A TOOL FOR COMPUTER-SUPPORTED DEVELOPMENT OF LEARNING SCENARIOS

Marina Zunic[1], Martina Holenko Dlab[2]

[1]University of Rijeka, Department of Informatics (CROATIA)
[2] University of Rijeka, Department of Informatics (CROATIA)

Abstract

In the contemporary education, teachers need ICT and digital skills to enhance the teaching and learning processes. This includes familiarity with the various digital tools they could use to carry out learning activities. Despite a large number of available tools for education, the new ones are constantly being developed so it can be challenging for teachers to choose the tools that best suit their needs. To support teachers in the process of planning learning activities, including the decision on which tools to use, tools for designing learning scenarios are being developed. Such specialized tools help the teachers to plan the learning process for different school subjects starting from stating the learning objectives and selecting teaching methods and strategies. The learning scenario also includes a description of all planned learning activities with the list of needed learning resources, tools, and equipment.

The advantage of tools for computer-supported development of learning scenarios is that they have the potential to guide teachers in the process of designing learning activities that will enable achievement of learning objectives of a particular school subject as well as the development of generic skills like collaboration, creative thinking, or computational thinking skills. The analysis of existing tools showed that their interfaces and features are not always suited to teachers that have only basic ICT skills, such as primary school teachers, but require advanced knowledge and skills.

This paper presents ongoing research with the aim to develop an easy to use digital tool for supporting junior grade primary school teachers in the process of planning learning activities. A prototype of a web tool that will guide teachers in the process of creating innovative learning scenarios for different school subjects is proposed.

It is expected that the proposed tool will help teachers to choose appropriate activities and tools for their students, and at the same time encourage them to develop their digital skills related to the use of digital tools and educational games.

Keywords: Learning design, learning scenario, game based learning, computational thinking

1   INTRODUCTION

The process of designing innovative learning scenarios which includes the choice of appropriate teaching methods and strategies as well as tools and equipment can often be very challenging for teachers. Digital tools that are being developed include different features that are in line with the contemporary approaches to teaching and learning, and teachers might find it difficult to keep up with the upcoming novel technology. Also, there are many teaching methods and strategies that can be used, so teachers need digital pedagogical competences to choose the most effective one for their students to design innovative learning activities.

By using tools that enable the creation of learning scenarios, teachers can plan the teaching process in a more efficient way. It is important to distinguish the concepts *learning activity* and *learning scenario*. A learning scenario is an instructional design model that is created for a specific subject and a specific situation. It describes activities and required resources and tools for involved participants (teachers, learners, and others) as well as the underlying pedagogical approach [1]. Learning activities are an integral part of a learning scenario, and one learning scenario may contain more than one learning activity. Each learning activity includes a task for students that is directed towards achieving learning outcomes [2].

Since contemporary teaching aims to actively involve students in the learning process, the design of the learning activities should be innovative to motivate students for active participation and engagement. Learning activities should also foster the development of generic competencies such as

computational thinking. Computational thinking is a problem-solving process that includes skills such as decomposition, abstraction, algorithms, debugging, iteration, and generalization. Using these skills, students are taught to break the problem into smaller parts, to logically organize and analyze data, and to identify and apply solutions to the problem [3]. These skills are useful for students during their education as well as for their future employment. Therefore, the development of computational thinking should be encouraged from an early age [4], and this can be achieved by using game-based learning activities. Primary school students find activities that involve playing or designing games very interesting, so it is more than desirable to include these activities in learning scenarios at this level of education [5]. The types of games that are most commonly used to encourage computational thinking skills are memory, mazes, and board games [6].

This paper presents ongoing research on how to create easy to use tool for computer-supported development of learning scenarios. This tool will support junior grade primary school teachers in the creation of learning scenarios through elaborate templates that will consist of general information and learning activities to develop computational thinking skills.

2   METHODOLOGY

The ADDIE model is chosen to develop the tool for creating learning scenarios because it is considered to be one of the most effective and commonly used models of instructional design. In line with the ADDIE model (Fig. 1), the tool will be developed through five phases - Analysis, Design, Development, Implementation, and Evaluation [7].
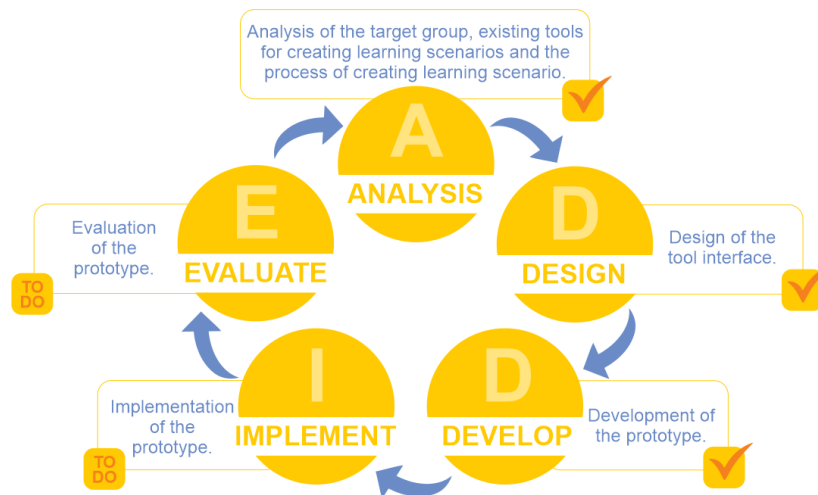

Figure 1. Process of the tool development

The goal of the first phase is to analyze the target group that will use this tool, specify requirements, and identify tool functionalities according to the needs of the target group. In this phase, the process of creating a learning scenario and existing tools for creating learning scenarios are analyzed to determine how to compensate their shortcomings with a new tool.

In the second phase, the tool is designed according to the identified needs determined through analysis. Entity-relationship diagram is designed to display the data structure that will be implemented in a database, and user interface is designed in order to create a simple and intuitive interface for users.

When all tool elements are designed in the previous phase, the third phase in which the development of a tool takes place is started. The goal of this phase is to create a prototype that will include all the specified functionalities.

In the phase of implementation, the prototype that has been created should be tested in a real environment by the group of teachers. The last phase of a model includes evaluation where feedback provided by the teachers informs the development of a new, improved version of the tool.

By using design-based research [8], the tool for creating learning scenarios will be developed in two cycles. In the first cycle, a simpler version of the tool will be developed, while in the second cycle some new functionalities will be added such as providing recommendations and enabled social networking for better communication within the tool.

3 RESULTS

So far in the research, the first three phases of the ADDIE model have been completed, and the implementation and evaluation will soon follow.

3.1 Analysis

Through analysis of existing systems with the same purpose of creating learning scenarios, it is concluded that existing tools are not suitable for teachers with a lack of advanced ICT skills or knowledge. Also, most analyzed tools have certain limitations, such as very simple templates for creating learning scenarios that does not cover all teaching process parts or they offer a small collection of recommended digital tools to include in the teaching process.

According to the findings, it is important to develop a tool that will be easy to use. Besides the need for a user-friendly interface, the process of creating learning scenarios should also be better supported.

A tool to support the teachers in process of creating learning scenarios should be aligned with the steps included in that process. Those are [9], [10], [11]:

- Define learners – The first step is to decide on the target group of students so that created learning scenarios can be relevant, engaging, and effective.

- Analyze the curriculum – By analyzing the curriculum, teachers are choosing content to teach or skills to develop.

- Define learning outcomes – It is important to define learning outcomes because they represent learning goals that should be achieved with the help of a created learning scenario. Learning outcomes are the objectives (one or more) that express what a student needs to know, understand, and/or is going to be able to show after completing a learning process.

- Design learning scenario – Design of the learning scenario starts with the specification of the general information about the context and the learning process. Besides learning outcomes that are mentioned before, grade, teaching subject, duration, correlation, and interdisciplinarity are also included in the general information of the learning scenario.

- Plan learning activities – After specifying general information about the learning scenario, defined learning activities are planned. Learning activities should be carefully aligned with the learning outcomes. Within one learning scenario there may be one or more activities performed one after the other or in parallel (Fig 2.).
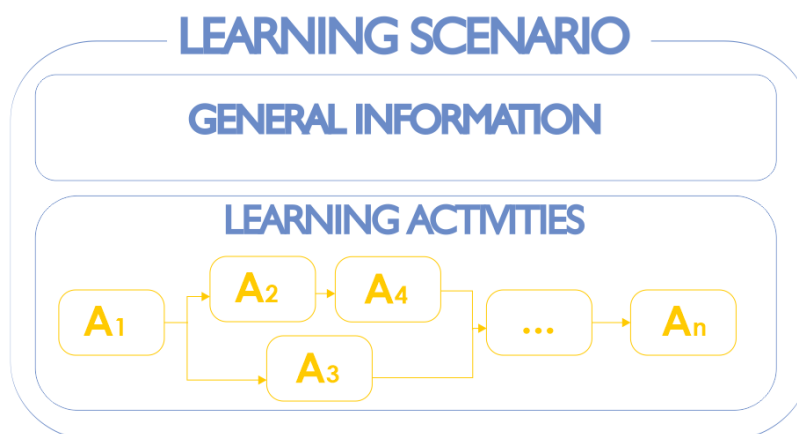

Figure 2. Main elements of a learning scenario

Every learning activity included in the learning scenario should be categorized in one of the three types – motivation activities, implementation activities, or reflection and evaluation activities [2]. By using these activity types teachers are encouraged to plan motivation at the beginning of the learning scenario, as well as reflection and evaluation at the end of the learning scenario. Motivation is a crucial factor in determining students' learning activity and achievement, and by including it the quality of the teaching process is increased, students become more involved in it and they achieve better

results [12]. All mentioned benefits also apply to include reflection and evaluation at the end of a learning activity, as it also contributes to better knowledge acquisition [13]. The second activity type, implementation, usually denotes activities that take place in the middle of a teaching process where students acquire new knowledge.

Figure 3. Desirable sequence of learning activities types in a learning scenario

Regarding the requirements for the tool that will be developed, the target group was determined first. The target group is junior primary school teachers. Besides the teachers, the tool will be used by administrators. The task of administrators is to manage user accounts, ensure security and efficiency of the tool, provide technical support to users, but also to review learning scenarios in order to indicate high-quality learning scenarios with the label „Example of good practice". This label can only be assigned by an administrator, and it helps teachers to distinguish learning scenarios that can serve them as examples of good practice.

Within the tool, teachers should be able to view their own and other publicly available learning scenarios. Every learning scenario will be displayed with all general information along with the visualization of all included learning activities. A search engine will help the teachers in finding learning scenarios based on one or more criteria such as keywords, teaching subject, grade, date of publication, the label „Example of good practice", and popularity (number of views and uses of a particular learning scenario). This way they can find an existing scenario to use it as a template or they can remix individual activities from different scenarios to create a new learning scenario according to their needs. With this possibility, teachers have a chance to get inspiration from the work of their colleagues and they can implement examples of good practice in their own classroom. Scenarios that are developed based on the existing ones should contain information about the original learning scenarios/activities and their authors (Acknowledgment).

3.2   Design of the tool

The purpose of the tool is to support the planning of the teaching process. The tool is designed to guide teachers through defining all of the essential elements of a learning scenario in a few steps. By using this tool, teachers should have the possibility to design high-quality learning scenarios in an efficient way.

Design of the tool first includes database design which is achieved by creating entity-relationship diagram. Entity-relationship diagram represents data or entity types and the relation between them. A relationship between entities is represented with a connection of one entity's primary key to another entity's foreign key [14]. Diagram consists of two main entities, *Learning scenario* and *Learning activity*. The *Learning scenario* entity describes the learning scenario through belonging attributes and is further described by the connected entities *User*, *Keyword*, *Learning outcome*, *Learning outcome (CT)*, *Learning scenario type*, and *Teaching Subject,* an entity that is double connected to the *Learning scenario*, as a teaching subject of a learning scenario as well as correlation and interdisciplinarity. The *Learning activity* entity is also described with belonging attributes and is further described with the connected entities *Learning activity type*, *Strategy and method*, *Teaching aid*, *Collaboration*, and *Performance.*

The second part of a tool design is the design of a simple interface for users without advanced ICT knowledge. The interface includes a display of a user profile and a section related to the development of learning scenarios. By selecting the correct components of a user interface an attempt was made to create an acceptable and comprehensible interface for users. During the design, it was planned to use basic elements such as input elements (select menus, radio buttons, checkboxes, text fields, and buttons), informational components (icons, notifications, and message boxes), and navigation that would include search field.

Tab. 1 shows all data elements related to the user profile of a tool. The table is divided into two parts, first part contains data that is required to enter when a new user registers, and the second part contains data that, in addition to the data from the first part (besides user password), is displayed when viewing the profile of an existing user.

Table 1. Data elements related to the users

No.	Label	Description	Control type	Type and length
Required data input to register a new user				
1.	First name	User first name	Text input	nvarchar (50)
2.	Last name	User last name	Text input	nvarchar (50)
3.	Username	Username	Text input	nvarchar (20)
4.	Email	User email	Text input	nvarchar (50)
5.	Password	User password	Text input	nvarchar (20)
Additional data retrieved from database and displayed when viewing an existing user profile				
6.	Registration date	User registration date (the current date is saved)	-	datetime
7.	User type	Admin or user (user is default type value)	-	bit
8.	Profile picture	User picture (optional)	Image upload	nvarchar (100)

Once the user registers, they can create a learning scenario. When creating a learning scenario from the scratch, users should enter the required data in three steps. In the first step general information about the learning scenario (Tab. 2, elements 1 to 5), in the second step learning outcomes (Tab. 2, elements 6 and 7), and in the last step keywords, correlation and interdisciplinarity are entered (Tab. 2, elements 8 and 9). Some additional information related to the scenario should be displayed when the users browse through existing scenarios (Tab. 2, element 10-16).

Table 2. Data elements related to the learning scenarios

No.	Label	Description	Control type	Type and length
Required data input to create a new learning scenario				
1.	Name	Name of the learning scenario	Text input	nvarchar (100)
2.	Description	Description of the scenario	Text input	nvarchar (MAX)
3.	Learning scenario type	Private or public scenario	Radio buttons	bit
4.	Teaching subject	Related teaching subjects	Select menu	nvarchar (50)
5.	Grade	Grade for which the learning scenario is intended	Select menu	int
6.	Learning outcomes	Related learning outcomes	Text input	nvarchar (MAX)
7.	Learning outcomes (CT)	Learning outcomes related to computational thinking	Text input	nvarchar (MAX)
8.	Keywords	List of keywords	Text input	nvarchar (100)
9.	Correlation and interdisciplinarity	Correlated teaching subjects	Select menu	nvarchar (50)
Additional data retrieved from database and displayed when viewing an existing learning scenario				
10.	Author	Author of a learning scenario	-	nvarchar (20)
11.	Acknowledgment	Author of the original learning	-	nvarchar (20)

No.	Label	Description		Type and length
		scenario (if applicable)		
12.	Duration	Sum of duration of all learning activities	-	tinyint
13.	Strategies and methods	Learning and teaching strategies and methods that are used in all activities of a learning scenario	-	nvarchar (100)
14.	Collaboration	Level of collaboration included in all activities of a learning scenario (individual, pair, small groups, class-based)	-	nvarchar (50)
15.	Teaching aids for teachers	Teaching aids needed for implementation in all activities of a learning scenario	-	nvarchar (100)
16.	Teaching aids for students	Teaching aids needed for implementation in all activities of a learning scenario (students)	-	nvarchar (100)

The second part of creating a learning scenario includes entering information about all learning activities. As mentioned before, learning scenario can have from *1* to *n* activities that can take place one after the other or in parallel. To specify the order in which activities are performed, the ordinal number of each activity should be entered. In this way user can define whether the activities are performed one after the other (if the ordinal number is higher by one from the previous one), or in parallel (if the ordinal number is the same as the previous one). Other required data to create a learning activity are described in Tab. 3.

Table 3. Data elements related to the learning activities

No.	Label	Description	Control type	Type and length
		Required data input to create a new learning activity		
1.	Ordinal number	Ordinal number of a learning activity	Select menu	tinyint
2.	Activity type	The type of a learning activity	Select menu	nvarchar (50)
3.	Use of digital technology	With or without digital technology	Radio button	bit
4.	Performance	Way of activity performance	Radio button	nvarchar (30)
5.	Name	The name of a learning activity	Text input	nvarchar (200)
6.	Description	Description of a learning activity	Text input	nvarchar (MAX)
7.	Duration	Duration of a learning activity	Text input	time (7)
8.	Strategies and methods	Used learning and teaching strategies	Select menu	nvarchar (100)
9.	Collaboration	Level of collaboration (individual, pairs, small groups, class-based)	Select menu	nvarchar (50)
10.	Teaching aids for teachers	Teaching aids used by teachers in learning activity	Select menu	nvarchar (100)
11.	Teaching aids for students	Teaching aids used by students in learning activity	Select menu	nvarchar (100)
		Additional data retrieved from database and displayed when viewing an existing learning activity		
12.	Acknowledgment	Author of the original learning activity (if applicable)	-	nvarchar (20)
13.	Author	Author of a learning activity	-	nvarchar (20)

3.3 Development of the prototype

To create the tool prototype, first, the database and a data model were created followed by the creation of a web application where a simple, intuitive, and responsive user interface was created.

The database was created in Microsoft SQL Server Management Studio [15] and was accessed by using the object-relational mapping framework Entity Framework Core [16]. To develop a web application, Net Core cross-platform [17], version of .NET, was used and the programming language C#.

In Fig. 4, an interface for entering a new learning scenario in three steps is displayed.

Figure 4. Interface for creating new learning scenario

4 CONCLUSIONS AND PLANS FOR FUTURE WORK

In the first cycle of the development, a simple prototype of a tool has been developed that allows users to create learning scenarios based on the existing scenarios or from the scratch. The prototype will be published on the web, and a target group of teachers will test it in a real environment. At the very end of the testing period, teachers will provide feedback about the tool by drawing attention to the functionalities that are missing or should be improved.

In the second cycle of the development, the tool will be upgraded with recommendations regarding certain types of activities that teachers could include in the learning scenario like activities for motivation and reflection as well as recommendations of digital tools that could be used to perform learning activities. Using recommendations, teachers will also be encouraged to include game-based learning activities in their learning scenarios, especially those for the development of computational thinking skills. To further enhance the process of learning scenarios development, the tool will include social networking features like sharing and commenting on learning scenarios. After the implementation of the designed learning scenario, teachers will have the opportunity to write impressions and observations. This information could help other teachers who want to carry out the same or similar learning activities with their students.

REFERENCES

[1]   A. Rojas-López i F. J. García-Peñalvo, „Learning Scenarios for the Subject Methodology of Programming From Evaluating the Computational Thinking of New Students", *IEEE Rev. Iberoam. Tecnol. del Aprendiz.*, sv. 13, izd. 1, Feb. 2018, str. 30–36, 2018.

[2]   J. Mezak i P. Pejic Papak, „Problem based learning for primary school junior grade students using digital tools", *2019 42nd Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2019 - Proc.*, str. 697–702, 2019.

[3]   D. Barr, J. Harrison, i L. Conery, „Computational Thinking: A Digital Age Skill for Everyone", *Learn. Lead. with Technol.*, sv. 38, izd. 6, str. 20–23, 2011.

[4]   V. J. Shute, C. Sun, i J. Asbell-Clarke, „Demystifying computational thinking", *Educ. Res. Rev.*, sv. 22, izd. November 2017, str. 142–158, 2017.

[5]   I. Botički, D. Pivalica, i P. Seow, „The Use of Computational Thinking Concepts in Early Primary School", *Proc. Int. Conf. Comput. Think. Educ.*, str. 6, 2018.

[6]   K. Tsarava, K. Moeller, N. Pinkwart, M. Butz, U. Trautwein, i M. Ninaus, „Training computational thinking: Game-based unplugged and plugged-in activities in primary school", *Proc. 11th Eur. Conf. Games Based Learn. ECGBL 2017*, izd. October, str. 687–695, 2017.

[7]   C. Peterson, „Bringing ADDIE to life: instructional design at its best", *J. Educ. Multimed. Hypermedia*, sv. 12, izd. 3, str. 1–5, 2003.

[8]   F. Wang i M. J. Hannafin, „Design-Based Research and Technology-Enhanced Learning Environments", *Springer, Educ. Technol. Res. Dev.*, sv. 53, izd. 4, str. 5–23, 2005.

[9]   R. Clark, „Accelerating expertise with scenario based learning", *Am. Soc. Teach. Dev.*, 2009.

[10]   T. de Jong *i ostali*, „Using scenarios to design complex technology-enhanced learning environments", *Educ. Technol. Res. Dev.*, sv. 60, izd. 5, str. 883–901, 2012.

[11]   N. Hoić-Božić, M. Holenko Dlab, S. Ursić Budim, i J. Mezak, „Development of computational thinking skills in primary school through digital storytelling with Scratch", *Proc. 10th Int. Conf. e Learn.*, izd. September, str. 114–119, 2019.

[12]   S. E. Indriwati, H. Susilo, i I. M. S. Hermawan, „Improving students' motivation and collaborative skills through Remap Jigsaw learning combined with modelling activities", *J. Pendidik. Biol. Indones.*, sv. 5, izd. 2, str. 177–184, 2019.

[13]   G. Zuckerman, „Development of reflection through learning activity", *Eur. J. Psychol. Educ.*, sv. XIX, str. 9–18, 2004.

[14]   S. Bagui i R. Earp, *Database Design Using Entity-Relationship Diagrams*, Second edi. CRC Press, 2012.

[15]   Microsoft, „SQL Server 2019", 2019. [Na internetu]. Dostupno na: https://www.microsoft.com/en-us/sql-server/sql-server-2019.

[16]   Microsoft, „Entity Framework Core", 2016. [Na internetu]. Dostupno na: https://docs.microsoft.com/en-us/ef/core/.

[17]   A. Troelsen i P. Japikse, *Pro C# 7: With .NET and .NET Core*. Apress, 2017.