Formal Model Of A Pedagogical Pattern Language

Emil Hadzhikolev, Stanka Hadzhikoleva, Emil Yonchev, Todor Rachovski

Abstract: Pedagogical patterns provide a convenient way for an abstract description of pedagogical situations and tasks, as well as possible solutions. They are an appropriate tool for transferring pedagogical experience and sharing good practices. Multiple patterns suitable for use in a common problem area and described in the same format form a pattern language. In recent years, experiments have been conducted to build e-learning systems based on pedagogical patterns. The question arises of combining and matching patterns in a way that allows effective training. The paper proposes a formal model of pedagogical pattern language. It includes a multi-layered formal model for pattern descriptions that makes it suitable for use in different contexts, including for software implementation, e-learning and more. A model of a pattern instance is suggested. It provides a way to describe a specific case of pattern use and it is appropriate for software implementation.

Index Terms: pedagogical patterns, pedagogical pattern model, model of a pedagogical pattern instance, pedagogical pattern language.

1. INTRODUCTION

PATTERN is a term first used by the Australian architect Christopher Alexander in the description of reusable architectural solutions. According to him, "Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice" [1]. In this context, Bergin defines patterns as a method of describing and sharing knowledge. This is because they offer a way to share good practices in a way that allows each practice to be used by many people in many different ways [2]. Patterns are widely used in various subject areas - software engineering, architectural design, business, and more. In recent years, they have been used in pedagogical science as well.Pedagogical Patterns are models for describing pedagogical tasks along with possible solutions. They cover expertise on the practice of teaching and learning. The aim is to capture the essence of practice in a compact form that can be easily provided to those in need of knowledge [2].Internet technologies have led to irreversible changes in education. Nowadays there are many software environments for conducting e-learning, online courses, digital educational resources, and more. Technology however cannot replace teachers. Many scientific teams work on tasks aimed at finding a way to make e-learning more attractive to learners - through various technologies for adaptability of training, gamification and more. The teacher's role in training is significant, which is confirmed by the fact that many learners prefer a combination of online education with face-to-face training. This brings up the question of using pedagogical patterns in online education. The construction of online courses based on pedagogical patterns could diversify the educational process, provoke the interest of learners and motivate them for more active and productive work.The article presents our work on the creation of a formal language for pedagogical patterns. It aims at defining a framework appropriate for creating educational software systems based on pedagogical patterns. The first version of the model of pedagogical pattern and pattern instance is described in [3]. This article presents the models expanded by adding new descriptive features that allow better use of patterns and instances, incl. the necessary competencies the teacher and learners must have to be able to use the patterns; classifications according to different taxonomies, which allows classifying the patterns according to the pedagogical goals set, respective construction of appropriate pattern sequences in line with the different levels of knowledge of the learners, etc.

- *Emil Hadzhikolev, Assoc. Prof.University of Plovdiv "Paisii Hilendarski", Department of Mathematics and Informatics, E-mail: emil.hadjikolev@gmail.com.*
- *Stanka Hadzhikoleva, Assoc. Prof.University of Plovdiv "Paisii Hilendarski", Department of Mathematics and Informatics, E-mail: stankah@gmail.com.*
- *Emil Yonchev, Ph.D. student University of Plovdiv "Paisii Hilendarski", Department of Mathematics and Informatics, E-mail: etj_bg@abv.bg.*
- *Todor Rachovski, Ph.D.University of Plovdiv "Paisii Hilendarski", Department of Mathematics and Informatics, E-mail: todormr@gmail.com.*

2 PEDAGOGICAL PATTERNS

Pedagogical patterns are high-level models that have been applied in various areas of teaching and learning. Generally speaking, pedagogical patterns are a tool for describing pedagogical experiences and good practices.

Weisburgh conducts a discussion that aims to identify an approach to documenting good practices in education and training by means of pedagogical patterns [4]. He summarizes 9 aspects - Name, Problem, Context, Forces, Solution, Examples, Resulting Context, Rationale, and Related Patterns. Their use is illustrated by the following statement:

1. If I find myself in some Context like Examples, and I face this Problem, with these Forces or constraints, but my situation is different from these Related Patterns,
2. Then I should think like this Rationale. If I want this Resulting Context, then I should follow this Solution.
3. And here is a Name to help me remember this scenario.

Weisburgh distinguishes 9 aspects of pattern description:

- Name – word or short phrase that refers to the model. This allows quick association and search of a pattern.
- Problem – identifying a problem, including the goal or desired result, and information that indicates that this problem exists.
- Context – prerequisites that must exist so that this problem can arise; it is often a situation.
- Forces – a description of the preconditions or

limitations and how they interact. Some of the preconditions may be contradictory.
- Solution – instructions that may include different variations. The solution could contain pictures, diagrams, prose, or other media.
- Examples – sample applications and solutions, analogies, visual examples and well-known applications that can help the user understand the context;
- Resulting Context – results after the pattern application, including post-conditions and unwanted consequences. This may include new issues resulting from the solution of the original problem.
- Rationale – logical reasoning that provokes the choice of the respective pattern. The rationale includes an explanation of why this pattern works, how to solve the problems and overcome the constraints to achieve the desired result.
- Related Patterns – differences and connections with other patterns that solve similar problems.

A set of pedagogical patterns focused on the same problem area and described in the same format form a pattern language. There are numerous languages for pedagogical patterns described in literature – for learning foreign languages [5], for learning object-oriented design templates [6], for training on object technologies [7], for creative learners [8], for active learners [9], for a differentiated approach to training [10], for technology-assisted teaching and learning [11], for e-learning [12], [13], [14] and others.Iba and Miyake offer pattern languages which they call "Learning Patterns". Initially, they were designed for use by students but were subsequently adapted to all types of learners in various fields - engineering, business, science and more. Particularly significant is their work on creating a pattern language for creative learners [8] and a pattern language for active learners [9]. Each of the two languages includes about 40 patterns. They are organized in 3 layers depending on the level of abstraction. In the top layer is the root pattern - "learning design". The second layer is formed by three fundamental patterns. They show the main behavior categorizing the specific patterns in the third layer. The third layer contains the patterns describing specific learning techniques. Each pattern is described in a format that consists of the following elements: "Pattern number", "Pattern name", "Introduction", "Illustration", "Context", "Problem", "Solution", "Related Patterns". Context-based categorization is set to enable quick search for a suitable pattern. 5 categories have been defined – "At beginning," "For goal setting," "In activity," "For output" and "At dead end". Each category consists of 4 contexts to which the specific patterns are classified. This allows users to search for patterns depending on the situation in which they find themselves - e.g. want to start learning something new, write an article, acquire a new skill, improve skills and so on.Chatteur and colleagues offer design for e-learning pedagogical patterns. They adapt the classical structure of Alexander's pattern by adding a new pedagogical layer. In this way, they shift the emphasis of the e-learning design from usability to pedagogy. The methodology for the development of patterns aims to put forward the theory of learning through constructivism and experiments. Once a problem has been defined, it requires the seeking of solutions, teaching strategies and alternative approaches in line with the pedagogical theory [13]. Pattern language KISTA consists of 28 patterns typified in 6 categories - "Digitization of Educational Materials", "Digital Learning Environments", "Open Classrooms", "Motivating Classroom", "Training and Documentation Technology", "Students' Digital Arenas" [15]. Here the focus is on the didactic and pedagogical aspects of technology use in teaching and learning.Patterns for technology-supported teaching and learning are presented in [11]. The described patterns are 29, grouped in 4 groups: "Learner-centered Design", "Communities of Learners", "Social Media and Learner Interaction in Social Spaces", and "Evaluation and Feedback". Each group includes a set of patterns along with the cases they are derived from and scenarios in which these patterns are applied in a new, different context.The conducted survey shows that pedagogical pattern languages use pattern-description formats similar to the ones in Weisburgh's model and the classical model of Christopher Alexander. They are appropriate for describing pedagogical experience and sharing good practices but cannot be used to support online learning.

3 PEDAGOGICAL PATTERN MODEL

The use of pedagogical patterns in the new context of conducting e-learning adds new requirements for their description. It is necessary to specify additional characteristics meant to lay the basis for their automated processing, incl. pattern classification, necessary technological prerequisites and requirements for their use, descriptive metadata, etc. The model we propose has 4 layers (fig. 1). It includes a basic, classification, technological, and a meta-layer. Each layer contains many specific characteristics that are used for different purposes. The basic layer includes the main characteristics of the pattern that are typically used for pedagogical patterns description.
- Pattern name – concise description of the problem and its solution;
- Pattern language the pattern belongs to;
- Abstract – a short paragraph outlining the key elements;
- Problem – detailed description of the problem and its context. It may include a list of conditions that need to be met so that the pattern use makes sense;
- Analysis – analysis of the problem and possible solutions;
- Known solutions – best practices in problem solving, good examples of theoretical or practical solutions to the problem;
- Context – situations in which the pattern is appropriate to use;
- Consequences - consequences of use, possible implementation problems and relevant solutions, other remarks;
- Related patterns – patterns that are appropriate to combine with.

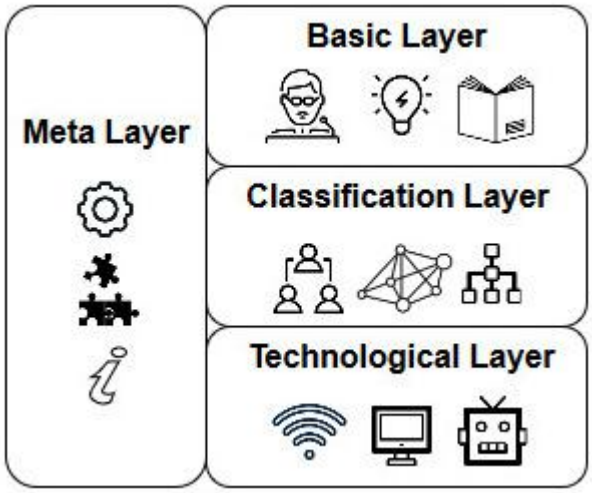

Fig. 1. Conceptual model of a pedagogical pattern

The classification layer includes numerous different classification types that define methodological, pedagogical and other aspects of the pattern use. The main characteristics of the layer are as follows:

- Type of learning – indicates whether the pattern is applicable to face-to face, mixed or online training;
- Classification according to the pattern's primary purpose - for training or for assessing the learners' knowledge and skills;
- Classification according to the type of learning - perception by verbal, paper or digital methods; discovery learning; learning through practical activities; learning through creative work etc.;
- Pattern classification according to Bloom's Taxonomy - Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation [16]. It is possible to maintain information about more than one taxonomy in the different software implementations, e.g. SOLO Taxonomy [17], Finks Taxonomy [18], Webb's Depth of Knowledge [19] and so on. Bloom's taxonomy is chosen because it is widely used in pedagogy to determine the learning objectives;
- Classification according to the main stages of the training process - introductory; for initial introduction to new material; for new knowledge and formation of concepts; for applying the new knowledge in practice; for forming habits; for repetition and summary, etc.;
- Classification according to the main didactic goal - for knowledge development and acquisition; for skill and habit formation; for knowledge summarizing and systematizing; for knowledge, skills and habits application; for knowledge, skills and habits control and assessment; combined pattern and more;
- Classification according to the organization of the students' activities - for individual work, work in pairs, work in groups, course work, etc.;
- User-defined classifications, specific to the particular pattern language.

The technological layer defines the technical and technological requirements for the pattern use, as well as the necessary competencies of the trainer and trainee.

- Specific learner competencies – requirements for specific competencies that the learner needs to be able to use the pattern, such as computer literacy, programming skills, etc.;
- Specific trainer competencies – requirements for specific competencies the trainer needs, such as work with specialized hardware, software, interactive technologies, etc.;
- Technical resources for the teacher – what the teacher needs to use the pattern – an e-learning environment, tools for online testing, applications for developing interactive educational resources, specialized software, other necessary resources, etc.;
- Technical resources for the learner - what the learner needs to use the pattern – device (computer, tablet, smartphone, PDA device), internet access, browser, virtual machine, operating system, specialized software, etc.

The Meta layer supports descriptive information about the pattern:

- Pattern source – link to the original source of the pattern (if the pattern is published elsewhere);
- Alternative pattern names – names under which the pattern is known in literature;
- Pattern significance (according to the author) - indicates whether the pattern is widely or less commonly used;
- User rating - user assessment on the pattern usefulness and applicability;
- Accumulated rating - rating calculated on the basis of the ratings of pattern instances;
- Language – natural language used in the pattern description – English, German and so on;
- Author – names of the pattern author;
- Contact details of the author;
- Creation date of the pattern;
- Date of latest update of the pattern;
- Pattern version.

Here it should be noted that it is not necessary to specify all the characteristics when describing a pattern. Besides, a characteristic may have multiple values. For example, the Technical Resources for the Learner characteristic may have multiple values for a particular pattern – computer, internet access, software application for business process design, etc. User rating and Accumulated rating are dynamic features and their value is calculated depending on the user opinion and assessment of the pattern usefulness and applicability.

4 MODEL OF A PEDAGOGICAL PATTERN INSTANCE

Pedagogical patterns make abstract descriptions of problems and their solutions but do not provide specific ways of pattern use. With the term pattern instance, we define a specific case of using a pedagogical pattern along with the educational resources used. The instance is a description of good practice shared by a learner or a trainer. If we look at the pattern as an idea to solve a common problem, then the pattern instance can be defined as a specific solution to a specific problem. It's appropriate to make the following analogy with the object-oriented programming (OOP) languages: a pattern refers to a pattern instance the same way that a class refers to a class instance. Unlike OOP however, where the instance has the

same structure as the inherited class, the pattern instances have a structure slightly different from the pattern structure. The pattern instance, like the pattern, has 4 layers - basic, classification, technological and meta-layer. The classification and technological layers of the instance have the same characteristics as the pattern. Moreover, the sets of possible values described in the pattern can be specified in them. Different characteristics only exist in the basic and meta layers, as described below. The basic layer contains information for identifying the instance and for the specific use case:

- Identifier - Pattern instance identifier;
- Implemented pattern;
- Description - how the pattern is used (situations, specific ways);
- Resources - link to educational resources used or developed for the instance needs;
- Subject area - in which field the instance is applicable – discipline, subject, scientific field, etc.
- Learners – age and education-related characteristics of the learners using the instance.

The Meta-layer supports descriptive information about the pattern instance:

- Educational institution - in which educational institution the pattern instance has been developed;
- Date of use of the instance;
- User rating – user assessment of the pattern usefulness and applicability;
- Language – the natural language used for the pattern description of - English, German, etc.;
- Author - names of the author who developed the pattern instance;
- Contact details of the lecturer;
- Creation date of the instance;
- Date of latest update of the instance.

The proposed pattern instance model is suitable for sharing good pedagogical practices described in different ways and related to various fields, incl. for modelling serious games for education [20], difficulties in the programming training [21], ideas to improve learner experience [22], to study learner profiles [23], etc.

5 FORMAL MODEL OF PEDAGOGICAL PATTERN LANGUAGE

Pedagogical pattern languages described in the scientific literature are usually focused on a single problem area. The patterns described in them use the same format that is different for the different languages. The expanded models of patterns and pattern instances provided in the article add a number of additional characteristics that can be used as a universal basis for software presentation of both existing and newly created patterns and instances.Building a common online catalog of patterns and instances will enhance the competencies of lecturers by giving them the opportunity to share, explore and apply good practices. The first version of the catalog created with our first version of models of patterns and instances is described in [24].The next step is the creation of a learning management system based on patterns. It sets additional requirements for functional extension of the catalog, including advanced pattern and instance models. The new classification features added to the pattern and instance models presented in this article help identify and classify the patterns, and respectively, work with them in the learning environment.Using the classification characteristics we define the term pattern type: Def. 1. A pattern type is defined by name and consists of all patterns of the pattern language that meet specified classification characteristics. Let us denote the set of all patterns with P.We describe $T \subseteq P$, the set of all patterns of a certain type T, as T (P, C), where T is the type name, P - the set of patterns of the respective type, C – constraints enforced on the patterns and instances. A distinctive feature is that restrictions can be imposed on both patterns and their instances existing in the system.As an expansion of the pattern type concept, we can say that one type can be defined upon another by enforcing additional restrictions. For example, type T_2 can be defined upon an already existing type T_1 in the following way: $T_2(T_1, C)$. Here are some exemplary restrictions for patterns of a given type described informally: "Classification according to the main stages of the training process: introductory", "Classification according to the organization of the students' activity: individual work", "scientific field: biology", and more. The main stages in the creation of pattern-based courses and lessons are:

- Determining the main sequence in which the themes / lessons are presented in the course.
- Defining the lesson structure. Lessons often have a similar structural sequence of exposition. An example of an elementary logical structure of a lesson is: introduction, main part and conclusion. For a pattern-based lesson, the basic linear structure of a lesson can be considered as a linear sequence of patterns: LessonType = (T_1, T_2,… T_n) where T_i are pattern types, for i = 1..n.
- Creating lessons. An appropriate LessonType lesson structure is selected for each lesson, and for each lesson element, a pattern is chosen according to the pattern type, as well as an instance that corresponds to it. For a specific lesson or a set of lessons, specific patterns can be selected, corresponding to the sets of pattern types: LessonPatterns = (P_1:T_1, P_2:T_2,… P_n:T_n), where P_i are type T_i patterns, for i=1..n. Respectively, we describe a specific lesson as a linear sequence of instances of the selected patterns: Lesson = (PI_1:P_1, PI_2:P_2,… PI_n:P_n) where PI_i are instances of the corresponding P_i patterns, for i=1..n.

Often patterns described in a certain pattern language are suitable for use not only in the problem area recommended for that language, but in others problem areas as well. Assuming this, we define the concept of dynamic formal pattern languages. These are pattern languages that consist of patterns from different pattern languages, refer to a new subject area, and have logic for describing sequences of patterns and their respective instances. Similarly to natural and formal languages, each formal pattern language includes a set of patterns and rules for combining them [25]. Pattern types are an analogy of parts of speech in a natural language, and specific instances - specific words in the formal language. The rules determine the possible combinations of words in the language and, respectively, the possible specific lessons composed of pattern instances. In this context, we consider pattern instances as words that are indivisible structural elements of lessons.Def. 2. Formal pattern language PL is a set of patterns P of certain pattern types and rules R for

combining them: PL = (P:T, R).Existing pattern languages satisfy the definition of a formal pattern language, with the rules for combining patterns being empty sets. On the other hand, based of this broad definition we could create many new formal pattern languages to have at hand in various cases. As a mandatory restriction, for example when planning lessons, we can indicate the constraint – pattern instances in all lessons must belong to the same formal pattern language. Then, by applying the rules of the formal pattern language, we create specific sentences using specific instances of the language patterns. Thus, we expand the above-described fixed structure of a lesson with the possibility of creating many lessons with different structure based on the same formal pattern language. Def. 3. A lesson is a combination of pattern instances belonging to a certain formal pattern language PL and satisfying the PL language rules for combining them: Lesson = (PI_1:P_1, PI_2:P_2,… PI_n:P_n) where PI_i are instances of the relevant patterns P_i, for i = 1..n, P_i belongs to PL and the order of instances PI_i is determined by the PL rules.Creating lessons based on pre-defined formal pattern languages would facilitate the work of authors of learning resources. By complying with the rules for combining patterns set out in the pattern language, the learning environment can automatically offer only the required set of pattern instances appropriate for creating the new lesson.

6 CONCLUSION

Pedagogical patterns provide a way for systematic description of various principles, methods, forms, means and procedures for conducting educational activities. They can be used as a tool for constructing an effective educational process.The model of a pedagogical pattern presented in the article expands the existing ones by adding a number of characteristics that facilitate the use of patterns, incl. for building software systems. The model of a pattern instance provides a way to describe a specific case of pattern use. The two models contain many characteristics related to the classification of patterns and instances from the perspective of the educational process and in accordance with the main stages of the training process, the main didactic goal, the organization of the learners' activity, the type of learning, the Bloom Taxonomy and others. This allows for choice of patterns and their arrangement in a way suitable for the construction of attractive training courses.The use of formal pedagogical pattern languages will reveal new perspectives for e-learning. This includes the opportunity to automate the creation of courses compiled by interchangeable pattern instances, multiple use of pattern instances, sharing pedagogical experience, and more.

ACKNOWLEDGMENT

The work is funded by the SP19-FMI-004 project entitled "Forming scientific competencies in the Students School for ICT Innovations" at the Research Fund of the Plovdiv University "Paisii Hilendarski".

REFERENCES

[1] C. Alexander, The Timeless Way of Building. New York: Oxford University Press, 1979.
[2] J. Bergin, J. Eckstein, M. Manns, H. Sharp, etc., Pedagogical Patterns: Advice For Educators. CreateSpace Independent Publishing Platform, 2012.
[3] St. Hadzhikoleva, E. Hadzhikolev, and T. Rachovski, "Pedagogical Patterns and Information Technologies", Announcements of Union of Scientists - Sliven Journal, vol. 32 (1), 2017, pp. 63-69.
[4] M. Weisburgh, "Documenting good education and training practices through design patterns", Formal Discussion Initiation, International Forum of Educational Technology & Society, 2004.
[5] C. Köppe and M. Nijsten, "A pattern language for teaching in a foreign language: part 1", EuroPLoP'12 17th European Conference on Pattern Languages of Programs, Irsee, Germany — July 11 - 15, 2012.
[6] C. Köppe, „A Pattern Language for Teaching Design Patterns. Transactions on Pattern Languages of Programming III". Lecture Notes in Computer Science, vol 7840, Springer, Berlin, Heidelberg, 2013.
[7] Pedagogical Patterns Project, 2012, available at http://www.pedagogicalpatterns.org.
[8] T. Iba and T. Miyake, „Learning Patterns: a pattern language for creative learners II", Proceedings of the 1st Asian Conference on Pattern Languages of Programs, Tokyo, Japan, Mar 16-17, 2010.
[9] T. Iba, T. Miyake, M. Naruse, and N. Yotsumoto, „Learning Patterns: a pattern language for active learners", PloP'2009 Proceedings of the Pattern Languages of Programming 2009. Chicago, Illinois, USA, August 28 – 30, 2009.
[10] Vl. Tzvetkov and S. Hadzhikoleva, "Pedagogical patterns for a differentiated approach to learning", International Scientific Conference UNITECH'2018, Gabrovo, Bulgaria, 2018, pp. 158-163.
[11] Y. Mor, H. Mellar, S. Warburton, and N. Winters, Practical Design Patterns for Teaching and Learning with Technology, Rotterdam: Sense Publishers, 2014.
[12] E-LEN project, 2005, available at http://www2.tisip.no/E-LEN/.
[13] F. Chatteur, L. Carvalho, A. Dong, „Design for Pedagogy Patterns for E-Learning", Eighth IEEE International Conference on Advanced Learning Technologies, Santander, Cantabria, Spain, 2008.
[14] N. Angelova and G. Kiryakova, „Sharing good practices and solutions in e-learning through pedagogical patterns", 7th National Conference on e-learning in higher education Borovets, Bulgaria, 2018.
[15] O. Knutsson, R. Ramberg, "Collaborative Pattern Language Representation of Designs for Learning", Proceedings of the Designs for Learning Conference, Copenhagen, Denmark, 2016.
[16] B. Bloom, M. Engelhart, E. Furst, et. al. Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain. New York: David McKay Company, 1956.
[17] J. Biggs and K. Collis, Evaluating the Quality of Learning. The SOLO taxonomy. New York: Academic Press, 1982.
[18] D. Fink, "A self directed guide to designing course for significant learning". San Francisco: Jossey – Bass, 2009, available at https://www.bu.edu/sph/files/2014/03/www.deefinkandassociates.com_GuidetoCourseDesignAug05.pdf.
[19] K. Hess, "A Guide for Using Webb's Depth of Knowledge with Common Core State Standards", Common Core Institute, 2013, available at http://aimc.alpineschools.org/wp-content/uploads/sites/6/2014/06/Webs-Depth-of-

Knowledge-Flip-Chart1.pdf.
[20] C. Barrera, R. Peytcheva-Forsyth and P. Petris, “SimAULA: Creating a Serious Game to Engage Teachers in Lifelong Learning”, Proceedings of the Online Educa Berlin. Berlin, 2012.
[21] M. Todorova and D. Orozova, “Training difficulties in deductive methods of verification and synthesis of program”, International Journal of Advanced Computer Science and Applications, Vol. 9(7), 2018, pp. 18-22.
[22] D. Paneva-Marinova, R. Pavlov, “Improving learner experience within educational nooks in digital libraries”, Learner Experience and Usability in Online Education, 2018, pp 193-174.
[23] I. Peneva and K. Yordzhev, “The Assertive Profile of the Bulgarian Students in Computer Science and Computer Engineering”, I. J. Education and Management Engineering, 2014, pp. 1-8.
[24] T. Rachovski, V. Tzvetkov, E. Hadzhikolev, and S. Hadzhikoleva, “Digital Repository of Pedagogical Patterns”, Proceedings of the University of Ruse, volume 56, book 3.3, (2017), pp. 16-21.
[25] Formal language, Wikipedia, 2019, available at https://en.wikipedia.org/wiki/Formal_language.