Co-funded by the
Erasmus+ Programme
of the European Union

O3/A4 IMPROVED METHODOLOGICAL LEARNING FRAMEWORK

Elaborated by EU-Track (Italy) and University of Thessaly (Greece)

TABLE OF CONTENTS

TABLE OF FIGURES

INTRODUCTION

The content of this report is referring to the document *O1/A6 - A game-based, design-thinking learning framework enhancing programming skills in secondary education* prepared in the *O1 - Methodological Learning Framework*. It aimed at developing the methodological framework to be used in the course of the Coding4Girls (abbreviated as C4G) project on how to adapt design thinking approaches which are inherently user-centered to ICT learning practices all the while motivating a fringe of the general population that is currently less interested in coding. The result was the designing of game-based learning experiences for secondary school students to build their programming skills to be delivered through two different products and complementary products: the Teachers' Platform and the Students' Game Environment.

During the validation processes of both the methodological framework and the C4G project software organized in the partners' country (Slovenia, Italy, Greece, Bulgaria, Portugal and Turkey), the project team has been collecting both qualitative and quantitative data with the help of several instruments, such as questionnaires and grids. The aim was to verify if the proposed pedagogical framework meets the target groups' needs in terms of relevance, acceptance, usability, and effectiveness.

Several additional learning materials were prepared for teachers in order to provide them more familiar with both the C4G methodology and tools as follows:

- Instructions for C4G platforms and software including a video tutorial in Italian;
- Learning scenarios, videos tutorial, interactive worksheets in Croatian;
- Online course including 16 hours in videoconference environments of MS Teams. All the recorded sessions were published in the LMS Moodle. The course is available in Bulgarian language at https://edugames.swu.bg/moodle/course/view.php?id=18.

(Please, see the program in annex).

Following the results achieved from the piloting tests, the current report describes how based on the main feedback and comments received by experts, teachers and students, to possibly improve and update the learning scenarios, the Teachers' Platform and the Students' Game Environment.

1. THE OPERABILITY AND INTEGRATION OF THE TWO C4G PLATFORMS

The project team has developed the C4G software consisting of two different interconnected parts. The first one is the Teachers' Platform and the second is the Students' Game Environment.

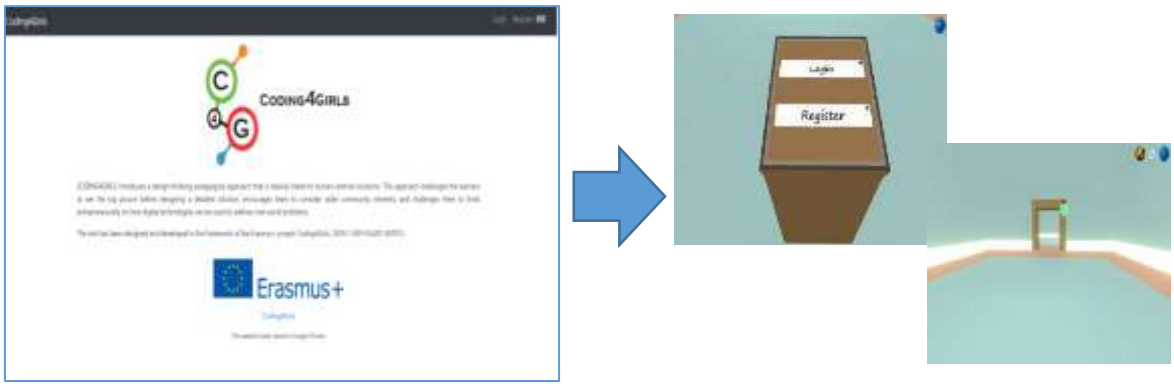


Figure 1. *The Teachers' Platform interconnected with the Students' Game Environment.*

The students are encouraged to design and code games that address specific needs or issues (depending on teachers' choice). It is a "low entry high ceiling approach" allowing students to start with easy problems until engaging more challenging tasks. Teachers design and present to their students "half-baked" scenarios in which a solution is partially ready by challenging them to complete the task. Therefore, teachers can prepare small and manageable modules by using Coding4Girls (C4G) software suitable for their students' needs and their knowledge level.[1]

[1] Dochshanov, A., & Tramonti, M. (2020). Computational Design Thinking and Physical Computing: Preliminary Observations of a Pilot Study. *Robotics*, *9*(3), 71.

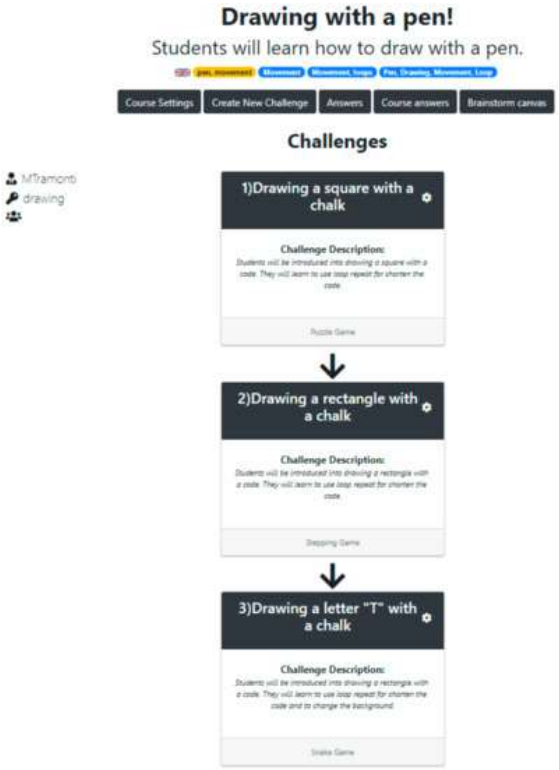


Figure 2. *An example of challenges in a coding course in the Teachers' Platform.*

The tasks can be solved individually and collaboratively by fostering group discussion in the classroom or, virtually via the Students' Game Environment. The software offers a brainstorming space where students can discuss and share their ideas by placing and managing multimedia-capable post-it notes.

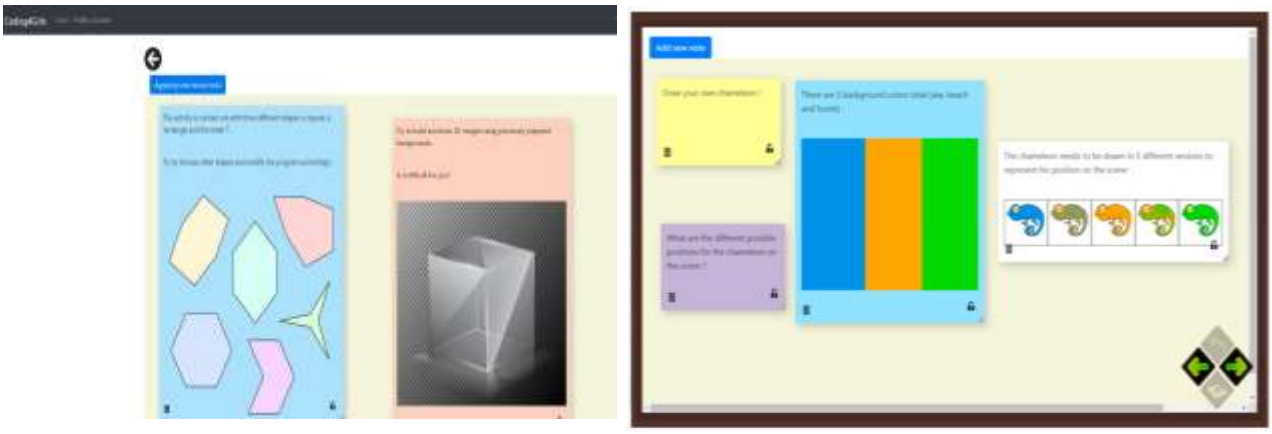


Figure 3. *Brainstorm zone prepared in the Teachers' platform and the Students' Game Environment for students' contribution and discussion*

Teachers and experts who have validated the C4G approach and the two education environments, overwhelmingly find the developed tools interesting and useful, especially to support the achievement of the learning objectives of students. This holds especially true for students who might present pre-existing learning difficulties.

The strong point of these tools is that they allow the customization and personalization of the learning path which can be designed. Courses can be diversified and tailored based on the students' age and the desired overall required level of difficulty.

In this context, the learning activities become captivating by increasing the motivation and the engagement of the students. As result, the learning process is encouraging and helping them to think creatively.

Each student has tools that, based on his learning methods, his cognitive style, his preferences, are more adequate than others to ensure the success of the teaching activity.

However, the role of the teacher remains central to facilitate and to prepare the learning scenarios to be delivered to the students involved.

One of the weak points underlined is that the teachers need specific training on how to use this software and products to make them master them and then to have the capability to exploit all their potentialities. In fact, in the first stage, its usability can seem to be too complicated if a teacher or a content creator is exposed to the platform without sufficient training.

For this reason, the C4G project created a YouTube channel where the teachers have at their disposal many videos serving as guidelines describing how to use and exploit the C4G products in their teaching practices. The videos are available in English with subtitles in different languages, currently Slovenian, Italian, Portuguese, Croatian and Turkish and Bulgarian. More subtitles are coming soon.

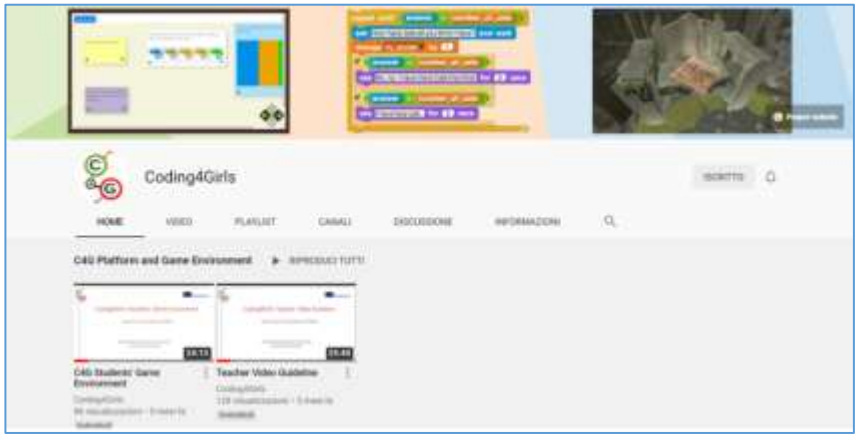


Figure 4. *The video-guidelines on how to use both the Teachers' platform and Students' Game Environment*

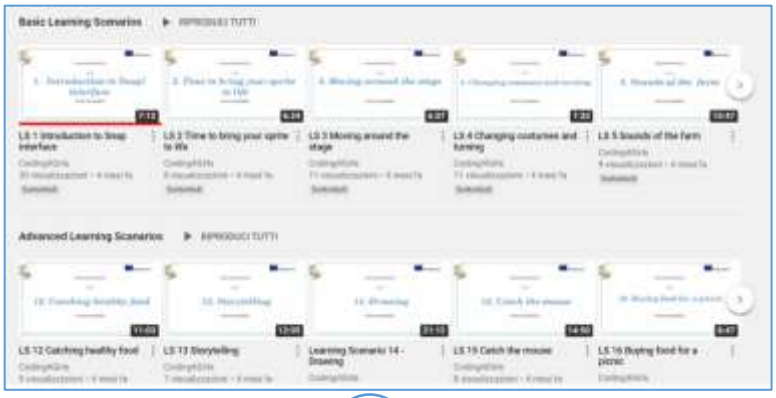
.

2. THE SELECTED PLATFORM

To help teachers master the C4G software, the project team prepared 21 distinct learning scenarios with varying levels, from basic coding practice to more advanced ones for more capable students.

These scenarios are available both on paper and in digital form on the C4G platform.

1



2

Figure 5. *1-Collection of game design-based learning sheets targeting teachers in paper-based [2]and 2-Collection of basic and advanced learning scenarios in multimedia form[3].*

These learning scenarios were developed by using the Snap! Software and were re-adapted to the design thinking approach and the structure of the C4G software constituting of two parts: the Teacher's Training Platform and the Student Game Environment. These courses and learning scenarios were structured into challenges to incentivizing students on solving a specific coding issue. The outcome *A1.1 Learning Scenarios Code*[4] reports two groups of the tables of the learning scenarios/training course with their code (where is available) in the Teacher's Training Platform and, consequently, in the Student Game Environment. They are in English, Slovenian, Italian, Croatian, Bulgarian, Turkish and Greek.

[2] The *Collection of game design-based learning sheets targeting teachers* is available on the project website in different partner languages: https://www.coding4girls.eu/results_03.php

[3] The collection of the instructional videos on both basic and advanced learning scenarios are available on the C4G YouTube Channel at https://www.youtube.com/channel/UC0DEdwkV9PsJ4Fb70MAJQrg

[4] https://www.coding4girls.eu/results_03.php

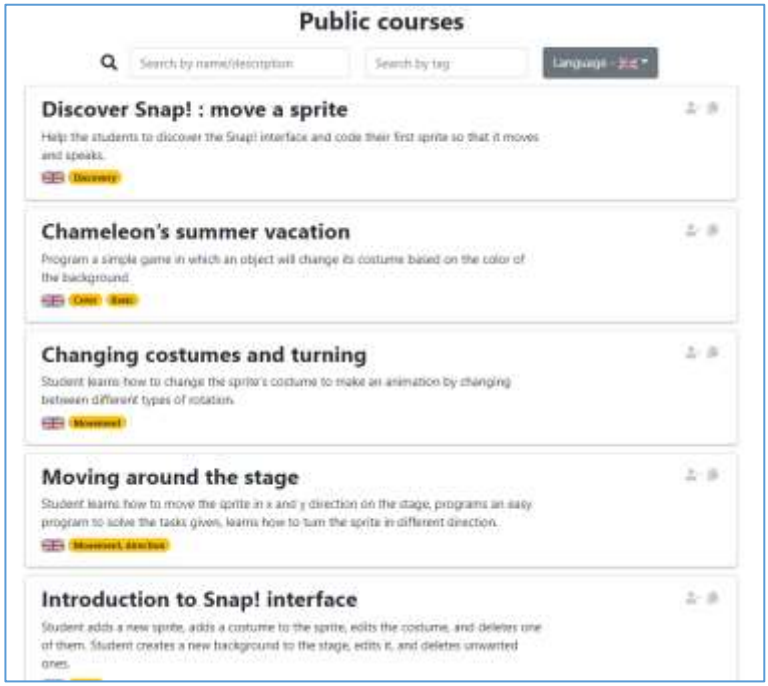


Figure 6. *The learning scenarios available in both Teachers' Platform and Students' Game Environment.*

These courses are constituted as a grouping space for thematically related activities, all connected to an overarching issue. The problem is presented at the very beginning of the course to the students who can brainstorm together to collaboratively elaborate tentative solutions. They can use post-its on a virtual board to discuss their ideas by using text, images or video (Figure 3). After the brainstorming phase, the students are given, in a step-by-step fashion; specific activities (presented in consecutive order) designed to present the tools necessary to solve the overarching problem. These coding activities will be presented by using Snap! canvas, through both the half-baked tasks to challenge the students in the problem solution and the final solution presentation.

The grouped activities in the course are called challenges, which are presented to students in a specific order set by the teacher (Figure 7). For example, if a teacher wants to create a course on basic programming knowledge, the first activity could concern the concept of Booleans, the second could concern conditional structures and the last one, the loops. This will allow teachers to prepare customized learning steps for their students in the Teachers' Platform and the students to be led gradually to the final learning objective in the Students' Game Environment.[5]

[5] Alden, D., & Tramonti, M. (2020). *Op.Cit.*

As previously underlined, the platform for Coding Snap! was chosen. Snap! uses a very similar language of Scratch. The main difference is that Scratch allows the teacher to explore much more in-depth and advanced programming concepts compared to Scratch, such as nested functions, lambda calculus and high order functions.

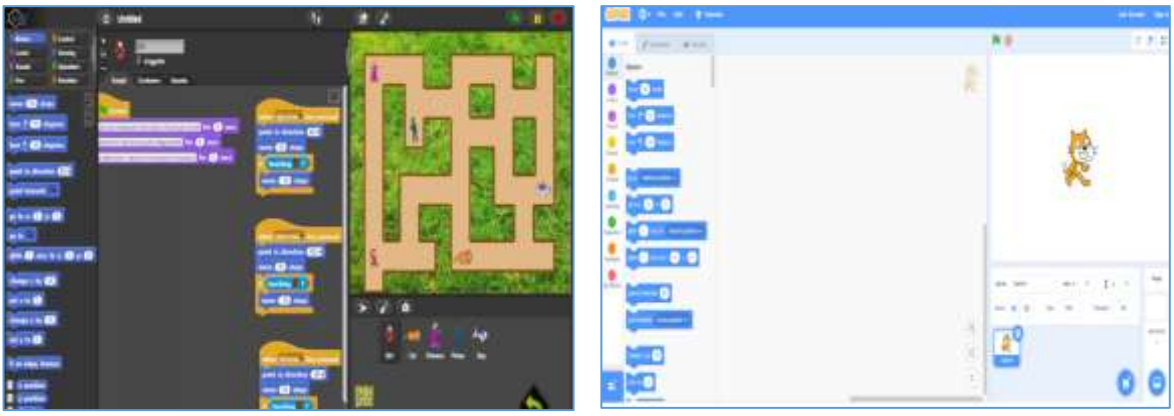


Figure 7. *Snap! and Scratch User Interfaces.*

Snap!, previously known as *Build Your Own Blocks*, is an extended reimplementation of Scratch featuring first-class procedures, first-class lists, and first-class sprites with inheritance. An earlier version, BYOB, was a modification of the Scratch 1.4 source code, but the current version is an entirely separate program, even though its user interface looks like Scratch 1.4 and it includes almost all of its primitive blocks.[6]

Despite the similarities between the platforms of Snap! and Scratch, some teachers prefer going on by using the Scratch platform, mainly by habit as that is a software that they know very well now, having used it for years. More superficial criticism was also raised, such as the black color of the background being not appropriate for young students. This kind of criticism can be linked to a lack of training as in Snap! The background can be changed from black to white being going into options and selecting flat design.

It was also noted that the graphics used in Snap! are more suitable for students of all ages.

[6] https://scratch.mit.edu/

2.1 The implementation

The students can access these challenges created by the teachers/content creators by downloading the Students' Game Environment file. The file contains a Unity 3D videogame where students can discover and complete the courses prepared by their teachers in a fun, engaging, and playful manner.

All the minigames offered (currently 11) to the students are related to and try to exemplify the actual programming concepts upon which the C4G courses and scenarios have been designed (Figure 8-16).

Figure 8. *Match3 Game*

Figure 9. *Dice Game*

Figure 10. *Multiple Choice Questions*

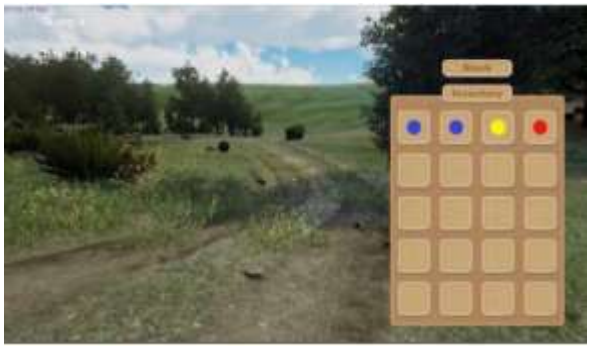


Figure 11. *Inventory*

Figure 12. *Stepping Game*

Figure 13. *Sound Game*

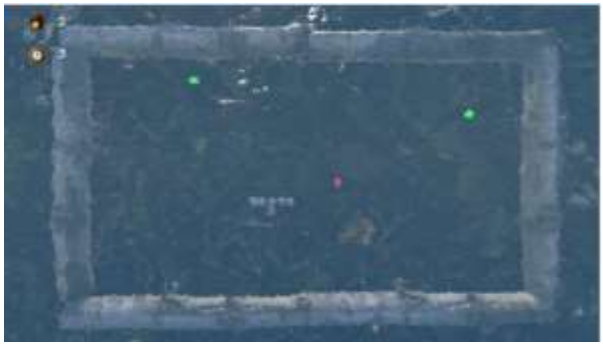


Figure 14. *Snake Game*

Figure 15. *Puzzle Game*

Figure 16. *Pattern Matching Game*

The courses, using elements of the design thinking approach, present to the students an overarching issue to solve and present the tools to solve it in a step-by-step approach. The courses as mentioned above, are prepared by teachers and published in the Teachers' Platform, but students can access them through the Students' Game Environment. Figure 17 shows the panel of challenges in the Students' Game Environment of the course created by

the teacher in the Teachers' Platform. The number from 0 to 3 represents the challenges. When selecting one of them, the topic to be studied in the selected challenge appears; for example, in our case is the "Conditional challenge". Besides the challenge topic, the system shows the 3D mini-game associated with it, e.g., "Find your path".

Figure 17. *The panel of challenges in the Students' Game Environment*

In the Students' Game Environment, each challenge is structured as follows: one introductory mini-game illustrating the coding concept; one instructional page for the task to be fulfilled in Snap!; two Snap! Canvases: one to be used to solve the task and the other to display the activity solution (if the teacher/content creator wished it so).

The teachers can decide which mini-game (optional) their students will play by selecting one of the existing mini-games.

The instructional page is in HTML, usually enriched by images or videos, to present the context and specific aim of the learning task to be fulfilled in Snap!

This page will be followed by a Snap! canvas, based on a template provided by the teacher containing the coding activity or problem to be executed or solved by students. Another Snap! canvas will be provided to display the solution of the challenge proposed. However, only the teacher can decide if solution canvas is shown or not. It will depend on the teaching process management selected by the instructor.

Since every course assembles more challenges, these steps will be repeated as many times as the number of the total challenges identified by the teachers. This allows breaking one complex coding activity into simple elementary steps where the answer and/or the solution to the preceding activity become the template in which the next activity is to be executed.

In the end, the course is constituted of a certain number of challenges which present the programming teaching through an incremental scaffolding process.

Once the students have completed all the challenges of the course, they are led back to the initial coding problem and asked to synthesize the new knowledge they just acquired. At the very end of the course, the players can see also all the solutions to the problem proposed by the other students by arousing confrontation moments.

From the feedback and comments collected, the use of the challenges for the students is considered to be very effective, because this system helps them see the big picture of the problem to be solved before designing a detailed solution.

Therefore, they are encouraged to learn more through fun but at the same time, they learn how to think entrepreneurially about digital technologies and how to use them to solve real life problems.

The C4G approach is a good concept and a great way of meeting the students in a universe they are very familiar with, the gaming world.

The game-based learning approach, as designed by the C4G software, can help students develop their basic skills and concepts without making them focus too much on the theoretical knowledge which is behind the game-exercise. For this reason, teachers think that these games shouldn't be too specific, but they should simply be preparatory and introductory for some disciplines.

Thanks to the diversification of the actions' settings, scenarios and challenges, the motivational aspect of the students remains high in C4G methodology by supporting them to strengthen their interest.

Besides, some students have difficulties with the traditional teaching methods; others arrive at second-grade high school with few digital skills, so, for example, sometimes they struggle to even follow the instructions to download the digital version of the textbook. In this context, the introduction of the C4G approach and products into the teaching practice would allow them to overcome these difficulties.

However, if on one hand, the designed tools and games' setting are effective for primary school students; on the other hand, according to the feedbacks collected, the learning scenarios are not very suitable for achieving the learning objectives of secondary school students.

This is because some learning scenarios are not very close to the secondary school students' interest and that the graphics of some games should be more appealing as students are often exposed to the more cutting edge version of 3D open world games on PC or gaming consoles.

In relation to the 3D open world aspect of the C4G software, concerns have been often raised about performance issues on the computer used by the students, especially in the context of an in-school computer lab. The C4G software requires quite some space and some computing power, and even though a whole array of tools (graphical quality settings, FPS display, anti-aliasing settings, etc.) are available to the user in order to try and improve the performance of the game, it will work quite slowly on older computers which can be quite common in school settings. Efforts have been made on that complex issue during the whole development phase and should be continued in order to provide the best experience possible to all end users in all settings.

3. GIRLS GAMING PREFERENCE

In the framework of the C4G project, instructional support contents were designed and developed to provide educators and teachers with the necessary how-to good practices allowing them to integrate the proposed learning methodologies and tools into their teaching and learning experiences. Therefore, the C4G team designed 21 learning scenarios including instructions for both teachers and students.

These learning materials aim to facilitate the more effective adoption of a project methodology for developing programming skills among girls through serious games into wider, blended learning school practices. They provide teachers with the necessary information for reinforcing their teaching on programming through the proposed serious games approach and design thinking learning methodologies.

They support the teachers' digital skills by increasing their familiarity in emerging ICT, mainly, in serious games.

Therefore, the learning sheets provided describe end-to-end blended learning activities that deploy the CODING4GIRLS serious game and design thinking approaches. They are accompanied also by visual aids, how-to videos, explaining the features and the functionality of the CODING4GIRLS serious game.

Besides, a user guide on the CODING4GIRLS serious game approach for building programming skills among girls through design thinking approaches was produced in English as well as the national languages of project partners – Bulgarian, Croatian, Greek, Italian, Portuguese, Slovenian and Turkish.

The learning sheets include information for each learning activity to be developed for building programming skills for girls and boys and, in particular, they underline the following features:

- The overall educational objective of the corresponding learning activity
- Concepts covered by the learning activity
- Specific learning objectives
- Expected learning outcomes
- Step-by-step use of the CODING4GIRLS game design-based learning approach

- Assessment methods for evaluating the knowledge developed
- Questions for initiating discussion among learners in the context of class collaboration.

These learning sheets are divided into two groups, basic and advanced, to lead the students from simply programming concepts and functionalities towards more complex ones with a special focus on the girls' preferences and interests. In particular, they cover both the generic functionality of the proposed serious game, including user interaction processes and feedback generation as well as descriptions of all learning activities that will be implemented in the proposed serious game.

Besides the learning sheets, also the C4G software is designed taking into account the features which allow increasing motivation and interest in building programming skills in girls.

In fact, according to Lastly, Alserri et al (2018), the C4G conceptual model for the serious game was developed for gender-based engagement by meeting the following elements[7]:

1) Learning Elements: these are the elements that distinguish entertainment games from educational games (Alserri et al., 2018);
2) Female Preferences for digital games: these are the preferences specific to girls that have to be incorporated into the design to motivate and engage them. According to the literature review conducted by the author, these preferences consist in exploration, character customization, storyline, social interaction, collaboration, challenges, fun, control, and feedback (Alserri et al., 2018);
3) Flow state theory: some of these elements are also female preference elements. These elements should also be incorporated to obtain engagement and motivation: challenges, fun, control, feedback, concentration, clear goals, skill and immersion (Alserri et al., 2018);
4) Female game types and genres: according to the authors, these would be fantasy and role-playing games.
5) Social gender factors: parental, peers and teacher influence.

[7] Ref. *O1/A4 Game-based Design-Thinking Learning Framework for enhancing programming skills in secondary education*

Therefore, the games settings and features were adapted to these characteristics to incentivize the girls' preferences.

After the validation activities, on the base of the feedback and comments gathered, the project team have reached an overall good positive acceptance of both the methodology and tools from both teachers and students sides.

Concerning gender issue, the data show that there are no relevant differences between boys and girls towards the acceptance and grade of the involvement.

The target group underlines how the C4G methodology and products can facilitate in the girls to a closer approach to the world of programming without the initial fear that most students normally have.

4. TEACHERS' LEVELS OF INVOLVEMENT

The platform has offered to the teachers a high level of involvement into the customization of the students' experience thanks to the interaction of the two platforms: Teachers' Platform and Students' Game Environment.

The design thinking and game-based approaches are considered effective as, in an involvement way, teachers can design a learning process leading students to the problem-solving process occurring in the game environment.

Figure 18. *The correspondence between the challenges created in the Teachers' Platform and the ones available in the panel of challenges in the Students' Game Environment.*

One of the advantages of C4G software often mentioned is that once the problem is defined, the students can collect their ideas, thanks to the brainstorming zone. This can encourage their reflection, ideas discovery and activity planning. After checking their choice's functionality made is verified, the students can proceed to apply it to the initial problem.

In this context, from the teacher's point view, the use of C4G approach allows to carry out effectively projects where students must necessarily work in a team to facilitate the creation of new ideas by listening to others and stimulating the use of different skills.

Therefore, if, from one hand, the C4G software can encourage the development of new ideas and digital skills; from the other hand it can incentivize also the collaboration among teachers even of different disciplines, mainly in the learning scenarios creation.

Besides, the system is considered innovative with high potentialities in both teaching and learning activities focused on the development of an intuitive and creative process.

However, some of the feedback and comments collected underline that the software can be a bit too complex for some teachers, mainly for those who have low ICT skills. This can imply the necessity to have more complete and comprehensive training for activities teachers in order for them to become more familiar with both the platforms before enabling their integration into everyday school practices.

As a result, the efficiency (and therefore the usability) of the C4G products is however extremely linked to the available/used hardware and the availability of time which is certainly important, mainly, in the initial phase for the teachers training.

This means that the project products could be very useful if they are synergistically integrated with other types of activities.

CONCLUSION

On the base of the experimentation results, the feedback received shows that early interventions can undoubtedly reduce the gap between male and female participation in computer science teaching and related professions. The C4G project will allow in the future bolstering and facilitating women's engagement in IT by making it more accessible to all. Along with a better accessibility, a better perception of what roles and professional careers the IT domain can offer is also fostered.

In this context, the C4G approach and software favours the increasing of the interest in ICT disciplines and the development of specific IT skills thanks to the possibility to design personalized learning contents that can be integrated into teaching practices.

In fact, most teachers, involved in the testing phase, have shown their interest in deepening of the use of C4G approach even after the validation activities organized by the project team.

In brief, the strengths of both C4G approach and products:

- Personalized learning scenarios for programming skills developing;
- Development of the students' specific skills focused on the problem analysis and solving, creating new solutions and ideas, verifying new opportunities and communicating with others in team working;
- The serious games designed can facilitate emotional involvement and development and promote more effective learning strategies, mainly in girls.

While among the features to be improved, the data underline mainly the following points:

- The software should be more stable and intuitive to reduce the training time that teachers need before implementing the C4G software in their practice;
- Improving the graphics of the games to make them more appealing for secondary school students;
- Increasing the number and type of the serious mini games available in both Teachers' Platform and Students' Game Environment.

REFERENCES

Ahmad, G., Soomro, T., & Naqvi, S. M. (2016). *AN OVERVIEW: MERITS OF AGILE PROJECT MANAGEMENT OVER TRADITIONAL PROJECT MANAGEMENT IN SOFTWARE DEVELOPMENT* (Vol. 10).

Aitken, A., & Ilango, V. (2013). *A comparative analysis of traditional software engineering and agile software development.* Paper presented at the 2013 46th Hawaii International Conference on System Sciences.

Alserri, S. A., Zin, N. A. M., & Wook, T. S. M. T. (2018). Gender-based Engagement Model for Serious Games. *International Journal on Advanced Science, Engineering and Information Technology, 8*(4), 1350-1357. doi: 10.18517/ijaseit.8.4.6490

Balaji, S., & Murugaiyan, M. S. (2012). Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management, 2*(1), 26-30.

Carmichael, G. (2008). Girls, computer science, and games. *ACM SIGCSE Bulletin, 40*(4), 107-110. doi: 10.1145/1473195.1473233

Dochshanov, A., & Tramonti, M. (2020). Computational Design Thinking and Physical Computing: Preliminary Observations of a Pilot Study. *Robotics*, *9*(3), 71.

Hosein, A. (2019). Girls' video gaming behaviour and undergraduate degree selection: A secondary data analysis approach. *Computers in Human Behavior, 91*, 226-235. doi: https://doi.org/10.1016/j.chb.2018.10.001

Jhajharia, S., kannan, v., & Verma, S. (2014). *Agile vs waterfall: A Comparative Analysis* (Vol. 3).

Kafai, Y. B. (2006). Playing and Making Games for Learning: Instructionist and Constructionist Perspectives for Game Studies. *Games and Culture, 1*(1), 36-40. doi: 10.1177/1555412005281767

Long, P. D. (2004). Encyclopedia of Distributed Learning. Thousand Oaks

Thousand Oaks, California: SAGE Publications, Inc.

Martin, R. C. (2002). *Agile software development: principles, patterns, and practices*: Prentice Hall.

Osborn, G. (2017). Male and Female Gamers: How Their Similarities and Differences Shape the Games Market., from https://newzoo.com/insights/articles/male-and-female-gamers-how-their-similarities-and-differences-shape-the-games-market/

Ow, S. (2009). *Review of Agile Methodologies in Software Development* (Vol. 1).

Peteranetz, M. S., Flanigan, A. E., Shell, D. F., & Soh, L. K. (2017). Computational Creativity Exercises: An Avenue for Promoting Learning in Computer Science. *Ieee Transactions on Education, 60*(4), 305-313. doi: 10.1109/te.2017.2705152

Petersen, K., Wohlin, C., & Baca, D. (2009). *The waterfall model in large-scale development.* Paper presented at the International Conference on Product-Focused Software Process Improvement.

Royce, W. W. (1987). *Managing the development of large software systems: concepts and techniques.* Paper presented at the Proceedings of the 9th international conference on Software Engineering.

Unhelkar, B. (2016). *The art of agile practice: A composite approach for projects and organizations*: Auerbach Publications.

Vermeulen, L., Looy, J. V., Courtois, C., & Grove, F. D. (2011). *Girls will be girls : a study into differences in game design preferences across gender and player types*. Paper presented at the Under the mask: perspectives on the gamer, Luton, UK. http://hdl.handle.net/1854/LU-1886961

Weintrop, D., & Wilensky, U. (2015). *To block or not to block, that is the question: students' perceptions of blocks-based programming*. Paper presented at the Proceedings of the 14th International Conference on Interaction Design and Children, Boston, Massachusetts.

Weisert, C. (2003). There's no such thing as the Waterfall Approach!(and there never was)'. *Information Disciplines, Inc*.

ANNEX- SYLLABUS FOR A QUALIFICATION COURSE FOR TEACHERS GAME-BASED PROGRAMMING TRAINING TECHNOLOGIES

SOUTH-WEST UNIVERSITY “NEOFIT RILSKI”

BLAGOEVGRAD

Approved by:

Dean of FMNS: / Assoc. Prof. El. Каращранова, PhD /

S Y L L A B U S

for a qualification course for teachers

GAME-BASED PROGRAMMING TRAINING TECHNOLOGIES

CODING4GIRLS PROJECT

Period of Tuition:	32 hours
Form of Tuition:	Partial on-site attendance/distance
Tuition-Providing Faculty:	Faculty of Mathematics and Natural Sciences (FMNS)
Tuition-Providing Department:	Informatics (in FMNS)

BLAGOEVGRAD

2020 г.

Syllabus

The syllabus and the teaching and learning materials were designed within the Coding4Girls project, co-financed together with ERASMUS+, KA201

Name of Programme
GAME-BASED CODING INSTRUCTION TECHNOLOGIES

Programme Target: (*more than one type of educationalists can be chosen*)	
X	Teachers – primary school
X	Teachers – lower-secondary school
X	Teachers – upper-secondary school
	Head-teachers
	Deputy Head-teachers
X	Head of ICT Sector
	Pedagogical consultants, psychologists
	Pedagogical staff in dormitories
	Speech therapists, rehabilitators, resource teachers
	Répétiteur, choreographers, coaches
	Other (please specify)

Syllabus annotation
Brief description:
The course is designed to target teachers of computer modelling at primary level, IT teachers at lower-secondary level, and IT and Informatics teachers at upper-secondary level, as well as head of ICT sector. Course graduates will be able to organise successfully digital competences development classes within the framework of the *Education for Tomorrow* project and other extracurricular activities in the field of teaching programming. The syllabus includes the following key topics: 1. Fundamental concepts – serious games, game-based learning, design thinking, learning theories and their association with serious games. 2. Game environments for the teaching of programming. Overview. 3. Snap! block environment for programming. Major characteristics and blocks. Introducing

basic structures through computer games design.

4. Game environment for programming, designed through the *Coding4girls* project. Teacher interface and student interface. Application of the environment in the process of acquiring new content and skills, digital competences evaluation, independent studies.

5. Creating learning scenarios for teaching coding to children through computer games.

The form of tuition is partial attendance – 16 classes of attendance is required, as well as 16 classes if distance education in the form of webinars and independent assignments. Practice sessions involve group work and production of resource materials. Should attendance at the university is impossible (in accordance with instructions of the Ministry of Education and Science /MES/), classes will be held in the form of synchronous learning through synchronous distance learning technologies: on-line videoconference platforms.

Aims and Objectives:

The tuition will result in improving the trainees' knowledge and skills in:

Academic competence:

- Knowledge in the sphere of the teaching and learning in computer modeling and coding, as well as coding teaching methodology. (1.1.)
- Evaluation of digital competencies through contemporary means and technologies (1.2.)
- Development of communicative skills, critical and design thinking for the purpose of efficient search, deduction and selection of information from a variety of sources. (1.7)

Pedagogical competence:

- Planning of activities, in the sphere of extracurricular education in coding (1.1)
- Realization of disciplinary and interdisciplinary connections, forecasting of the expected learning outcomes. (1.2)
- Application of innovative methods for teaching and evaluating students' results. (1.5)
- Supporting and motivating students in the formation of digital skills (2.4)
- Implementing the requirements for safe learning, education and work conditions and providing students with a safe and secure environment, including work on the Internet (2.10)
- Using innovative methods and tools to promote student progress. (3.2)

Administrative competence

- Organization of coding training in a game environment.

Methods of teaching:

The following training methods will be applied:

Lecture, discussion, group work, independent assignment.

Material, technical and information resources specific to the program:

Laptop, video projector, screen, internet access, computer room with internet access for students, e-learning platform Moodle, online video conferencing environment, study materials: user guides, videos, sample lesson scenarios, developed within the Coding4Girsl project.

Relationship between the theoretical and practical parts:
Theoretical part - 30%, practical part - 70%

Competencies to be acquired: **(one may choose one or more types of competencies)**	
X	Academic
X	Pedagogical
	Organisational
	Communicative
X	Administrative

Forms of tuition:	
	On-site attendance
X	Distance
X	Partial on-site attendance

Duration of tuition: (*number of classes*)
32 (16 on-site attendance and 16 distance education)

Number of qualification credit points: (*number of points*)
2 credit points

Final evaluation (elective):	
X	Defending a lesson project
	Test
	Presentation
	Other(please, specify):
	Methodological case-study

Trainers included in the program:

(list the names and PIN of the trainers)
Daniela Ivanova Tuparova **Boyana Garkova** **Rositsa Georgieva**

Contact person:	
Name:	**Daniela Ivanova Tuparova**
Current address (by ID card):	
Tel:	
E-mail:	ddureva@swu.bg

Blagoevgrad Lecturer:

2020 г. / Prof. Daniela Tuparova, PhD /

The syllabus was discussed and approved by the Department Council of the Department of Informatics on …… .....… protocol №

Head of Department:

/Assoc. Prof. Stefan Stefanov, PhD/

The syllabus was approved by the Faculty Council of the Faculty of Natural Sciences and Mathematics on ……..; protocol No….