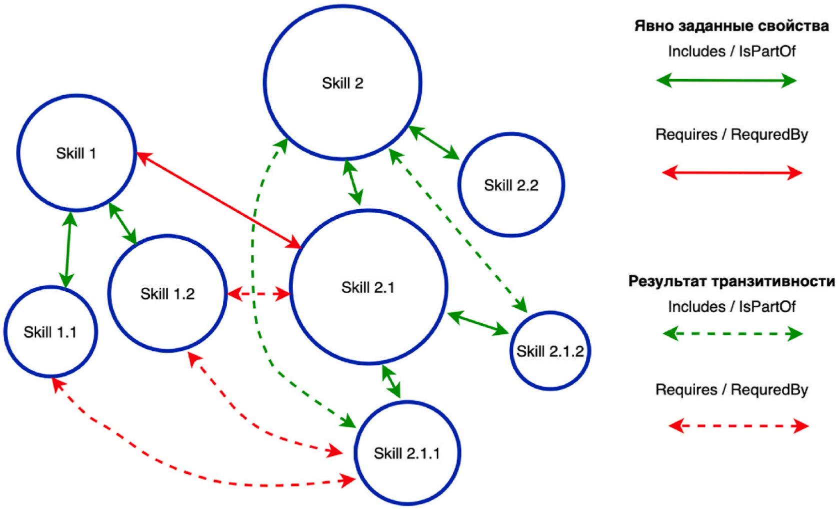

Fig. 3. Transitive connections between competencies.

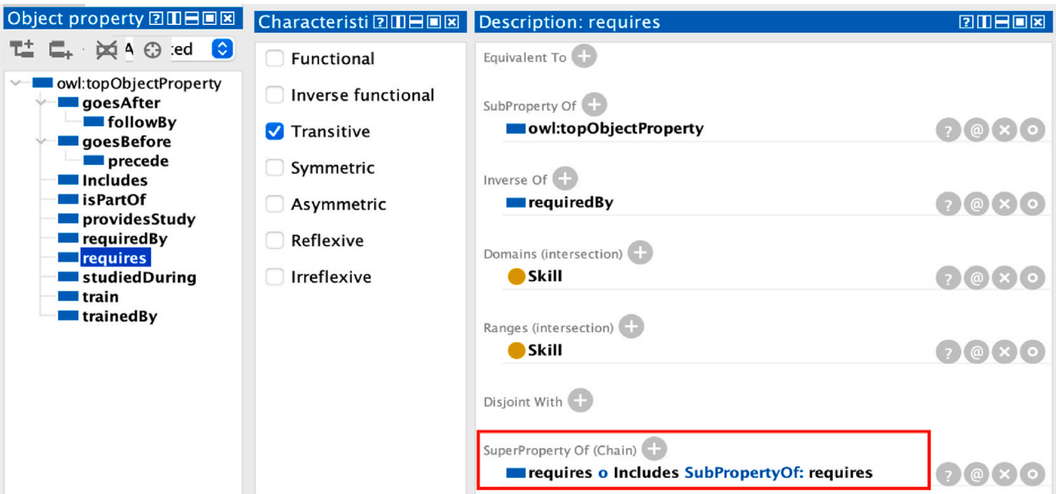

Fig. 4. Defining a chain of object properties for linking nested competencies with the 'require' property.

5. Implementation of an ontological model

After developing the ontology structure, educational program data were integrated into it and described in Python as the following set of lists:

- Competencies – list of competencies, with each element comprising a list of three elements: a competency identifier, the competency's title, and the identifier of the parent competency;
- Dependencies – a list of dependencies between competencies, which in the ontology model are reflected by the object property (relations) requires/requiredBy. Each element of the list is also a list of two elements, where the first element indicates the identifier of the dependent competency, and the second element represents the identifier of the required competency;
- Courses – a list of educational courses, each element of which is a list consisting of a course identifier, course name, number of credits in the course, the assessment method (credit/exam), and a semester identifier;
- Discipline_competency_mapping – a list containing information on the development of competencies by educational courses. Each element of the list represents a tuple consisting of a course identifier and the identifier of the competency it forms.

To load the lists into the ontology model, a Python program code was written that uses the Owlready2 library to work with ontologies [55,56,57]:

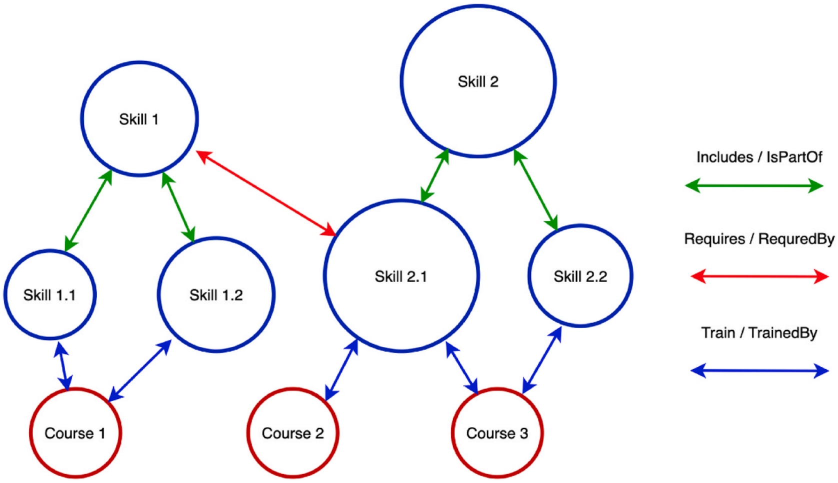

Fig. 5. The connection of educational courses with competencies.

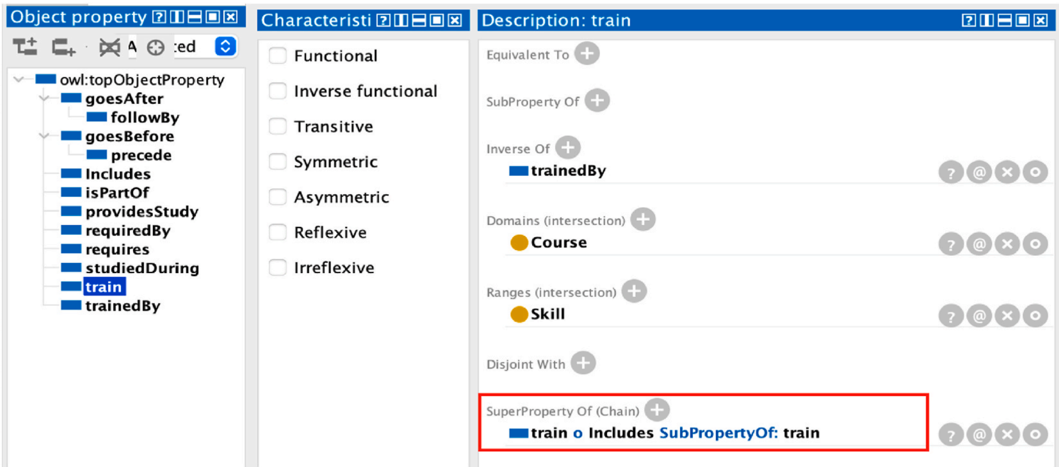

Fig. 6. Defining the chain of object properties for linking nested competencies with the "train" property.

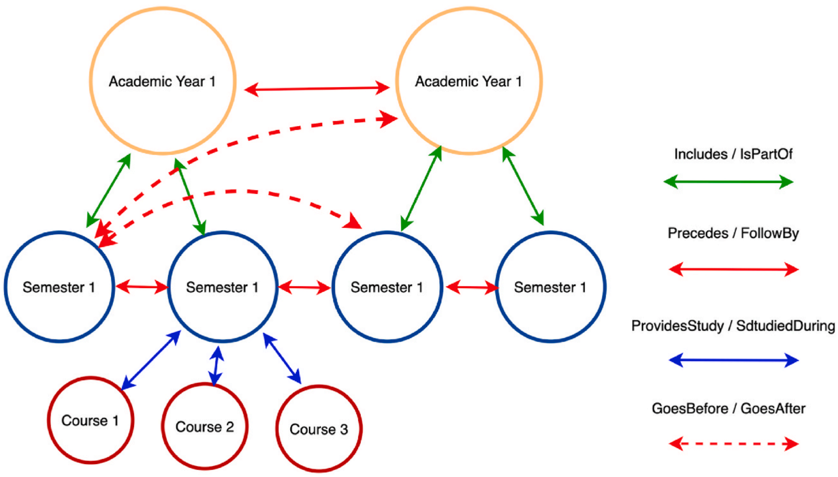

Fig. 7. Relationships between concepts - time periods.

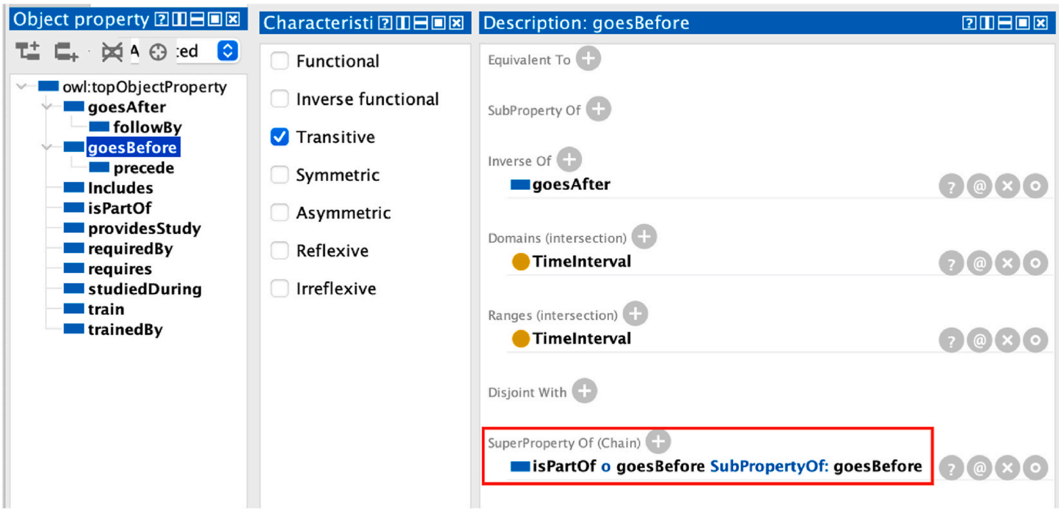

Fig. 8. Defining an Object Properties chain to link AcademicYear and Semester over time.

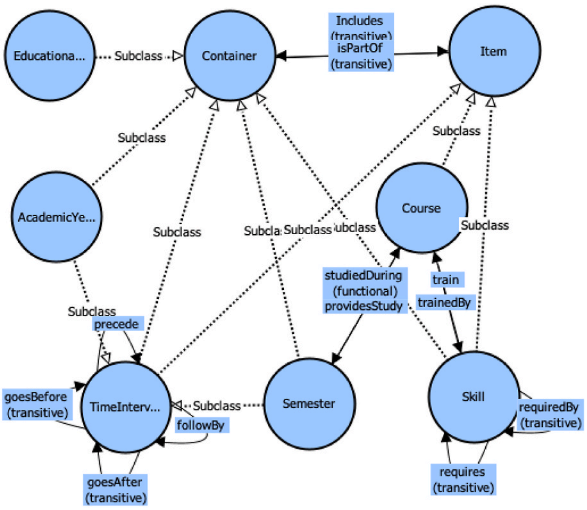

Fig. 9. Ontology graph of the competency model and educational program.

- The ontology of the competence model of the educational program, containing classes, properties, and instances of study periods, is loaded from an RDF file using the get_ontology().load(). method ();
- For each competency from the external list of competencies, an individual representing that competency is created. Additionally, a link with the parent competency is established, if available, through the object property (relations) includes/isPartOf;
- For competencies, the required skills (preceding ones) are identified from the list of dependencies, and this information is incorporated into the ontology as an object property (relationship) 'requires/requiredBy';
- In the ontology, an individual is generated for each course from the list of courses, and attributes such as the number of credits, assessment method, and the duration of the study are included;
- For every course created in the ontology, the competencies it generates are linked to it using the 'train/trainedBy' property, based on the data from the 'discipline_competency_mapping' list.

To check the ontology's integrity, logical inference engines are employed.

After all individuals and properties have been created in the ontology, program access to ontology objects can be obtained using the Owlready2 library [55,56,57]. The primary means of data extraction is the SPARQL query language. The Python program code cycles through all semesters of the curriculum, courses, and the competencies they generate. An example of the output data generated by this code is presented in Fig. 10.

A distinctive feature of program processing of competencies is the fact that the competence model is represented as a graph (tree)

```
==================================================================
Semester 3
==================================================================
------------------------------------------------------------------
Course: Operating Systems  Credits: 4  Evaluation: Exam
------------------------------------------------------------------
Trained competencies:
  1.Cloud Platforms
  2.Google Cloud Platform (GCP)
  3.Microsoft Azure
  4.Amazon Web Services (AWS)
  5.Serverless Architecture
------------------------------------------------------------------
Course: Databases and SQL  Credits: 5  Evaluation: Exam
------------------------------------------------------------------
Trained competencies:
  1.NoSQL Databases
  2.Data Migrations
  3.Databases
  4.Object-Relational Mapping (ORM)
------------------------------------------------------------------
Course: Web Development and HTML/CSS  Credits: 5  Evaluation: Exam
------------------------------------------------------------------
Trained competencies:
  1.JavaScript
  2.HTML and CSS
------------------------------------------------------------------
Course: Interface Development  Credits: 4  Evaluation: Test
------------------------------------------------------------------
Trained competencies:
  1.Frontend Development
  2.Frontend Frameworks
  3.JavaScript
  4.User Interfaces and User Experience
  5.Responsive Design
  6.HTML and CSS
```

Fig. 10. Example of information output from the ontology.

with an arbitrary number of nested levels. Due to the fact that hierarchical links between competencies isPartOf/Includes are transitive, with the help of standard tools, such as SPARQL query, it is easy to get all subsidiary elements of any branch of the tree. However, they cannot be used to obtain information about the hierarchy level at which the received elements are located. To obtain all nodes of the competence tree while preserving the hierarchical structure with an unknown number of nesting levels, a recursive algorithm is required. The program code generates a list of top-level competencies, which act as the tree's vertices. Subsequently, for each top-level competence a recursive function is derived, which is passed the parent competence and the integer number of the level necessary to display the nesting of competencies. A fragment of the result produced by the recursive algorithm is presented in Fig. 11.

The analysis of the educational program, using the developed competence model, is carried out through the following algorithm, implemented in the Python programming language. For each semester, a set of competencies is constructed using SPARQL queries. This set includes competencies acquired by students during the study of courses from all previous semesters (C), the set of competencies formed by the courses in the current semester (N), and the set of competencies that serve as prerequisites for their study (P). Subsequently, the Python software calculates the difference P - N. This resulting set represents the competencies that are missing due to inconsistencies in the educational program. The analysis of the educational program is performed after enabling logical inference (reasoner synchronization). The program's results for the first semester are presented in Fig. 12.

6. Results and discussion

The main results of the work performed are as follows:

An analysis of scientific publications devoted to the issue of applying a competency-based approach in assessing learning outcomes and the composition of higher education programs has been conducted. In the majority of these works, the authors highlight the difficulties in conducting a comparative analysis of competencies across various educational programs, which arise within the context of academic mobility. The main problem lies in the absence of a unified standard for representing a competence model. Additionally, there is a lack of technological solutions to implement such a standard.

An ontology of competencies has been developed, its structure forming an acyclic graph with an arbitrary number of nesting levels. At the top level of the hierarchy, nodes correspond to competency types. Typically, each graph node has several subsidiary competencies that provide detailed specifications of its content. The proposed model is implemented as an ontology and integrated with the curriculum ontology.

A Python program code has been developed using the Owlready2 library for populating the competency ontology with data and extracting data using a recursive algorithm.

An algorithm for checking the consistency of the educational program has been implemented using SPARQL queries and Python programming code. This algorithm involves determining the difference between the set of required competencies for studying courses in a semester and the set of competencies already acquired by the students.

The main challenge in implementing the scientific and technical solutions proposed in the article is the difficulty of constructing an adequate and balanced model of competencies. Considering that in scale, it will correspond to a group of educational programs and must contain a comprehensive set of competencies for hundreds or even several hundred courses. Such a scale does not pose technical difficulties in implementing the software, but it can present a problem for organizing the work of the expert community, whose function will include updating the competency tree. Increasing the detail of competencies will lead to a constant increase in their

```
3. Professional competencies
    3.1. Software development
      3.1.1. Programming language selection
        3.1.1.1. Python proficiency
        3.1.1.2. Java proficiency
        3.1.1.3. C++ proficiency
        3.1.1.4. JavaScript proficiency
      3.1.3. Data Structures and Algorithms
        3.1.3.1. Sorting algorithms
        3.1.3.2. Searching algorithms
        3.1.3.3. Algorithm complexity
        3.1.3.4. Graphs and trees
        3.1.3.5. Hash tables
        3.1.3.6. Dynamic programming
      3.1.2. Object-Oriented Programming and Design Patterns
        3.1.2.1. Principles of OOP
        3.1.2.2. Singleton pattern
        3.1.2.3. Factory Method pattern
        3.1.2.4. Observer pattern
        3.1.2.5. MVC pattern
        3.1.2.6. MVVM pattern
      3.1.4. Testing and debugging
        3.1.4.1. Unit testing
        3.1.4.2. Integration testing
        3.1.4.3. Debugging and profiling
        3.1.4.4. Security testing
        3.1.4.5. Performance testing
    3.2. Web Application Development
      3.2.1. Frontend Development
        3.2.1.1. HTML and CSS
        3.2.1.2. JavaScript
        3.2.1.3. Frontend Frameworks
        3.2.1.4. Responsive Design
        3.2.1.5. User Interfaces and User Experience
```

Fig. 11. Fragment of competence tree visualization using a recursive algorithm.

```
======================================================================
Semester 1
======================================================================
Skills obtained in previous semesters:
       Skill_0.1 Russian language
       Skill_0.2 Profiled mathematics
       Skill_0.3 Informatics and ICT
       Skill_1.1.1 Oral communication
       Skill_1.1.2 Written communication
       Skill_1.1.3 Visual communication
       Skill_1.1.4 Intercultural communication
Skills required:
       Skill_1.1 Communication skills
       Skill_1.1.1 Oral communication
       Skill_1.1.3 Visual communication
       Skill_1.1.2 Written communication
       Skill_1.1.4 Intercultural communication
       Skill_3.1.3.4 Graphs and trees
       Skill_3.1.3.6 Dynamic programming
       Skill_3.1.3.2 Searching algorithms
       Skill_3.1.3.1 Sorting algorithms
       Skill_3.1.3.3 Algorithm complexity
       Skill_3.1.3.5 Hash tables
       Skill_3.1.3 Data Structures and Algorithms
MISSING SKILLS:
       Skill_1.1 Communication skills
       Skill_3.1.3.4 Graphs and trees
       Skill_3.1.3.6 Dynamic programming
       Skill_3.1.3.2 Searching algorithms
       Skill_3.1.3.1 Sorting algorithms
       Skill_3.1.3.3 Algorithm complexity
       Skill_3.1.3.5 Hash tables
       Skill_3.1.3 Data Structures and Algorithms
```

Fig. 12. The results of the analysis of missing competencies in the first semester.

quantity and create risks of increasing the complexity and time required for experts to coordinate the changes made.

The conducted research made it possible to answer the research questions posed.

1) Is it possible to construct an ontological model of educational program competencies that takes into account the hierarchical embedding of competencies, and integrate this model with an ontological model of courses, learning periods, and descriptions of prerequisites? The answer to this research question is YES, such a model has been constructed.
2) Is it possible to implement logical deduction procedures for the analysis of the alignment between entry requirements and learning outcomes, while considering the hierarchical competency model? The answer to this research question is YES; this task has been resolved. This is because the ontological model and logical inference mechanisms are best suited for processing graph-like data structures, which include the hierarchical competence tree.

3) Are there effective tools for manipulating data in the OWL/RDF format using the Python programming language that enable the resolution of challenges associated with the manipulation of ontological competency models? The answer to this research question is YES, as the authors utilized the Owlready2 library as such a tool.
4) In what manner can the software processing of an acyclic competency graph be structured, accounting for an unrestricted quantity of hierarchical levels contained within it? The answer to this research question is YES; for processing the competency graph, the authors developed and tested a recursive algorithm, which represents the most concise software implementation of this task.

7. Conclusion

In this article, the authors describe the developed ontological model of competencies, which is applicable to any higher educational program and is intended to assist in the practical implementation of the advantages of the competency-based approach to curriculum design. The primary distinguishing features of the developed model are as follows:

- The aggregate of all competencies within the educational program constitutes a flat acyclic graph (tree) with an unlimited number of hierarchical levels.
- The nodes of the competence tree that serve as children detail the content of the top-level competency.
- Any node in the competence tree can be associated with a set of requirements for the learner's competency (prerequisites) – a list of other competencies within the educational program, the prior acquisition of which is mandatory;

The developed model integrated with the ontological model of the educational program includes descriptions of study periods and educational courses. The advantages of this model are the ability for unlimited detailing of competencies through hierarchical nesting and a single data model for the description of learning outcomes and prerequisites, which provides transparency and clarity in designing the curriculum.

The model has a universal structure and can be applied to educational programs in technical, natural science, and humanities fields. The applicability condition of the model is the classical structure of the curriculum, consisting of individual courses tied to periods of study. However, for some educational programs, particularly medical postgraduate programs [5,58], direct application of this model is not possible due to differences in approaches to developing competency-based curricula.

Continuous program assessment is of paramount importance in the implementation of competency-based educational programs to ensure compliance with evolving industry standards, technological advancements, and cutting-edge educational practices. A shared competency model facilitates the effectiveness of such assessment by allowing stakeholders to "speak the same language." This ensures that the educational program meets the needs of all involved parties and reflects current industry requirements.

Adherence to educational standards is crucial for the acceptance and recognition of competency-based educational programs. Collaboration with regulatory bodies and adherence to their criteria ensure that graduates meet the necessary requirements for licensure, certification, or professional practice.

The reproducibility of the model is ensured by open access to the model's code, as well as to the source code of the developed tools. All files necessary for implementing the model in the process of designing competency-based educational programs are provided in an open repository at https://github.com/AyzhanNaz/OntoSkills.

For the computer processing of the ontological model, the Python programming language was utilized, along with the Owlready2 library [55,56,57]. Program code was developed to create individuals and their properties in the ontology based on Python lists, as well as program code for extracting data from the educational program. To obtain all nodes of the competence tree while preserving the hierarchical structure with an unknown number of nesting levels, a recursive algorithm was employed.

The analysis of the educational program's consistency using the developed competence model was also implemented through Python programming code [55,56,57]. For each semester, a list of missing competencies was obtained using SPARQL queries - these are competencies that serve as prerequisites for the courses in the semester but have not been acquired by the students in previous semesters.

The possibilities of the developed ontological model of competencies can be used not only for analyzing the consistency of the curriculum but also for aligning different educational programs. This task arises when transferring a student between various educational institutions or transitioning to a different educational program. In this context, the development and updating of a competency model that covers extensive knowledge domains and is shared among all educational institutions within the academic mobility environment become relevant. It is evident that its development can only be achieved by large groups of specialists organized on a crowdsource principle. As mentioned earlier, the ontological approach is well-suited for the implementation of such projects, as the principle of distribution is fundamental to the concept of the Semantic Web. Defining the architecture and means of implementing a 'global' competency model is a subject for future research.

CRediT authorship contribution statement

Marek Milosz: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Conceptualization. **Aizhan Nazyrova:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Funding acquisition, Conceptualization. **Assel Mukanova:** Resources, Investigation, Funding acquisition, Formal analysis. **Gulmira Bekmanova:** Writing – original draft, Validation, Software, Data curation. **Dmitrii Kuzin:** Visualization, Software, Project administration, Methodology, Conceptualization. **Gaukhar Aimicheva:** Writing – original draft, Validation, Resources, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] H. Vargas, et al., Automated assessment and monitoring support for competency-based courses, IEEE Access 7 (2019) 41043–41051. C.

[2] S. Gottipati, V. Shankararaman, Competency analytics tool: analyzing curriculum using course competencies, Educ. Inf. Technol. 23 (2018) 41–60.

[3] R. Malhotra, M. Massoudi, R. Jindal, Shifting from Traditional Engineering Education towards Competency-Based Approach: the Most Recommended Approach-Review//Education and Information Technologies, 2023, pp. 1–31. C.

[4] L.R. Betts, et al., Developing a competency-based education training programme for university tutors, The Journal of Competency-Based Education (2019). T. 4. – №. 4. – C. e01200.

[5] B.Y.M. Kwan, A. Mbanwi, N. Cofie, C. Rogoza, O. Islam, A.D. Chung, N. Dalgarno, D. Dagnone, X. Wang, B. Mussari, Creating a competency-based medical education curriculum for Canadian diagnostic radiology residency (Queen's Fundamental Innovations in Residency Education)-Part 1: transition to Discipline and Foundation of Discipline Stages, Can. Assoc. Radiol. J. 72 (3) (2021 Aug) 372–380, https://doi.org/10.1177/0846537119894723. Epub 2020 Mar 4. PMID: 32126802.

[6] M. Fuster, Education Fast Forward: Building a Future that Works for All, OECD Publishing, 2022.

[7] R.L. Colby, Competency-based Education: A New Architecture for K-12 Schooling, Harvard Education Press, 2019.

[8] A. Clear, et al., Computing Curricula 2020 (CC2020): Paradigms for Future Computing Curricula//ACM, IEEE Computer Society, NY, 2020.

[9] C. Servin, et al., CS2023: ACM/IEEE-CS/AAAI Computer Science Curricula-Specialized Platform Development, 2023.

[10] D.S. Bowers, A. Hayes, T. Prickett, T. Crick, K. Streater, C. Sharp, The Institute of Coding Accreditation Standard: Exploring the Use of a Professional Skills Framework to Address the UK Skills Gap, in: Proceedings of the 2023 Conference on United Kingdom & Ireland Computing Education Research, 2023, pp. 1–7. September.

[11] A. Clear, A. Parrish, P. Ciancarini, S. Frezza, J. Gal-Ezer, J. Impagliazzo, M. Zhang, Computing Curricula 2020 (CC2020): Paradigms for Global Computing Education, 2020. New York, https://www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2020.pdf.

[12] ABET, Inc, Criteria for Accrediting Computing Programs, 2023, pp. 2023–2024. https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2023-2024/.

[13] R. Raj, A. Kumar, M. Sabin, J. Impagliazzo, Interpreting the ABET Computer Science Criteria Using Competencies, in: Proceedings of the 53rd Acm Technical Symposium on Computer Science Education, vol. 1, Association for Computing Machinery, New York, NY, USA, 2022, pp. 906–912, https://doi.org/10.1145/3478431.3499293.

[14] S. Frezza, M. Daniels, A. Pears, r. Cajander, V. Kann, A. Kapoor, C. Wallace, Modelling Competencies for Computing Education beyond 2020: A Research Based Approach to Defining Competencies in the Computing Disciplines, in: Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education, ACM, New York, NY, USA, 2018, pp. 148–174.

[15] A. Clear, T. Clear, A. Vichare, T. Charles, S. Frezza, M. Gutica, J. Szynkiewicz, Designing Computer Science Competency Statements: A Process and Curriculum Model for the 21st Century, in: Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education, ACM, New York, NY, USA, 2020, pp. 211–246.

[16] David Bowers, Mihaela Sabin, Rajendra Raj, John Impagliazzo, Computing Competencies: Mapping CC2020 Dispositions to SFIA Responsibility Characteristics, 2022, pp. 428–437, https://doi.org/10.1109/EDUCON52537.2022.9766565.

[17] P. Leys, J. Mincer-Daszkiewicz, Erasmus Without Paper: Dream becoming reality, EPiC Ser. Comput. 86 (2022) 66–73.

[18] K. Cheung, B. Li, P. Benz, K.M. Chow, J.T.D. Ng, W.Y.Y. Kwok, H. Tsang, D.N.H. Leung, J.K.Y. Lui, Y. Na Li, et al., Prototype development of a cross-institutional credit transfer information system for community college transfer students, Sustainability 13 (2021) 9398.

[19] A. Heppner, A. Pawar, D. Kivi, V. Mago, Automating articulation: Applying natural language processing to post-secondary credit transfer, IEEE Access 7 (2019) 48295–48306.

[20] M.S. Giani, J.L. Taylor, S. Kauppila, Examining the educational and employment outcomes of reverse credit transfer, AERA Open 7 (2021) 1–15.

[21] N. Kumar, Managing skill gaps through competency mapping—A strategic tool for competitive edge, Int. J. Manag. 3 (2014) 134–139.

[22] N. Anisha, Competency mapping of the employees, Int. J. Adv. Res. Technol. 1 (2012) 713–720.

[23] M.I. Uddin, K.R. Tanchi, M.N. Alam, Competency mapping: A tool for HR excellence, Eur. J. Bus. Manag. 4 (2012) 90–98.

[24] B. Misnevs, Software Engineering Competence Evaluation Portal//Procedia Computer Science, 2015, pp. 11–17. T. 43. – C.

[25] S. Sanghi, The Handbook of Competency Mapping: Understanding, Designing and Implementing Competency Models in Organizations, SAGE Publications, New Delhi, India, 2016.

[26] S. Mäses, O. Maennel, S. Sütterlin, Using competency mapping for skills assessment in an introductory cybersecurity course, in: Proceedings of the International Conference on Interactive Collaborative Learning, Tallinn, Estonia, September 2020, pp. 23–25.

[27] V.K. Jain, Competency Mapping in Indian Industries-A Case Study, Int. J. Emerg. Res. Manag. Technol. 2 (2013) 10–21.

[28] M. Abelha, et al., Graduate Employability and Competence Development in Higher Education—A Systematic Literature Review Using PRISMA//Sustainability, 2020. T. 12. – №. 15. – C. 5900.

[29] J. Kansal, N. Jain, Development of competency model and mapping of employees competencies for organizational development: A new approach, J. Sci. Ind. Res. 78 (2019) 22–25.

[30] L. Zaitseva, B. Misnevs, Competency-based Approach Teaching Software Engineering//Smart Education and E-Learning 2019, Springer Singapore, 2019, pp. 231–241. C.

[31] L. Zaitseva, e-CF Competences in Software Engineering Course//Appl, Comput. Syst. (2019) 32–37. T. 24. – №. 1. – C.

[32] V.K. Jain, Competency Mapping in Indian Industries-A Case Study, Int. J. Emerg. Res. Manag. Technol. 2 (2013) 10–21.

[33] J. Kaur, V. Kumar, Competency mapping: A gap Analysis, Int. J. Educ. Res. 1 (2013) 1–9.

[34] D. Russo, Competency measurement model, in: Proceedings of the European Conference on Quality in Official Statistics, Madrid, Spain, 2016, 31 May 2016–3 June.

[35] D. Breen, G. Shorten, A. Aboulafia, D. Zhang, C. Hockemeyer, D. Albert, Defining a competency map for a practical skill, Clin. Teach. 11 (2014) 531–536.

[36] V. Shankararaman, S. Gottipati, Mapping information systems student skills to industry skills framework, Abu Dhabi, United Arab Emirates, in: Proceedings of the 2016 IEEE Global Engineering Education Conference (EDUCON), 2016, pp. 11–13. April.

[37] L.M. Kipper, S. Iepsen, A.J. Dal Forno, R. Frozza, L. Furstenau, J. Agnes, D. Cossul, Scientific mapping to identify competencies required by industry 4.0, Technol. Soc. 64 (2021) 101454.

[38] O. Ozyurt, F. Gurcan, G.G.M. Dalveren, M. Derawi, Career in Cloud Computing: Exploratory Analysis of In-Demand Competency Areas and Skill Sets, Appl. Sci. 12 (2022) 9787.

[39] S. Gottipati, V. Shankararaman, Competency analytics tool: Analyzing curriculum using course competencies, Educ. Inf. Technol. 23 (2018) 41–60.

[40] J. Wang, Y. Dong, Measurement of text similarity: A survey, Information 11 (2020) 421.

[41] A.N. Garman, M.O. Standish, D.H. Kim, Enhancing efficiency, reliability, and rigor in competency model analysis using natural language processing, J. Competency Based Educ. 3 (2018) e01164 [Green Version].

[42] P.R. Kowligi, P. Prajapati, F.R. Jones, M.A. Mardis, Comparing Florida's advanced manufacturing curriculum framework to the department of labor competency model, in: Proceedings of the 2020 ASEE Virtual Annual Conference Content Access, Virtual, 2021, 22 June 2020–26 June.
[43] A. Margienė, S. Ramanauskaitė, J. Nugaras, P. Stefanovič, Automated Transformation from Competency List to Tree: Way to Competency-Based Adaptive Knowledge E-Evaluation, Appl. Sci. 12 (2022) 1582, https://doi.org/10.3390/app12031582.
[44] A.A. Silva, et al., Using Concept Maps and Ontology Alignment for Learning Assessment//IEEE Technology and Engineering Education, 2012, pp. 33–40. T. 7. – №. 3. – C.
[45] S. Ramanauskaitė, A. Slotkienė, Hierarchy-Based Competency Structure and Its Application in E-Evaluation, Appl. Sci. 9 (2019) 3478, https://doi.org/10.3390/app9173478.
[46] A. Nazyrova, M. Milosz, G. Bekmanova, A. Omarbekova, A. Mukanova, G. Aimicheva, Analysis of the Consistency of Prerequisites and Learning Outcomes of Educational Programme Courses by Using the Ontological Approach, Appl. Sci. 13 (2023) 2661, https://doi.org/10.3390/app13042661.
[47] L. Romero, M. Gutiérrez, Software design for the evaluation of competency based learning in engineering careers Ontological approach for modeling, in: 2020 15th Iberian Conference on Information Systems and Technologies (CISTI), IEEE, 2020, pp. 1–6. C.
[48] M. Zouri, A. Ferworn, An ontology-based approach for curriculum mapping in higher education. 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), IEEE, 2021. C. 0141-0147.
[49] G. Bekmanova, et al., The Model of Curriculum Constructor//Computational Science and its Applications–ICCSA 2021: 21st International Conference, Cagliari, Italy, September 13–16, 2021, Proceedings, Part II 21, C, Springer International Publishing, 2021, pp. 459–470.
[50] G.H.A. Ahmed, L. Kovács, Development of ontology-based model to support learning process in LMS, Indonesian Journal of Electrical Engineering and Computer Science T. 24 (1) (2021) 507–518. C.
[51] A. Rajak, A.K. Shrivastava, D.P. Shrivastava, Automating Outcome Based Education for the Attainment of Course and Program Outcomes, in: 2018 Fifth HCT Information Technology Trends (ITT), Dubai, United Arab Emirates, 2018, pp. 373–376, https://doi.org/10.1109/CTIT.2018.8649532.
[52] P. Stolic, D. Milosevic, Z. Stevic, I. Radovanovic, Ontology Development for Creating Identical Software Environments to Improve Learning Outcomes in Higher Education Institutions, Electronics 12 (2023) 3057, https://doi.org/10.3390/electronics12143057.
[53] S. Perera, S.O. Babatunde, L. Zhou, J. Pearson, D. Ekundayo, Competency mapping framework for regulating professionally oriented degree programmes in higher education, Stud. High Educ. 42 (2017) 2316–2342.
[54] M. Deev, A. Finogeev, Application of the convergent education model in the development of a smart learning environment, Telematics and Informatics Reports 10 (2023) 100051, https://doi.org/10.1016/j.teler.2023.100051.
[55] B. Steenwinckel, et al., pyRDF2Vec: A Python Implementation and Extension of RDF2Vec//European Semantic Web Conference, C, Springer Nature Switzerland, Cham, 2023, pp. 471–483.
[56] J.B. Lamy, Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies, Artif. Intell. Med. 80 (2017) 11–28, https://doi.org/10.1016/j.artmed.2017.07.002.
[57] L. Jean-Baptiste, Ontologies with python, Apress, Berkeley, CA, 2021.
[58] L. Wong, A.D. Chung, C. Rogoza, B.Y.M. Kwan, Peering Into the Future: A First Look at the CBME Transition to Practice Stage in Diagnostic Radiology, Acad. Radiol. 30 (10) (2023 Oct) 2406–2417, https://doi.org/10.1016/j.acra.2023.06.013. Epub 2023 Jul 14. PMID: 37453881.