A New Collection of Educational Scratch Projects Produced by Computer Science Students

Mile Jovanov, Emil Stankov and Bojan Ilijoski
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
{mile.jovanov, emil.stankov, bojan.ilijoski}@finki.ukim.mk

***Abstract*—Scratch is a free educational programming language specifically designed to be convenient for students of very young age to learn programming. The course “e-Learning Systems” is an elective course for the students at our institution – Faculty of Computer Science and Engineering (FCSE) in Skopje, primarily offered to the students enrolled in the “Computer Science Education” track, in the third semester of the undergraduate studies. Starting from the current academic year (2016/17), students of this course are asked, as a project activity, to implement an educational game or story in Scratch. In this paper we present the new website we have developed for the purposes of the course, which contains the student projects developed throughout the winter semester of 2016/17, as part of the course activities. We describe the form of the student projects and the collection of projects present on the website, as well as their functionalities and application.**

Keywords—development of educational Scratch projects; e-learning systems course; block-based programming.

I. Introduction

The course “e-Learning Systems” is an elective course for the students at our institution – Faculty of Computer Science and Engineering (FCSE) in Skopje, primarily offered to the students enrolled in the “Computer Science Education” track, and the “Applied e-Technologies” track, in the third semester of the undergraduate studies [1]. According to our implementation of ECTS (European Credit Transfer and Accumulation System), it can also be elected by students from other tracks in a higher semester.

The goals of the course include familiarizing students with the basic concepts of e-learning systems, their usage in education, as well as the functionalities that they offer to their users. Upon successful completion of this course the students will be able: to understand the role and new trends in technology assisted learning environments; to evaluate the learning process in technology assisted learning environments; to evaluate the effectiveness of the courses that are performed with the assistance of systems for technology assisted learning; to identify the needs of students in technology assisted learning environments; to understand the transition from traditional to technology assisted learning environments; and independently – to create an environment for collaborative creation of educational content.

The course contents include: models of learning, traditional learning environments, tools used in traditional learning environments, integration of technology into the learning process, the basic models of using ICT in education, technology aided learning environments, tools for technology assisted learning, comparison between traditional learning environments and technology assisted learning environments, interaction with technology assisted learning, interactive learning through the media, blended learning, designing digital content for learning technology and assisted learning, systems for content management, student oriented learning, collaborative learning and teaching, e-learning 2.0, learning through play, application of ICT in education of: preschool children, children and people with disabilities, gifted children, tools for knowledge management.

The course includes lectures, projects, discussions and laboratory exercises (weekly: 2 hours of lectures, 2 on discussions/exercises and 2 hours in lab). The students pass the course with 50% scored points from all the activities, including work on exercises, short exams and work on a project. Starting from the academic year of 2016/17, students are asked, as a project activity, to implement an educational project in Scratch.

In this paper we present the new website we have developed as e result of the “e-Learning Systems” course held in the winter semester of the current academic year (2016/2017). This website contains all the student projects developed throughout the semester as part of the course activities. Furthermore, we describe the form of the student projects and we go into details for three selected projects from the collection present on the website, with description of their functionalities and application in an educational environment.

II. Organization of the Work on Projects in the “E-Learning Systems” Course

Student projects can be boring and far from challenging to students. They may need to spend a lot of time on research about topics out of their interest, and still not be required to do any kind of practical work or work that will challenge them adequately on an intellectual level. For the course “e-Learning Systems”, we always try to assign students interesting and challenging projects.

The work-on-project in the course is organized throughout the semester. At the beginning of the semester, students are

The research presented in this paper is partly supported by the Faculty of Computer Science and Engineering, at Ss. Cyril and Methodius University in Skopje.

divided into teams (groups) of four to five members, and they generally work on the project outside the classes – although some of the course's lab classes are specifically dedicated to implementation of projects. Each team must select one member to take the role of team leader – a person responsible for corresponding with the teachers. Since usually there are different project types offered to the students of the course, the team leader also has the responsibility to select an appropriate project type for his/her team – after consulting the team's members. This means that the teams are allowed to "set the bar" for themselves, i.e. to decide on the project difficulty level and the requirements that they will have to meet in order to complete the project.

As a project activity for the course held in the winter semester of the current academic year, students were required to implement an educational game or story in Scratch. Regardless of the students' choice (game or story), as a mandatory part, the project had to include a process of learning and examination of the acquired knowledge on the topic under consideration. The topic for this year's project was "Work in Scratch and basic concepts of programming". The project had to be finished by the end of the semester and presented in public, in front of all the students and teachers on the course.

This year, three types of projects were offered to students. More precisely, it was a single base project (type 1), with added requirements (for type 2 and 3). The project types differed in complexity: type 1 project was the simplest one and type 3 was the most complex one. Of course, they were assigned different amount of points, according to the complexity: type 1 was assigned 7 points, type 2 – 10 points, and type 3 was assigned 13 points.

For the type 1 project, students were expected to implement an e-learning game or story, which had to rely on the basic principles of development and contain the basic elements of a game or a story, as described in [2]. The developed game/story had to look aesthetically satisfying and have at least three different levels. Furthermore, students were required to provide a form of examination (through questions/answers, or some other form that they would find to be appropriate) in the project. The type 2 project had the same basic requirements as type 1. Additionally, students had to create animations, as well as to provide gradation in the examination process (e.g. if they used questions/answers, they had to arrange them so that the easiest question would appear in the first level and the hardest one would appear in the last level). They also had to provide at least two different main characters from which the user would be able to choose at the beginning of the game/story. The type 3 project included everything from the previous type, and plus there had to be a start screen with some animation and a field where the user could write his/her name. Furthermore, students were expected to implement an end screen where the current user's points and the high scores would be displayed. Of course, different types of questions were expected to be assigned different points (e.g. the hardest question might be assigned the most points and the easiest question – the least number of points). Additional important requirement for this type of project was to provide two different storylines (for a story) or gameplays (for a game). In this way, by choosing a different main character from a list of (at least) two characters, the user would have the opportunity to play the game in a different way or to follow a different story, and learn some new concepts on the topic.

The assessment of projects and students was conducted in the following way. Each project, developed by a particular team, won at most the number of points defined as the maximum number of points for the corresponding type of project (7, 10 or 13), depending on how far the requirements for that project type were met. The final grade of the project was based on presentation, gameplay, storytelling, and the learning impact of the game/story. On the other hand, each team member (student) won the points that his/her team got, plus a particular amount of individual points. The individual points were assigned to each student according to the impression he/she had left to the course teachers at the public presentation of projects (at the end of the semester). This impression was formed based on the answers given by the student to 2-3 questions asked by the teachers regarding the implementation of a particular conceptual part of the project. The individual points were also different for the three different project types, and each student could win at most the number of points defined as the maximum number of individual points for the corresponding project type: 5 (for type 1), 6 (for type 2), and 7 (for type 3).

In the following section we explain our decision to use Scratch as a tool, and in Section IV we describe the website we have developed for the purposes of the "e-Learning Systems" course, which contains all the student projects developed throughout the semester. We also describe some examples of successful student projects that are present on the website.

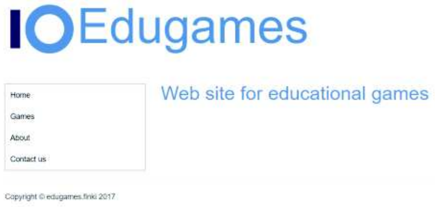
III. Why Scratch?

Scratch ([3], [4]) is a free, very popular, educational programming language specifically designed to be convenient for students of very young age (ages 8 to 16, or from third grade of primary education to high school students) to learn programming. Scratch was developed by the Lifelong Kindergarten Group at the Massachusetts Institute of Technology (MIT). It provides tools for creation of interactive stories, games, art, simulations, and all that using block-based programming [5]. As stated in [4], "Scratch helps young people learn to think creatively, reason systematically, and work collaboratively – essential skills for life in the 21st century". Programming in Scratch is done by dragging blocks from a built-in block palette and attaching them to other blocks to form a program structure called a script – a method known as "drag-and-drop programming" [6]. The Scratch project began in 2003, and the Scratch software and website were publicly launched in 2007 [3]. Currently, there are more than 12 million registered Scratch users, and this statistic grows rapidly each day.

Programming in Scratch introduces students and enables them to understand some of the basic principles of programming in general, such as: sequential execution of statements, conditional execution of statements, repetition of statements, variables and storing values, operators and operands, parallel execution of statements, using and passing messages as a means of communication between objects in a program, event handling, reacting to input and/or other events, etc. Scratch programming also enables development of some basic IT skills, since creating a typical Scratch application often includes: drawing sprites (characters), inserting photos or other types of files created using a graphics development software as background images, recording sounds and saving them in the MP3 file format, adding different textual fonts to the application, moving sprites to different positions on the screen using graphical commands, manipulation and applying digital effects to graphical objects, etc. Crook [7] argues that besides the fact that Scratch can be used as a core element to teach digital literacy in schools, it also "offers a creative environment to empower children to program computers and could thereby extend the ICT curriculum beyond the teaching of basic office skills". Scratch can also help embed the usage of computers throughout the curriculum and expand their usage beyond the classes in informatics only – which is a common situation in both the primary and the secondary education nowadays.

IV. The Edugames Website: Repository of Succsessfull Student Projects

In order to have a single space where we can gather all the projects developed by students of the "e-Learning Systems" course, and to provide a way for the students' work to be publically available, we have created a website where we have uploaded all the projects implemented in the current academic year. The website's name is Edugames [8]. The home page of the website is shown in Fig. 1. Currently, the structure of the website is very simple – it contains only 4 different main pages which can be reached through the appropriate hyperlinks on the home page, as can be seen in Fig. 1. Clicking on the Games hyperlink opens up the most important one – the page that contains a listing of hyperlinks to each of the projects developed by this year's teams of students. This page is shown in Fig. 2. Clicking on any one of the displayed hyperlinks loads the Scratch project implemented by the corresponding team.



IOEdugames

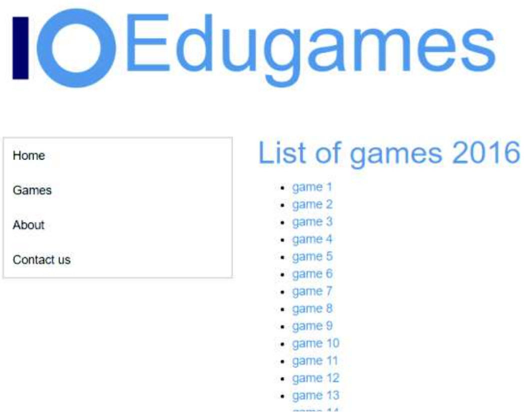
Home
Games
About
Contact us

Web site for educational games

Copyright © edugames.finki 2017

Fig. 1. The home page of the Edugames website.

The students made a serious effort in realizing their projects, and worked continuously throughout the semester – on the course's lab classes dedicated to projects implementation, as well as outside of them. This is also confirmed by the fact that almost 90% of the projects were completed on schedule. Some of the students showed exceptional commitment to the activity and produced educational projects of significantly high quality level. In the following subsections we will describe a few of the most successful projects developed by the students of the course in the current academic year.



IOEdugames

Home
Games
About
Contact us

List of games 2016

- game 1
- game 2
- game 3
- game 4
- game 5
- game 6
- game 7
- game 8
- game 9
- game 10
- game 11
- game 12
- game 13

Fig. 2. Listing of projects developed by students that are present at the Edugames website (excerpt).

A. "Coins collector" Game

The main idea of the "Coins collector" game is to use some of the Scratch elements in order to make a movement of the main character (Fig. 3). In the game, the player has to move the character over the scene using the available Scratch blocks (controls) in order to collect all the coins placed on it. After collecting the coins, the character must reach the goal position, which opens up the next, more complex level of the game. Each new level of the game includes a different, larger set of controls that can be used by the player, so besides the standard controls for moving forward/backward and rotating, there are also controls for door opening, using a piece of weaponry, etc. Additionally, for each different level there is a time constraint for completion, so the player must play quickly and efficiently.

The game offers an interesting way of introduction to the elements of Scratch.

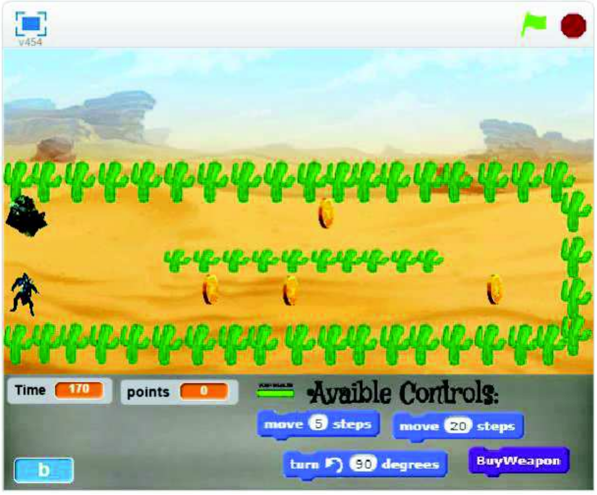


Fig. 3. "Coins collector" game – an example of a successful educational project developed by students of the course "e-Learning Systems", in the winter semester of 2016/2017. The game offers an interesting way of introducing the elements of Scratch.

B. *"The programming course" Game*

The game shown in Fig. 4 represents a classroom simulation. The idea here is that the teacher, through short messages and animation of writing on the blackboard, introduces the student to the contents to be learned. Then, he asks questions to which the student must provide correct answers in order to proceed with the game. The game has many levels, each of which covers different programming concepts, such as variables, conditions, loops, etc. Furthermore, the difficulty of questions rises with each new level.

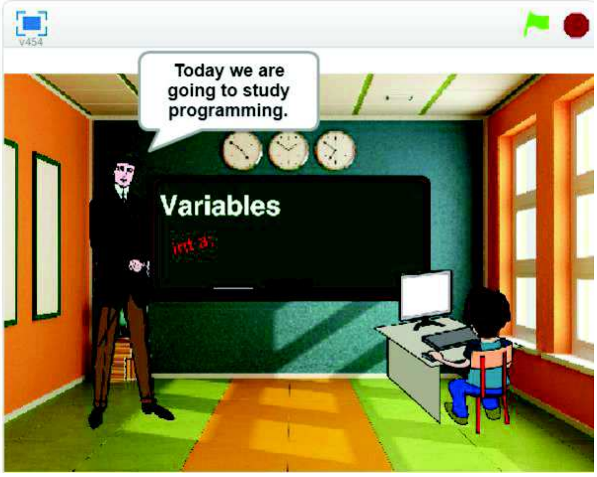


Fig. 4. "The programming course" game – second example of a successful educational project developed by students.

C. *"Code me" Game*

Another successful educational project, which simulates the process of learning a programming language, is the "Code me" game, shown in Fig. 5. The main scene consists of 4 parts. The first part is the scene where the main character can be moved, as well as some additional objects. The other 3 parts are used to present the description of the story, to present the available controls and to provide a space where the user can write the programming code. First, the story is set in the upper left part named "Instructions" (Fig. 5), and the events that need to happen are described. In the middle part, named "Commands", the controls that are available for the current scene (level) are displayed. There are controls for moving to the left/right, jumping, taking a particular object, leaving an object, using an object, etc. The appropriate controls must be placed in the correct logical order in the upper right part, named "Program", so that executing them will realize the described story successfully. The game has several levels and it becomes more and more complex from level to level: more controls are introduced and the program (sequence of controls) to be executed in order to realize the story becomes larger.

"Coins collector", "The programming course" and "Code me" are three examples of games produced by our students. The benefit from this approach, besides the knowledge gained by the students, is that there now exists a valuable collection of short educational games that may help other students to learn a particular topic through a game. This year's topic was 'Introduction to programming', and the produced games may be beneficial for the younger pupils, having in mind the trend of introduction of computational thinking and programming in the early stages of education in our country [9], as well as in many other countries worldwide.

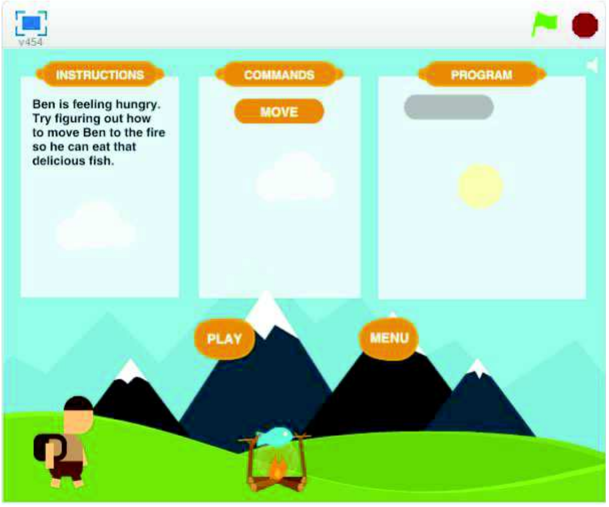

Ben is feeling hungry. Try figuring out how to move Ben to the fire so he can eat that delicious fish.

Fig. 5. "Code me" game – third example of a successful educational project developed by students. The appropriate controls (middle section) must be placed in the correct logical order in the right section, named "Program", so that executing them will realize the described story (left section) successfully.

V. Conclusion and Future Work

Our general impression is that students tend to show high interest in Scratch as an e-learning platform. Although some of the students were skeptical at the beginning and shared an opinion that Scratch is a programming environment for children, later they captured our primary idea, which was not to teach them to program in Scratch (students at that age are already capable to program in much more complex environments and languages), but rather to enable them to understand the concept of an environment for learning programming which is intended for children. This is confirmed by the successfully implemented projects, a large part of which were systems for learning programming and/or mathematics suitable for children of very young age. Of course, without doubt, through the process of creation of the project students also learned Scratch as well.

In the future, we plan to continue with this type of project activity for the "e-Learning Systems" course. Our goal is to inspire the future generations of students to create projects of even higher quality, and to publish all these projects and expand the Edugames website. We also plan to group projects by theme and by intended users' age on the website, in order to make them easily accessible for the users for which they were originally intended. This may be done by listing the games by their names and attaching few labels for each game which will allow sorting and fast access to a particular group of games.

References

[1] Website of the Faculty of Computer Science and Engineering (FCSE), http://www.finki.ukim.mk, accessed March 2017.

[2] A. Majumdar, "5 game elements that create effective learning games," eLearning Industry 13 of June 2016 article,

https://elearningindustry.com/5-game-elements-create-effective-learning-games, accessed March 2017.

[3] J. Maloney, M. Resnick, N. Rusk, B. Silverman, and E. Eastmond, "The Scratch programming language and environment," in: ACM Trans. Comput. Educ., vol. 10, no. 4, article 16, pp. 1-15, November 2010.

[4] Website of the Scratch programming language, https://scratch.mit.edu, accessed March 2017.

[5] D. Weintrop, "Minding the gap between blocks-based and text-based programming," in: Proceedings of the 46th ACM Technical Symposium on Computer Science Education, pp. 720. New York, NY, USA: ACM.

[6] B. DiSalvo, "Graphical qualities of educational technology: Using drag-and-drop and text-based programs for introductory computer science," in: IEEE Computer Graphics and Applications, vol. 6, pp. 12-15, 2014.

[7] S. Crook, "Embedding Scratch in the classroom," in: International Journal of Learning and Media, vol. 1, no. 4, pp. 17-21, 2009.

[8] The Edugames website, http://edugames.finki.ukim.mk, accessed March 2017.

[9] M. Jovanov, E. Stankov, M. Mihova, S. Ristov, and M. Gusev, "Computing as a new compulsory subject in the Macedonian primary schools curriculum," in: 2016 IEEE Global Engineering Education Conference (EDUCON), pp. 680-685. IEEE, 2016.