International Journal of Engineering Education Vol. 35, No. 4, pp. 1176–1193, 2019
Printed in Great Britain

0949-149X/91 $3.00+0.00

A Framework for Introducing Personalisation into STEM-Driven Computer Science Education*

VYTAUTAS ŠTUIKYS, RENATA BURBAITĖ, VIDA DRĄSUTĖ, GIEDRIUS ZIBERKAS and SIGITAS DRĄSUTIS
Software Engineering Department, Kaunas University of Technology, Studentų 50, Kaunas, Lithuania.
E-mail: vytautas.stuikys@ktu.edu, renata.burbaite@ktu.lt, vida.drasute@ktu.lt, giedrius.ziberkas@ktu.lt, sigitas.drasutis@ktu.lt

Currently two approaches, personalised learning and STEM, are intensively researched worldwide; however, we still know little about how they should or could be integrated seamlessly. This paper is just about that, proposing a framework for introducing personalised learning in STEM-driven Computer Science (CS) education. We motivate the framework by presenting the methodology and theoretical background for creating personalised content. This framework outlines basic activities relevant to personalised learning in STEM and focuses on the content personalisation and learner's knowledge assessment and self-assessment. We propose a generic structure of Personalised Learning Objects (PLOs) in three categories: component-based LO, generative LO and smart LO (the latter is a combination of the first two). The generic structure integrates those entities with the assessment modules and specifies the distributed interface for connecting them with digital libraries. Firstly, we have developed the learner's assessment model that integrates attributes defined by the revised BLOOM taxonomy and computational thinking skills with the adequate tasks. Then, using this model and applying meta-programming techniques, we have implemented the assessment modules and integrated them with PLOs. We illustrate and motivate this approach by presenting two case studies taken from the real educational setting at the high school. Finally, we evaluate our approach. As STEM relates to technology and engineering disciplines and CS-based modules are within most engineering curricula, our approach contributes to engineering education too.

Keywords: STEM-driven CS education; robotics; personalisation; personalised content

1. Introduction

Currently two separate paradigms, the personalised learning and STEM (STEAM), are widely discussed by educational strategists, individual researchers, practitioners and organizations. This interest goes from the numerous efforts to improve and advance education in the 21st century at all levels worldwide, starting from the primary school and ending with the university or even lifelong learning. Personalised learning (PL) places the learner's needs at the centre of education. That is a long-standing vision in education [1]. PL seeks for a higher motivation and engagement, taking into account the learner's differentiation and preferences in the continuous education cycle. This vision, as highly believed, leads to a faster and deeper knowledge. The STEM-driven education, on the other hand, brings the interdisciplinary knowledge so needed in the modern age. This knowledge have to ensure a better preparedness of former learners to enter the modern labour market fluently after graduating school, college or university [2].

A majority of writers on PL accepts the formal definition provided by the U.S. Department of Education in the 2017 National Education Technology Plan Update [3]. This document defines PL as "*instruction in which the pace of learning and the instructional approach are optimized for the needs of each learner. Learning objectives, instructional approaches, and instructional content (and its sequencing) may all vary based on learner needs. In addition, learning activities are meaningful and relevant to learners, driven by their interests, and often self-initiated*". Typically, researchers define STEM as an interdisciplinary approach to learning, where "*rigorous academic concepts are coupled with real world lessons as students apply science, technology, engineering, and mathematics in contexts that make connections between school, community, work, and the global enterprise, enabling the development of STEM literacy and with it the ability to compete in the new economy*" [4].

Despite of the broad stream of research in each field, we still know little on how it would be possible to combine both approaches into a coherent methodology seamlessly, aiming at winning benefits from each. Therefore, in this paper, we propose a framework, explaining the way for introducing PL into STEM-driven CS education. This framework outlines basic activities relevant to PL in STEM and focuses on two essential attributes of PL, i.e., the content personalisation and learner's knowledge assessment and self-assessment. The aim is to re-design the former developed content for CS education using the STEM paradigm [5, 6], taking into account the requirements for the explicit personali-

* Accepted 19 April 2019.

sation of this content and the learner's knowledge and skills explicit assessment.

The contribution of this paper is *a generic structure of personalised learning objects* (LOs) in three categories: component-based LO, generative LO and smart LO (the latter is a combination of the first two [7]). The generic structure integrates those entities with the assessment modules and specifies the distributed interface for connecting them with digital libraries. The other contribution is the *learner's knowledge assessment tool* implementing the model that integrates attributes defined by the revised BLOOM taxonomy [8] and computational thinking skills [9, 10] with the adequate questionnaires or solving the exam tasks we have developed. The basis of our methodology is the recognition, extraction and explicit representation and then implementing of the *STEM-driven learning variability* in four dimensions, i.e., social, pedagogical, technological and content.

The structure of this paper is as follows. In Section 2, we analyse the related work. In Section 3, we formulate requirements for personalised STEM-driven CS education and two research questions. In Section 4, we present the basic idea of the approach and methodology applied. In Section 5, we outline the theoretical background to motivate the research questions, proposed solutions and outcomes. In Section 6, we present the proposed framework for introducing PL into STEM-driven CS education with the focus on the generic structure for representing personalised content and assessment facilities. In Section 7, we present and analyse two case studies and some outcomes from practice. In Section 8, we provide a summarized discussion and evaluation of this approach, also indicating on some drawbacks. Finally, in Section 9, we conclude on outcomes achieved and indicate on the future work.

2. Related work

We categorize related work into two streams: **A**. Analysis of personalised learning taxonomies and attributes with the focus on definitions and defining attributes of this learning process; **B**. STEM-driven CS education with the focus on content, its retrieval and personalised environments in general.

A. Despite of some hype in this field, the term 'personalised learning' is yet not well understood with multiple definitions proposed so far. In addition to what we have presented in Section 1, the Bill & Melinda Gates Foundation [11] states: "*Personalised learning seeks to accelerate learning by tailoring the instructional environment—what, when, how and where students learn—to address the individual needs, skills and interests of each student. Students can take ownership of their own learning, while also developing deep personal connections with each other, their parents and other adults.*" The White paper [12] concludes that (1) personalised learning (PL) is a conceptual subset of the student-centred learning and (2) indicates on four defining structural elements of the PL system. They include *competency-based learning*, *multiple paths* of study, *the use of variable time*, and *inclusion of meaningful assessment* and accountability. The report [12] highlights only three components that should form the core of PL structures: (i) Learner Profiles (they convey how a student learns best using customized learning environment); (ii) Customized Learning Paths based on individual interests, strengths and learning styles; (iii) Proficiency-Based Progress while advancement is tied to performance, not seat time or credit. At a 2010 symposium on PL, titled "Innovative to Educate: [Re] Design for Personalised learning", experts identified five essential elements central to PL: (i) Flexible, Anytime/Everywhere Learning; (ii) Redefine Teacher Role and Expand "Teacher"; (iii) Project-Based and Authentic Learning Opportunities; (iv) Student-Driven Learning Path; (v) Mastery— or Competency-based Progression/Pace.

Centre for Curriculum Redesign (CCR) [14] presents an extensive overview and focuses on a holistic approach, offering a complete framework across the four dimensions of an education, i.e., knowledge, skills, character, and meta-learning. According to the report, knowledge must reveal a better balance between traditional and modern subjects, including interdisciplinary subjects. Skills should relate to the use of knowledge and engage in a feedback loop with knowledge. Character qualities describe how one engages with and behaves in the world. Meta-learning fosters the process of self-reflection and learning how to learn, as well as the building of the other three dimensions. This report also delivers an extended list of technology used in PL. The paper [15] indicates five research areas to focus on advancing PL: (i) how educators and researchers use data; (ii) how technology is designed to support learners and associated pedagogical practice; (iii) how to educate personnel who are prepared to work in personalised settings; (iv) how content are designed and (v) how curriculum are designed to support PL. In addition, this paper argues that PL requires "a unique approach to the design, implementation, and assessment of learning". When implementing PL, "teachers become designers or engineers of learning". They can design environments that "meet the parameters of success for all learners, and when these environments fail, must work to identify, solve, and test solutions through an iterative design process".

A group of experts [16] proposes the "working definition" containing four attributes of PL: (1) *Competency-based progression*, meaning that each student advances and gains credit as soon as he/she demonstrates mastery through the continually provided assessment. (2) *Flexible learning environments*, meaning that the student needs to drive the learning environment, i.e., all operational elements respond and adapt to support students in achieving their goal. (3) *Personal learning paths*, meaning that each student follows a customized path that responds and adapts based on his/her individual learning progress, motivations, and goals. (4) *Learner profiles*, meaning that each student has an up-to-date record of his/her individual strengths, needs, motivations, and goals. The paper [17] addresses the problem of dynamically selecting a knowledge route that suits best to the individual learner's needs and profile through a set of learning resources. This paper engages parameters of the learner's cognitive style to create a multi-criteria utility model that evaluates available didactic methods, using initial evaluations upon a set of basic cognitive categories. The paper [18] examines seven critical dimensions of PL. They include (i) development of key skills, which are often domain-specific; (ii) levelling the educational playing field through guidance for the improvement of students' learning skills and motivation; (iii) encouragement of learning through "motivational scaffolding". The remaining ones include (iv) collaboration in knowledge-building; (v) development of new models of assessment; (vi) use of technology as a personal cognitive and social tool; (vii) the new role of teachers in better integration of education within the learning society. The paper [19] first observes that adapting a game to enhance its educational benefit endangers its intrinsic motivation and flow. Next, having this in mind, this paper proposes a novel approach for non-invasively adapting a game to enable a personalised learning experience. The paper [20] presents a different view on personalisation than what typically occurs in this field. The paper states "personalisation occurs when learning turns out to become personal in the learner's mind", meaning that there is the need of a special focus on confronting learners with tracked information.

Dockterman [1] presents insights to guide contemporary efforts in PL. He considers a brief overview of historical efforts to create a scaled system of education for all, also with the acknowledgement of individual learner variability. He proposes a concept of the *personalisation-based pedagogy*. The latter should focus on variability across multiple dimensions, not just domain knowledge and skill. In this regard, he states the following:

"*Instructional design and materials, informed by data and learning science, can focus on the anticipated variability within the target population that will matter most for learning and demonstrating competence with the academic goals*".

B. There are many challenges and issues in STEM-oriented learning. Among others, those include: (i) motivating and engaging students to participate in STEM-oriented learning [21, 22] and (ii) integrating STEM-oriented aspects in the school curriculum [23]; (iii) selecting adequate tools [24]; (iv) providing students' research to enforce computational thinking [25] (see also Chapter 1 in [6], for a more extensive analysis of challenges). W. Gander [26] observes that CS is the leading science of the twenty-first century. Similarly to mathematics, practically all sciences use CS approaches. According to the author, it has to be a part of general knowledge in education. One can find the use of the term "smart" very often in the scientific literature on STEM and CS education now. However, researchers try to assign a different meaning to this term, depending on the context. On this account, for example, Brusilovsky and his colleagues [27] state: "*Computer science educators are increasingly using interactive learning content to enrich and enhance the pedagogy of their courses. A plethora of such learning content, specifically designed for computer science education, such as visualization, simulation and web-based environments for learning programming, are now available for various courses. We call such content smart learning content*". In [5], we have defined smart GLO as an entity with the enhanced functionality, which implements the learning variability. Note that Boyle and his colleagues pioneered in this field, by introducing the GLO concept yet in 2004 [28]. Now there are other followers of this approach [29, 30]. Smart GLOs evolve over time [7] (for the evolution curve, see p.p. 138-140 in [6]). In this paper, we have yet enforced the GLO "smartness" by changing its structure and introducing the concept smart learning object. Regarding other characteristics of GLO, there is a generic attribute (we mean *computational thinking*) to characterize learner's ability in getting knowledge in CS or other disciplines. In this regard, J.M. Wing observes in [31] that (1) "*Computational thinking is a fundamental skill for everyone, not just for computer scientists*". (2) "*To reading, writing, and arithmetic, we should add computational thinking to every child's analytical ability*". (3) "*Computational thinking involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science*".

Computational thinking is about separations of concepts; it is about the use of abstractions and generalisations.

The paper [32] extends the understanding of the concept "computational thinking" to "AI thinking" and presents a Cloud-eLab education platform. It delivers a personalised content for each student with flexibility to repeat the experiments at their own pace which allow the learner to be in control of the whole learning process. The paper [33] considers social and technological aspects and emphasizes two important issues of personal learning environments (PLEs). The first is that PLEs "provide learners with their own spaces under their own control to develop and share their ideas". The second is that PLEs "are not an application but rather a new approach to the use of new technologies for learning". The research paper [34] considers principles for designing a personalised e-learning system, taking into account aspects of cultural backgrounds influences, i.e., differences among ethnic groups from Eastern and Western countries, on student learning approaches and learning styles. In doing that, the following needs to be considered: the issues of educational value differences, educational, cultural background differences, cultural communication differences, language usage differences and students' individual learning style preferences. The study [35] takes a broader view on PLEs and provides an extensive literature review aiming at creating a better understanding of PLEs and developing a knowledge base to inform further research and effective practice. The authors treat PLEs as "a concept related to the use of technology for learning, focusing on the appropriation of tools and resources by the learner". The paper [36] defines a PLE as a "structure and process that helps learners organize the influx of information, resources and interactions that they are faced with on a daily basis into a personalised learning space or experience". Using a PLE, the student is able to develop "an individualized digital identity" through the perceptual cues and cognitive affordances that the PLE provides. The paper [37] reviews the variety of efforts and approaches on how to establish a PLE, and suggests a categorization for them. The article [38] first clarifies key concepts and assumptions for personalised learning environments. Then it summarizes the authors' critique on the contemporary models for personalised adaptive learning. Subsequently, it proposes an alternative concept of a mash-up personal learning environment that provides adaptation mechanisms for learning environment construction and maintenance. The paper [39] discusses the paradigm shift towards personalised learning, from the educational, technological and standardisation perspectives. The paper [40] presents the motivation behind, the workflow supported by and the evaluation of the learning object Generator, a tool that offers personalised support and scaffolding for users. The paper emphasises that users are not necessarily content creation or pedagogical experts to provide assembling of pedagogically sound personalised Learning Objects. The paper [41] discusses an adaptive intelligent personalised learning environment. The findings of this research are the development of a model and intelligent algorithms for personalised learning. This model bases on the premise of an ideal system being one, which does not just consider the individual, but also considers groupings of likeminded individuals and their power to influence learner's choice.

An important conceptual and methodological step towards the integration and possible personalisation within different educational paradigms is the recognition that since 2015 CS is as a part of STEM in the U.S. education system [42]. This paper appeals for building the infrastructure for CS classes and providing steps towards pervasive computing education in order to reach the pervasiveness of mathematics and science education. The report [43] presents an extensive exploration and extended vision of CS education in the framework of the K-12 program in the U.S. This document also highlights a tight relationship between CS and STEM.

The next reviewed papers consider modern approaches to support personalized search, retrieval and delivery of personalized content that are important issues in general, for not only STEM or CS. The paper [44] introduces a multi-agent model based on learning styles and a word analysis technique to create a LO recommendation system to filter out unsuitable learning concepts from a given course. This model classifies learners into eight styles and implements compatible computational methods consisting of three recommendations (i) non-personalised, (ii) preferred feature-based, and (iii) neighbour-based collaborative filtering. Algorithm implementing the second recommendation has been experimentally proven as the best in terms of the preference error. The paper [45] proposes a hybrid recommendation method to assist user's personal needs in the search and selection processes of learning objects in Learning Objects Repositories. The proposed method uses a combination of different filtering techniques, such as content comparison, and collaborative and demographic searches. To achieve this goal, metadata information, management activities of resources and user profiles are used. The paper [46] presents an extensive review of recommender systems aiming at supporting the educational community by personalising the learning process. The review includes 82 systems developed over 2000–2014 years and classifies them into seven clusters according to their characteristics and for their contribution to the evolution of this research field. In the context of using Learning Management Systems (LMS), the paper [47] proposes a method and techniques on how LMS can deliver personalised material suited to the learner's learning requirements and learning style. The paper [48] surveys the existing approaches for the authoring and engineering of personalisation and adaptation in e-learning systems. This study enables to provide the comparison of various methods and techniques and facilitates their integration and reuse. In the context of personalisation of e-learning environments, the paper [49] discusses the learning preferences according to VAK model that classifies learners as visual, auditory or kinaesthetic. The findings of this experimental research show that learning styles as determined by self-assessment of using the VAK model do not necessarily improve performances. The other observation is that "working towards more flexibility and adaptability of the environment might be a better approach rather than to work on the adaptability of the environment".

In summary, there is a very broad and extremely intensive research work in both fields, i.e., personalized learning and STEM (SC) education now, though we were able to reveal that only partially in the given format. Both fields are highly heterogeneous in conceptual, pedagogical, social, technological, methodological and other aspects. Personalization concepts range, for example, from learner's needs [1, 16], different learner's profiles used [17, 18], to personalized content search, retrieval and delivery [45–48] and personalized environments [35–38]. STEM and CS education, on the other hand, are highly related fields [42], especially in terms of applying robotics [6]. Furthermore, there is some intersection or commonality between personalized and STEM (CS) approaches in terms of students' engagement and improving the quality of education [19–23]. Our overview is by no means comprehensive. Nevertheless, there is a large room for further research, especially in terms of integrative and quality ensuring aspects. This motivates our next steps well in presenting concepts we propose and consider throughout this paper.

3. Requirements for personalised STEM-driven CS learning and research questions

Personalised learning (PL) places the learner at the centre of the educational processes. As our literature review shows, the following attributes are essential to characterize the personalisation in learning [16]:

(1) The role of the teacher in providing a guidance of the *educational process is minimised*, and student, to some extent, is ready to accept and provide *self-initiation* and *self-guidance* of the process.
(2) There is the need to measure *the competency-based progression*, meaning that each student advances and gains credit as soon as he/she demonstrates mastery through the continually *provided assessment*.
(3) There are multiple choices in forming personal learning paths, meaning that each student follows a customized path that responds and adapts based on his/her individual learning progress, motivation, and goals.
(4) The learner is aware about *his/her profile*, meaning that each student has an up-to-date record of his/her individual strengths, abilities, level of previous knowledge, needs, motivation, and goals.
(5) There is a *flexible learning environment*, meaning that the student needs drive the learning environment, i.e., all operational elements respond and adapt to support students in achieving their goal.
(6) Technology stands for the essential factor predetermining the capabilities of PL.

These attributes are common and do not much depend on the learning paradigms used; however, the latter two attributes may be a highly specific, for example, in case of STEM-driven CS education, using robotics and other smart devices [6]. The listed attributes enable us to formulate requirements for personalised learning using the STEM-driven paradigm. The basic requirements (R) for PL follow.

R1. Personalised learning (PL) and environments in STEM-driven CS education should rely on the model-driven processes, meaning that a variety of models to support the functioning of those personalised processes and environments are applied.

R2. The learner's profile model (shortly learner's model) should be as generic as possible to enable extracting from it a concrete model for each student or a group of students.

R3. The learner's model as a part of the pedagogical-social model must have the highest priority with regard to the remaining models (technological and content) in PL.

R4. Since aforementioned attributes define the variability aspects in each category, though implicitly, we need to focus on the explicit representation of those variability aspects in order it would be possible to create a large space to form the personalised choices in selecting and managing personalised learning paths.

R5. The adequate personalised learners' assessment models and tools should be developed or selected. The assessment process has to be at least at two levels: (a) during learning process; (b) after completing this process (https://hundred.org/en/innovations/personalized-learning-paths).

R6. The PL and its environments should be designed so that it would be possible to provide, to analyse, to investigate, e.g., through applying the inquiry-based approach, or sometimes to change the personalised processes. For this purpose and due to the complexity of this problem, we state the need of considering the personalised learning system (i.e., its processes, functionality and structure) at three levels: *component level*, *sub-system level*, and *system level*.

R7. The personalised processes of the component level include the lowest-level tools only, i.e., personalised Component-Based (CB) Learning Objects (shortly PCB LOs), personalised Generative Learning Objects (PGLOs), and personalised Smart Learning Objects (PSLOs), each having a separate/individual extension for providing the learner's assessment procedure.

R8. The personalised processes of the sub-system level include additionally such sub-systems as personalised libraries.

R9. The personalised processes of the system level include all types of components and sub-systems within the smart learning environment [4], now extended with facilities for personalised learning.

R10. A specific focus has to be taken to implement multiple feedback links in dealing with PL.

R11. Ensuring the simplicity and flexibility in redesigning existing entities, for example, providing changes in the interface part, preserving the same functionality in the body.

Based on the formulated requirements and taking into account the continuous evolution of our approaches [5, 6], we consider the following research questions (RQs) in this paper.

RQ1: Transforming the structure of previously developed GLOs/SLOs into the personalised structure (PGLOs/PSLOs) enriched by the assessment modules.

RQ2: Developing a component-level framework for integrating PCB LOs, PGLOs and PSLOs into the personalised learning process.

4. Basic idea and methodology

The previously developed and discussed entities represent Component Based (CB) Learning Objects (LOs), Generative Learning Objects (GLOs) and Smart Learning Objects (SLOs) [6]. Roughly, one should understand those entities in this way here. CB LO is an instant of a content piece. It may represent a visual material (e.g., film for enhancing motivation), instructional text (e.g., fragment of theory, explaining robot's control program algorithm or physical characteristics of robots or other smart devices or their parts), etc. Typically, it is a fine-grained component retrieved from an external repository, though the teacher is able to modify, or even to create it from scratch if needed. GLO represents a set of the related instances (CS related teaching content, typically control programs for educational robots or microcontrollers) woven together with pedagogical—social and technological (e.g., robot characteristics) and content aspects in the same specification using heterogeneous meta-programming techniques [5]. Therefore, GLO along with a meta-language processor is a content generator or tool to produce instances on the learner's demand. SLO is a pre-specified set of both CB LOs and GLOs [7]. We have designed those entities so that they not only cover all topics of the curriculum completely, but also do that with a great degree of the surplus. For example, a topic may have a few CB LOs, GLOs or even SLOs. This is the result of the continuing evolution of the concept GLO [6, pp.138–140]. We have introduced SLO in the context of STEM-driven robot-based CS education aiming at extending the capabilities of inquiry-based learning in this paradigm.

We have designed those GLOs and SLOs previously without explicit requirements for personalisation and explicit self-assessment, though they included some social aspects integrated along with pedagogical aspects. In order to allow learners having a more flexibility in forming and managing their personalised learning paths, it was necessary to make three innovations. The first refers to the explicit separation of learner's profile and pedagogy from the remaining aspects, i.e., technology and content. That relates to the need of changing the interface within the GLO and SLO specifications without changing their functionality and developing of the unified interfacing structure for external storing of those content entities (CB LOs, GLOs and SLOs). The second innovation refers to external changes by introducing self-assessment capabilities for learners after the interpretation of those content entities. Finally, the third innovation is the introduction of multiple feedbacks in order make possible the measurement of a progress in the learner's skill development during the learning process. By introducing those innovations, we were able to create *a truly personalised learning entity* renaming it adequately as personalised CB LO (shortly PCB LO), personalised GLO (PGLO), and personalised SLO (PSLO).

We explain the idea on how it was possible to integrate introduced innovation into the process seamlessly in this way. We have developed a framework reflecting the processes, personalised learning paths, the use of personalised entities, assessment and self-assessment activities all together closed by multiple feedback links, as we will explain that in more detail later. The basis of our methodology in designing GLO/SLO is STEM-driven learning variability as a compound of pedagogical-social variability, technological variability, content variability and interaction variability among the enlisted variabilities [6]. This compound, in fact, is an anticipated variability predefined by modelling. We treat it as a *static variability*, having no explicit features with the run time learning processes. By re-designing GLO/SLO, we have extended and reconstructed this variability space in *the process dimension* through adding new features related to personalisation such as for self-assessment. In addition, we have extended the learning variability space by adding the personalised learning path variability. Therefore, after introducing innovations, we have within the learning space a *predefined static variability* and *dynamic variability* those result from our efforts at achieving aims of personalisation. With regard to the role of variability aspects in PL, reader needs to look at the papers in Sect. 2A once again, especially at the extract from [1].

To exploit the variability aspects as fully and efficiently as possible, we need first to use well-defined design principles. Those are the *separation of concepts* and *reusability*, followed by analysis and modelling. We have adopted them from software engineering and incorporated in our content design and re-design methodology. Analysis and modelling give us the possibility to recognize and extract variability in a variety of its kinds and then to represent it by adequate models explicitly. Finally, generative reuse brings the technology to implement the constituents of the framework as effectively as possible. A more extensive description of the methodology we use, one can find in [50, 51]. Note that it is the same for the assessment modules development. Largely, this methodology is for researchers, content designers and, of course, for smart CS teachers who are able to play a role of the content designers. In the next section, we present a background of our approach.

5. Theoretical background

We describe the background by providing definitions of the basic terms, their properties, and relevant models to specify adequate entities more precisely as follows.

Definition 1. Personalised learning (PL) is the approach that places the learner's needs at the centre of learning, uses *personalised learning objects* and exploits through learning activities the *attributes* that enforce the *personalisation* (e.g., self-guidance, self-assessment, use of personalised learning paths, etc.) as much as possible (revised from [16]).

Property 1. In case of the personalised STEM-driven CS education, there are three types of the personalised learning objects, i.e., Personalised Component-Based Learning Objects (shortly PCB LOs), Personalised Generative Learning Objects (PGLOs) and Personalised Smart Learning Objects (PSLOs).

Definition 2. In terms of the IEEE definition of LO [52], Component-Based Learning Object (CB LO) is an instance, typically digital, either retrieved from external sources, or modified/created by the user.

Definition 3. PCB LO is the structure consisting of two entities, i.e., the CB LO and the module that provides the learner's knowledge and skills assessment and measurement of progress in learning, using this CB LO (revised from [52]). Simply, PCB LO = CB LO + LKSAM (where LKSAM – learner's knowledge and skills assessment module).

Definition 4. GLO is the specification that implements the *pre-defined learning variability aspects* (i.e., pedagogical, social, technological and content), using heterogeneous meta-programming techniques [53]. This specification consists of two interrelated parts, i.e., *meta-interface* and *meta-body*. The first stands for delivering parameters and their values. The second stands for implementing the functionality, i.e., learning variability aspects woven together specifically.

Property 2. GLO is a domain-specific *heterogeneous meta-program*; in other words, the latter, along with the meta-language processor, is the *learning content generator* on the user's demand. The user (teacher or learner) operates with the meta-interface, seeing parameters and their values (that represents the variability space for choice). The meta-body is completely hidden from the user. The system operates with it only.

Property 3. GLOs evolve over time in terms of their types, numbers, structure or even functionality (see evolution curve in [6, pp. 138–140], for more details). The earlier developed GLOs have the *integrated meta-interface*. The currently developed GLOs are PGLOs, having the *distributed meta-interface*, i.e., social (learner's) and pedagogical parameters are placed within the metadata in the local library, while the remaining parameters, i.e., technological and content, are within the specification itself being stored in the external repository.

Definition 5. The heterogeneous meta-program is the specification that is implemented using at least two languages, i.e., meta-language and target language (languages). The latter stands for delivering the base functionality through a domain (target) program, for example, a robot control program in our case. The first stands for expressing the generalization, i.e., learning variability aspects [6, p. 116].

Property 4. It is possible to use any programming language as a meta-language in *the mode of the structured programming* [54]. One can find an approval of this property in [53]. Therefore, designers (e.g., CS teachers) have a broad possibility in creating GLOs/SLOs.

Example 1. For a long time, we use PHP as a meta-language and C, RobotC, SQL as target languages.

Property 5. The following expression (1) estimates the possible number of instances N that the user, using a meta-language processor, can derive from the given GLO (PGLO) specification:

$$N \le |P_1| \times |P_2| \times \ldots \times |P_i| \times \ldots \times |P_n|, \quad (1)$$

where $|P_i|$ – the number of values of the parameter P_i, n – the total number of parameters.

Note that the equality sign (=) in the expression (1) holds when all parameters are independent, i.e., not interacting [6].

Example 2. Let us have a GLO (PGLO) with six independent parameters, each having five values. In this case, the variability space of possible instances N =5*5*5*5*5*5= 3125.

Property 6. The variability space predefined by (1) also specifies the possibility for the learner's choices in forming personalised learning paths while using a single PGLO. As, in personalised learning, the learner uses some set of PGLOs even during a short session such as the lesson time, there is indeed a huge space for making choices to form personalised learning paths.

Note that, typically, the size (in symbols) of a single GLO (PGLO) is the only 3-5 times larger than the instance derived from this specification. Considering this observation and Property 3, the following property holds.

Property 7. GLO (1) *contributes* to saving the space within a digital library, (2) *substitutes* the instance search procedure by the *generation* procedure; (3) the latter *brings* the LO instance you intend to receive, i.e., resolves the problem of synonyms.

Definition 6. PGLO is the structure of two entities, i.e., the GLO and the module that provides the learner's knowledge and skills assessment and progress in learning, using this GLO. Simply, PGLO = GLO + LKSAM (where LKSAM – learner's knowledge and skills assessment module).

Definition 7. Pre-designed SLO is a set consisting of two subsets taken from the available sources: (1) pre-specified subset of CB LOs (PCB LOs) and (2) pre-specified subset of GLOs (PGLOs) (revised from [7]) and additionally containing the cumulative assessment module CAM. Formally,

$$PSLO_{pre-designed} = \{PCB_1, \ldots, PCB_m\} \cup \{PGLO_1, \ldots, PGLO_k\} \cup CAM,$$

where $\{PCB_1, \ldots, PCB_m\}$ – the m-size pre-specified subset of PCB LOs, $\{PGLO_1, \ldots, PGLO_k\}$ – the k-size pre-specified subset of PGLOs

Definition 8. PSLO is the structure consisting of two entities, i.e., the SLO and the module that provides the learner's knowledge and skills assessment and progress in learning, using this SLO. Simply, PSLO = SLO + LKSAM (where LKSAM—learner's knowledge and skills assessment module).

Property 8. Structure and functionality of the special part of LKSAM are different for each category of the personalised learning entities (we will explain that in more details later).

Definition 9. Learner's knowledge and skills assessment model (shortly assessment model) is the structure consisting of generic and specific parts. The generic part represents and integrates concepts taken from the revised BLOOM taxonomy [8] with the ones that characterize computational thinking skill [9, 10]. The specific part represents a set of tasks related to the personalised content components (LOs).

Definition 10. Feature-based assessment model is the feature diagram that consists of the following elements: (i) features; (ii) parent-child relationships of features and (iii) constrains (Fig. 1).

Property 9. The learner's knowledge and skills assessment module LKSAM is the implementation of the feature-based assessment model, for example, using meta-programming techniques and the methodology [51, 53].

Definition 11. Cumulative assessment is the process that involves a continuous monitoring of the student progress and is composed of partial assessments retrieved from LKSAMs.

Definition 12. Learning variability is the characteristic of personalised learning that covers the following attributes: *social variability* (learner's variation in the profiles, demands, etc.), *pedagogical variability* (STEM pedagogy, including inquiry-based approaches, etc.), and *technological variability* (variation in technology types, characteristics, e.g., software-based, hardware-based, such as robots and smart devices) and *content variability* (data, algorithms, programs) (revised from [5], p. 106).

Property 10. Social variability and pedagogical variability are the attributes of a higher priority with respect to the remaining variability types. Therefore, the social variability and pedagogical varia-

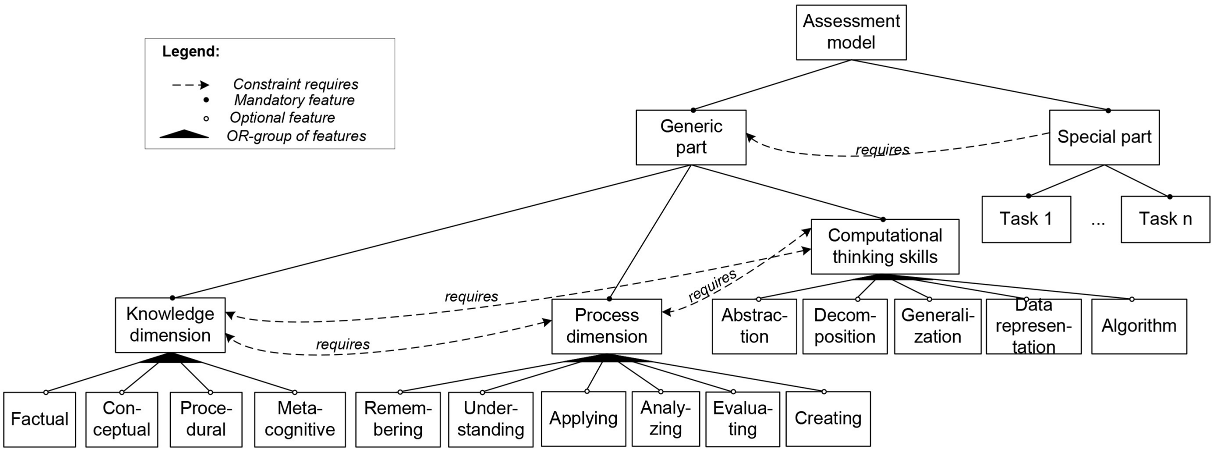


Fig. 1. Feature-based learner's knowledge assessment model based on using attributes from [55].

bility are the context to the remaining variability types.

Property 11. Personalised learning path is the expression of possibilities in making a choice between the processes and activities predefined by the educational environment that implements and supports the learning variability in order to achieve the learning objectives.

6. A framework to implement personalised STEM-driven CS education

In Fig. 2, we present a framework that schematically outlines the way for implementing personalised learning using personalised LOs. The framework includes the following constituents: actors, learning task and plan, learning activities and processes, resources used, learning paths with feedbacks, learner's assessment and learner's progress measurement. A rectangular given by fat lines represents learning activities and processes while rectangular with thin lines represents resources. As in personalised learning, there is a large differentiation in the learner's profile, the teacher's role changes significantly. It moves from the knowledge provider towards the personalised content creator along with learning activities in helping students as a facilitator, moderator and so on. At the very begin-

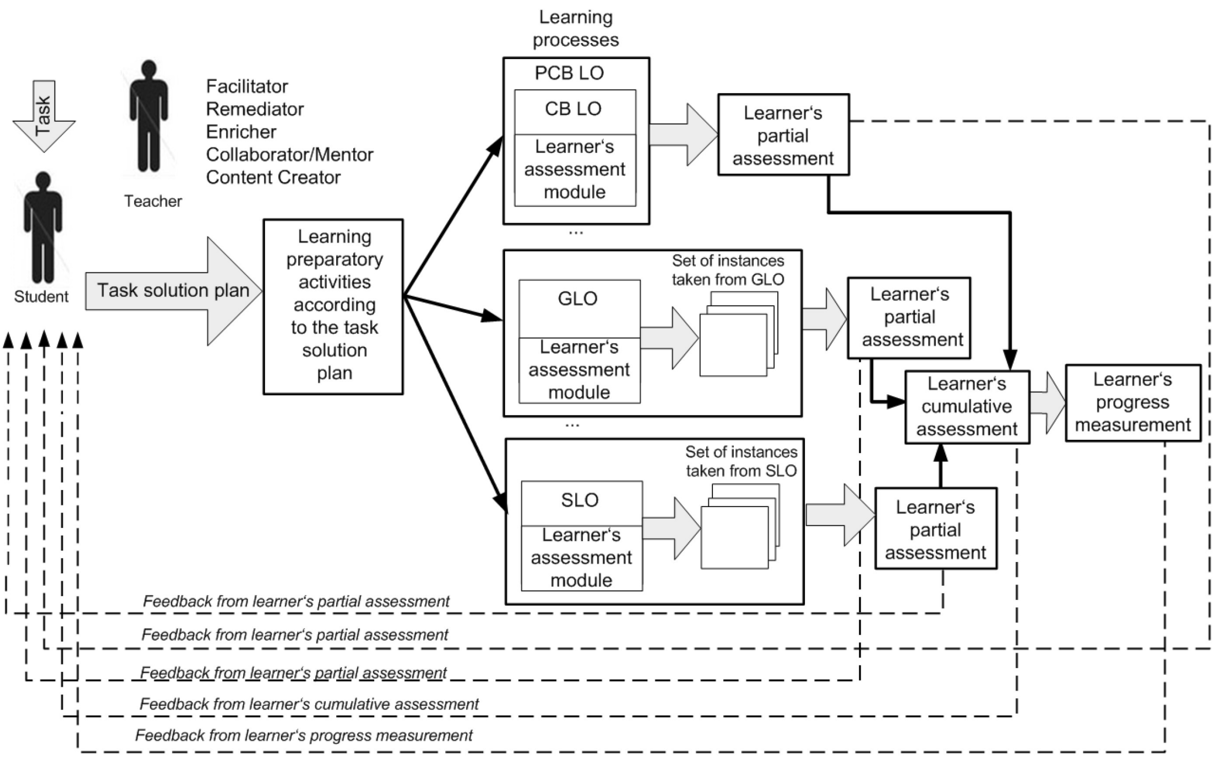


Fig. 2. A schematic view of the proposed framework outlining activities, processes and feedbacks.

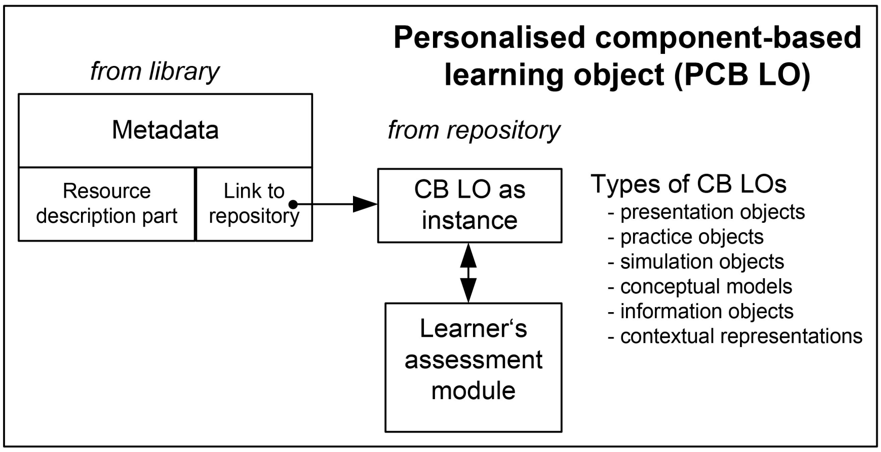


Fig. 3. Structure of personalised component-based LOs.

ning, a learner having a task and the plan for its interpretation needs to recognize the category he/she belongs (beginner, intermediate, advanced) for selecting the adequate learning path.

First, we outline the personalised content in sub-section 6.1. Then we will describe processes and activities through the personalised learning paths in sub-section 6.2.

6.1 Structural models of personalised learning objects

With regard to the interfacing capabilities, all types of personalised LOs have the unified structure we identify as a generic one here. The *generic structure* includes metadata part and connection with the learner's knowledge and skills assessment module. However, the metadata part and special part of assessment modules are different for each type of entities (cp. Figs. 3, 4(b) and 5). The commonality of this structure is that they all have to be stored in the same way for retrieving at the use time. In Fig. 3, we present the structure of PCB LOs. This structure corresponds to *Definition 2* (see Section 5).

In general, those components taken from external repositories may include the following types: presentation objects, practice objects, simulation objects, conceptual models, information objects and contextual representation [56]. In STEM-driven CS education, they include those that are specific for robotics (such as electrical circuits, guides for mechanical designing of robots, etc.) and, additionally, learners may create own LOs through his/her personalised learning paths and processes and collect them in personalised libraries for the future use and sharing.

In Fig. 4(a), we present the structure of the former GLO more extensively researched in [5, 6]. Its specification has the *integrated interface* and meta-body (see *Definition 4* and Properties 2 and 3 in Section 5). The meta-body describes the implementation of functionality aspects. The meta-body hides them from the user (learner or teacher). Therefore, it is darkened (see also Property 2 in Section 5).

In Fig. 4(b), we present the structure of the personalised GLO with the distributed interface. Note that, in developing phase of those structures,

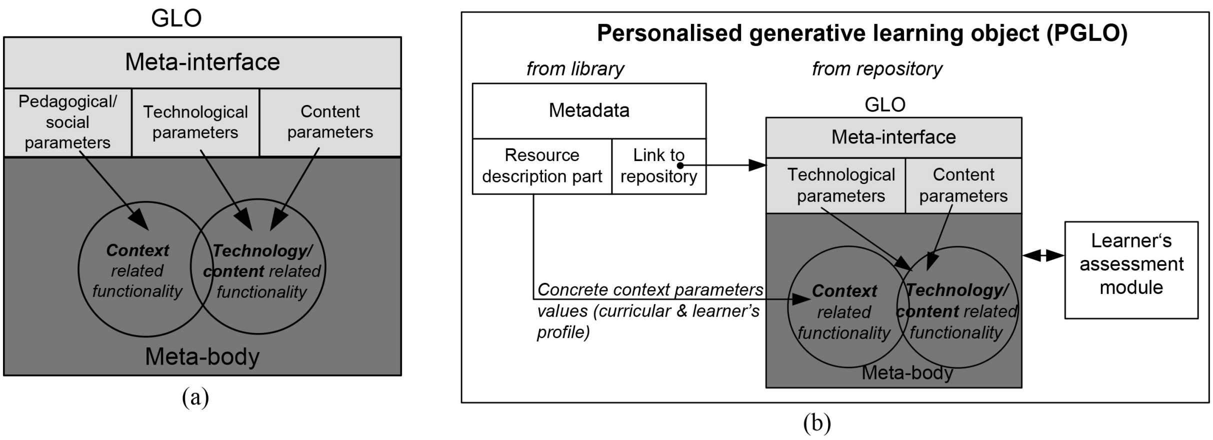


Fig. 4. Structure of GLO with the integrated interface (a) and PGLO with distributed interface (b).

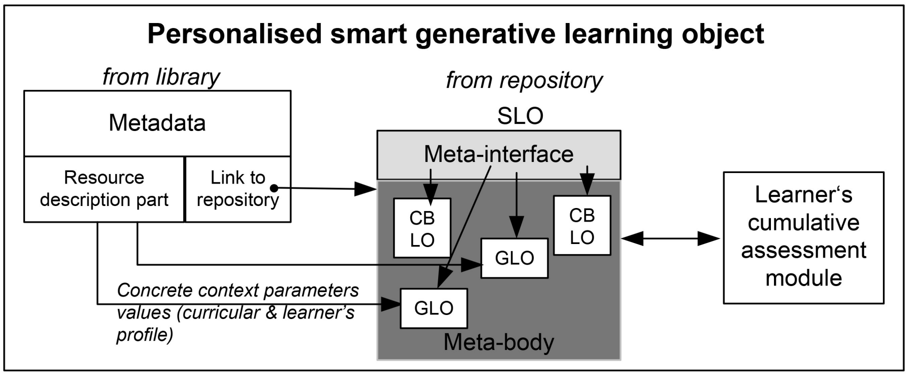


Fig. 5. Structure of personalised smart LO.

we widely exploited design principles formulated in Section 4. Indeed, one can see the separation of the following concepts: meta-interface from the meta-body, integrated interface from distributed interface, context parameters from remaining parameters and the local library from external repositories. On using reuse-based aspects and principles in designing GLOs, one can learn more from [50, 51].

In Fig. 5, we present the structure of the personalised SLO. As specified by *Definition 7*, PSLO consists of a pre-designed set of CB LOs and GLOs. Why we need to pre-design those sets? The main reason is the flexibility in making choices to form a *variety of personalised learning paths* for different students. The teacher prepares those sets in advance. The size of the set, the number of entities of the different types (CB LO or GLO) within the set depends on a topic and learning objectives.

In STEM-driven CS education, CB LOs represent the content that is used to enhance motivation, different tutorials and user guides, including hardware-oriented ones, such as electrical circuits, robot designing instructions, etc. GLOs represent generic control programs for robots and micro-controllers.

6.2 Personalised processes and activities within the framework

In describing processes and activities, we consciously omit the STEM learning semantics, because we have discussed that more extensively in [6]. The personalised learning paths predefine PL. The process starts when the student receives the task given by the teacher. Then he/she develops the task solution plan. For that, the teacher's role may be different: (a) teacher provides an extensive support, e.g., for the beginner; (b) teacher provides a little support, e.g., for the intermediate student; (c) there is no help at all, e.g., for the advanced student. The teacher's help may include explaining the task and clarifying learning objectives and providing the student with keywords as a part of metadata for retrieving the personalized content. Having this plan, the learner is able to develop the learning scenario with little to no teacher interference. In Fig. 2, we indicate on that as preparatory activities. There are three constituent parts of personalized learning paths for each student: (1) *before learning*; it includes planning and retrieving the needed resources; (2) *during learning*; it includes activities and processes with the personalized content; (3) *after learning*; it covers assessment activities combined with multiple feedback links. Due to the structure of the personalized LOs, due to the multiple assessment possibilities, there is a huge space to make choices in forming personalized learning paths. In addition, a learner has the possibility to repeat the path or to modify it and the process itself through feedbacks based on the assessment and self-assessment. The learner drives those actions, fully or partially, taking ownership of his/her own learning. How that works in practice, we will show in Section 7. In sub-section 6.3, we outline software tools to support the implementation of the proposed framework.

6.3 Software tools to implement the proposed framework

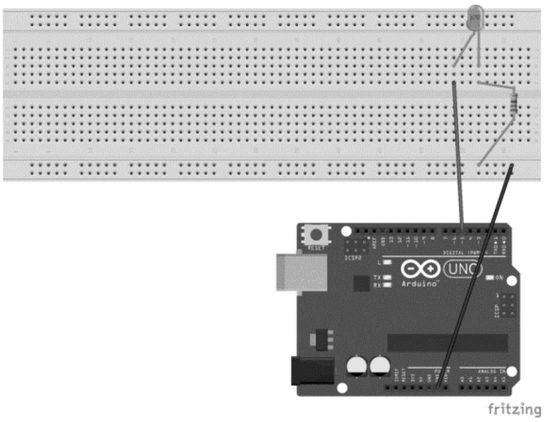
For creating personalized content, we use the following tools. For the content (PGLO, PSLO) specification at a high level of abstraction using feature models, we use SPLOT (Software Product Lines Online Tools, www.splot-research.org). For the generative content implementation, we apply heterogeneous meta-programming approaches [6, 53]. For component-based parts (CB LOs), we apply a general purpose SW or/and domain-specific SW (Fritzing for circuits modelling, fritzing.org; LEGO Digital Designer for LECO robots' modelling, https://www.lego.com/en-us/ldd). In the use mode of the proposed approach, the general-purpose SW include Browsers, PDF readers, text Editors, Spreadsheets, different video-audio players, etc. Regarding domain-specific SW, we use environments for robot programming (ArduinoC, RobotC, Java). For assessment purposes, we use online quiz makers (Socrative, https://socrative.com; Google Forms).

7. Case studies

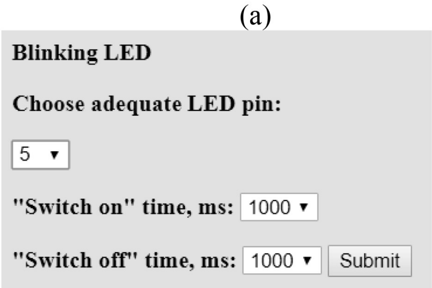
We present two case studies to demonstrate the gaining STEM-driven CS knowledge and skills to program LEDs functionality, using ARDUINO microcontrollers. Using the developed PSLOs, learners improve knowledge and skills in Science (Physics, e.g., electricity; Computer Science, e.g., programming), Technology (microcontrollers), Engineering (Electrical Engineering) and some aspects of Math (e.g., representing decimal numbers in the binary numeral system). From the viewpoint of personalisation, the presented PSLOs are extended by adding learner's assessment modules aiming at monitoring the student's progress and ensuring effective feedback. By introducing two case studies, we aim at showing a variety of personalised learning paths and different possibilities to make assessment. In Case Study 1, for example, the special part of the learner's assessment module presents a set of the multiple-choice questions; while in Case Study 2, we present the special part by practical tasks. Note that in those case studies, learner views the lower part of the PGLO interface only, for selecting the values of technological and content parameters.

7.1 Personalised SLO "Blinking LED"

The aim of the PSLO "Blinking LED" is to explain the programming principles of the LED and to demonstrate principles of procedural programming. PSLO "Blinking LED" consists of CB LO (see Fig. 6(a)) that represents an electrical circuit or physical part of PSLO, and GLO (see Fig. 6(b)) that represents a generic control program of LED. First, the learner constructs the electrical circuit, then generates the LED control program (see Fig. 6(c)). The learner can make changes in the electrical circuit and choose adequate values of parameters in the meta-interface of GLO working at his/her own pace. After completing the planned tasks, the student answers the questions (see Fig. 7(b), (c)) that determine the student's achievements, according to the assessment model generic part (see Fig. 7(a)). Each question covers some aspects of knowledge, cognitive processes and computational thinking skills dimensions. For example, the question presented in Fig. 7(c) covers aspects of factual and conceptual knowledge, involves cognitive processes, such as understanding, applying and analysing and includes computational thinking skills, such



(a)



(b)

```
//Blinking LED

int ledPin = 5;

void setup()
{
   pinMode(ledPin, OUTPUT);
}

void loop()
{
   digitalWrite(ledPin, HIGH);
   delay(1000);
   digitalWrite(ledPin, LOW);
   delay(1000);
}
```

(c)

Fig. 6. Blinking LED: (a)—physical part of SLO (CB LO), (b)—meta-interface of generalized control program (GLO), (c)—generated instance of the LED control program.

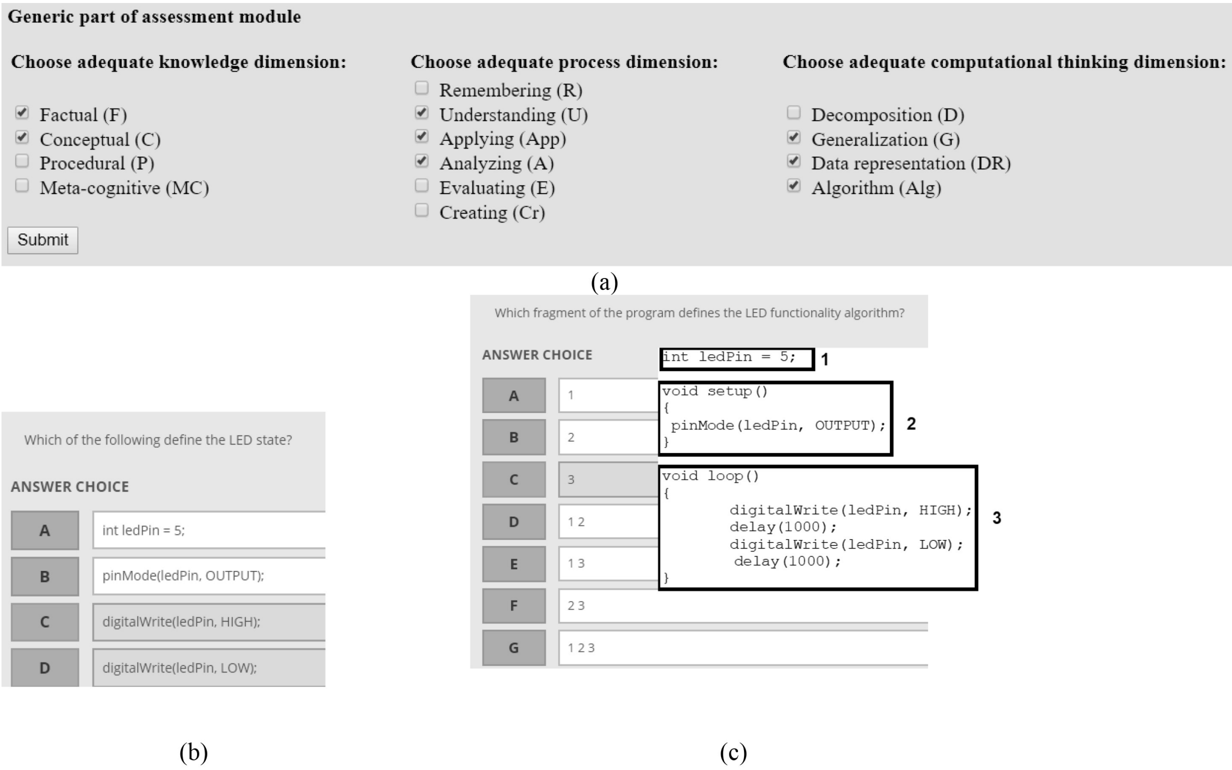
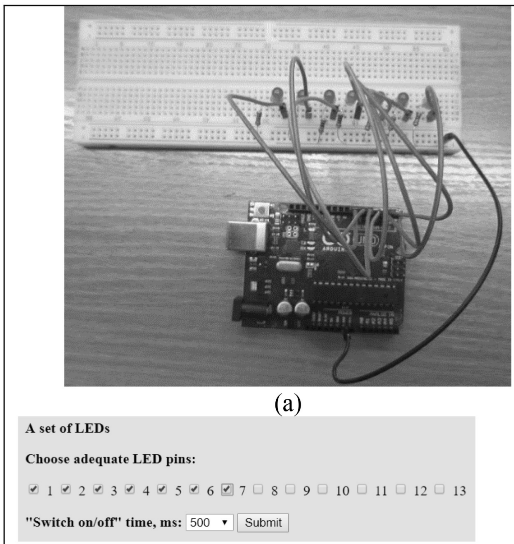


Fig. 7. Learner's assessment module related to SLO "Blinking LED": (a)—generic part of the assessment module (see also Fig. 1); (b) & (c) – items of special part of the assessment module (see also Fig. 1): (b)—question covering F, C; U, App, A; G, DR; (c)—question covering F, C; U, App, A; G, DR, Alg.

as generalisation, data representation and algorithm.

7.2 *Personalised SLO "A set of LEDs"*

The aim of the Case Study 2 "A set of LEDs" is to explain the concept of the array and to acquire appropriate skills in applying loops and functions. PSLO "A set of LEDs" consists of CB LO (see Fig. 8(a)) that represents a physical part of PSLO, and GLO (see Fig. 8(b)) that represents a generic control program of LEDs. First, the learner constructs the electrical circuit, then generates LEDs control pro-



(b)

```
//A set of LEDs

int ledPins[] = {1,2,3,4,5,6,7};
//--------------------------------------------
void setup()
{
  for(int i = 0; i < 7; i++)
  {
     pinMode(ledPins[i],OUTPUT);
  }
}
//--------------------------------------------
void loop()
{
   int delayTime = 500;
   for (int i = 0; i < 7; i++)
     digitalWrite(ledPins[i], HIGH);
   delay(delayTime);
   for (int i = 6; i >= 0; i--)
     digitalWrite(ledPins[i], LOW);
   delay(delayTime);
}
```

(c)

Fig. 8. A set of LEDs: (a)—physical part of SLO (CB LO), (b)—meta-interface of generalized control program (GLO), (c)—generated instance of the LEDs control program.

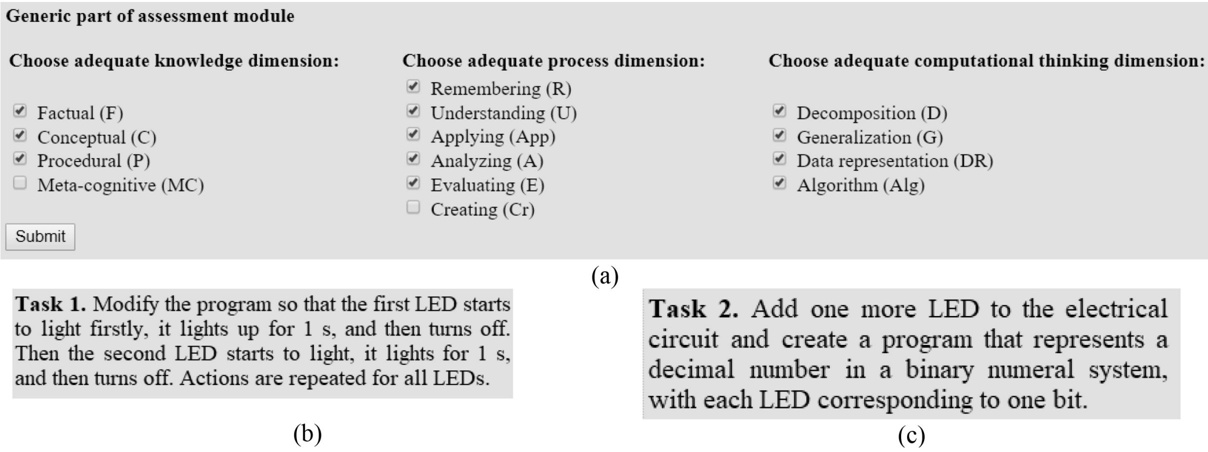


Fig. 9. Learner's assessment module related to SLO "A set of LEDs": a—generic part of assessment module (see also Fig. 1); b & c—items of special part of assessment module (see also Fig. 1): (b)—practical task covering F, C; U, App, A; G, DR, Alg; (c)—practical task covering F, C, P; R, U, App, A, E; D, G, DR, Alg.

gram (see Fig. 8(c)). The learner should make changes in the electrical circuit and choose adequate values of parameters in the meta-interface of GLO working at his/her own pace. After completing the planned tasks, the student performs additional practical tasks (see Fig. 9(b), (c)) that check the student's progress, according to the assessment model generic part (see Fig. 9(a)). Each practical task covers some aspects of knowledge, cognitive processes and computational thinking skills dimensions. For example, the task presented in Fig. 9(c) covers aspects of factual, conceptual and procedural knowledge, involves cognitive processes, such as remembering, understanding, applying, analysing and evaluating and includes computational thinking skills, such as decomposition, generalisation, data representation and algorithm.

8. Discussion

The personalised learning and STEM (STEAM) stand for the central topics among educational strategists, researchers and practitioners now. That is so because of the urgent needs to improve education at all levels, responding to the existing and emerging new economic and social challenges in the 21st century. In this paper, we have discussed the approach on how to combine both paradigms in case of STEM-driven CS education using robotics and microcontrollers at the high school. We have focused on the content personalisation and learner's knowledge and skills assessment problems here. Our aim was *to make the personalisation aspects explicit* of the *integrated content* researched in its variety of forms extensively in [5, 6]. By the integrated content, we mean *generative and smart learning objects* that integrate the social, pedagogical, technological and content variability aspects in the same specification using meta-programming techniques. We have formulated requirements, outlined the methodology and theoretical background used. On this basis, we have developed the framework explaining on how it was possible to deal with and solve the personalisation problems (research questions) in this case. Two aspects are at the centre of this proposal. The first is the personalising of the so-called Component-Based Learning Objects (CB LOs), Generative Learning Objects (GLOs) and Smart LOs (SLOs). The second is the extended facilities of the learner's knowledge and skills assessment through partial, cumulative assessment and progress measurement combined with multiple feedbacks. To build this, we have proposed the learner's knowledge assessment model that integrates two attributes taken from known approaches (revised BLOOM taxonomy and computational thinking skills) along with the adequate tasks. We have implemented the model by developing the adequate tool for assessment the learner's knowledge. Therefore, we can summarize the contribution of this paper as follows. (1) The proposed personalised GLO has the improved structure, i.e., distributed interface (meta-interface) as compared to integrated interface in the structure of the former GLO (compare Figs. 4(a) and (b)). (2) The personalised SLO enables to enlarge the space for learners to choose the personalised learning paths significantly. (3) There are possibilities for measuring the learner's progress through multiple assessments. (4) We have introduced a generic (common) structure for all types of personalised LOs as well as combined them with assessment facilities. The basis of our methodology is the concept of learning variability covering the pedagogical, social (with the focus on learners), technological and content aspects. The personalisation attributes we have taken into

Table 1. How our approach supports trends of engineering education

Trends of Engineering Education (EE) as defined in [57]	**Explanation on how our approach supports these trends**
Delivery of authentic, active learning to large student cohorts	The proposed framework is adaptable to students with different levels of knowledge, because of: (1) the developed learning content can be personalised according to the student's preferences; (2) learning resources support learning-by-doing, inquiry-based, problem solving learning approaches that ensure active learning; (3) the technological and content tools are applicable to the large student number. This framework is implemented at the high school and 135 students use content and technological resources during CS lessons as well as in informal learning. We tested our resources in workshop (57 participants). The students and participants of workshop rated the content and activities as very interesting and engaging.
Increase in flexibility, choice and diversification	The variety of content and of technological resources enable students to choose topics and activities that fit their learning styles and preferences, and to create their own learning paths.
A multi-disciplinary, global and societally focused approach	The topics composed of created resources cover many interrelated fields, such as CS, mechanical and electrical engineering, mathematical modelling. The topics enable to create the preconditions for solving complex problems through modelling, designing and programming prototypes of autonomous vehicles, industrial and domestic robots, etc.
Integrating and embedding key learning experiences	The proposed approach creates conditions for self-directed, contextualized learning, provide appropriate learning opportunities, support learning transfer to workplace, create the preconditions for effective learning, develop teamwork skills.

account extend the space and possibilities in forming personalised learning paths significantly.

We use robotics as technology and tools for practice in STEM-driven CS education. Robotics enables to provide the interdisciplinary knowledge. In a broader context, among other aspects, our approach introduces the E-knowledge and T-knowledge by applying physical components such as sensors, microcontrollers, educational robots [7]. Learning activities related to using the learning-by-doing and inquiry-based models cover multiple learners' actions such constructing and investigating characteristics of those components. Those activities require a great deal of modelling and applying other approaches, for example, specific software tools. The use of real-world tasks or their prototypes, on the other hand, strengthens the learners' engagement into their activities. By solving those tasks, it is possible to introduce knowledge from physics (e.g., to investigate properties of light while dealing with light sensors, etc.), mechanical and electrical engineering (e.g., construction a robot from mechanical parts and designing electrical circuits, estimating the voltage level to ensure the needed speed of robots' motors, etc.). What is common for a more complex real wold tasks is that physical components require *control programs* to ensure the functionality of those components. Learning on how to construct and use those programs is not only a matter of CS teaching and learning, but of the most engineering disciplines too. Therefore, we can state that our approach contributes to engineering education as well. However, in our approach, as it was presented in this paper, STEM-driven CS education stands at the centre while robotics and engineering aspects form a context to this approach. In terms of possibilities to apply our approach in engineering education, we have evaluated the approach with regard to the future directions of engineering education [57] (see Table 1).

This approach, to some extent, is applicable to other teaching disciplines and other schools too. There are a few possibilities to do that. Firstly, the developed component-based, and generative smart learning objects could be used in the mode use-as-is, for example, for further modification of generated instances for learning to create robot control programs. Secondly, it is possible to apply those entities in physical and engineering experiments. Next, the assessment model can be adapted in the context of other subjects and schools.

The restriction of personalised education, in terms of how we presented it here, is that it requires additional collaborative efforts and considering of a broader educational context. That is so because of the interdisciplinary nature of the STEM paradigm, the diversity of its own possibilities and those introduced through personalisation, diversity of skills and results achieved by different learners through their personalised learning paths. There is a need to compose the solution of the real-world task consisting of sub-tasks defined by fulfilling the separate personalised learning paths. All these factors result in the need of sharing the gained knowledge, providing discussions with others, including teachers, parents and other stakeholders. At the current state of this research, we have had a restricted possibility to provide a more extensive researching on assessment issues in terms of learning progress measurements.

Note that the extended personalised smart learning objects will be introduced as a new type of learning resources into the curricula "Programming Basics" at the high school soon.

9. Conclusion and future work

A deep and wide feedback initiated and controlled by learner drives the personalised learning process and contributes to forming a variety of personalised learning paths. The introduced generic structure of the personalised learning objects (LOs) enables a flexible integration of those entities into personalised libraries or repositories and supports providing the managing procedures flexibly. The personalised smart LO, in fact, is a mini scenario to support forming personalised learning paths by the learner and, in this way, driving the learning process. Personalisation of STEM-driven CS education nominates a new way for developing computational thinking skills in the process of gaining the interdisciplinary knowledge; it contributes in achieving a faster and deeper knowledge for decision-making skills and measuring progress through multiple assessments. This approach, by no means, does not exclude the teacher from the education process. Rather, it enforces the need of teacher's participation, however, with a changed role. In this case, the teacher becomes an initiator and moderator of the process; he/she stands for a provider of the needed support and as a designer of the personalised content. The discussed approach, to some extent, also contributes to engineering education. It is so because of the interdisciplinary nature of STEM and CS as a leading science to bring the fundamental knowledge so needed in the modern age.

The future work will include connecting this framework with personalised libraries and extending those ideas to the complete personalised environment.

References

1. D. Dockterman, Insights from 200+ years of personalised learning, *NPJ Science of Learning*, **3**(15), pp. 1–6, 2018.
2. The Future of Jobs Report 2018, World Economic Forum, Centre for the New Economy and Society, http://www3.weforum.org/docs/WEF_Future_of_Jobs_2018.pdf, Accessed 21 November 2018.
3. Reimagining the Role of Technology in Education: 2017 National Education Technology Plan Update, U.S. Department of Education, http://tech.ed.gov, Accessed 6 February 2018.
4. Southwest Regional STEM Network, Southwest Pennsylvania STEM network long-range plan (2009–2018): plan summary. Southwest Regional STEM Network, Pittsburgh, p. 15, 2009.
5. V. Štuikys, *Smart Learning Objects for Smart Computer Science Education: Theory, Methodology and Robot-based Implementation*, Springer, 2015.
6. V. Štuikys and R. Burbaitė, *Smart STEM-Driven Computer Science Education: Theory, Methodology and Robot-based Practices*, Springer, 2018.
7. V. Štuikys, R. Burbaitė, T. Blažauskas, D. Barisas and M. Binkis, Model for introducing STEM into high school computer science education, *International Journal of Engineering Education*, **33**(5), pp. 1684–1698, 2017.
8. L. W. Anderson, D. R. Krathwohl, P. Airasian, K. Cruikshank, R. Mayer, P. Pintrich, J. Raths and M. C. Wittrock, A taxonomy for learning, teaching and assessing: A revision of Bloom's taxonomy, New York, Longman Publishing, 2001.
9. K. Cummins, Teaching Digital Technologies & STEM: Computational Thinking, coding and robotics in the classroom, Amazon.com, Accessed 22 January 2019.
10. S. Atmatzidou and S. Demetriadis, Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences, *Robotics and Autonomous Systems*, **75**, pp. 661–670, 2016.
11. J. F. Pane, E. D. Steiner, M. D. Baird, L. S. Hamilton and J. D. Pane, Informing Progress: Insights on Personalised Learning Implementation and Effects, 2017. https://www.rand.org/content/dam/rand/pubs/research_reports/RR2000/RR, Accessed 23 November 2018.
12. A Look to the Future: Personalized Learning in Connecticut, White Paper on Personalized Learning, https://www.capss.org/uploaded/2014_Redesign/Educational_Transformation/CAPSS_Whitepaper_FINAL_12-23-14_copy_2.pdf, Accessed 23 November 2018.
13. Design Principles for Personalised Learning Environments, The Institute at CESA #1, Cooperative Educational Service Agency #1. http://www.wasb.org/websites/convention/File/2013, Accessed 20 December 2018.
14. J. S. Groff, Personalised Learning: The State of the Field & Future Directions, Center for Curriculum Redesign, 2017, www.curriculumredesign.org, Accessed 23 November 2018.
15. J. D. Basham, T. E. Hall, R. A. Carter, M. William and W.M. Stahl, An Operationalized Understanding of Personalised Learning, *Journal of Special Education Technology*, **31**(3), pp. 126–136, 2016.
16. Personalised Learning: A Working Definition (October 22, 2014, Education Week), https://www.edweek.org/ew/collections/personalised-learning/, Accessed 5 February 2019.
17. N. Manouselis and D. Sampson, Dynamic knowledge route selection for personalised learning environments using multiple criteria, *Applied Informatics Proceedings*, No. 1, pp. 448–453, 2002.
18. S. Järvelä, Personalised learning? New insights into fostering learning capacity. *ocde-ceri (eds.), Personalising Education*, París: ocde/ceri, pp. 31–46, 2006.
19. N. Peirce, O. Conlan and V. Wade, Adaptive educational games: Providing non-invasive personalised learning experiences, *Digital Games and Intelligent Toys Based Education*, Second IEEE International Conference, pp. 28–35, 2008.
20. D. Verpoorten, C. Glahn, M. Kravcik, S. Ternier and M. Specht, Personalisation of learning in virtual learning environments, *European Conference on Technology Enhanced Learning*, Springer, Berlin, Heidelberg, 2009.
21. J. Ardies, S. De Maeyer, D. Gijbels and H. van Keulen, Students attitudes towards technology, *International Journal Technology Design Education*, **25**(1), pp. 43–65, 2015.
22. C. Kim, D. Kim, J. Yuan, R. B. Hill, P. Doshi and C. N. Thai, Robotics to promote elementary education pre-service teachers' STEM engagement, learning, and teaching, *Computers & Education*, **91**, pp. 14–31, 2015.
23. C. Robertson, Restructuring high school science curriculum: a program evaluation, 2015, http://scholarworks.waldenu.edu/dissertations/270/, Accessed 6 February 2019.
24. N. Arshavsky, J. Edmunds, K. Mooney, B. Thrift, L. Wynn, S. Center, K. Samonte and L. Janda, Race to the top STEM affinity network, 2014, http://cerenc.org/wp-content/uploads/2014/12/FINAL-STEM-final-report-12-4-14.pdf, Accessed 6 February 2019.
25. B. C. Gamse, A. Martinez, L. Bozzi, H. Didriksen, Defining a research agenda for STEM corps: working white paper, Abt Associates, Cambridge, MA, 2014.
26. W. Gander, Informatics—New Basic Subject, *Bulletin of EATCS*, **2**(116), 2015.
27. P. Brusilovsky, S. Edwards, A. Kumar, L. Malmi, L. Benotti,

D. Buck and J. Urquiza, Increasing adoption of smart learning content for computer science education, *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference*, ACM, pp. 31–57, 2014.
28. T. Boyle, D. Leeder and H. Chase, To boldly GLO—towards the next generation of learning objects, *World conference on eLearning in Corporate*, Government, Healthcare and Higher Education, Washington USA, November 2004.
29. C. B. Chirila, H. Ciocârlie and L. Stoicu-Tivadar, Generative learning objects instantiated with random numbers based expressions, *BRAIN Broad Research in Artificial Intelligence and Neuroscience*, **6**(1–2), pp. 70–83, 2015.
30. C. B. Chirila, Generative learning object assessment items for a set of computer science disciplines, in Balas V, Jain LC, Kovačević B (eds.), Soft computing applications, *Advances in intelligent systems and computing*, **356**, Springer, Cham, 2016.
31. J. M. Wing, Computational thinking, *Communications of ACM*, **49**(3), pp. 33–35, 2006.
32. P. Rad, M. Roopaei, N. Beebe, M. Shadaram and A. Yoris, AI Thinking for Cloud Education Platform with Personalised Learning, *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.
33. G. Attwell, Personal Learning Environments-the future of eLearning?, *Elearning papers*, **2**(1), pp. 1–8, 2007.
34. R. Boondao, A. J. Hurst and J. I. Sheard, Understanding cultural influences: Principles for personalised e-learning systems, *World Academy of Science, Engineering and Technology*, **48**, pp. 1326–1330, 2008.
35. I. Buchem, G. Attwell and R. Torres, *Understanding personal learning environments: Literature review and synthesis through the activity theory lens*, 2011.
36. L. Castañeda, N. Dabbagh and R. Torres-Kompen, *Personal learning environments: Research-Based practices, frameworks and challenges*, 2017.
37. A. Kiy and U. Lucke, Technical Approaches for Personal Learning Environments: Identifying archetypes from a literature review, *Advanced Learning Technologies (ICALT* 2016), IEEE, 16th International Conference, pp. 473–477, 2016.
38. F. Wild, F. Mödritscher and S. Sigurdarson, Designing for change: mash-up personal learning environments, *eLearning Papers*, **9**, 2008.
39. D. Sampson and C. Karagiannidis, Personalised learning: Educational, technological and standardisation perspective, *Interactive educational multimedia: IEM*, (4), pp. 24–39, 2002.
40. A. Brady, O. Conlan, V. Wade and D. Dagger, Supporting users in creating pedagogically sound personalised learning objects, *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, July 2008, Springer, Berlin, Heidelberg, pp. 52–61, 2008.
41. R. Costello and D. P. Mundy, The adaptive intelligent personalised learning environment, Advanced Learning Technologies (ICALT2009), *Ninth IEEE International Conference*, July 2009, pp. 606–610, 2009.
42. M. Guzdial and B. Morrison, Growing Computer Science Education into a STEM Education Discipline, *Communications of the ACM*, **59**(11), pp. 31–33, 2016.
43. K-12 Computer Science Framework Steering Committee, K-12 Computer Science Framework, Technical Report, ACM, 2016.
44. N. Pukkhem and W. Vatanawood, Personalised learning object based on multi-agent model and learners' learning styles, *Maejo International Journal of Science and Technology*, **5**(3), p. 292, 2011.
45. A. Zapata, V. H. Menendez, M. E. Prieto and C. Romero, A hybrid recommender method for learning objects, *IJCA proceedings on design and evaluation of digital content for education (DEDCE)*, **1**, pp. 1–7, 2011.
46. H. Drachsler, K. Verbert, O. C. Santos and N. Manouselis, Panorama of recommender systems to support learning, *Recommender systems handbook*, Springer, Boston, MA, pp. 421–451, 2015.
47. L. Sun, S. Williams, K. Ousmanou and J. Lubega, Building personalised functions into dynamic content packaging to support individual learners, *Proceedings of the 2nd European Conference on e-Learning*, Glasgow, pp. 439–448, 2003.
48. P. Dolog, M. Kravcik, A. Cristea, D. Burgos, P. Bra, S. Ceri and E. Melis, Specification, authoring and prototyping of personalised workplace learning solutions, *International Journal of Learning Technology*, **3**(3), pp. 286–308, 2007.
49. M. I. Santally and A. Senteni, Effectiveness of Personalised Learning Paths on Students Learning Experiences in an e-Learning Environment, *European Journal of Open, Distance and E-learning*, **16**(1), pp. 36–52, 2013.
50. R. Burbaitė, K. Bespalova, R. Damaševičius and V. Štuikys, Context-aware generative learning objects for teaching computer science, *International Journal of Engineering Education*, **30**(4), pp. 929–936, 2014.
51. V. Štuikys, R. Burbaitė, K. Bespalova and G. Ziberkas, Model-driven processes and tools to design robot-based generative learning objects for computer science education, *Science of Computer Programming*, Elsevier, **129**, pp. 48–71, 2016.
52. The learning Object Metadata Standard, IEEE Learning Technology Standards Committee, 2007 Revision, Accessed 18 January 2019.
53. V. Štuikys and R. Damaševičius, *Meta-Programming and Model-Driven Meta-Program Development: Principles Processes and Techniques*, Springer, 2013.
54. E.W. Dijkstra, Notes on Structured Programming, Dahl OJ, Dijkstra EW Hoare CAR (eds.), *Structured Programming*, Academic, London, 1972.
55. R. Burbaitė, V. Drąsutė and V. Štuikys, Integration of computational thinking skills in STEM-driven computer science education, IEEE, *IEEE Global Engineering Education Conference (EDUCON)*, pp. 1824–1832, 2018.
56. D. Churchill, Towards a useful classification of learning objects, *Educational Technology Research and Development*, **55**(5), pp. 479–497, 2007.
57. R. Graham, The global state-of-the-art in engineering education (Outcomes of Phase 1 benchmarking study), MIT School of Engineering, 2018. Retrieved on April 10 2019 from eet.mit.edu/wp-content/uploads/2018/03/MIT_NEET_GlobalStateEngineeringEducation2018.pdf

Vytautas Štuikys, Professor, holds PhD and Doctor Habilitatus degrees in CS and Software Engineering since 1970 and 2003 adequately. His research interests include design of educational software systems (eco-systems) based on using high-level modelling, model transformations (including meta-programming), knowledge-based and Learning Analytics approaches. He is author/co-author of three monographs published by Springer in 2013, 2015, 2018.

Renata Burbaitė, PhD is currently an Associated Professor at the Software Engineering Department of Kaunas University of Technology and Informatics teacher at the High School. Her research interests include CS educational domain modelling and STEM-oriented robot-based CS education methodologies. She is a co-author of the monograph, Smart STEM-Driven Computer Science Education: Theory, Methodology, and Robot-Based Practices", published by Springer in 2018.

Vida Drąsutė is a Project manager in the Faculty of Informatics at Kaunas University of Technology. Since 2006 she developed more than 30 educational international projects, in 5 of them was as coordinator. Her doctoral thesis is

concerned with intelligent systems in education, personalised learning, pedagogical and technological agents, and rational provision of learning syllabus. Area of Expertise: Formal and non-formal education, intelligent systems, e. learning processes, personalised learning, STEAM, education improvement and development.

Giedrius Ziberkas received PhD degree in information technology in 2001 from Kaunas University of Technology. Currently he is an Associated Professor at Software Engineering Department of Kaunas University of Technology. His research interests include automated program generation, metaprogramming, and development of learning systems. In addition, he has published a few works in the energy efficiency area. Since 2011, he is accumulating experience in the field of personal data security.

Sigitas Drąsutis, PhD is an Associated Professor at Kaunas University of Technology, Software Engineering Department. He holds PhD in Computer Science since 2006. His research interests cover analysis, specification and development of information systems for studies, databases, e-testing systems. He is lecturing subjects concerning with computer science, programming in JAVA, PHP, communication and collaboration technologies.