DIRECTORATE GENERAL FOR
INNOVATION AND EDUCATIONAL
TECHNOLOGIES

EDU SIM STEAM

STEAM Framework for Teacher Training

2021

EDUSIMSTEAM | Erasmus+ KA3 Forward Looking Cooperation Project

With the support of the Erasmus+ Programme of the European Union

Title	**STEAM Framework for Teacher Training**
Version	E-report
Authors	Ligita Zailskaite (Kaunas University of Technology) Renata Burbaite (Kaunas University of Technology) Remigijus Kutas (Kaunas University of Technology)
Editors	Dr. İpek SARALAR-ARAS (MoNE) Sümeyye Hatice ERAL (MoNE)
Peer Review	Michael Hallissy (H2 Learning) Maria Fojk (H2 Learning) Miguel Goncalves (CTEM Academy) Prof. Dr. Piet Kommers (Both Social) Dr. İpek SARALAR-ARAS (MoNE) Sümeyye Hatice ERAL (MoNE)
Design	Emre TURAN (MoNE)
Publication Date	2021
Dissemination	Public
eISBN Numarası Deliverable Number	TBC TBC

Disclaimer

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

CONTENT

Introduction

The project, “Fostering STEAM Education in Schools” (EDUSIMSTEAM), promotes an effective STEAM approach in education and develops the related teachers’ skills and curriculum.

This paper aims to propose a framework, which is oriented to the r-learning and can be used in other educational areas as well. Educational STEAM programs and methodologies, such as educational robotics, can aid in the acquisition of new skills and competencies needed to solve complex societal problems. Since the degree of interaction offered by robots is thought to be conducive to learning, the physical tangibility of robots necessitates a transition to more imaginative and efficient teaching methods.

That variety of scenarios in the STEAM field is important for several reasons:

- In every project partner country, we have different educational programs, therefore it’s important to give teachers the guidelines for STEAM education and they will have the freedom to prepare appropriate scenarios, based on their learners’ knowledge, experience, and in relationship with particular region problems.
- The learners can be educated in main principles or methods in a STEM field, but when we integrate Art and creativity, we see that there are no boundaries for possible real-life problem solutions. Every learner or learner team can propose their own knowledge and experience-based solutions and solve them. Sometimes it is impossible to solve some open-ended problems, therefore learners and teachers have to use open-ended problem-solving approaches. This is the case, because learners must be convinced of the validity of their steady progress toward the end target. Even though we only have a partial solution, we believe that advanced problem implementation and testing will aid us in reaching the final solution.
- Teachers' roles in the educational process require a holistic approach in the STEAM sector. Teachers must comprehend the shift from a teacher-centered to a learner-centered approach, as well as the role of learners in a learning environment and their own role in this context. Teachers should have strong pedagogical knowledge and be capable of adopting them. Therefore, we proposed framework, pedagogical aspects of STEAM, examples of learning scenarios in order to cover all the essential fields in STEAM education process.
- The r-learning requires, that the teacher should be experienced in teaching technology, be familiar with programming, and at the same time have a lot of soft skills: to motivate learners, to inspire, understand learners’ needs and etc. All this complex of teachers’ competencies for STEAM teaching should be built on previous teachers’ competencies. The process is unique and the insights in this document will give possibility for teachers to involve in teaching process such aspects which are necessary
- The learners’ assessment methods in the STEAM field should be adopted according to the scenarios, content, tasks and at the same time should be engaging. Damaševicius, Narbutaite, Plauska, & Blažauskas, (2017) identified that learners’ participation in the r-learning course, when learners have the possibility to use tangible robotics kits, increased, but this didn’t have a positive impact on the exam results. After the analysis of learners' projects (practical knowledge) and exams (theoretical knowledge), it was identified the high correlation between mentioned variables. Therefore, for the learners' knowledge assessment in the STEAM field, it’s important to incorporate practical and theoretical knowledge assessment methods.

WP2 report contains of six parts:

- The first part presents the framework for teachers’ professional development analysis and presents the developed model through the WP2 implementation period.
- The second part describes the pedagogical aspects of STEAM education. This part gives an understanding to the teachers what are the most important components in the STEAM educational environment.

- The chapter "scenario in STEAM field" defines theoretical aspects of generic scenario development and provides an example of scenario creation in the educational robotic field.
- The fourth part, the collaborative learning model in STEAM, emphasizes the main aspects of work in groups. Collaborative learning is using a learner-oriented approach; therefore, the teacher's role becomes as mentor or advisor.
- The fifth part, the assessment model (pedagogy), presents the main concepts which are important for the learners' knowledge evaluation in the STEAM context.
- The six-chapter presents scenarios assessment methodology. The experts or partners can be invited for the evaluation.

This document is prepared during the period 01-11-2020 to 30-04-2021 after an analysis of literature and discussions in meetings held on 20/11/2020, 15/01/2021, 04/03/2021. The presented frameworks are based on a common agreement of the project partners from Turkey, Spain, Portugal, Lithuania, Ireland, and the Netherlands and will be used for the professional development of teachers. The analysis is particularly focused on determining teachers' needs for STEAM education and 21st-century skills. More information is provided on the project website: http://edusimsteam.eba.gov.tr/.

Keywords: STEAM scenarios, pedagogy, conceptual model, 21st-century skills, educational robotics, collaborative learning.

1. Frameworks for Teachers' Professional Development in STEAM field: An Overview

There is a lot of hope that using integrated STEAM education approaches can help the next generation of learners to solve real-world problems, to ensure collaboration during learning process, to adopt interdisciplinary teaching and etc. Therefore, it is important for the appropriate teachers' education in the STEAM, to give them a deeper understanding of the main educational aspects and how it is better to integrate them with educational robotics.

Arts disciplines together with STEM (Science, Technology, Engineering, and Mathematics) can help learners to investigate the intersection r-learning. The arts have the power to open up new perspectives, ways of thinking, and ways of learning. The r-learning is the Arts ("A") component that contributes to the effectiveness of STEAM education. The robot's usage in education can promote the acquisition of transdisciplinary expertise in social and humanistic sciences (Damaševičius, Maskeliūnas, & Blažauskas, 2018).

Learners' ways of thinking, engaging with others, making choices, and living are expected to be transformed by interactive technology-enriched environments. In this sense, r-learning incorporates a variety of educational robots as interactive environments, as well as other educational media such as digital materials, software programs, e-books, and websites, so that young children will grow up to be competent and capable individuals in their rapidly evolving technological environments (Jung and Han, 2020). R-learning will provide educationally relevant opportunities for learning about new forms of technology for learners (Tocháček, Lapeš, & Fuglík, 2016).

According Han (2010), r-learning can be defined as learning in which is used educational and assistive robots. In this report we narrow down the concept of r-learning and define r-learning as learning that includes various forms of educational robots.

Leoste and Heidmets (2019) emphasize the robots' ability to teach mathematics and other disciplines, but they agree that implementing r-learning is difficult. Teachers should have prior experience teaching technology and programming. Its critical pedagogical and methodological criteria in the robotic domain (hardware and software parts). This teaching technique can be extended to several subjects (Damasevicius, Narbutaite, Plauska, & Blazauskas, 2017).

1.1. Frameworks for Teachers' Professional Development Analysis

There are various frameworks that could be used when planning the STEAM professional development platforms' curriculum (see Table 1). The framework for teacher training curriculum in EDUSIMSTEAM was amended after careful analysis of different frameworks. Some components were integrated into the framework according to the needs of teachers (see Needs Analysis Report http://edusimsteam.eba.gov.tr/?p=337&lang=en) and discussed during the project meetings.

The following part presents the frameworks used in the project.

Table 1. The variety of STEAM professional development frameworks for curriculum building

Name	Main aspects		Authors
Digital Competence Framework for Educators (DigCompEdu)	Main domain	Professional engagement, digital resources, Teaching and learning, assessment, Empowering learners, facilitating learners' digital competence	https://ec.europa.eu/jrc/en/digcompedu

	Sub-domains	Digital competences, Subject-specific competencies, Transversal competencies as educators' and learners' competences	
A Highly Structured Collaborative STEAM Program: Enacting a Professional Development Framework	Design and development	Common vision and design, targets (teachers' orientation, knowledge and practices, which are closely related to the learner outcomes), context (individual contexts, environment and high-reliability organizations);	Bush, Cook, Ronau, Rakes, Mohr-Schroeder, & Saderholm (2016),
	Implementation phase	whole group engagement, classroom implementation and four phases of active implementation (plan, make, study, act)	
	Evaluation	Design, contexts, cycles, connections, measures and assessment, outcomes	
	Research	The main aspects are teacher's knowledge and teachers' orientation and measure of these components.	
Kolb's Experiential Learning Cycle as a Base of Teacher Training Framework	Phases	Concrete Experience, Reflexive Observation (RO), Abstract Conceptualizing; Active Experimenting	https://www.skillshub.com/what-are-kolbs-learning-styles/ Kolb 1984; Kolb, & Kolb, 2009
STEM-driven conceptual model	Components	Pedagogy driven activities, technology-driven processes, knowledge transfer channels; educational environment (tools, STEM library and etc.); learning outcomes	Burbaitė, Drąsutė, & Štuikys, 2018; Štuikys, Burbaite, Blažauskas, Barisas, & Binkis, 2017

1.1.1. DigiCompEdu.

The framework DigiCompEdu focuses on is digital technologies used for enhancement and innovativeness of education and training. The framework consists of six domains: they are professional engagement, digital resources, teaching and learning, assessment, empowering learners, facilitating learners' digital competence. Furthermore, it also includes a number of sub-domains such as digital competences, subject-specific competencies, and transversal competencies as educators and learners' competencies. The framework consists of the six main domains and specific skills are defined in each of these domains (Figure 1). The competencies in these areas range from levels A1 to C2 (see Redecker, 2017).

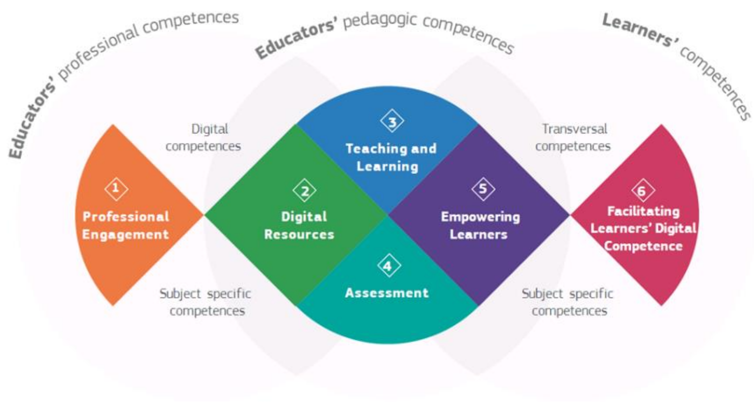


Figure 1. DigiCompEdu Framework (**Redecker, 2017**)

The model presents how the teachers‘skills are important in the STEAM field and emphasize the teachers‘ ability to raise professional competencies as educators, pedagogical competences, or learners' competencies. This leads us to the approach that we have to raise different competences of teachers and at the same time to ensure that he or she will be competent in the digital field. When we look at r-learning (Jung, S. E., & Han, J. (2020) we understand that the teacher ideally should the competencies mentioned above and should be experienced in teaching technology, be familiar with programming, and be able to inspire learners to participate in project development.

1.1.2. A highly structured collaborative STEAM program.

Bush, Cook, Ronau, Rakes, Mohr-Schroeder, & Saderholm (2016), presented the highly structured Mathematics-Science Partnership (MSP) professional development (PD) program for the integration of STEAM in elementary mathematics and science. The framework was developed seeking to promote teachers' ability to integrate STEAM into the classrooms and was based on the previous frameworks (Rakes, Bush, Ronau, Mohr-Schroeder, & Saderholm, 2017; Saderholm, Ronau, Rakes, Bush, & Mohr-Schroeder, 2017) The model examines four phases (design and development, implementation, evaluation and research) of a PD program in details and the relationship between these phases.

The first phase, "design and development", involves common vision and design, targets (teachers' orientation, knowledge and practices, which are closely related to the learner outcomes), context (individual contexts, environment and high-reliability organizations); the second phase involves whole group engagement, classroom implementation and four phases plan, do, study, act; third phase evaluation involves design, contexts, cycles and connections, measures and assessment, outcomes; in the research phase, the main aspects are teachers knowledge and teachers orientation and the measurement of booth components.

While all these phases are essential in the STEAM field, the framework appears to be missing teachers' ability to use particular technological tools, digital tools, and learners' preferences in the context of r-learning. According to Weintrop et al. (2015), the use of computational resources and activities in mathematics and science classrooms give learners a more realistic understanding of these fields, better-preparing learners for careers in these fields.

1.1.3. Kolb's Experiential Learning Cycle.

Well-known frameworks, such as Kolb's Experiential Learning Cycle, emphasize a personal oriented educational process. Kolb's experiential model shows how experience is converted into ideas and concepts, which then are used for active experimenting (Kolb, 1984; Kolb, & Kolb, 2009); separate learning components are correlated with the varying stages of the cycle. The learner's engagement in experience formation helps to learn from their mistakes.

The learning circle consists of four phases:

- Concrete Experience, here learners obtain actively experiences in laboratory session, field class;
- Reflective Observation, here the learner expresses their experience obtained during the lessons;
- Abstract Conceptualization, where the learner tries to conceptualize a model or theory what was observed or what is planned to observe;
- Active Experimentation related with active planning or testing of theory or plan for a future experience (Figure 2). Teacher works as mentor, while learners plan their activities.

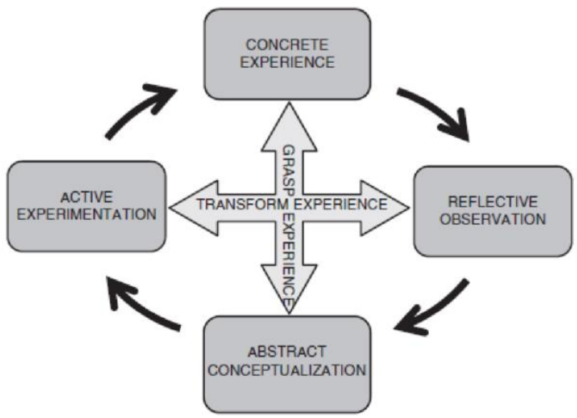


Figure 2. Kolb's Experiential Learning Cycle (Kolb and Kolb, 2009)

In this model, we may see some correlations with computational thinking[11] skills, such capability to think in abstract ways and divide broad views into small pieces (Selby, & Woollard, 2013) These aspects are important in r-learning as well.

1.1.4. STEM-driven conceptual model of the computational thinking (CS) curriculum.

Burbaitė, Drąsutė, & Štuikys, (2018); Štuikys, Burbaite, Blažauskas, Barisas, & Binkis, (2017) proposed a model for STEM-driven computational thinking curriculum from teachers' perspective. Computational thinking is especially emphasized in the computer science field and it compliments critical thinking as a way of reasoning to solve problems, make decisions and interact with our world. The model involves five components such as pedagogy-driven activities, technology-driven processes, knowledge transfer channels; educational environment (tools, STEM library, etc.); learning outcomes and these components are connected with two-sided arrows.

[1] Computational thinking can be defined thought a) process b) the concept of abstraction, c) the concept of decomposition: a) how people think to solve problems; b) breaking down big problems into smaller ones by functionality; c) to provide simultaneous consideration for multiple layers of abstraction as well as consideration for specifying the layers' interfaces.

This model was proposed for the partners as a framework for the teachers' framework creation (Figure 4). This proposition was based on several reasons. It is constrained from the teachers' approach. This framework is applicable in other disciplines, not only in the computer science field (Figure 3).

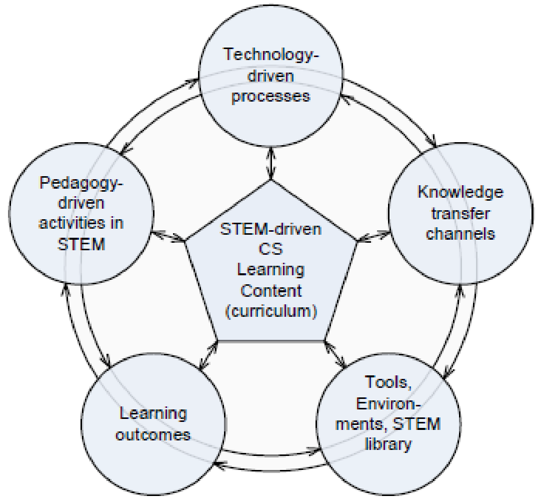


Figure 3. A framework to implement STEM-driven conceptual model of the Computer Science curriculum (Burbaitė, Drąsutė, & Štuikys, 2018; Štuikys, Burbaite, Blažauskas, Barisas, & Binkis, 2017).

The models/frameworks which have briefly outlined (see Table 1) above are applicable for different STEAM subjects. However, the final framework (figure 3) appears to be better suited to the robotics field. From the pedagogical perspective, robotics can make STEAM courses more alive since learners can build robots by themselves as well as to program them and can therefore learn directly from them as well as be creative.

In the next subchapters, we will present other theories and models applicable in the STEAM field and a modified framework to implement the STEM-driven conceptual model of the Computer Science curriculum.

1.2.Other theories and models adaptable in the STEAM context

The STEAM field requires a holistic approach for teachers' role in the educational process. The teachers have to understand the changes from teachers-oriented approach to learners-oriented approach, learners' role in a learning context and their own role in this context.

In this subchapter, we will review the theories and models which can supplement the educational STEAM field (Table 2).

Table 2. Main components of theories and models applicable for STEAM education.

Name model/theory	Main aspects		Authors
The model consists of main components: and	Active	Manipulative, observant	Jonassen, Peck, & Wilson (1999)
	Constructive	Articulate, reflective	
	Cooperative	Cooperative, conversational	
	Authentic	Contextualized, complex	
	Intentional	Reflective, regulatory	
Activity Theory	Subject,	Who implementing activity	Engestrom (1987) Engeström (2001)

	Object	Task, which leads to an outcome	
	Community	Participants of activity	
	Rules	Defines how problem-solving and decision making will be organized	
	Specific tools (instruments)	Collaborative tools	
	Division of labor	Divides labor between parties	

1.2.1. Meaningful learning model.

The criteria for experiences within the field of meaningful learning[22] are clarified by Jonassen, Peck, & Wilson (1999) and derived from active learning (Dewey, 1938). According to the nature of this concept, a learner is learning when he or she is doing something. When technologies engage learners in these structures, meaningful learning can be developed. Real learning is related to learners' active actions, the learners who are engaged in a meaningful activity that allows them to control objects and their surroundings where they are working.

It is crucial to present learners with interesting, important, and engaging problems, and these problems should be unstructured. Learners may recognise that common textbook problems or online content are prescriptive and well-structured, and therefore have little motivation or ability to solve them. In contract to well-structured online content there are issues with bad-structured content with some omitted components (Sigrén, 2003).

Collaboration necessitates a lot of discussion among the participants. Learners should be responsible for their own experience, but even though you agree with the collaborative learning principles, the most difficult aspect of putting the values into practice could be assessing the learners.

The main components of the model are: active (manipulative, observant), constructive (articulate, reflective), cooperative (cooperative, conversational), authentic (contextualized, complex), and intentional (reflective, regulatory) presented in Figure 4.

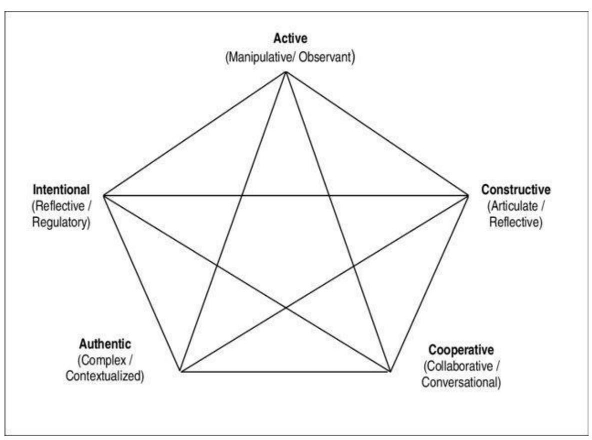


Figure 4. Five attributes of meaningful learning are interdependent.

[2] Meaningful learning is a process when the learner is learning something by doing. When technologies engage the learner in the process, meaningful learning can be created.

This model is useful in the STEAM field and r-learning as well, while it emphasizes the learners' own responsibility to be engaged in different activities, collaborative learning, etc. In the STEAM field, we emphasize the learner's-centric approach where the teacher is a mentor in the learner group; the teacher and the learner share the responsibility for the learning goals.

1.2.2. Activity Theory.

In the sense of understanding human behavior, relationships with societies, and dynamics against various social actors, activity theory is one of the most applied and studied theories.

The first mediation triangle, which discloses interaction between subject, object and community, was proposed by Engeström in 1987. Later it was presented as a triangle with social rules, specific tools (instruments), and division of labour (Figure 5).

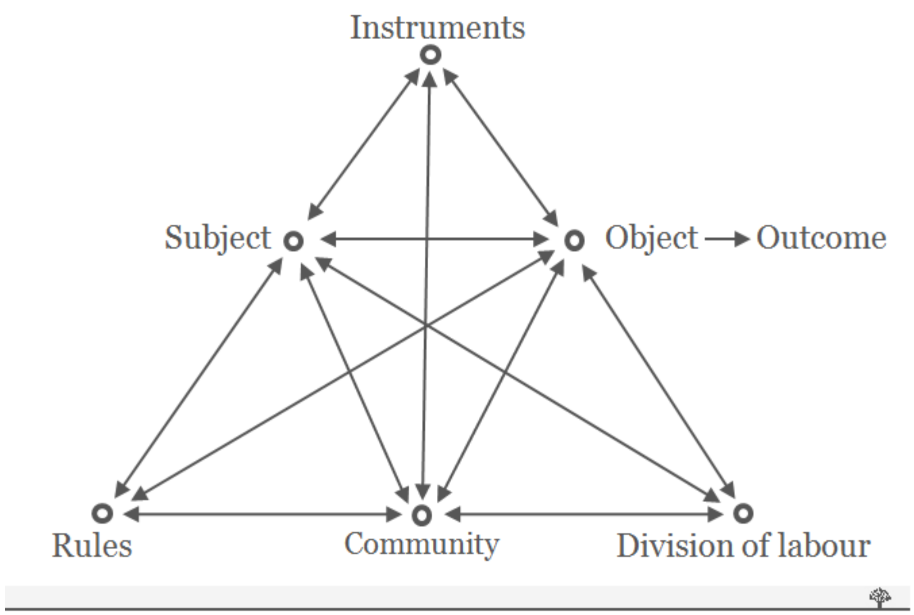


Figure 5. The general model of an activity system (Engestrom, 1987; Engeström, 2001).

Activity theory provides options for comprehending computer program usage and device design, as well as other facets of job activity, which are continually reconstructed to meet the complex demands of any organization. Members of the activity system hold various positions and backgrounds, resulting in a variety of angles or perspectives on their common object.

Activity theory is used in the engineering, human-computer interaction field and others STEM fields.

In this section, we present the models, which can be useful for developing our STEAM model. The suggested model is presented in the 1.3 sub-chapter.

1.3. Suggested framework for WP-2

The presented framework is based on the “A framework to implement STEM-driven conceptual model of the Computer Science curriculum” (Figure 3) and discloses STEM-driven computational thinking curriculum from teachers’ perspective; involves all five components important for STEAM learning curriculum activities such as pedagogy-driven activities, technology-driven processes, knowledge transfers channels; educational environment, learning outcomes.

The suggested draft model for WP-2 after the project meeting is presented in Figure 6.

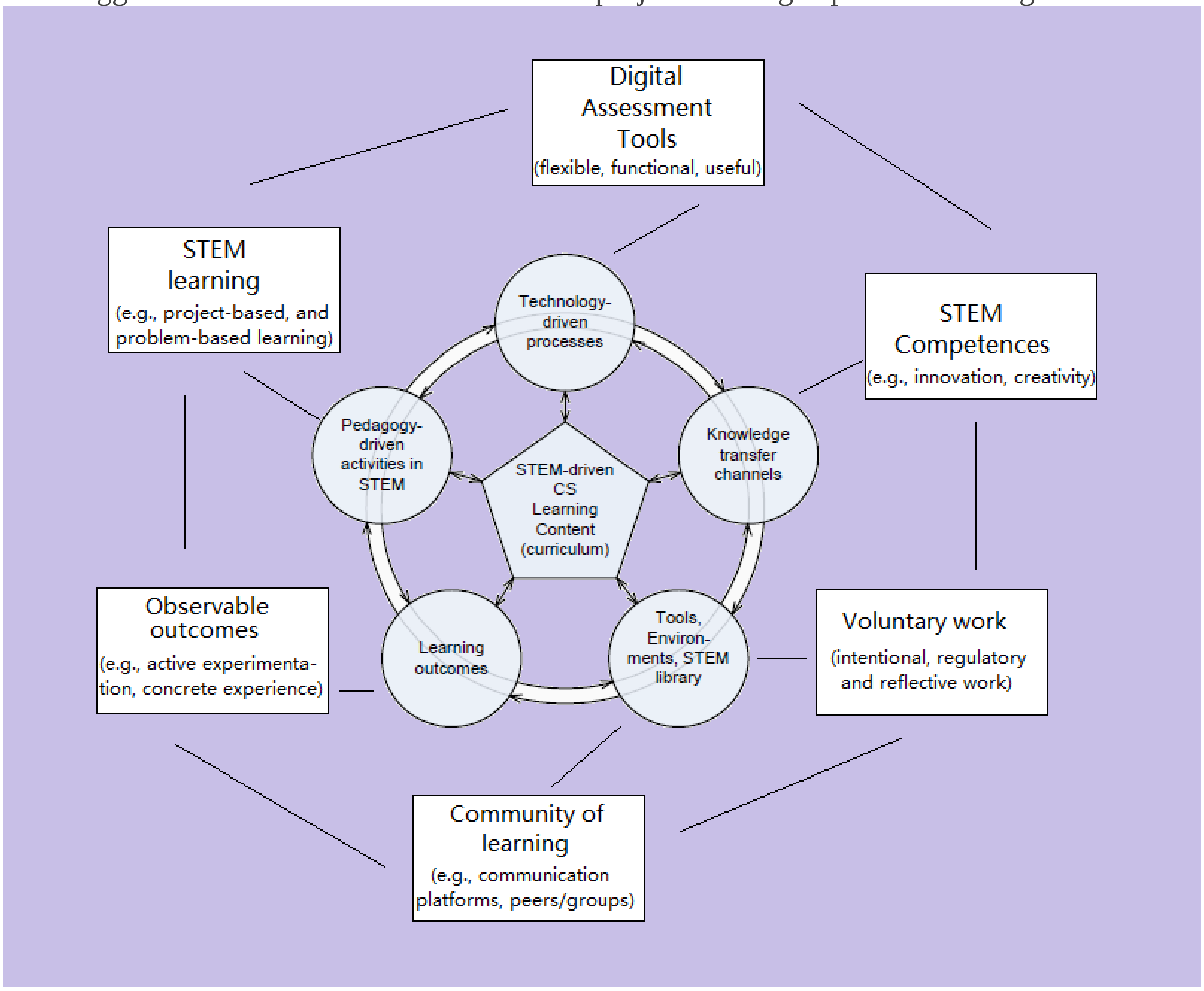


Figure 6. Components of the suggested draft framework for WP-2.

Furthermore, a training program in which teachers prepare, present, exchange, and receive input on their STEAM lesson plans, STEAM integrated course events, learning scenarios, or interdisciplinary project activities/ideas may be integrated into the course design at the end of the course (or at regular intervals during the course).

Finally, a STEM library may include not only articles and books but also lesson plan templates and guiding material (Figure 7). The Figure 7 presents the Figure 6 in more detailed way.

The first cycle involves the main components of STEAM-driven Learning Content (curriculum): The model involves five components as pedagogy-driven activities, technology-driven processes, knowledge transfer channels; educational environment (tools, STEM library, etc.); learning outcomes. The first cycle components are connected with two-sided arrows.

The second cycle involves sub-components such as Digital Assessment, STEAM Competences, Voluntary work, Community of Learning, Observable Outcomes, STEAM learning Types, Learners Identity and Needs (Table 3).

For the construction of the framework, we used layered learning design proposed by Boyle (2009). The pedagogical aspects are in surface layer (No. 2). STEAM learning components are included into the deeper layer (No.1). The two layers are not connected with the arrows. The components of the surface layer can be connected to the different components of deeper layer.

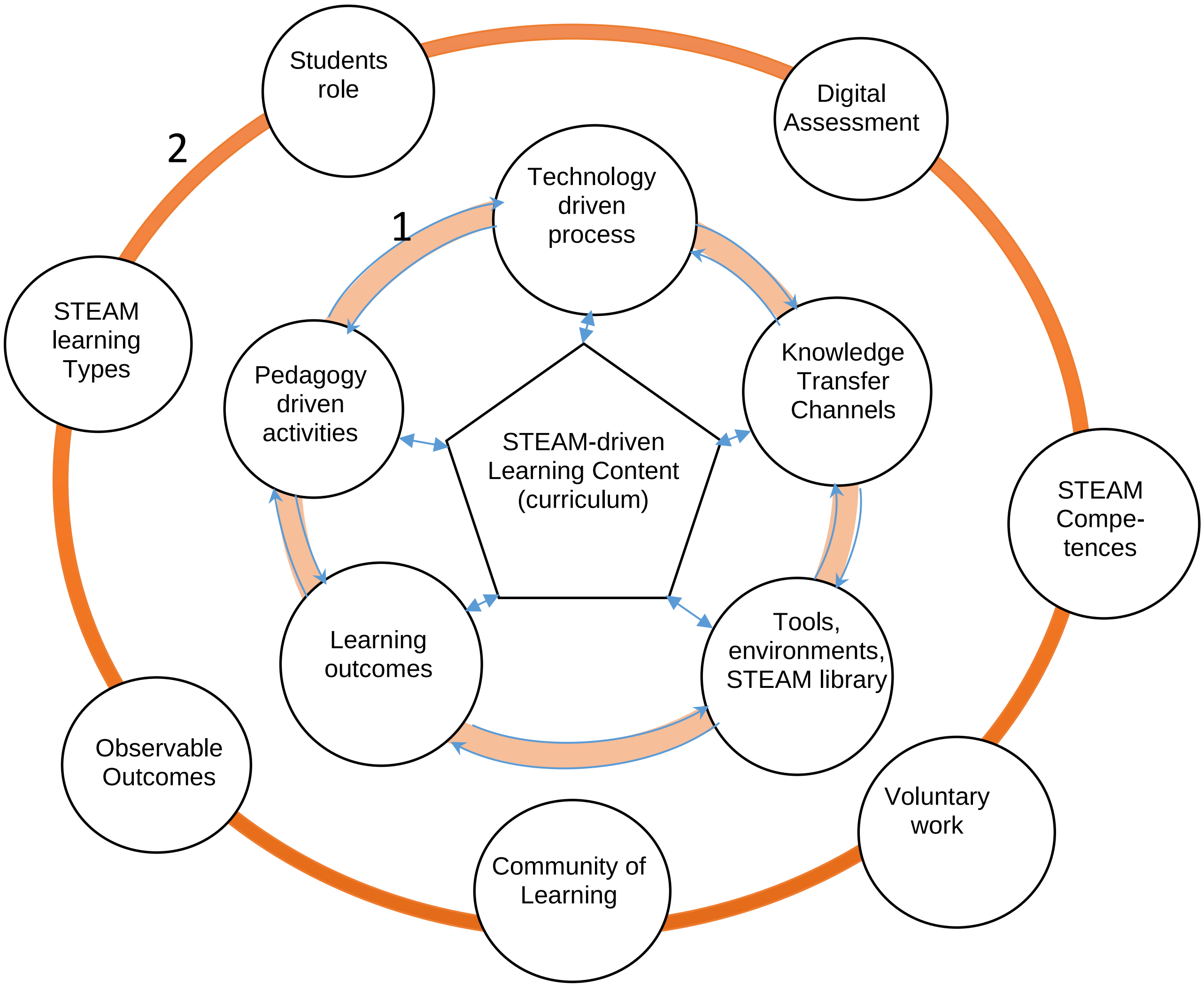


Figure 7. Two-cycle STEM-driven conceptual model (2CSTEAM) (final model).

Despite that we prepared the model for teachers' professional development in the STEAM field, we include as important component – learners' roles, which is related to learners' preferences; needs which make an impact on learning motivation and learners' identity:

- Learners' **preferences** for the course depend on learners' previous experience, what they learned, what they would like to learn, the learning style that can better fit them.
- **Needs and motivation**. Learners have different needs for teachers' support and teacher's support can make an impact on learners' motivation.
- **Identity** is complex in thinking about relationships with others and oneself in the learning process.

Table 3. 2nd Cycle components of 2CSTEAM.

No.	Sub-component	Detailed
1	STEAM learning types (educational approaches)	Problem-based Project-based Inquiry-based Design-based Storytelling Others
2	Learners’ role	Preferences Identity Needs and motivation
3	Digital Assessment	Flexible Functional Useful Free Interactive
4	Teachers Competences in STEAM field	Innovativeness Creativity Creation of a learning environment; Usage of varied teaching strategies Ability to identify learners’ needs Good communication abilities Ability to collaborate Capability to interact with all learners
5	Voluntary work	Intentional; Regulatory; Reflective work
6	Community of Learning	Platforms for collaboration and communication Relationship with peers/ groups nurturing
7	Observable Outcomes	Developed critical thinking Collaboration skills in groups Project development Technical knowledge New educational methods New content creation skills

The model 2CSTEAM can be extended, whereas it can be used in different STEAM disciplines and the second cycle can be added with some particular components important in a particular STEAM discipline (Figure 8).

The implementation of the proposed framework has five stages.

1. The first one involves the need for analysis, requirements, participants’ capabilities. The learners have to have a main understanding of the STEAM field as well (Chapter 1);

2. Training objectives – In this stage, the learners' profile and the learning space is identified, and the pedagogical aspects are taken into consideration (Chapter 2); tools and online learning space, technology options, repository usages, external library usage (it's optional);
3. Curriculum creation (prototype creation) (Chapter 3);
4. Training implementation using collaborative learning model and learners' knowledge assessment (see chapter 4 and 5);
5. Training Evaluation (Chapter 6)

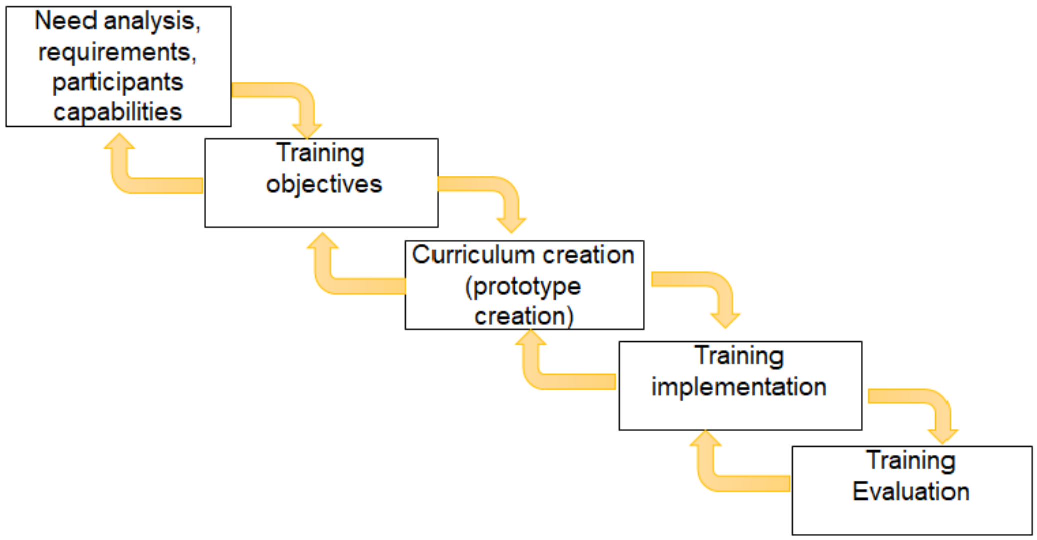


Figure 8. Implementation stages of STEM-driven conceptual model (2CSTEAM).

As a reminder, the returning arrows show that after implementation of every stage, we can revisit the previous stage to make further changes.

From our point of view, this model can be adapted for different disciplines, though it is based on the STEM-driven conceptual model of the computer science curriculum. This correlation between a particular discipline and computational thinking can make a positive impact on different disciplines' curriculum. For example, bioinformatics and computational biology are two distinct areas that benefit from the combination of biology and computer science. The former entails gathering and processing biological data. Simulating biological structures and processes is part of the latter.

We propose that this model could be well applicable for r-learning as well. Educational STEAM initiatives and methodologies, including educational robotics, will help to learn new skills and competencies needed to solve complex problems affecting human society. The physical tangibility of robots necessitates a shift to more innovative and effective teaching methods, as the level of engagement provided by robots is might be conducive to learning (Damaševičius, Maskeliūnas, Blažauskas, 2018).

In the next chapter, we will present the pedagogical aspects of STEAM education which are applicable in different disciplines and the r-learning context.

2. Pedagogical aspects of STEAM education

2.1 Technology, pedagogy and content integration

For a long time, educational technology was criticized for theoretical justification, until Mishra and Koehler (2006) proposed Technological Pedagogical and Content Knowledge (TPACK) (originally TPCK, now known as TPACK) (see Figure 9). This model involves three main domains such as content knowledge, pedagogical knowledge, and technological knowledge.

Content knowledge. Teachers' awareness of knowledge (CK) refers to their comprehension of the subject matter to be learned or taught. The value of content knowledge for teachers is very important. This knowledge will be needed, as will knowledge of concepts, theories, ideas, organizational frameworks, facts and evidence, as well as established practices and approaches to the development of such knowledge. Teachers should understand the basic principles in the disciplines in which they teach because knowledge and the essence of study differ greatly across fields. For example, in science, this could include understanding scientific facts and theories, scientific methods, and proof-based reasoning. Awareness of art history, popular paintings, sculptures, artists and their historical backgrounds, as well as aesthetic and psychological theories for the assessment of art in the case of art appreciation, are examples of such knowledge.

Pedagogical knowledge (PK)**.** Teachers' pedagogical knowledge is related to their in-depth understanding of teaching and learning practices, processes, and methods. They contain, among other things, overall educational objectives, values, and goals. Understanding how learners learn, general classroom management skills, lesson planning, and learner assessment are all examples of this generic type of expertise. It requires an understanding of what kind of techniques and methods should be used in the classroom, the nature of the target group, and strategies for assessing learner comprehension. The teacher with extensive pedagogical understanding knows how learners acquire knowledge and skills, as well as how they develop mental habits and effective learning strategies. Therefore, pedagogical expertise necessitates comprehension of learning cognitive, social, and developmental theories, as well as how they apply to classroom learners.

Technological knowledge (TK)**.** Technology knowledge is still changing in the TPCK sense, more so than the other two core knowledge areas, content and pedagogy. Therefore, explaining it is notoriously difficult. The absolute concept of technology awareness is under consideration. However, some ways of thinking about and interacting with technology will extend to all technology tools and services.

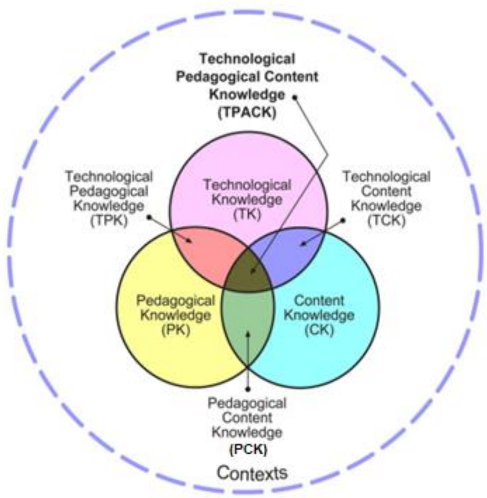


Figure 9. Components of Technological Pedagogical and Content Knowledge (TPACK) (Mishra and Koehler, 2006)

Pedagogical content knowledge (PCK). Shulman's definition of pedagogical knowledge that is applicable to content teaching is consistent with and like Pedagogical Content Knowledge (PCK). The idea of subject matter transformation for teaching is central to Shulman's understanding of PCK. According to Shulman (1986), this transition occurs as the teacher interprets the subject matter, tries several ways to present it, and adapts the instructional materials to the learners' different conceptions and prior experience. PCK addresses the core issues of teaching, learning, training, assessment, and reporting, as well as the factors that promote learning and the connections between pedagogy, curriculum and pedagogy.

Technological content knowledge (TCK). The relationship between technology and content comprehension has a long history. The advent of new technologies that enables new and fruitful ways of representing and manipulating data has coincided with advancements in fields as diverse as medicine, history, archaeology, and physics. Consider how the invention of the digital computer altered the fundamentals of physics and mathematics, emphasizing the importance of simulation in the study of phenomena. Technical advancements have also created new metaphors for understanding the setting. Technologies are making the invisible visible such as seeing the heart as a pump or the brain as an information retrieval device, to name a few examples. These representational and metaphoric associations are not superficial. They've also influenced major changes in the discipline's history. Recognizing the impact of technology on a discipline's activities and skills is critical when developing appropriate technological tools for educational purposes. Content decisions will limit the types of technologies that can be used. Technology can restrict the types of learning content that can be generated, but it can also allow the creation of new and more diverse learning content.

Furthermore, methods for navigating through these learning content can provide a greater degree of flexibility. TCK, then, is an understanding of how technology and content communicate and restrict one another. Teachers must have a deep understanding of how the subject matter (or the types of representations that can be constructed) can be altered using technologies in addition to mastering the subject matter they are teaching. Teachers should think about how the technologies in their domains are best suited to talk about subject-matter learning and how the content influences or even changes the technology or vice versa.

Technological Pedagogical Knowledge (TPK). TPK is an understanding of how teaching and learning can evolve when specific tools are used in various ways. This entails understanding the pedagogical affordances and limitations of several technological instruments as they relate to disciplinarily and developmentally appropriate pedagogical designs and methods. To improve TPK, a better understanding of the constraints and affordances of technologies, as well as the disciplinary contexts in which they operate, is needed. The TPACK (TPCK) framework summarizes the intersection of the Technological knowledge, Pedagogical knowledge and Content Knowledge within the overall educational context. This framework, in fact, is the conceptual model to understand any education domain. Štuikys and Burbaitė (2018) presented a framework of defining STEM contexts. This model is based on Mishra and Koehler's (2006) TPACK (originally TPCK, now known as TPACK) model (Figure 10) and is treated as conceptual model to understand STEM domain.

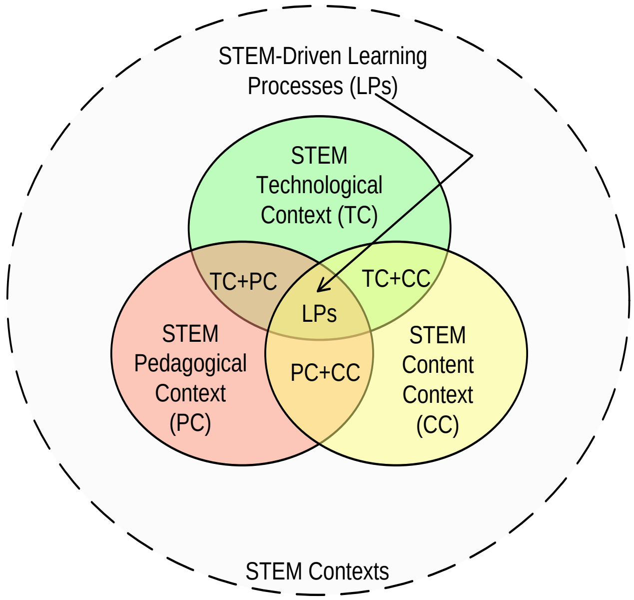


Figure 10. A framework of defining STEM contexts (Štuikys, & Burbaitė, 2018).

Conradty and Bogner (2018) emphasize creativity's importance in the STEAM field in comparison with the STEM concept. The STEAM can be related to active communication and learner teamwork during the lessons. Learners learn to speak and present their ideas and are not afraid to share their thoughts. Children spend most of their time testing and developing their projects rather than sitting at their desks. STEAM lessons should engage learners and keep in touch with teachers. Learners are more interested in science and technology when they are busy building robots, bridges, and buildings. Learners gain science education skills as well as STEAM competencies. Therefore, it is important to prepare appropriate STEAM pedagogy for the education process and to ensure teachers plan.

2.2. Pedagogical STEAM driven model

Usually, teachers are qualified to teach only one subject in secondary education, so involving an integrated, transdisciplinary, or interdisciplinary approach into STEAM education is of great importance. STEAM is a teaching and learning approach that integrates STEM disciplines with Art and can improve learners' inquiry skills, problem-solving skills, and creative thinking. The movement from STEM to STEAM movement can provide new insights and vocabulary in transdisciplinary thinking. In this part, we will present the main components of the pedagogical STEAM-driven approach.

The conceptual pedagogical STEAM-driven model is based on Štuikys & Burbaitė (2018) framework of defining STEM contexts. There are three components of this model are pedagogical aspects, technological aspects, and content aspects (Figure 11).

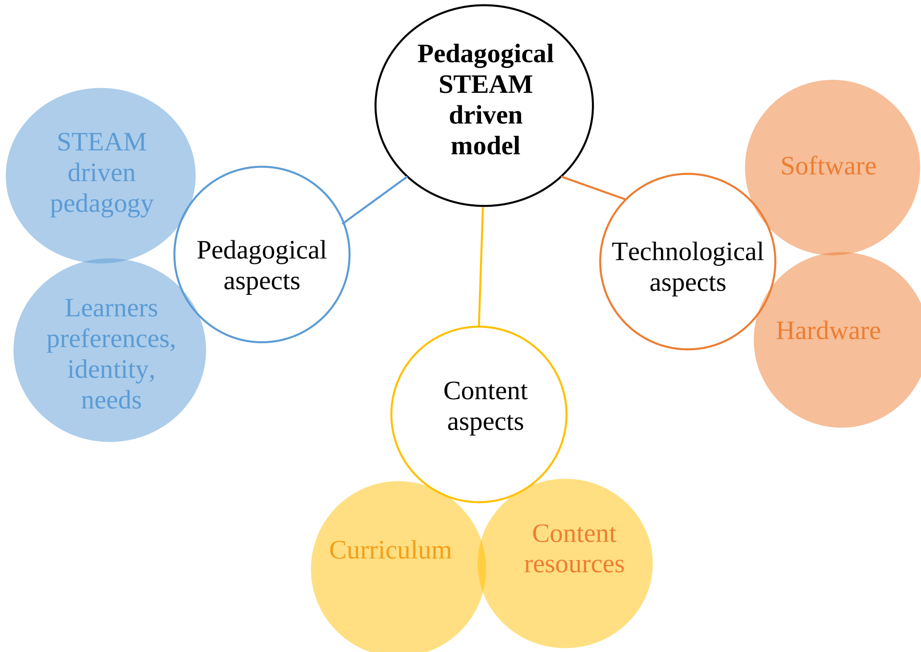


Figure 11. Conceptual pedagogical STEAM-driven model.

Pedagogical aspects consist of STEAM-driven pedagogy and learners' preferences and identity. This sub-component involves motivational attributes which define teachers' role in all educational processes. In the STEAM field, the teacher becomes a mentor who can help solve tasks or advisors for learners on what kind of technological devices to choose in order to solve a particular problem.

In the STEAM field, learner-centered learning is of great importance (Cornelius-White & Harbaugh, 2009) because a successful learning process depends on the learners' preferences, identity, and needs leading to learners' motivation. Therefore, looking from teachers' perspective is important to prepare pedagogical resources that will be adopted according to learners' learning level. If the material is too easy, then it will not be interesting to study. If it is too difficult – it will frighten learners. Cognitive skills should be developed through the thinking, learning, remembering, problem-solving process (Figure 12).

In Figure 12, we present an attribute-based model that includes attributes of two main components, i.e. the STEAM-driven pedagogy and learners' preferences. STEAM-driven attributes related to the motivation, pedagogical approaches specifically extended to be relevant for the STEAM paradigm, while the remaining attributes (assessment and learner's preferences) are generic.

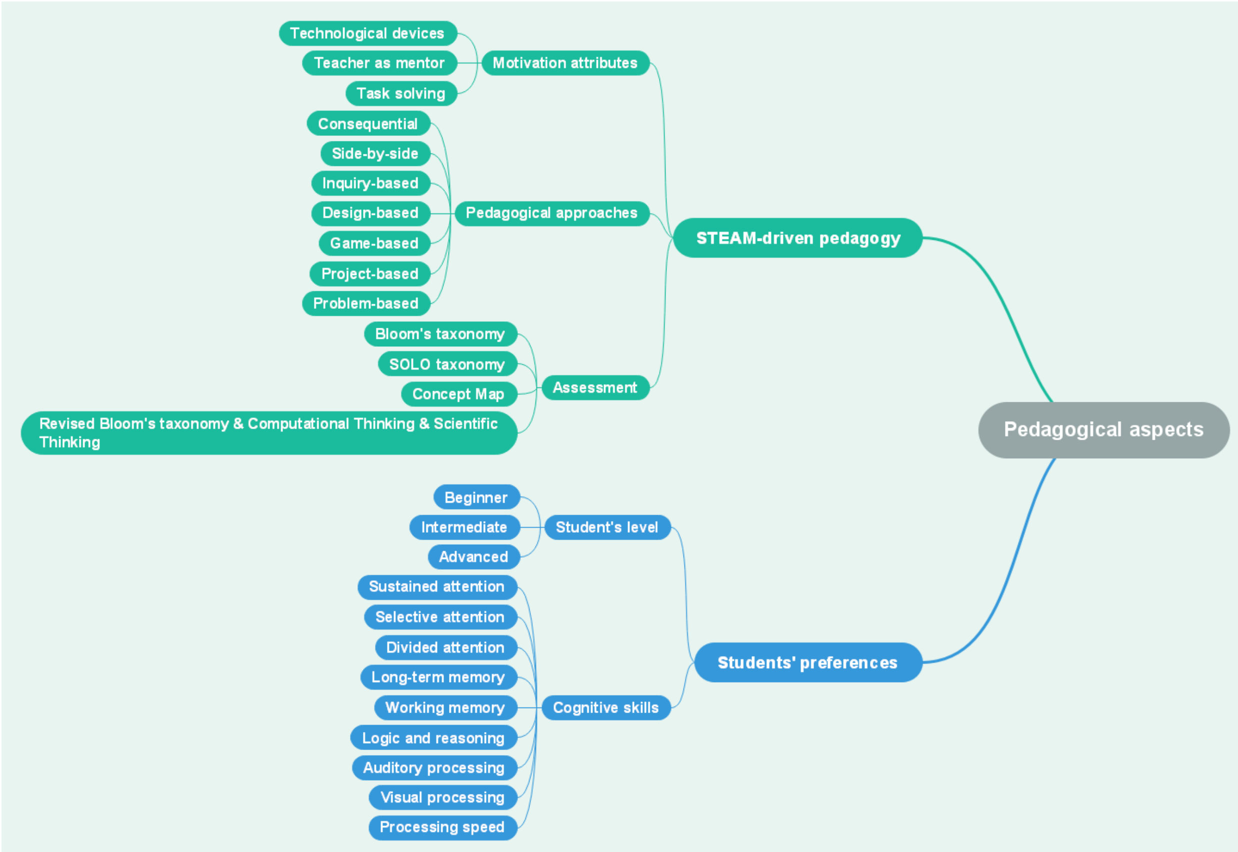


Figure 12. Conceptual pedagogical STEAM-driven model: pedagogical aspects.

2.3. The model of technological and content aspects

The importance of technological and content aspects is presented in Figures 10 and 11. This section presents a detailed model of the technological and content aspects of STEAM.

Technological aspects consist of software and hardware (Figure 13). The software components are related to general programs, which are important for content search and creation and other specific software programs and modelling tools.

Hardware is important, especially in the educational robot field, where understanding how particular components of smart devices can be used can lead to achievements in the learning process.

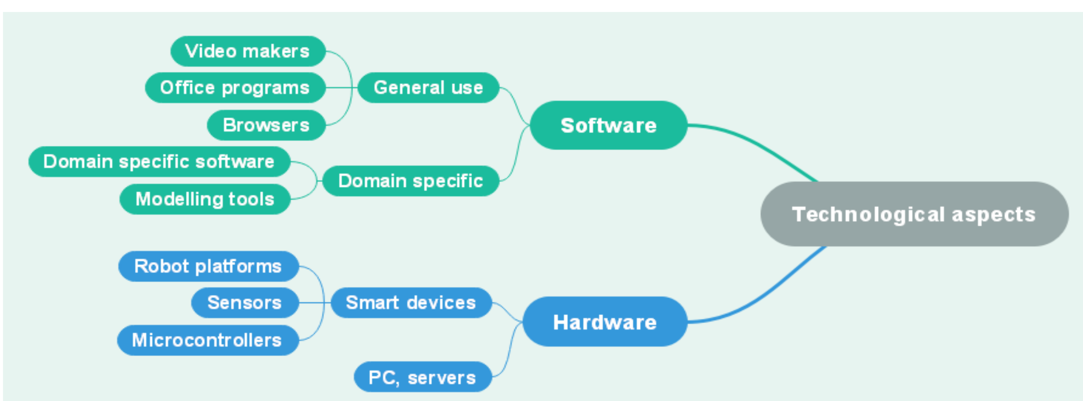


Figure 13. Conceptual pedagogical STEAM-driven model: technological aspects.

More detailed information about content aspects, curriculum creation stages will be presented in the next chapters. Figure 14 presents STEAM content aspects, including curriculum and content resources, that will derive from a personal generative library or from external libraries.

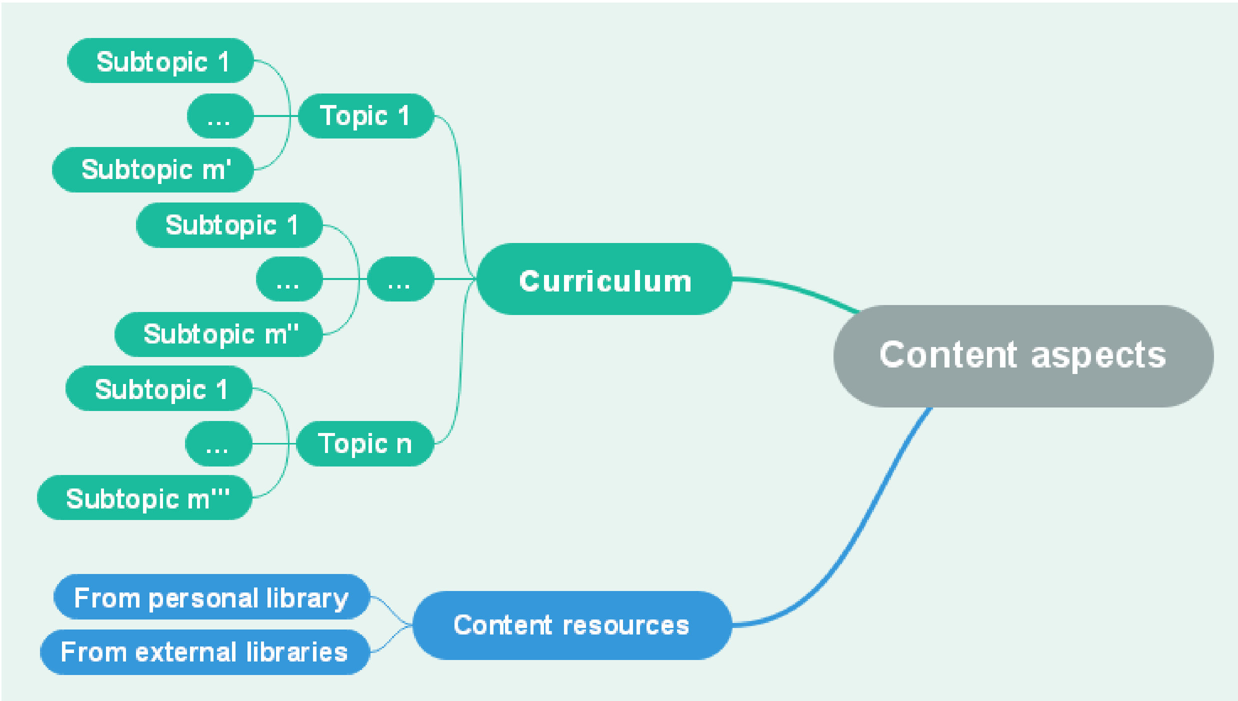


Figure 14. Conceptual pedagogical STEAM-driven model: content aspects

In this chapter, we presented pedagogical aspects of STEAM education involving a learner-centered learning approach. Using this approach, a learner becomes an active participant in the learning process and the teacher becomes a mentor who can help to solve, consult, direct, etc.

3. Learning scenario in STEAM field

3.1. The basis for learning scenario creation

Through the STEAM approach, the learner's mastery can be improved in individual disciplines and assist them in connecting disciplines. The learner's experience determines the development of understanding. The new knowledge builds on previous understanding, allowing it to be incorporated into the new context. As a result, learners should be able to relate what they learn in the classroom to what they see in the real world. When learners deepen their understanding, information is passed.

Deeper learning is described in the report (National Research Council, 2012) as "the process by which an individual can use what they learned in one situation to apply it in another". Deep learning requires acquiring information from others in the group for a person to acquire expertise in a specific field of knowledge and/or results.

The study goes through various types of information, including material knowledge and how to use it to answer questions and solve problems. Furthermore, the study stresses the importance of 21st-century competencies (cognitive, interpersonal, and intrapersonal domains). Cognitive competencies involve critical thinking, analysis and problem-solving. The most important conclusion is that: "The process of deeper learning is essential for the development of 21st-century competencies, and the application of transferable 21st-century competencies support the process of deeper learning in a virtuous cycle."

By design, integrated STEM educational environments require learners to participate in disciplinary knowledge transfer, with the goal of allowing them to transfer their knowledge to another area or activity in the future. Integrated STEM experiences differ depending on whether they are structured to facilitate knowledge convergence across disciplines or to target discipline-specific knowledge and skills. Although a particular background or behaviour can necessitate the use of methods from multiple disciplines, learners are required to demonstrate gains in only one discipline.

STEAM experiences are intended to help learners progress in more than one field of study in addition to enhancing their learning in that area, but they are not required to show an ability to make connections across disciplines. There are also a variety of interconnected experiences that assist learners in creating cross-disciplinary connections. Learners' experience in and major discipline should be taken into account in the curriculum so that they can make links between disciplines and build on what they already know.

Same study as referenced earlier on transition in the light of 21st-century skills, the National Research Council According to the National Research Council (2012): "there is little research on how to help learners transfer competencies learned in one discipline or topic area to another". The study identifies instructional features and characteristics that promote learning transfer:

- Concepts and tasks broad use;
- Elaboration and questioning encouragement;
- Challenging learners to motivate them;
- Teaching by using examples and relevant cases;
- Learners' interests' activation;
- Formative feedback usage.

Many of these safety features are present in integrated STEAM programs, but further research is required to evaluate how well they help the growth of both disciplinary expertise and interdisciplinary thought. Both of these features are relevant to the STEAM scenario dilemma and are compatible with it.

3.2. Variety of scenarios in e-learning

A learning scenario is a description of how a person learns in a particular setting. It explains how it is organized to ensure that a given field is well-understood. It specifies the tasks, activities, and resources, equipment, and services that must be used.

Lejeune and Pernin (2004) proposes a *scenario structure, taxonomy, and vocabulary*, which it then incorporates into a broader framework. The answers to questions like “What is aggregation?” and “How are aggregators made?” are given in this paper by Lejeune and Pernin (2004). Boudalis (2012) examines the educational examples used in the scenarios “Discover the COSMOS.” The paper contains two completed samples as well as an outline for “The Pedagogy of Inquiry Teaching: Strategies for Developing Inquiry as Part of Science Education”.

Rius, Sicilia, and García-Barriocanal (2009) talks about the specification for automating learning scenario usage. The *creation of a database of learning scenarios that can be used automatically to retrieve stored specifications is a major aspect of this issue*. The proposed method employs a domain-specific ontology that aids in the development of new scenarios as well as the monitoring of existing ones. *A real-life learning scenario is used to explain how to apply the applicable principles to the scenario and automate the process*. A typology of various scenario approaches is presented in the paper. There are three main areas of *focus (goals, design, and content)*, each with ten subcategories. This typology depicts how different scenario methods are used, as well as the variety of performance types and contexts in which they are used.

Dahlgren and Oberg (2001) focus on *problem-based learning* and *how the scenarios in a ten-week introductory course* in environmental science worked in terms of the structure and substance of the questions they elicited. The information is explored in the context of scenario design and learner perspectives.

Personalization of learning scenarios is proposed by Essalmi, Ayed, Jemni, & Graf (2010) based on two levels of *individual learning style preferences*. The first approach allows for *customized learning to extend learning*. The second level allows teachers to *tailor personalization approaches to their courses' precise requirements*. Web service technology and new e-Learning personalization technology converge to create an interoperable e-Learning personalization solution.

Vantroys and Peter (2003) describe a cooperative Open Workflow System (COW) that aims to bring Learning Paths into *motion within Learning Management Systems* (LMS). Educational modeling languages are used as a standard method of representing learning paths in learning management systems. The COW learning engine is oriented toward learning based on using the internet. In order to demonstrate how to decompose a model into constituent bits, the paper *contrasts EML and workflow approaches*. Boticario and Santos (2007) address the role of the entire e-Learning cycle (design, publication, usage, and auditing) in the e-learning process. Machine learning, formative evaluation, and complexity reduction methods are all used in this approach.

Multimedia learning scenarios, according to Nadolski, Hummel, Slootmaker, and Van der Vegt (2012), will promote lifelong learning by making a range of higher-order skills simpler and more engaging. Since current virtual worlds, game development platforms, and virtual reality devices are unsuitable for the proliferation of such scenarios. This research aims to find the most efficient architectures for creating them. The paper proposes a *methodology for achieving such architecture description and setup*. Luckin, Mavrikis, Avramides and Cukurova (2015) describe the *efforts to adapt collaborative problem-solving and problem-based learning to project-based learning, including structure and technology*.

Challco, Mizoguchi, Bittencourt and Isotani (2015) investigate the use *of gamification to boost learner motivation by altering their emotions in a positive way*. The paper proposes an ontology called OntoGaCLeS to provide a formal systematization of knowledge about gamification and its correct application.

Just a few papers dealing with STEM-related issues were found (Štuikys and Burbaitė, 2018). The paper focuses on the "Go-Lab" program, a STEM education initiative aimed at motivating and orienting learners to research STEM fields as part of their potential educational journey from an early age. This paper describes an inquiry-learning framework that enables teachers to use online labs customized to their classes and learners to learn scientific methodologies when performing experiments in the labs.

According to Kerven, Nagel, Smith, Abraham and Young (2017), scenario-based learning immerses learners in the real world, increasing their interest in computing. Komis, Romero and Misirli (2016) proposed a scenario-based guide to designing educational robotics activities. Costa (2014) employs scenario-based learning with robots to boost learner motivation to learn to program.

Personalized e-Learning scenarios, according to Essalmi, Ayed, Jemni, and Graf (2013), offer one of the most successful levels of personalization.

Zook, Lee-Urban, Riedl, Holden, Sottilare, and Brawner (2012) describe an automated scenario generator for military scenarios. The agent-based simulation of many health education scenarios is used by Gupta, Bertrand, Babu, Polgreen, and Segre (2012). Martin, Schatz, Bowers, Hughes, Fowlkes, and Nicholson (2009) explain how scenario-based analysis was used to construct the procedural model.

One of the greatest challenges in STEAM-driven education is to implement the most effective learning methods, resources, and tools to achieve learning goals. This is especially important in terms of integrating the advanced technology with STEAM-driven scenarios aiming to achieve a higher efficiency through systematization, integration, and automation.

3.3. A framework for creating scenarios

In this section, we present the methodology for creating a generic scenario for STEAM-driven education.

The framework for learning scenarios consists of three interrelated parts: header implementation, main part implementation and external libraries (Figure 15):

- **Header interface.** Before the material is posted on the learning platform, the teacher must think about curriculum scenario types that will lead to reachable objectives implementation and expected results. In this phase, it is important to think about pedagogical approaches and recuses which are used. It is important to think about a learner's identity and preferences.
- **The main part interface** is related to the content of selected resources preparation, main activities and task identification and implementation.
- **External libraries** are important for engaging content development.

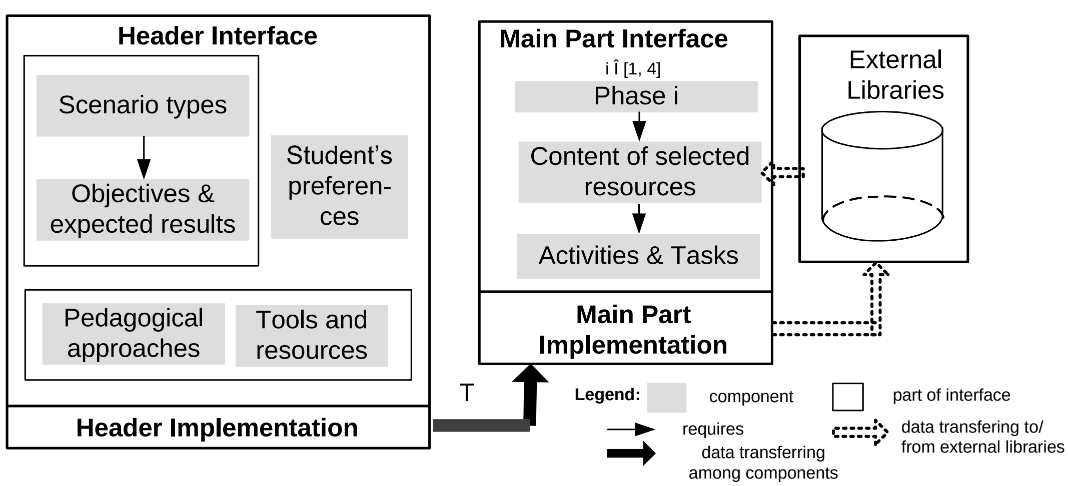


Figure 15. Learning scenario framework (Štuikys & Burbaitė, 2018),

Bybee, Carlson-Powell and Trowbridge (2008), and Boudalis (2012) focus on the structure of scenarios. The designed specification was regarded as a standard or generic specification in this case, but it was provided as a scenario prototype by the authors of Boudalis (2012).

The structure's content must be adjusted to the condition in which it will be used. First, we begin with minor changes to the original scenario. The two-level structure has remained unchanged, although the number of phases in the Main Part has been decreased from five to four (Figure. 16). We've also made several improvements to the headers, such as eliminating Learner Positions and replacing them with Planned Outcomes and Curriculum-related Goals.

There are five subtopics in the Header and four subtopics in the Main Part. The "Who's Who in Robotics" is completely dedicated to STEM-based CS education using robotics. The specificity of our system emerges in the next step when we differentiate between STEM-focused goals and STEM-related contexts.

The learning scenario example structure is presented in figure 16. An example of a scenario (Header) is presented in **Appendix 1.**

The Header has five products, while the Main Part has four (**Appendix 2).** The content of the products in both sections is completely unique and geared toward STEM-driven CS education with robotics. When we implement the scenario oriented to STEM-oriented goals and meaning in the next stage of our framework, we introduce the specificity.

Some situations can be omitted depending on the goals or other factors. However, the sequence discussed here embraces both problem-based and inquiry-based learning. The investigation, according to Barell (2010), is "the engine of complex thought during problem-solving." This method relies on the learner's prior experience to create the new ones on their own.

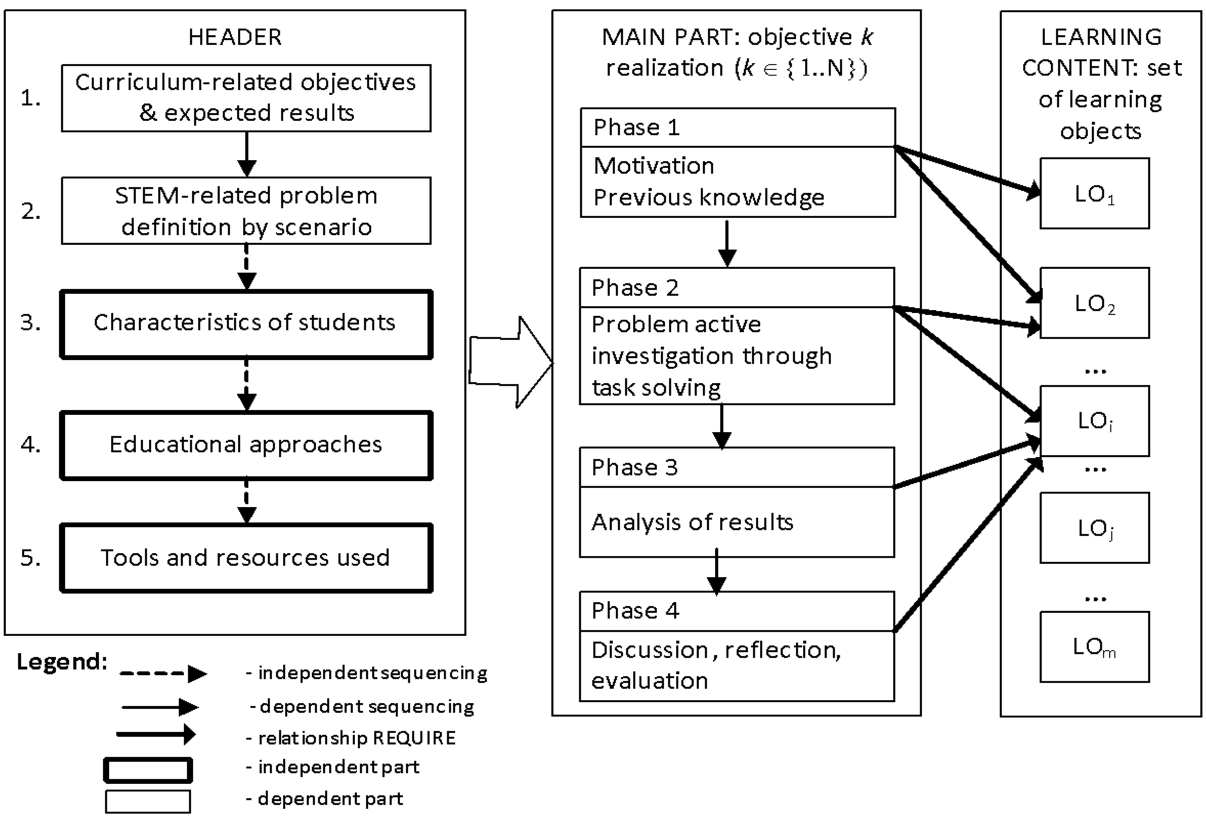


Figure 16. Learning scenario (Štuikys & Burbaitė, 2018).

Figure 16 depicts the various scenarios and applications of scenario analysis in education (meaning the duration of the year or half-year, i.e., semester). It is not necessary to do things in the order indicated. A few examples can be excluded depending on the project's objectives.

The investigation, according to Boudalis (2012), is "the driving force in complex thinking during problem-solving." This opinion relies on the learner's prior knowledge for them to create some new ones on their own. Consequently, the proposed series facilitates the incremental accumulation of previously learned information from previous circumstances.

The type defines what the activities and tasks can be implemented. It is possible to concentrate on a specific field of expertise in STEAM education through understanding and applying these fields of study.

The integration of General Instructional Design Phases (ADDIE model) and Pedagogical STEAM-driven model is presented in Figure 17.

Phases	**Key aspects**	**Key elements from P-STEAM**
Analyse	The problem definition and analysis, goals, success metrics, and overall objectives	
Design	Learning objectives, instructional methods and activities, storyboards, content, subject matter knowledge, lesson outlines, and media assets.	
Develop	Develop the content and learning interactions; visualization	
Implement	Content delivery to the platform	
Evaluate	Measurement tools (what to measure and how)	
Pedagogical aspects	Content aspects	Technological aspects

Figure 17. Integration General Instructional Design Phases and Pedagogical STEAM-driven model (P-STEAM)

Combining the five stages of the ADDIE model (Figure 17) and the Pedagogical STEAM-driven model (P-STEAM) (Figure 16), we see the integration of generalised scenarios creating model for STEAM implementation capabilities which can be as guidelines for the teachers. The table shows how pedagogical aspects and action are united, and theory and practice can be mixed, thanks to the creation of rich functional operation teaching activities in r-learning.

4. Collaborative learning model in a STEAM environment

Collaborative Learning generates a highly encouraging learning atmosphere in schools, promotes learner cooperation, and results in developing new information through a reflexive process mediated by the teacher. A technological environment in which learners engage actively, exchange experiences, and create knowledge is referred to as computer-assisted collaborative learning (Plauska & Damaševičius, 2014). Face-to-face communication creates a highly rewarding learning environment by altering classroom dynamics and facilitating learner collaboration to produce good outcomes.

Education in a virtual space or using different online platforms, mobile learning provides more challenges for teachers, but the essential principles of collaborative learning remain.

Collaborative Learning is very important in STEAM-driven education. It is a part of STEAM-driven pedagogy (see Figures 6, 7) and it discloses the student-centric learning approach We present the framework for Collaborative Learning in the context of r-Learning (Figure 18).

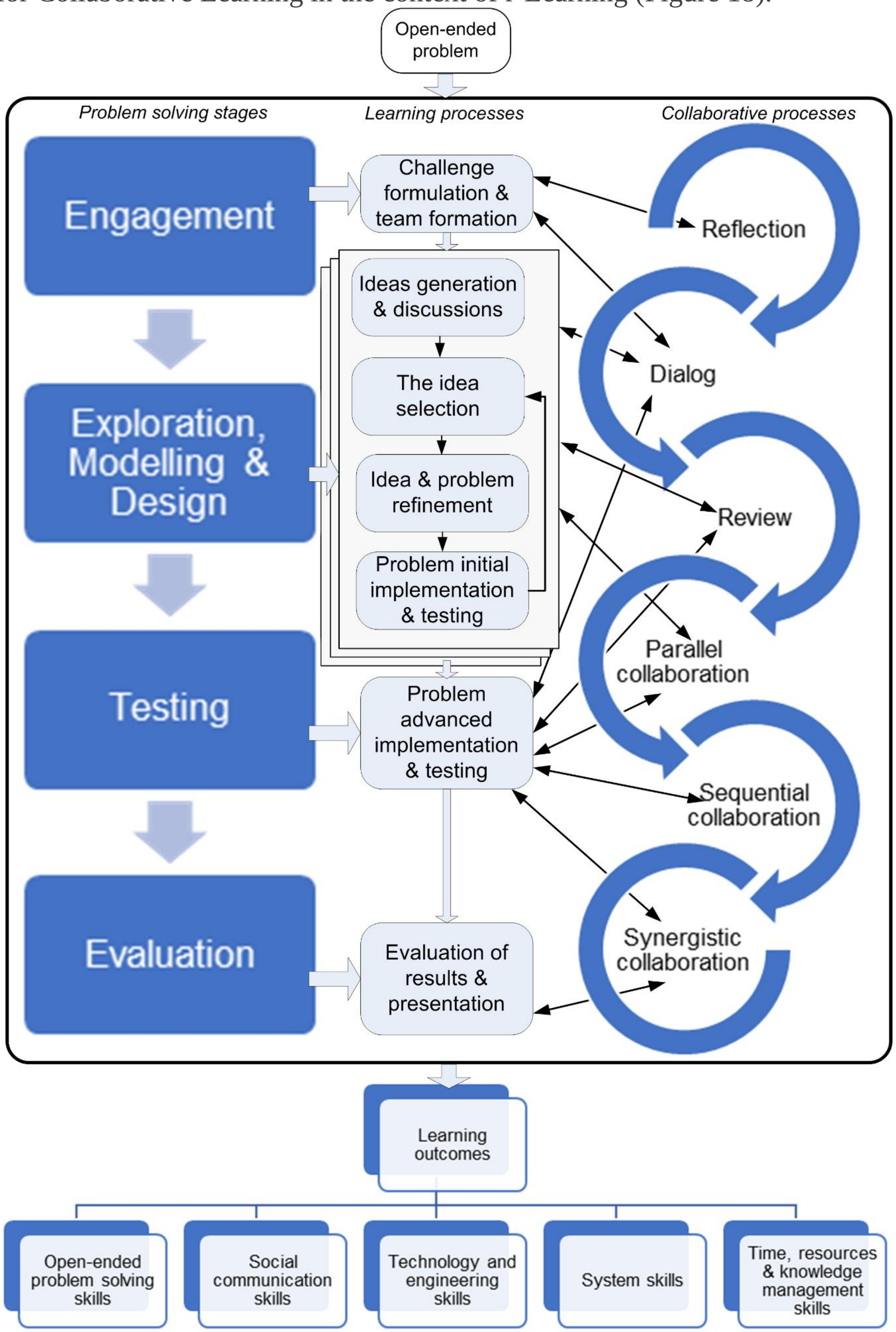


Figure 18. A Framework for investigating Collaborative Learning in the context of robot contests (Drąsutė, Burbaitė, Štuikys, & Drąsutis, 2020).

Figure 18 presents the three parts: problem-solving stages, learning process and collaborative process. We address learning processes in the middle of these sections by going through the entire cycle of a given open-ended problem.

The thicker arrows in the system denote top-level process sequencing, while the -directional arrows on the right often denote multiple feedbacks that occur during the execution of learning processes. It's worth noting that learning through complex problem-solving ifs a difficult method to predict, so we'll have to do some experimenting (see Table 4 and Table 5), which will require not only deep feedback but also incremental staging and approximation.

The actions that follow the challenge formulation and team formation (see Figure 19) at the top of the middle part). These actions include:

- The ideas generation and discussions;
- The selected idea
- The problem refinement;
- Problem initial implementation & testing (meaning the use of some simplifications, or approximation and exploration through multiple gradual trials and repetitions).

This, we conclude, is the most critical component of open-ended problem-solving. Since learners must be persuaded of the legitimacy of their steady progress toward the end goal, this is the case. We suggest that even though we only have a partial solution, advanced problem implementation and testing will help us get to the final solution.

The final stage of learning involves analyzing and representing the performance, which may vary slightly from those obtained during the planning and presentation phases of the competition. At the bottom of Figure 19, we summarize the learning outcomes. The following section goes into greater detail about the framework's implementation and features (Table 4; Table 5).

Table 4. Relationship model among STEM components and Collaboration in different stages of open-ended problem solving (OEPS) (Drąsutė, Burbaitė, Štuikys, & Drąsutis, 2020).

Stages of OEPS (Reid, 2002)	Engagement	Exploration	Modelling	Design	Testing	Evaluation
Collaboration aspects STEM sub-domains	Team building, sub-teams building	Agreements on how to solve sub-tasks in sub-teams & on how to join sub-tasks solutions in the complex task solution are made. Sub-teams prepare sub-tasks solution plans that include technical & content resources, provide learning activities to achieve initially defined goals. Sub-tasks solution plans of sub-teams & generalized task solution plans are adjusted if needed. Learning activities are executed according to tasks solution plans & in accordance with the agreements.				Evaluation of the involvement and contribution of sub-teams & team members
	Collaborative activities in CL framework					
Science	Challenge formulation, discussion about possible	Measuring physical characteristics of the components used	Modelling of mechanical movement of a robot	Corrections of the initial model	Identifying physical interactions among components	Suitability of physical characteristics of components

Computer Science	ways to solve challenges, motivating movies from previous challenges	Component's functionality programming for components physical characteristics measurement	A virtual model of robot	Robot control programs creating	Corrections of robot control programs	Suitability of robot control programs
Techno-logy		LEGO components & relevant software investigation	Selection of adequate components and software	Component list corrections, a search of alternative solutions	Use of adequate software & hardware for robot testing	Suitability of components & software used
Engineering		Simple prototypes for components physical characteristics measurement	The initial design of the robot	The final design of the robot	Corrections of the final robot	Suitability of mechanical robot design
Mathematics		Dependencies among component physical characteristics	Dependencies among physical characteristics of the initial robot	Dependencies among physical characteristics of the final robot	Sub-tasks performance accuracy measurements	Task performance accuracy measurements

Table 5. Model defining links among collaboration aspects in learning and STEM-driven Computer Science tasks.

Type of the STEM-driven CS tasks (Štuikys, & Burbaitė, 2018) Collaboration aspects	Component Testing of Smart Devices	Smart devices functionality modelling	Smart devices assembling & testing	Smart devices *use-as-is* for solving real tasks

(Cukurova, Avramides, Spikol, Luckin, & Mavrikis, 2016).				
Exploring & Understanding 1. Discovering perspectives & abilities of team members. 2. Discovering collaborative interaction to solve the problem along with goals. 3. Understanding roles in solving a problem.	Optimal team size – 2 members. Learning interaction: knowledge sharing, comments, suggestions, reflection. Learners collaborate on a pair; the teacher is a mentor.	Optimal sub-team size – 2-4 members. Depending on the complexity of the task, it can be addressed to several sub-teams. Learning interaction: knowledge sharing, comments, suggestions, reflection inside sub-team and among sub-teams. Learners collaborate in small teams & a whole-class community. The teacher works as a mentor.		
Representing & Formulating 4. Building a shared representation & negotiating the meaning of the problem. 5. Identifying & describing tasks to be completed. 6. Describing roles & team organization (communication, engagement rules).	Agreements on how to solve the problem are made. The tasks to be completed: components' parameter testing and evaluating. Components' testing and measurement results are evaluated by both team members. Dominating communication type is face-to-face.	Agreements on how to solve sub-tasks in sub-teams & on how to join sub-tasks solutions in the complex task solution are made. Sub-teams prepare sub-tasks solution plans that include technical & content resources, provide learning activities to achieve initially defined goals. Dominate different types of communication, such as face-to-face, online synchronous & asynchronous.		
Planning & Executing 1. Communicating with team members about the actions to be performed. 2. Enacting plans. 3. Following rules of engagement.	Communicating between team members & teacher, learning plan adjustment if needed. Learning activities are executed according to the plan & agreements.	Communicating between sub-team members & teachers; among sub-teams; among sub-teams & teachers. Learning plans of sub-teams & generalized task solution plans are adjusted if needed. Learning activities are executed according to learning plans & in accordance with the agreements.		
Monitoring & Reflecting 6. Monitoring & repairing the shared understanding. 7. Monitoring results of actions & evaluating success.	Monitoring of components testing & evaluating procedures. Comparison of the obtained results. Team members' roles: testing (tester),	Monitoring covers all learning processes starting with the task decomposition into sub-tasks and finishing with the obtained results evaluating. The roles (designer, constructor, programmer, tester, analyst) of sub-team members are also continuously monitored. Monitoring is performed by learners & teachers.		

8. Providing feedback & adapting roles of team members.	processing of the results (analyst).	

5. Learner's knowledge assessment model (pedagogy) is STEAM context

In the STEAM field, it is essential to determine learners' collaborative activities in order to define the obtained skills and provide effective problem-solving opportunities within instruction. In this chapter, we present the learners' assessment in STEAM context sub-models (Table 6, Table 7, Table 8).

Table 6. Revised Bloom's taxonomy: the Cognitive processes' dimension (adapted from Anderson, & Bloom, 2001).

A lower order thinking skills	Category	Cognitive processes
	Remembering – retrieving relevant knowledge from long-term memory.	Recognizing Recalling
	Understanding – determining the meaning of instructional messages, including oral, written, and graphic communication.	Interpreting Exemplifying Classifying Summarizing Inferring Comparing Explaining
	Applying – carrying out or using a procedure in a given situation.	Executing Implementing
	Analyzing – breaking material into its constituent parts and detecting how the parts relate to one another and to an overall structure or purpose.	Differentiating Organizing Attributing
Upper order thinking skills	Evaluating – making a judgment based on criteria and standards.	Checking Critiquing
	Creating – putting components together to form a novel, coherent whole or make an original product.	Generating Planning Producing

In Table 7, we present computational thinking skills derived from (Sheard, Simon, Hamilton, & Lönnberg, 2009; Pan, Polden, Larkin, Van Duin, & Norrish, 2012).

According to Psycharis (2018), many researchers in the field have proposed that CT is a universal capability, mindset, competency practice, and problem-solving strategy that affects virtually all disciplines. Such abilities are essential in the STEAM field for learners, as well it is important for teachers and can be used for problem-solving approach implementation.

The language, as well as basic information and components, make up factual knowledge. Classifications and categories, concepts and generalizations, philosophies, models, and systems are all examples of conceptual understanding. Subject-specific skills and algorithms, strategies and processes, and criteria for using suitable procedures are all described by procedural expertise.

Table 7. Computational thinking skills sub-model.

Skill	Explanation
Abstraction	It is the process while simplifying from the concrete (something complicated) to the general as solutions are developed (by leaving out irrelevant details, finding the relevant patterns, and separating ideas from tangible details).
Decomposition	It is the process of breaking down problems into smaller parts that may be more easily solved.
Generalisation/ Pattern recognition	It is transferring a problem-solving process to a wide variety of problems and allows an expansion of an existing solution in a given problem to cover more cases.
Data representation	It is any sequence of one or more symbols given meaning by specific act(s) of interpretation. It is something more fundamental than an algorithm.
Algorithm	It is a practice of writing step-by-step specific and explicit instructions for carrying out a process.

Metacognitive knowledge, described as "knowledge of one's own cognition and about oneself in relation to various subject matters" (Anderson and Bloom, 2001), is "knowledge of one's own cognition and about oneself in relation to various subject matters." The information styles mentioned vary from the concrete to the abstract. Computational Thinking (CT) is a problem-solving method that encompasses a variety of traits and dispositions - forming problems in such a way that we can use a computer and other resources to assist us in solving them (Figure 19).

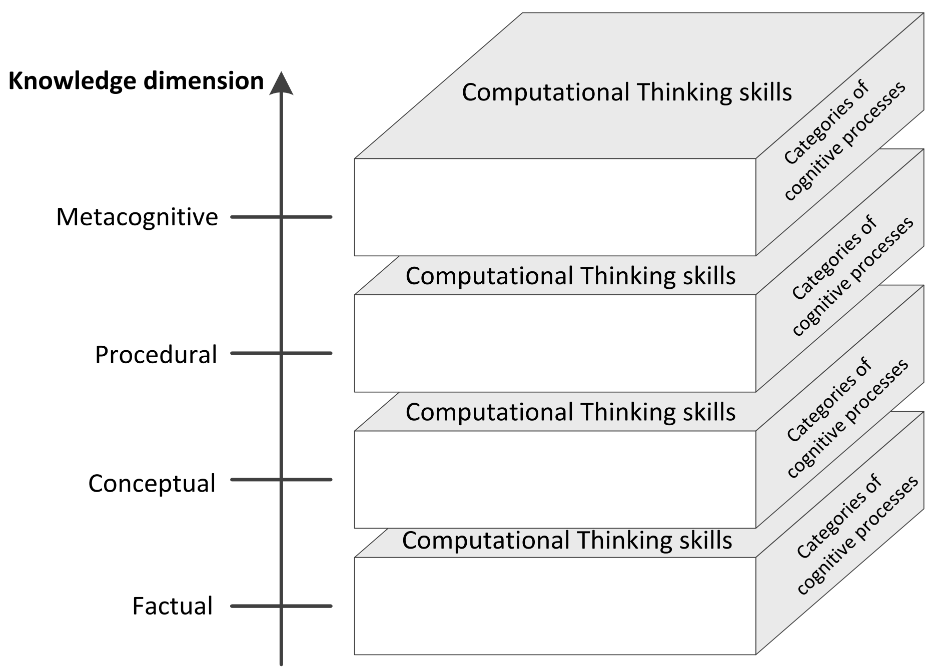


Figure 19. The connection between CT Skills, Knowledge Dimensions and Categories of Cognitive Processes.

6. Scenario's assessment methodology

The expert/ partners survey method was used to evaluate the effectiveness of the integrated model. The purpose of the assessment is to select the appropriate survey method and the appropriate qualifications of the experts involved in the survey.

The questionnaire was designed according to the Likert scale (Appendix 3).

Experts/partners assessment is a generalized opinion of a group of experts/partners describing the required skills of an expert in a certain field, an expert can be a source of qualitative information, and expert quality can be assessed as an aggregate indicator of objective and subjective status or compatibility factor:

$$k = 1 - \frac{\eta}{\eta_{max}} \qquad (1)$$

Where η is the number of conflicting assessments per expert/partner and ηmax is the maximum possible number of conflicting assessments.

The number of experts/partners was selected based on the assumptions formed in classical test theory, which state that the reliability of aggregated solutions and the number of experts/partners is related to the factor determining the effectiveness of the study.

The highest percentage of reliability is obtained with the evaluation of at least 7-10 experts/partners. More evaluators influence the percentage of reliability insignificantly. Therefore, 10 experts/partners were invited to evaluate the scenarios.

The questionnaire was to be validated before submission to experts/partners. The validity of the questionnaire was evaluated by experts: whether the questionnaire is representative and will measure what was intended to be measured. The purpose of the questionnaire is to gather the necessary information to obtain scientific conclusions. Validity shows whether what should be measured is really being measured.

Conclusions

In this study, we presented the training framework for Teachers' Professional Development in the STEAM field. The document consists of six parts: 1) Frameworks for Teachers' Professional Development in STEAM field: An Overview; 2) Pedagogical aspects of STEAM education; 3) Learning scenario in STEAM field; 4) Collaborative learning model in a STEAM environment; 5) Learners knowledge assessment model (pedagogy) is STEAM context; 6) Scenarios assessment methodology.

According to Rabalais (2014), the STEAM's approach is to investigate the connection between the impact on the arts and success in Science, Technology, Engineering, and Math (STEM). The STEAM-oriented atmosphere in school teaching is a method of encouraging learners to engage in educational projects involving science, technology, engineering, art, and mathematics (Yakman, 2008). The r-learning is the Arts ("A") component that contributes to the effectiveness of STEAM education.

R-learning can provide educationally relevant opportunities for learning about new forms of technology for learners (Tocháček, Lapeš, & Fuglík, 2016); it can help to integrate practical knowledge and theoretical knowledge for successful real life problem solutions (Damaševicius, Narbutaite, Plauska, & Blažauskas, 2017); it can promote the acquisition of transdisciplinary expertise in social and humanistic sciences (Damaševičius, Maskeliūnas, & Blažauskas, 2018); educational robots can be used as tools for motivating students to learn STEM subjects (Costa, 2014). Learners' contact with educational robots is regarded as successful participation in STEM-related skills.

We hope that presented framework for teaching content will be guidelines for teachers and will help teachers to obtain a deeper understanding of the scenarios creating stages, parts, and connections between scenarios implementation phases, learning content units, learners' assessment models and capabilities to evaluate prepared scenarios.

We hope that presented framework can be used for both synchronous and asynchronous sessions and it can be tailored to each partner country's educational system.

References

1. Anderson, L. W., & Bloom, B. S. (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Longman.
2. Barell, J. (2010). Problem-based learning: The foundation for 21st century skills. 21st century skills: Rethinking how learners learn, 175-199.
3. Boticario, J. G., & Santos, O. C. (2007). A Dynamic assistance approach to support the development and modelling of adaptive learning scenarios based on educational standards. In Fifth international workshop on authoring of adaptive and adaptable hypermedia. International conference on user modelling.
4. Boudalis, A (2012). Implementation scenarios definition and analysis. Discover the COSMOS Deliverable.
5. Burbaitė R (2014) Advanced generative learning objects in informatics education: the concept, models and implementation. Summary of doctoral dissertation, physical sciences, informatics (09P), Kaunas University of Technology, Kaunas
6. Burbaitė, R., Drąsutė, V., & Štuikys, V. (2018, April). Integration of computational thinking skills in STEM-driven computer science education. In *2018 IEEE Global Engineering Education Conference (EDUCON)* (pp. 1824-1832). IEEE.
7. Bush, S. B., Cook, K. L., Ronau, R. N., Rakes, C. R., Mohr-Schroeder, M. J., & Saderholm, J. (2016). A highly structured collaborative STEAM program: Enacting a professional development framework. *Journal of Research in STEM Education*, *2*(2), 106-125.
8. Bybee, R. W., Carlson-Powell, J., & Trowbridge, L. W. (2008). *Teaching secondary school science: Strategies for developing scientific literacy*. Columbus: Pearson/Merrill/Prentice Hall.
9. Challco, G. C., Mizoguchi, R., Bittencourt, I. I., & Isotani, S. (2015, June). Steps Towards the Gamification of Collaborative Learning Scenarios Supported by Ontologies. In International Conference on Artificial Intelligence in Education (pp. 554-557). Springer, Cham.
10. Cole, M., & Engeström, Y. (1993). A cultural-historical approach to distributed cognition. Distributed cognitions: Psychological and educational considerations, 1-46.
11. Costa, J. M. (2014, October). Using scenario-based learning with robots to increase the programming interest. In *Proceedings of the Second International Conference on Technological Ecosystems for Enhancing Multiculturality* (pp. 133-138).
12. Cukurova, M., Avramides, K., Spikol, D., Luckin, R., & Mavrikis, M. (2016, April). An analysis framework for collaborative problem-solving in practice-based learning activities: a mixed-method approach. In *Proceedings of the Sixth International Conference on Learning Analytics &* Knowledge (pp. 84-88).http://www.firstlegoleague.org/, Accessed 5 January 2020.
13. Dahlgren, M. A., & Öberg, G. (2001). Questioning to learn and learning to question: Structure and function of problem-based learning scenarios in environmental science education. Higher education, 41(3), 263-282.
14. Damaševičius, R., Maskeliūnas, R., & Blažauskas, T. (2018). Faster pedagogical framework for steam education based on educational robotics. International Journal of Engineering and Technology, 7(2.28), 138-142.
15. Damaševicius, R., Narbutaite, L., Plauska, I., & Blažauskas, T. (2017). Advances in the use of educational robots in project-based teaching. *TEM Journal*, *6*(2), 342.
16. Dewey, J. (1938). Experience and education, New York: The Macmillan company, 1938
17. Drąsutė, V, Burbaitė, R, Štuikys, V, & Drąsutis, S. (2020). Collaborative learning in stem-driven computer science education: a robot contest-based approach. INTED 2020: 14th Internatio*nal Technology, Education and Development Conference, 2-4 March, 2020, Valencia (Spain): Conference Proceedings*, 8149–8157. https://doi.org/10.21125/inted.2020.2223

18. Engestrdm, Y. (1987). Learning by expanding: An activity-theoretical approach to developmental research. Helsinki: Orienta-Konsultit.
19. Engeström, Y. (2001). Making expansive decisions: An activity-theoretical study of practitioners building collaborative medical care for children. In Decision making: Social and creative dimensions (pp. 281-301). Springer, Dordrecht.
20. Essalmi, F., Ayed, L. J. B., Jemni, M., & Graf, S. (2013, July). Automating the E-learning Personalization. In *International Workshop on Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data* (pp. 342-349). Springer, Berlin, Heidelberg.
21. Gupta, M., Bertrand, J. W., Babu, S. V., Polgreen, P., & Segre, A. M. (2012, January). An evolving multi-agent scenario generation framework for simulations in preventive medicine education. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium* (pp. 237-246).
22. Han, J. (2010, February). Robot-aided learning and r-learning services. In *Human-robot interaction* (p. 288). Rijeka: InTech. Jonassen, P., & Peck, K. L. Wilson. (1999). Learning with technology: a constructivist perspective.
23. Jonassen, P., & Peck, K. L. Wilson. (1999). Learning with technology: a constructivist perspective.
24. Jung, S. E., & Han, J. (2020). A Comprehensive Review on r-Learning: Authentic r-Learning Beyond the Fad of New Educational Technology. International journal of advanced smart convergence, 9(2), 28-37.
25. K. Cummins, Teaching Digital Technologies & STEM: Computational Thinking, coding and robotics in the classroom. Retrieved from Amazon.com, 2016.
26. Kerven, D., Nagel, K., Smith, S., Abraham, S., & Young, L. (2017, March). Scenario-based inquiry for engagement in general education computing. In Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education (pp. 303-308).
27. Koehler, M. J., & Mishra, P. (2009). What is technological pedagogical content knowledge? Contemporary Issues in Technology and Teacher Education, 9(1), 60-70.
28. Kolb, A. Y., & Kolb, D. A. (2009). The learning way: Meta-cognitive aspects of experiential learning. *Simulation & gaming*, *40*(3), 297-327.
29. Kolb, D. (1984). Experiential learning: Experience as the source of learning and development. Prentice-Hall.
30. Komis V, Romero M, Misirli A (2016) A Scenario-Based Approach for Designing Educational Robotics Activities for Co-creative Problem Solving. In International Conference EduRobotics 2016 (pp. 158-169). Springer, Cham
31. L. W. Anderson, D. R. Krathwohl, P. Airasian, K. Cruikshank, R. Mayer, P. Pintrich, et al., "A taxonomy for learning, teaching and assessing: A revision of Bloom's taxonomy," New York. Longman Publishing. Artz, AF, & Armour-Thomas, E.(1992). Development of a cognitive-metacognitive framework for protocol analysis of mathematical problem solving in small groups. Cognition and Instruction, vol. 9, pp. 137-175, 2001.
32. Lejeune, A., & Pernin, J. P. (2004). A Taxonomy for Scenario-Based Engineering. CELDA, 4, 249-256.
33. Martin, G., Schatz, S., Bowers, C., Hughes, C. E., Fowlkes, J., & Nicholson, D. (2009, October). Automatic scenario generation through procedural modeling for scenario-based training. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 53, No. 26, pp. 1949-1953). Sage CA: Los Angeles, CA: SAGE Publications.
34. Nadolski, R. J., Hummel, H. G., Slootmaker, A., & Van der Vegt, W. (2012). Architectures for developing multi user, immersive learning scenarios. Simulation & Gaming, 43(6), 825-852.
35. National Research Council (2012) Education for Life and Work: Developing Transferable Knowledge and Skills in the 21st Century. Washington, DC: The National Academies Press

36. Pan, Z., Polden, J., Larkin, N., Van Duin, S., & Norrish, J. (2012). Recent progress on programming methods for industrial robots. *Robotics and Computer-Integrated Manufacturing*, *28*(2), 87-94.
37. Plauska, I., & Damaševičius, R. (2014, October). Educational robots for internet-of-things supported collaborative learning. In International Conference on Information and Software Technologies (pp. 346-358). Springer, Cham.
38. Psycharis, S. (2018). STEAM in education: A literature review on the role of computational thinking, engineering epistemology and computational science. computational steam pedagogy (CSP). Scientific Culture, 4(2), 51-72.
39. Rabalais, M. E. (2014). *STEAM: A national study of the integration of the arts into STEM instruction and its impact on learner achievement*. University of Louisiana at Lafayette.
40. Rakes, C. R., Bush, S. B., Ronau, R. N., Mohr-Schroeder, M., & Saderholm, J. (2017). Making teacher PD effective using the PrimeD framework. New England Mathematics Journal, XLX, 52-62.
41. Redecker, C. (2017). European framework for the digital competence of educators: DigCompEdu (No. JRC107466). Joint Research Centre (Seville site).
42. Reid, J. A. (2002). Managing Small-Group Learning. Primary English Teaching Association, PO Box 3106, Marrickville NSW 2204, Australia (Stock No: pet071: $27 Non Members; $20 Members).
43. Rius, À., Sicilia, M. A., & García-Barriocanal, E. (2009) An ontology to automate learning scenarios? An approach to its knowledge domain. International Journal of Doctoral Studies, 4, 151-165
44. S. Atmatzidou and S. Demetriadis, "Advancing learners' computational thinking skills through educational robotics: A study on age and gender relevant differences," Robotics and Autonomous Systems, vol. 75, pp. 661-670, 2016.
45. Saderholm, J., Ronau, R. N., Rakes, C. R., Bush, S. B., & Mohr-Schroeder, M. (2017). The critical role of a wellarticulated conceptual framework to guide professional development: An evaluation of a state-wide two-week program for mathematics and science teachers. Professional Development in Education, 43, 789-818. doi: 10.1080/19415257.2016.1251485
46. Selby, C., & Woollard, J. (2013). Computational thinking: the developing definition.
47. Sheard, J., Simon, S., Hamilton, M., & Lönnberg, J. (2009, August). Analysis of research into the teaching and learning of programming. In *Proceedings of the fifth international workshop on Computing education research workshop* (pp. 93-104)
48. Sigrén, P. (2003). Open distance learning in higher educational systems: A technological approach within a social constructivist perspective. In *International Conference on Education and Information Systems-Technologies and Applications (EISTA 03), Orlando, FL, JUL 31-AUG 02, 2003*. INT INST INFORMATICS & SYSTEMICS, 14269 LORD BARCLAY DR, ORLANDO, FL 32837 USA.
49. Štuikys V (2015) Smart Learning Objects for Smart Education in Computer Science. Springer, New York
50. Štuikys, V. (2015). Smart Learning Objects for Smart Education in Computer Science: Springer.
51. Štuikys, V., & Burbaitė, R. (2018). *Smart STEM-Driven Computer Science Education: Theory, Methodology and Robot-based Practices*. Springer.
52. Štuikys, V., Burbaite, R., Blažauskas, T., Barisas, D., & Binkis, M. (2017). Model for introducing STEM into high school computer science education. *The International journal of engineering education*, *33*(5), 1684-1698.

53. Tocháček, D., Lapeš, J., & Fuglík, V. (2016). Developing technological knowledge and programming skills of secondary schools students through the educational robotics projects. *Procedia-Social and Behavioral Sciences*, *217*, 377-381.
54. Vantroys, T., & Peter, Y. (2003, September). COW, a flexible platform for the enactment of learning scenarios. In International Conference on Collaboration and Technology (pp. 168-182). Springer, Berlin, Heidelberg.
55. Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., &Wilensky, U. (2016). Defining Compu-tational Thinking for Mathematics and Science Classrooms. Journal of Science Education and Technol-ogy, 25(1), 127-147. DOI: 10.1007/s10956-015-9581-5
56. Yakman, G. (2008). ST ${\Sigma} $@ M Education: an overview of creating a model of integrative education, PATT.
57. Zook, A., Lee-Urban, S., Riedl, M. O., Holden, H. K., Sottilare, R. A., & Brawner, K. W. (2012, May). Automated scenario generation: toward tailored and optimized military training in virtual environments. In Proceedings of the international conference on the foundations of digital games (pp. 164-171).

Appendix 1

Scenario description: HEADER

Topic	Chatbots
Grade	7-8
Activity type	Individual work, group work
Integration	IT; Mathematics; Physics; Chemistry; Biology; Economics; Arts; Languages; History; Geography
Keywords	Chatbot; artificial intelligence; cloud technologies; queries
Annotation	Learners will build a rule-based chatbot from scratch using the Scratch visual programming language, applying IT, other subjects' knowledge and skills in real situations. Learners are provided with a description of the development of a simple rules-based robot, which learners will then use to create, test, and evaluate the functionality of their own chat robots. Learners will learn how to create conversational artificial intelligence-based chatbots, apply IT, native language, mathematics knowledge and skills, and acquire the basics of economics and entrepreneurship using specialized online chat robot development tools.
Aim	To get acquainted with the possibilities provided by chat robots, to learn how to create chat robots, to test them.
Tasks	1. Analyse the scope of chatbots and agree on criteria for assessing the quality of chatbots. 2. Find out the basic principles of developing rules-based chat robots, create chatbots for specific purposes, test them, evaluate the quality and possibilities of use. 3. Find out the basic principles of creating chat robots based on artificial intelligence, formulate a problem that would help to solve chat robots, choose appropriate tools, create chatbots for specific purposes, test them, evaluate their quality and usability. 4. Summarize and present the obtained results.
Expected results	Will be able to explain the scope and influence of chat robots. Will create chat robots based on rules and chat artificial intelligence to solve specific tasks. Will test the developed chat robots and evaluate their application possibilities. Will summarize and present the obtained results.

	** Team game "Let's talk". Teams come up, or choose a topic, creates and present chat robots according to the theme.*
Tools	Raspberry Pi microcomputer, Scratch visual programming language, specialized chatbots platforms online, such as: https://surveybot.io/features/surveys; https://snatchbot.me/; https://www.engati.com/; https://mobilemonkey.com/; https://telegram.org/; http://meokay.com/; https://flowxo.com/; https://botkit.ai/; https://dialogflow.com/; https://botsify.com/; https://chatfuel.com/; https://manychat.com/; https://wit.ai/
Learning activities	Research and analysis of the applications of chat robots; development (programming) of chat robots from scratch and using specialized chat robot development tools; testing and evaluation of developed robots, research of application possibilities.
Previous knowledge	Manage the basic tools of the Scratch program. Investigate the properties of objects, change them. Explore how objects change with the simplest commands. Write the actions of the procedure and apply them to different situations in the projects. Create an animated drawing, plan, and prepare an animated project using computer software and hardware. Use digital tutorials. Use computer and information technology terms correctly, describe the basic concepts.

Appendix 2

Scenario: MAIN PART

3 STAGE ◷ 25 min
DEVELOPING A RULES-BASED CALL ROBOT IN A SCRATCH ENVIRONMENT

By consistently following the steps below, we will create a chatbot named Ladybug. Ladybug will meet and talk to you.
When creating a chatbot, you can choose another character, background, questions.

1. Open the Scratch environment, remove the default, and select the new character Ladybug (Figure 1).

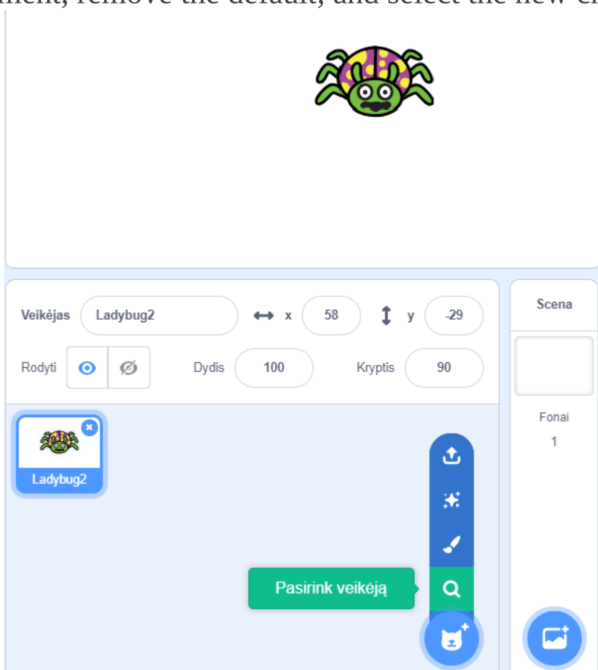


Figure 1. Choice of a new actor

2. Select a background (Figure 2).

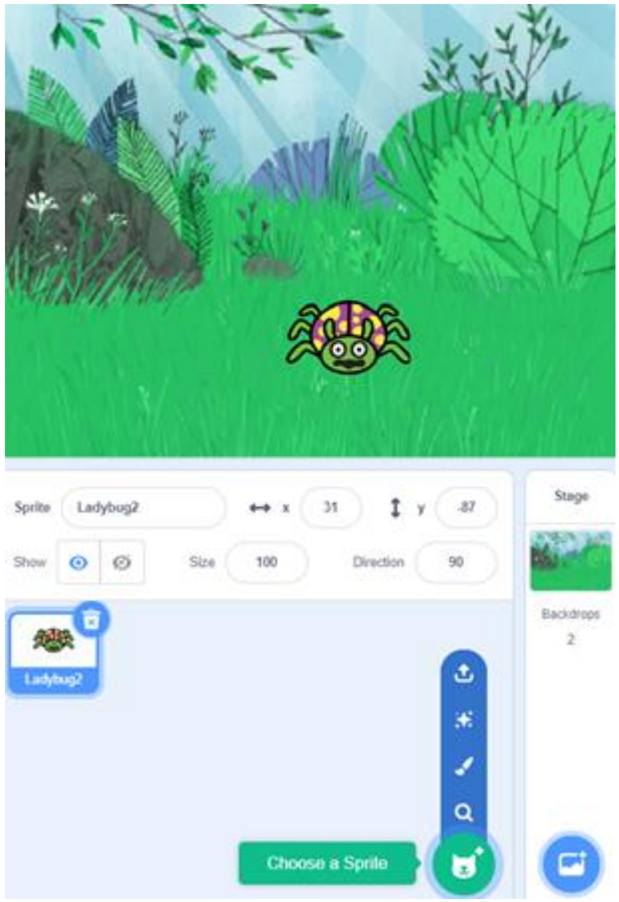


Figure 2. Background selection

3.Click on Ladybug and construct the Ladybug presentation and the first question (Figure 3).

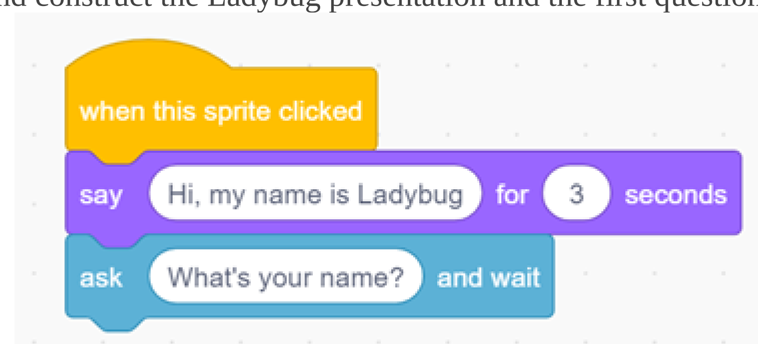


Figure 3. The introduction of the borage and the first question

4. Test the code: click  and then click on Ladybug. You should see Ladybug's introduction for the first 3 seconds (Figure. 4 a), after which Ladybug will ask you a question (Figure. 4 b).

a b

Figure 4. Introduction of the presentation (a) and the first question (b)

5. Create a Ladybug answer that mentions the name you entered. Because Ladybug will interact with people with different names, create a variable name (Figure 5).

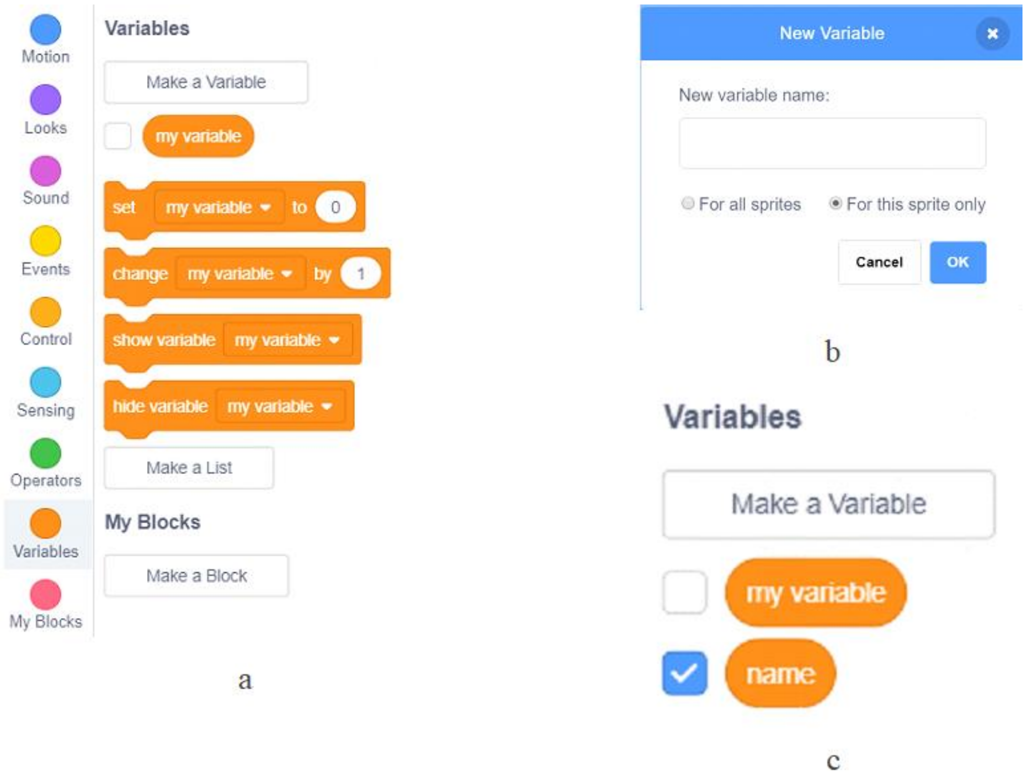
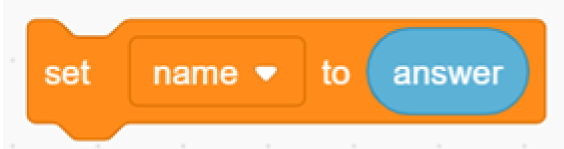


Figure 5. Creating a variable name: a - create a variable; b - enter the name of the variable and OK; c - the created variable appears in the list of variables

6. Add new components to the code - Ladybug's answer (Figure 6).

In a sentence 

the variable response is generated automatically. You will find it in the group 

.

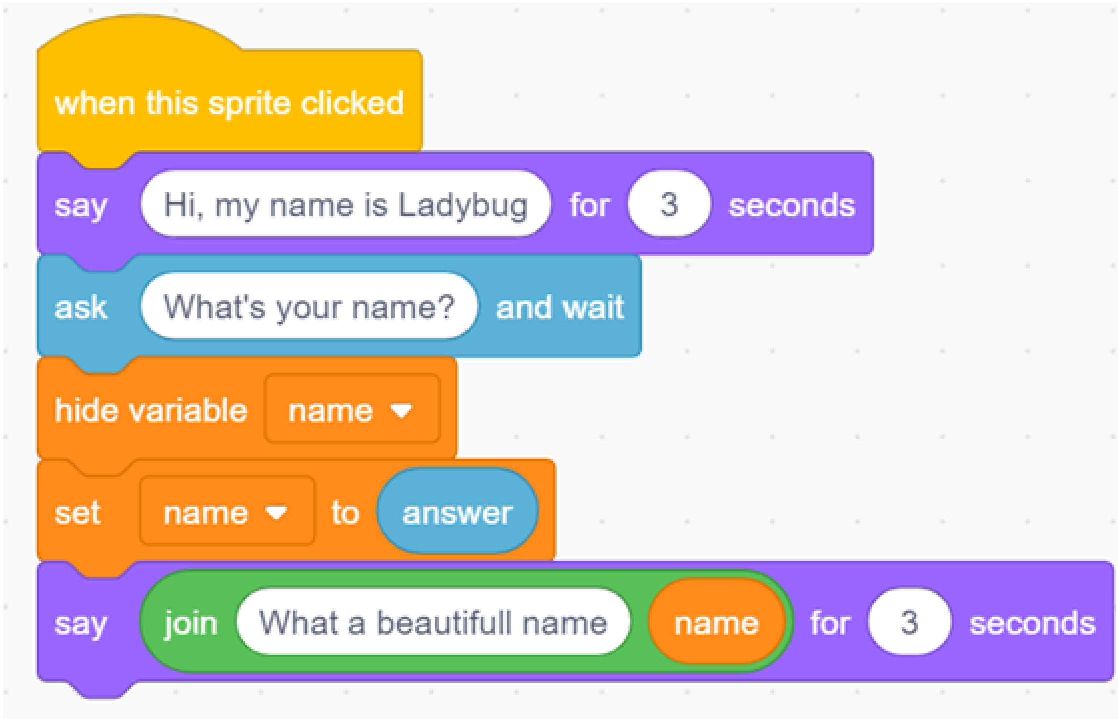


Figure 6. Ladybug's answer

7. Test Ladybug's answer. The test procedure is described in step 4. If you did everything right, you should see an image similar to Figure 7.

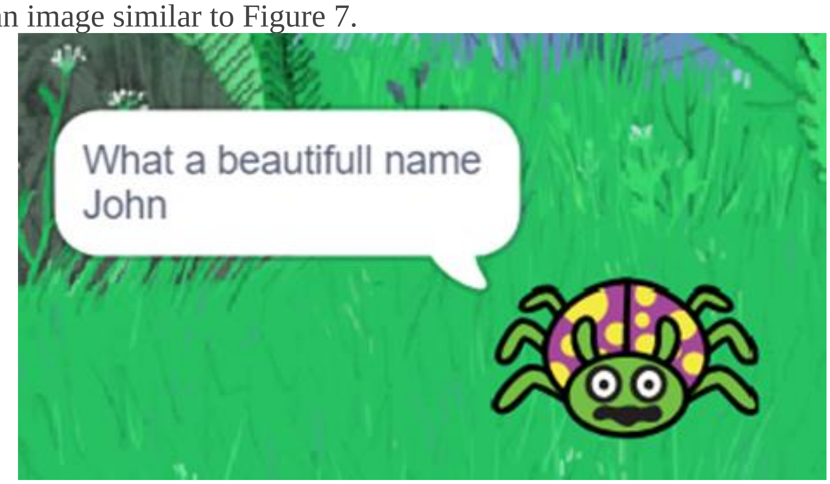


Figure 7. Ladybug response testing

1 self-study task

- Upgrade the chatbot you are developing to ask:
  1. Where do you live?
  2. What do you like to do in your free time?
- Formulate the answers to the questions as you see fit. If it is difficult to do it on your own, you can use the advice: formulate the answer to the first question as follows: "I have not been in" + the value of the location variable.
- Formulate the answer to the second question as follows: "I also like to"+ leisure activity variable value + ", as well.".

- Test the completed task.

---

9. Now, create a question to answer Yes or No. Depending on the answer, Ladybug will provide her comment. Let Ladybug ask: “Do you like to read?”. If the answer is Yes, then Ladybug will answer “Great”, and if No - “Sorry. I could offer an interesting book.” The code snippet is shown in Figure

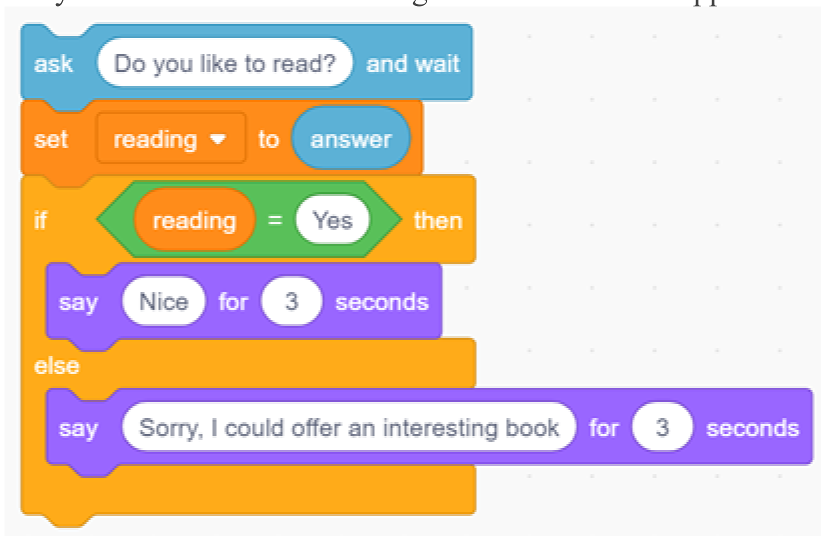


Figure 8. Different comments on the question

9. Test the generated code (Figure 9).

Figure 9. Testing

2 independent work tasks

• Supplement the chatbot being developed so that it asks questions to which the answers would be Yes or No, and depending on the answer, Ladybug will provide different comments. Formulate the questions and answers as you see fit, but if you find it difficult to do it on your own, you can get help.
• You can check that you have completed the task correctly.

• Help: Do you like to solve crossword puzzles? If the answer is yes, then the comment would be, "Great, we'll have time to decide." If the answer is no, then the comment would be, "Oh, I like it."
• Test the completed task.

Additional self-employment tasks for those who are doing very well

- Create additional questions that answer Yes or No, and the robot will ask additional questions related to the answer.
- Help: Do you like sports? If the answer is Yes, then the robot could ask, "What sport do you like the most?" And then comment, "Good choice." If the answer is No, then the robot could comment: "Sorry. Exercise is healthy."
- Help: Do you understand math well? If the answer is no, then the robot could ask, "What's the hardest thing about learning math for you?" And then comment, "Practice solving problems and everything will be fine." If the answer is Yes, then the robot could comment: "Great. Mathematics is a much-needed science."
- "Teach" the robot to treat uppercase and lowercase letters alike, i.e., the answers Yes and No, yes and no, YES and NO, the robot should understand.

Result that could be obtained by properly completing self-study task 1

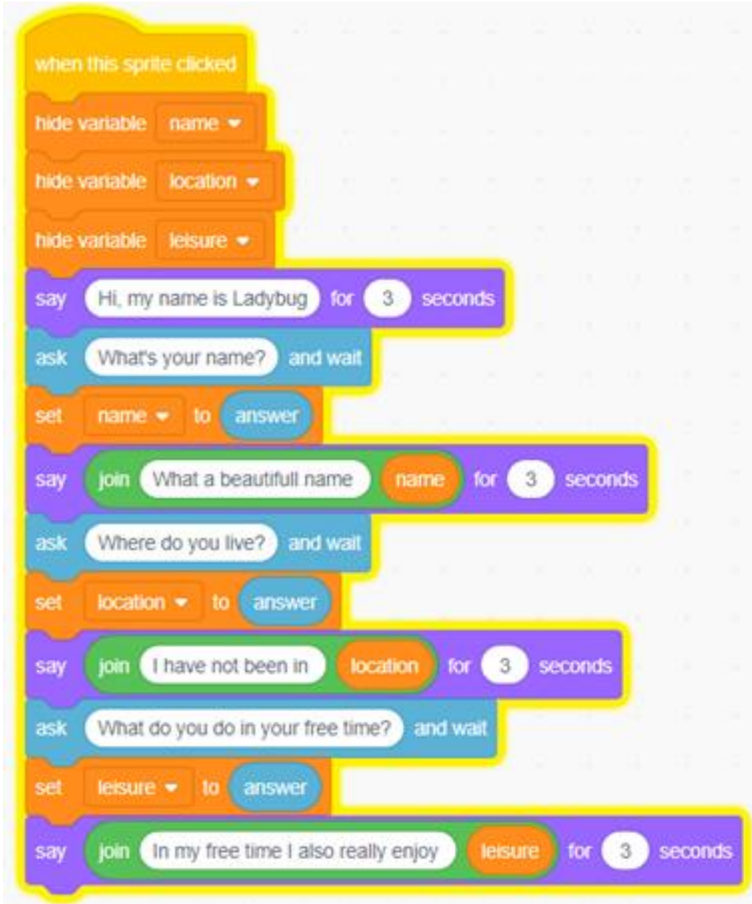


Figure 10. Result of 1 independent work task

The result that could be obtained by properly completing self-study task 2

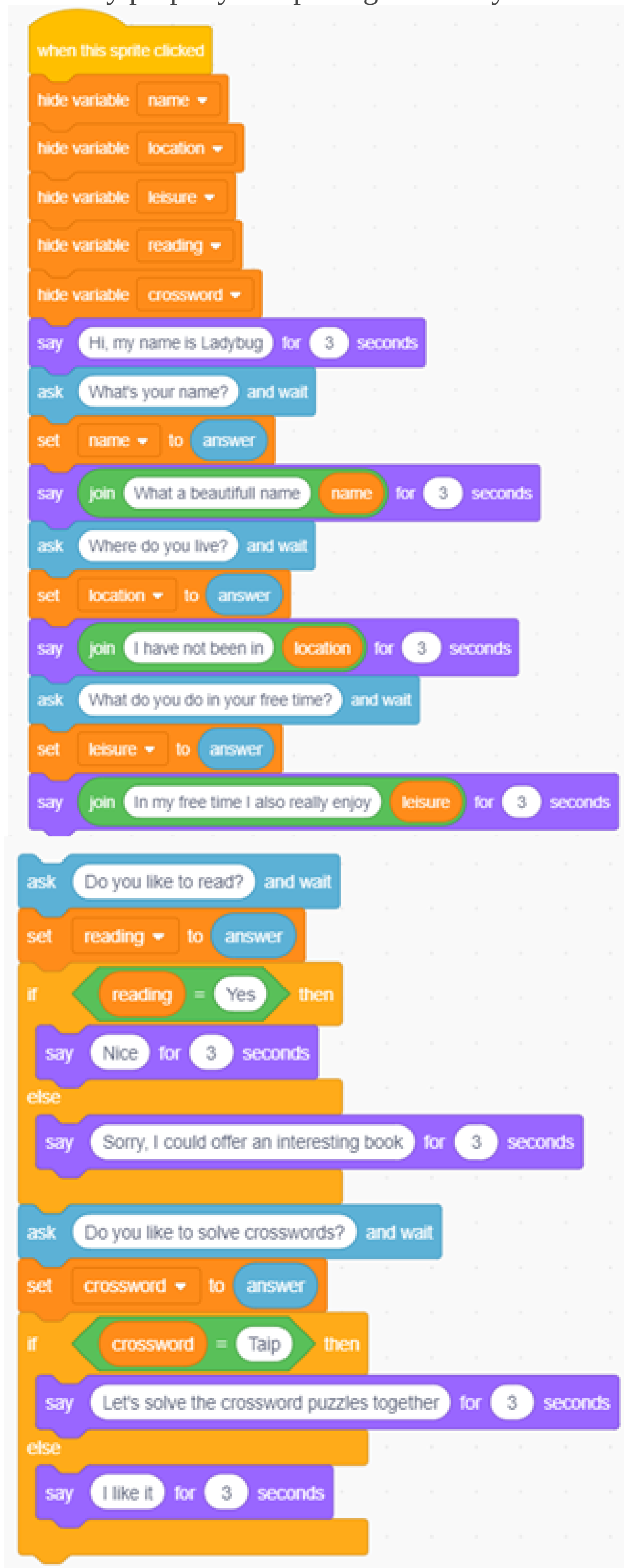


Figure 11. Result of independent work task 2

Appendix 3

LEARNING SCENARIO EVALUATION QUESTIONNAIRE

Learning Scenario ID: ____

About the completion of the presented learning scenario for general usability, please fill in the following survey so the research team can improve the user experience according to the received feedback. Such feedback will be used to create an action plan to enhance the STEAM *in-service* training platform aiming at the creation of an international teaching tool to spread the STEAM competences with a larger number of learners.
The obtained results regarding this survey will be shared with you in the near future as well as the reviewed and improved training platform, so please, share your electronics contact (E-mail), so we can deliver you this information.
Please take some time to analyze and answer the following questions, evaluating your user experience. For the rating ones, please choose and highlight a number between 1 (Strongly Disagree) and 7 (Strongly Agree) as it best matches your opinion. In the end, there is a *white space* where you can share with us your comments, suggestions, or even notes on the presented Learning Scenario.
The EduSimSTEAM team grants no data will be shared with your personal information and the privacy policy will be respected.

Name:		Age:
Email:		Gender: M☐ F☐
Province:	Sub-Province:	Country
School type*:	Date: / / (DD/MM/YYYY)	

*School type: Kindergarten, Primary School, Secondary School, High School or Vocational High School

1. The execution of the learning scenario is intuitive and can be performed with ease.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
2. The time required to perform this training is reasonable (not too short nor too long).								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
3. The provided support information (online help, messages, documentation) is adapted to the needs of the learning scenario.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
4. The presented learning scenario is focused on STEAM competences.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
5. This learning scenario appealed to robotics and algorithmic thinking.								

STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
6. The provided robotics resources were enough to achieve a good level of understanding and working with such devices and to improve algorithmic and problem-solving thinking.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
7. The lack of programming skills is an obstacle to the presented learning scenario.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
8. This learning scenario improves robotics literacy.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
9. This learning scenario induces skills of algorithmic thinking.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
10. This learning scenario is capable of developing skills of problem-solving.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
11. The tools provided are appropriate to the needs of the platform.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
12. The overall environment is suitable to develop STEAM competencies.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
13. There are enough resources on this scenario to create an enjoyable and teachable experience.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
14. By following the presented learning scenario, one can expect significant learning outcomes.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
15. The experience was pedagogically oriented.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
16. This learning scenario allows effective digital assessment.								
STRONGLY								**STRONGLY**

DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**AGREE**
17. Community learning can be achieved with this platform/scenario.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
18. The contents are appropriate to the learners' needs.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
19. This platform is innovative.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
20. Creativity is one of the outcomes of the learning experience.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
21. A good learning environment is achieved through the presented medium.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
22. This learning scenario allows the usage of varied teaching strategies.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
23. By starting a learning experience with this platform, one can determine the learners' needs.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
24. Communication is encouraged in the presented scenario.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
25. This learning scenario promotes collaboration.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
26. By using this learning scenario, teachers have an effective medium of communication with the learners.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**
27. This learning scenario promotes collaboration skills in groups.								
STRONGLY DISAGREE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**STRONGLY AGREE**

28. This learning scenario encourages project development. **STRONGLY DISAGREE 1 2 3 4 5 6 7 STRONGLY AGREE**
29. Technical knowledge is developed by using this learning scenario. **STRONGLY DISAGREE 1 2 3 4 5 6 7 STRONGLY AGREE**
30. This learning scenario can be a new educational method. **STRONGLY DISAGREE 1 2 3 4 5 6 7 STRONGLY AGREE**
31. Learners that experience such a learning environment are more likely to create new and innovative content. **STRONGLY DISAGREE 1 2 3 4 5 6 7 STRONGLY AGREE**
32. As a general overview of the presented learning scenario, how good do you think the platform is? **REALLY BAD 1 2 3 4 5 6 7 REALLY GOOD**
Define the *user experience* in one word/small sentence:
Suggestions/ comments/ notes: