Production Systems and Information Engineering
Volume 11 (3), pp. 142–153.
https://doi.org/10.32968/psaie.2023.3.10

ONTOLOGY METRICS AS UML METRICS ASPECT

ANITA AGÁRDI
University of Miskolc
Hungary Institute of Information Technology
agardianita@iit.uni-miskolc.hu

Abstract: The aim of the article is to define ontological metrics from the point of view of UML metrics. Many metrics have been developed for UML over the years. The article presents these metrics and then pr-sents an adaptation to ontological systems. The article presents the most important articles related to UML metrics, and then discusses which of these metrics can be applied to ontological systems and how. The article also presents a sample otology and analyzes it with the presented metrics.

Keywords: *ontology, metrics*

1. Introduction

The validation of systems is an important task nowadays. During the software development, people tried to design various metrics. Thus, a number of metrics were also defined for UML (Unified Modeling Language) [1]. In this article, I will present the adaptation to the ontology [2]. As the first step of the research, I presented, how many publications had already been published on the topic and how current the research topic was. I used the google scholar database for the search. I performed the searches on 07. 10. 2022 and in the interval 2010–2022, and set the following search parameters:

- Keyword=’software modelling’
- Keyword=’ontology’
- Keyword=’UML’
- Keyword=’Metrics’
- Keyword=’Ontology metrics’
- Keyword=’UML metrics’
- Keyword=’Ontology and UML metrics’

Results for keyword='software modelling'

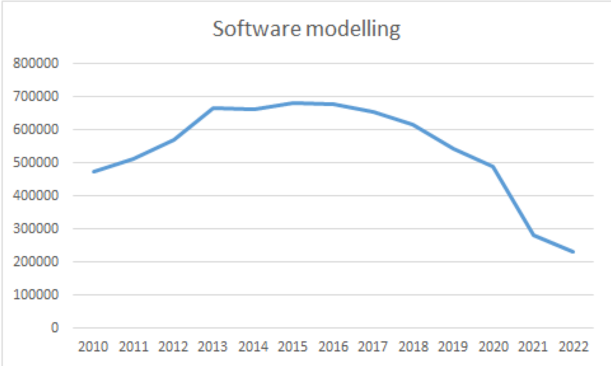

Figure 1
Results for software modelling

Figure 1 shows that the most scientific articles were written in the 2013–2017 period on the topic of software modeling. From 2018, fewer articles were published in this topic. The maximum number of publications was 680,000, which was made in 2015, and the minimum was 281,000, which was made in 2021. The average number of publications per year was 542,846. A total of 7,057,000 publications were written during this period.

Results for keyword='ontology'

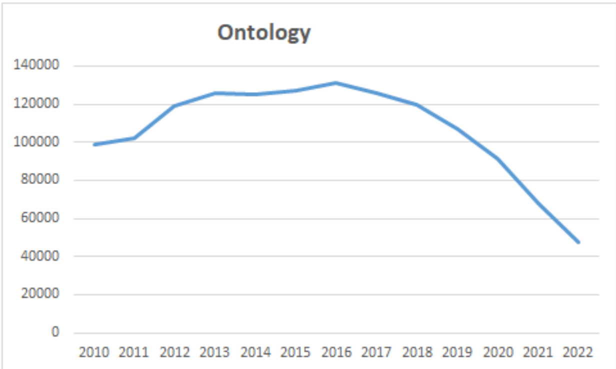

Figure 2
Results for Ontology

Figure 2 shows the search results for the keyword 'ontology'. Articles on the topic were very popular here from 2013 to 2018. As of 2018, the number of articles is constantly decreasing. The most publications were written in 2016, a total of 131,000. And the fewest publications were in 2021, when 67,800 were published. The average number of publications per year is 106,792. During this period, the authors wrote a total of 1,388,300 publications.

Results for keyword='UML'

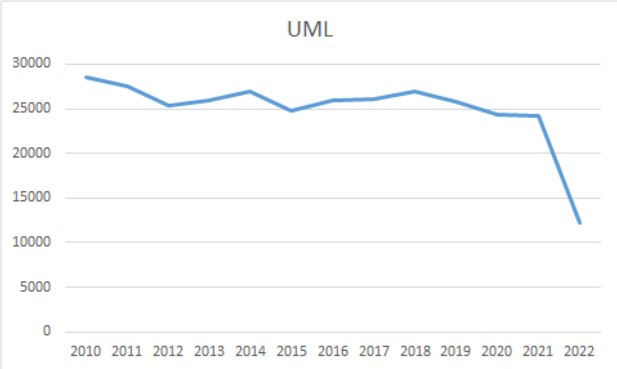

Figure 3
Results for UML

Figure 3 shows the results obtained for the UML keyword. As of 2020, the number of articles related to this modeling has decreased. The authors wrote the most articles in 2010, a total of 28,500 articles. And the fewest articles in 2021, a total of 24,200 pieces. The authors wrote an average of 24,985 articles per year. During this period, a total of 324,800 articles were published.

Results for keyword='Metrics'

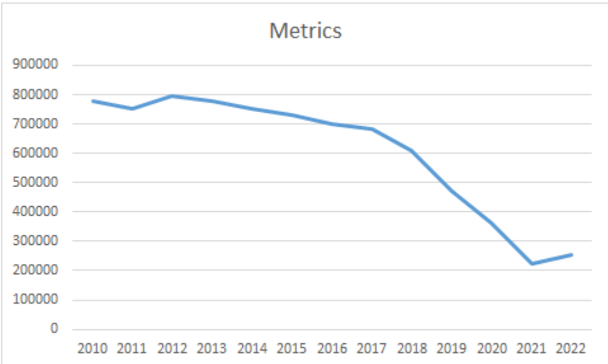

Figure 4
Results for Metrics

Figure 4 shows the results for the Metrics keyword. From 2018 onwards, the number of articles published in the topic is gradually falling. The authors wrote the most articles in 2012, 796,000 articles. And in 2021, the fewest, 223,000 publications were published. On average, 606,846 publications were published on the topic each year. And a total of 7,889,000 pieces, considering the examined period.

Results for keyword='Ontology metrics'

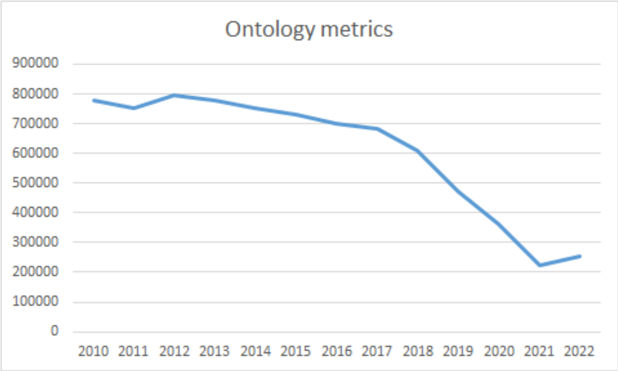

Figure 5
Results for Ontology metrics

Figure 5 illustrates the results of the Ontology metrics search. At the bottom of the result, the number of articles decreases from 2016, but it started to increase in 2022. In 2012, the authors wrote the most articles, 796,000 articles. The least number of articles is 223,000 pieces in 2021. On average, the authors wrote 606,846 articles per year. A total of 7,889,000 articles were published in the examined period.

Results for keyword='UML metrics'

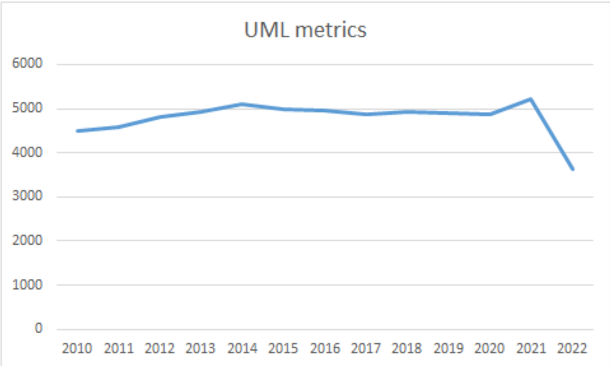

Figure 6
Results for UML metrics

Figure 6 illustrates the UML metrics search results. The number of published articles was approximately the same every year, the most articles were published in 2021, 5,220 articles, while the fewest articles were published in 2010, 4,510 articles. An average of 4,792 articles were published each year. A total of 62,290 articles were published in the examined period.

Results for keyword='Ontology and UML metrics'

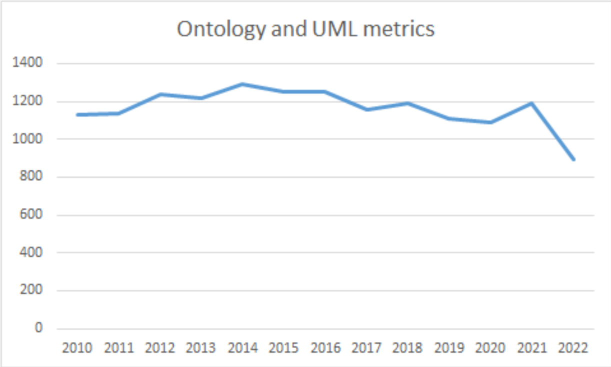

Figure 7

Results for Ontology and UML metrics

Figure 7 illustrates the Ontology and UML metrics search results. Based on the results, the authors wrote an almost unifying number of articles each year. The authors wrote the most articles in 2014, 1290 articles. The fewest articles were published in 2020, with 1,090 articles. On average, the authors wrote 1166 articles per year. A total of 15,154 articles were published in the examined period.

The summary results are illustrated in Figure 8.

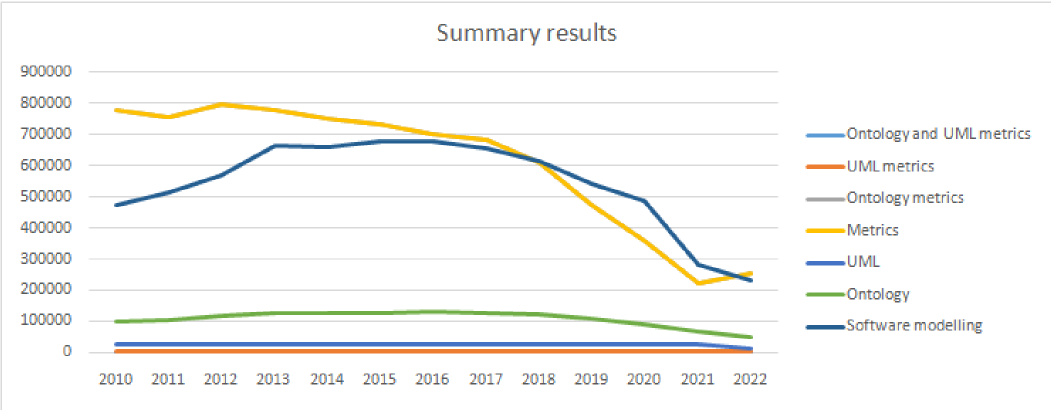

Figure 8

Summary results

Figure 8 shows that Metrics and Software modeling have the most publications. The number of publications will decrease from 2018.

2. UML metrics

In this chapter, I present the most important publications related to UML metrics.

Genero, M., Piattini, M. & Calero, C. [3] presents metrics that can be used in object-oriented software development. According to them, the quality of software products is determined by factors such as functionality, reliability, usability, efficiency, maintainability and portability. The following processes are important for object-oriented software designers:

- identify weak design points and correct them.
- choose objectively between design alternatives.
- modifying based on external quality characteristics and metrics, such as maintainability and reusability.
- The metric groups below are defined by the authors in the articles:
- CK metrics
- Li and Henry metrics
- MOOD metrics
- Lorenz and Kidd metrics
- Briand et al. metrics
- Marchesi's metrics
- Bansiya et al. metrics
- Genero et al. metrics

In the publication of Manso, M. E., Genero, M. & Piattini, M. [4], 8 metrics were used due to the use of UML relationships, and 3 metrics were used to measure their size. An algorithm based on principal component analysis is presented. The obtained results show that the metrics related to associations, aggregations, generalizations and dependencies are the most relevant, while those related to size are not so important.

The authors Chen, Y., Boehm, B.W., Madachy, R. & Valerdi, R. [5] present UML metrics through an eServices application, and analyze this application from a metric point of view.

Yi, T., Wu, F. & Gan, C. [6] presented different metrics for class diagrams. These metrics help software developers analyze the reliability, maintainability, and complexity of systems. This article examines the limitations of existing metrics. Performs a comprehensive analysis and comparison of the complexity of class diagrams. In addition, this paper performs a statistical analysis.

Genero, M., Miranda, D. & Piattini, M. [7] discuss UML metrics. Two types of metrics: size and structural complexity metrics.
Size metrics:

- NEntryA (Number of entry actions)
- NexitA (Number of exit actions)
- NA (Number of activities)

- NSS (Number of simple states)
- NCS (Number of composite states)
- NE (Number of events)
- NG (Number of guards)
- Structural metrics
- McCabe (Cyclomatic Number of McCabe)
- NT (Number of transitions)

Girgis, M. R., Mahmoud, T. M., & Nour, R. R. [8] describe a tool that automates the calculation of important metrics applicable to UML class diagrams. The article also describes some areas where OO design metrics can be applied to improve software quality. Finally, the article presents a case study.
The authors present the following metrics:

- WMC: Weighted Methods per Class
- APPM: The Average Parameters Per Method
- NAssoc: Number of Association
- NAgg: Number of Aggregation
- NDep: Number of Dependencies
- NGen: Number of Generalization
- NGenH: Number of Generalization Hierarchies
- NAggH: Number of Aggregation Hierarchies
- MaxDIT: Maximum DIT
- MaxHAgg: Maximum HAgg
- NAssocC: Number of Association per Class
- HAgg: Height of class within aggregation hierarchy
- NoDP: Number of Direct Parts
- NP: Number of Parts
- NW: Number of Wholes
- NAgg: Multiple Aggregation
- NDepIn: Number of Dependencies In
- NDepOut: Number of Dependencies Out
- NOM: The number of local methods
- SIZE2: #Attributes + # local methods
- PIM: Public Instance Methods
- NIM: All the public, protected, and private methods
- NIV: Number of Instance Variables
- DAC # attributes in a class that have another class
- DAC’ # different classes that are used as types
- OCAEC Class-attribute export coupling
- OCAIC Class-attribute import coupling
- ACAIC Ancestor class-attribute import coupling

- DCAEC Descendant class-attribute export coupling
- ACAEC Ancestor class-attribute export coupling
- DCAIC Descendant class-attribute import coupling
- ACMEC Ancestor class-method export coupling
- OCMIC Class-method import coupling
- DCMEC Descendant class-method export coupling
- OCMEC Class-method export coupling
- ACMIC Ancestor class-method import coupling
- DCMIC Descendant class-method import coupling
- NASS Number of Associations
- DCC Direct Class Coupling metric
- DIT Depth of inheritance
- NOC Number of childern
- AIF Attribute Inheritance Factor
- MIF Method Inheritance Factor
- NMO Number of methods overridden by a subclass
- NMI Method inherited by a subclass
- NMA Number of methods defined in a subclass
- SIX Specialization Index metric for each class

3. Case study with ontology system

In this chapter, I present the adaptation of individual UML metrics to ontological systems. I will also present an ontological system and then analyze the system from a metric point of view. I used the following metrics and their adaptations:

- WMC (Weighted Methods per Class): The number of properties for each class.
- Average WMC (Weighted Methods per Class): the average of the WMC values.
- DIT (Depth of Inheritance): For the individual classes that are neglected children in the tree.
- Average DIT (Depth of Inheritance): The average of the DIT values.
- NOC (Number Of Childrens): The number of children in each class.
- Average NOC (Number Of Childrens): The average of the NOC values.
- DAC: The number of properties of the class object.
- Average DAC: The average of the DAC values.
- OA1: Total number of classes.
- OA2: Total number of inheritance hierarchies.

Analyzing ontologies with UML metrics is a supplement to ontology metrics. Analyzes can be important in order to obtain more information about individual ontologies. With the metrics, we can evaluate how thoroughly the individual ontologies have been created, how complicated they are, how large the knowledge base is.

3.1. College Mngt Sys

The College Mngt Sys [9] ontology models a university or college. The 'College' class consists of the 'Department' and 'Researchgroup' classes. The system also includes a 'Course' class, which defines the type of courses. The following three types of courses are available in the system: 'PGCourse', 'PhD', 'UGCourse'. Events are organized at the university, these are represented by the 'Event' class, whose subclasses can be as follows according to the type of event: 'Conference', 'Meeting', 'Presentation', 'Workshop'. The 'Conference' can probably be a scientific conference or another promotional conference. A 'Meeting' is a small-scale event. 'Presentation' can be a presentation of a problem or topic. The 'Workshop' can be, for example, a job search workshop. The university also has a library, which can be of two types: 'PGLibrary', 'UGLibrary'. Regarding the educational institution, there are two types of people in the system: 'Employee' and 'Student'. A university-related project can be a 'DevelopmentProject' or a 'ResearchProject'. Publications written by lecturers, researchers and students have also been categorized, there are three different categories in the system: 'Article', 'Book' and 'Journal'. The system does not contain object properties. The system contains datatype properties, the following: 'abstract', 'address', 'applicationAreas', 'approaches', "can be reached at", 'date', 'dateOfLastModification', 'description', 'email', 'eventTitle', 'fax', 'firstName', 'firstPage', 'hasEndDate', 'hasEndTime', 'hasStartDate', 'hasStartTime', 'hasTenure', "is age", "is researching", 'keyword', 'lastName', 'lastPage', 'location', 'mailingLists', 'middleInitial', 'name', 'name', 'number', "office room No.", 'phone', 'photo', 'proceedingsTitle', 'productFAQ', 'productMailingList', 'productName', 'series', "telephone number", 'title', 'title', 'Title', 'type', 'volume', 'webpages', 'year', 'year'.

However, only a few of these are attached to a class, so they are actually used. The following properties are actually used by the system: 'hasEndDate', 'hasEndTime', 'hasStartDate', 'hasStartTime', 'hasTenure'. Events have these properties.

The system also contains 4 entities, which are as follows: 'AI', 'Graphics', 'HCI', 'Network'. However, these are not connected to classes either. The VOWL representation of the system is shown in Figure 9.

The evaluation of the metrics for the ontology is as follows. WMC (Weighted Methods per Class) and Average WMC (Weighted Methods per Class) are not available due to the lack of properties. DIT (Depth of Inheritance) values are shown in Figure 10.

The figure shows that almost all classes have this value of 0 or 1, which means that there are not many parent-child relationships. Average DIT (Depth of Inheritance) value is also 0.72. This is shown in Figure 11.

The NOC (Number Of Childrens) values are illustrated in Figure 12, according to which the class that has children has only one child. Average NOC (Number Of Childrens) is 0.28. OA1 value, i.e., the total number of classes is 25, and OA2 value is 1, the total number of inheritance hierarchies.

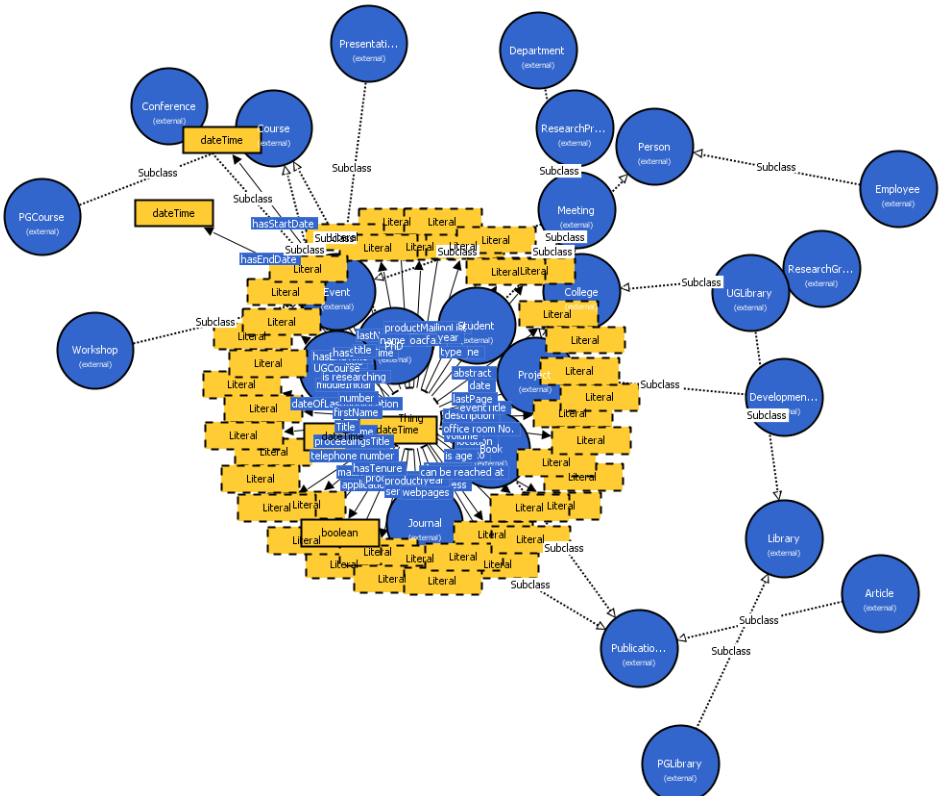

Figure 9
VOWL representation of College Mngt Sys

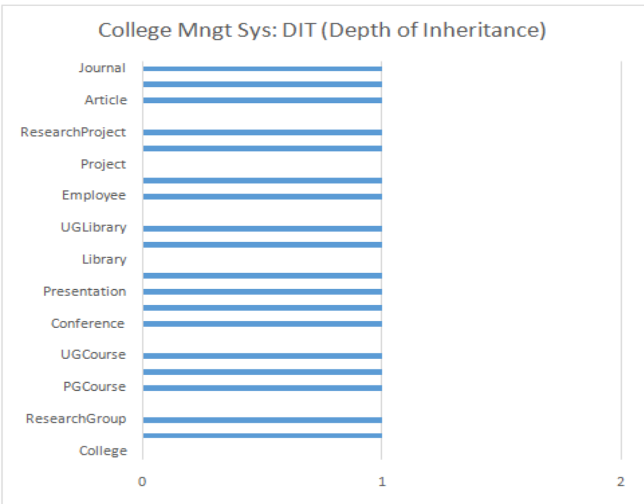

Figure 10
College Mngt Sys Depth of Inharicance values

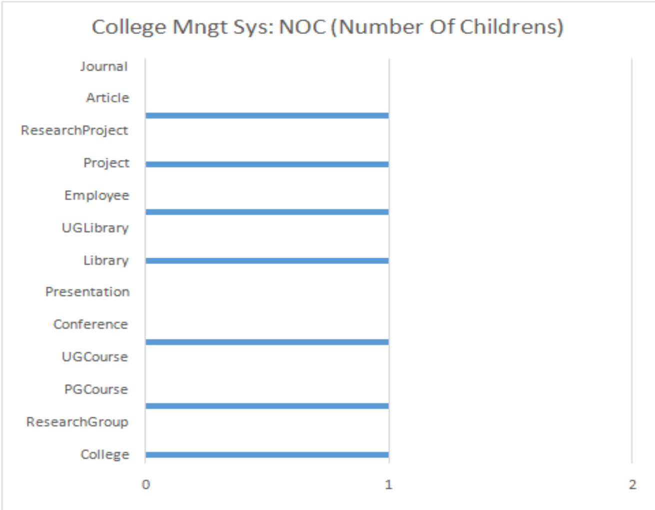

Figure 11

College Mngt Sys: NOC (Number Of Childrens)

Based on the metrics, the following conclusions can be drawn about the College Mngt Sys ontology. This ontology does not contain many classes, and the classes do not contain many subclasses either (NOC values 0 or 1 and DIT values 1). The ontology does not contain object properties, only data type properties. The number of properties of the data type is quite large compared to the number of classes, which means that each class is described in detail. This ontology does not contain individuals, so it can practically be considered a framework that can be supplemented by individual developers. The ontology is not complicated, it contains only a few classes, the class-subclass hierarchy is not large, so this ontology can be a good framework for further ontologies that describe the knowledge base in more detail. The ontology contains a minimal hierarchy and will probably be expanded in the future with more classes and properties, so that it can be used in more fields.

4. Conclusions and future work

In this article, the evaluation of ontology systems is presented from the point of view of a UML metric. The article first conducts a literature search. With the help of Google Scholar, the article examines how current the research topic is. The search was performed using the following keywords: ‘software modeling’, ‘ontology’, ‘UML’, ‘Metrics’, ‘Ontology metrics’, ‘UML metrics’, ‘Ontology and UML metrics’. Then some publications related to UML metrics were presented. Then the adaptation of UML metrics to ontological systems. After that, the selected sample system, the College Mngt Sys, was presented and then analyzed from the point of view of UML metrics. My future research is the creation of new metrics, metric analysis of other ontologies.

Acknowledgement

Supported by the ÚNKP-22-3 New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development and Innovation Fund."

References

[1] Uml, O. (2001). Unified modeling language. Object Management Group, 105.

[2] Smith, B. (2012). Ontology. In The furniture of the world. Brill, pp. 46–68.

[3] Genero, M., Piattini, M. & Calero, C. (2005). A survey of metrics for UML class diagrams. *Journal of object technology*, 4 (9), pp. 59–92.

[4] Manso, M. E., Genero, M. & Piattini, M. (2003). No-redundant metrics for UML class diagram structural complexity. In: *International Conference on Advanced Information Systems Engineering*, Springer, Berlin, Heidelberg, pp. 127–142. https://doi.org/10.1007/3-540-45017-3_11

[5] Chen, Y., Boehm, B. W., Madachy, R. & Valerdi, R. (2004). An empirical study of eServices product UML sizing metrics. In: *Proceedings. 2004 International Symposium on Empirical Software Engineering, 2004. ISESE'04*, IEEE, pp. 199–206. https://doi.org/10.1109/ISESE.2004.1334907

[6] Yi, T., Wu, F. & Gan, C. (2004). A comparison of metrics for UML class diagrams. *ACM SIGSOFT Software Engineering Notes*, 29 (5), pp. 1-6. https://doi.org/10.1145/1022494.1022523

[7] Genero, M., Miranda, D. & Piattini, M. (2003). Defining metrics for UML statechart diagrams in a methodological way. In: *International Conference on Conceptual Modeling*, Springer, Berlin, Heidelberg, pp. 118–128. https://doi.org/10.1007/978-3-540-39597-3_12

[8] Girgis, M. R., Mahmoud, T. M. & Nour, R. R. (2009). UML class diagram metrics tool. In: *2009 International Conference on Computer Engineering & Systems*, IEEE, pp. 423–428. https://doi.org/10.1109/ICCES.2009.5383226

[9] College Mngt Sys ontology: https://github.com/ayesha-banu79/Owl-Ontology, accessed: 02. 11. 2022.