Proceedings of the
12th European Conference on Games Based Learning

SKEMA Business School

Sophia Antipolis, France

4-5 October 2018

Edited by

Dr Mélanie Ciussi

A conference managed by ACPI, UK

Review Process

Papers submitted to this conference have been double-blind peer reviewed before final acceptance to the conference. Initially, abstracts were reviewed for relevance and accessibility and successful authors were invited to submit full papers. Many thanks to the reviewers who helped ensure the quality of all the submissions.

Ethics and Publication Malpractice Policy

ACPIL adheres to a strict ethics and publication malpractice policy for all publications – details of which can be found here:
http://www.academic-conferences.org/policies/ethics-policy-for-publishing-in-the-conference-proceedings-of-academic-conferences-and-publishing-international-limited/

Conference Proceedings

The Conference Proceedings is a book published with an ISBN and ISSN. The proceedings have been submitted to a number of accreditation, citation and indexing bodies including Thomson ISI Web of Science and Elsevier Scopus.

Author affiliation details in these proceedings have been reproduced as supplied by the authors themselves.

The Electronic version of the Conference Proceedings is available to download from DROPBOX http://tinyurl.com/ECGBL2018. Select Download and then Direct Download to access the Pdf file. Free download is available for conference participants for a period of 2 weeks after the conference.
The Conference Proceedings for this year and previous years can be purchased from http://academic-bookshop.com

E-Book ISBN: 978-1-912764-00-6
E-Book ISSN: 2049-100X
Book version ISBN: 978-1-911218-99-9
Book Version ISSN: 2049-0992

Published by Academic Conferences and Publishing International Limited
Reading
UK
Tel: +44-118-972-4148
www.academic-conferences.org

Game Programming in Primary Schools: Guidelines for a Project-Based Learning Path

Laura Freina, Rosa Bottino and Lucia Ferlino
Institute for Educational Technologies of the National Research Council, Genova, Italy
freina@itd.cnr.it
bottino@itd.cnr.it
ferlino@itd.cnr.it

Abstract: In the "Fate il vostro gioco" project, students in the last year of an Italian primary school have been involved in a game making activity, using Scratch. The aim of the project was to define guidelines for teachers to help them design a series of lessons aimed at developing basic computational thinking skills. According to the learning by doing approach, students learn through an active participation in project-based activities, allowing them to make new connections and generate new ideas. In the first phase of the project, an introduction to coding with Scratch was carried out. In the second phase, students, divided into groups developed a game according to the requests of a "client", related to game characteristics and the type of game. Each group chose the game they wished to develop among a restricted number of possibilities. At the end of the project, a presentation was organized involving families, other classes and teachers. Scaffolding was provided to all groups according to their needs: examples of complete games were shown, the game elements were identified and, in some cases, a closer guidance was given. A lesson involving the whole class was devoted to solving some of the most difficult issues related to the development of some games. Big differences with respect to students' abilities and level of independence in managing their work was detected. Qualitative data was collected during each meeting: two trainee teachers observed the students working in groups, and students were asked to fill in a diary after each meeting, stating their opinions, problems, plans, etc. Based on the collected data and the problems that arose, guidelines for primary teachers wanting to design an intervention aimed at game making for their class are being produced. These address the organization of the groups and of their work, indications for the observations as well as the level and type of scaffolding to be offered according to the planned activities and the students' needs.

Keywords: learning by doing, game making, computational thinking, coding, scaffolding

1. Introduction

Computational Thinking (CT) provides an approach to reasoning about problems that draws on concepts fundamental to computing (Wing, 2006): abstraction, algorithmic thinking, evaluation, decomposition, generalisation (Curzon et al., 2014).

Apparently, there is not a unique definition of CT. Román-González, Pérez-González and Jiménez-Fernández (2017) analyse several different definitions trying to determine the cognitive abilities underlying CT. In doing so, they identify frameworks for developing CT (and the underlying concepts and abilities) in a classroom. Among these, they report six different concepts identified by the organization Computing At School in the UK (logic, algorithms, decomposition, patterns, abstraction and evaluation) and five approaches to working in the classroom (tinkering, creating, debugging, persevering and collaborating).

With respect to the educational value of CT during the early school years, starting from Papert's research (1980), it has been acknowledged that coding could help to reify some of such concepts and can be a useful tool to learn how to think. This is especially true when situations in which to explore "powerful" ideas are created (such as, for example, differential geometry with the turtle micro world or feedback with Lego robots).

Currently, CT and coding are again in the pipeline of school education in many countries and are beginning to appear in policy documents on school curricula. Even if some initiatives have been performed to introduce CT at compulsory school level, they are still limited in time and scope. According to Lye and Koh's review (2014) there is a lack of studies focused on the introduction of CT in typical K-12 classrooms, they find that most interventions were after school programs based on a voluntary participation. A systematic Literature Review on the subject by Lockwood and Mooney (2017) states that the introduction of CT in school curricula is still in its infancy and teachers feel the need for detailed lesson plans and curriculum structure.

The National Plan for Digital Schools of the Italian Ministry of Education includes the development of CT competencies as one of the targets of compulsory school education (PNSD, 2015). In the Ministry guidelines, it is also stated that the most simple and amusing way to develop CT skills is through coding in a gaming environment. [1] Unfortunately, often schools are not ready and teachers have difficulty in introducing CT to the students and integrate it in their lessons.

The Institute for Educational Technologies of the National Research Council has a wide experience in using game based learning in support of the development of basic skills in primary school students (Bottino & Ott, 2006; Freina, Bottino, Ferlino & Tavella, 2017; Freina & Bottino, 2018). Starting from this experience, the "Fate il vostro gioco" project was set up based on game making activities.

To support adequately the coding activities, a complete set of scaffolding interventions was defined. The term "scaffolding", introduced by Wood, Bruner and Ross (1976), refers to all those activities that support the students in performing a specific task, which they would not be able to perform alone. Scaffolding implies the presence of an expert who can give the needed support and a task that is located in the student's Zone of Proximal Development (Vygotsky, 1978). Moreover, as Pea (2004) argues, scaffolding, to be defined as such, needs to fade away as the student's abilities increase and the tasks become more accessible. In the end, students become independent and no more support is needed.

In their exploration of scaffolding in classroom systems, Davis and Miyake (2004) report a previous work where, starting from the assumption that scaffolding involves adults working face-to-face with learners, four characteristics of scaffolding are identified:

- the adult takes responsibility for encouraging a child to become involved in a "meaningful and culturally desirable activity beyond the child's current understanding or control";
- the adult engages in a process of careful diagnosis of the learner's current level of understanding or proficiency and calibration of the appropriate support that needs to be provided;
- the adult can provide a range of types of support;
- the support is "assumed to be temporary" and thus fades over time.

In general, the constant presence of an expert working with the student is not the only way scaffolding can take place: in a classroom environment help can come from the class teacher, supporting teachers, peers, paper based artefacts, technology, etc.

The paper describes the project, detailing the organization of the activities and the used scaffolding. At the end, some guidelines for teachers are outlined.

2. The "Fate il vostro gioco" project

To address the need of the Italian school, a series of lessons aimed at developing basic CT skills were designed and tested in a case study involving a grade 5 primary class of an Italian school. The aim of the project was to foster CT skills through the development of a digital game using Scratch[2], a visual online programming environment (Resnick et al., 2009; Brennan & Resnick, 2012). The methodology at the basis of the intervention followed a learning by doing approach: students learned through an active participation and by means of project-based activities.

The project was organized with the objective of defining and testing a learning path for the introduction of game making activities for students in the last years of the Italian primary school. After the definition of the learning path, a case study was set up to test it.

2.1 Methodology

Students made a direct experience of the chosen programming environment by developing simple routines at first and more complex programs later, in a classroom collaborative environment where 'trying-things-out' (tinkering) was actively encouraged. Furthermore, specific attention was paid to motivating students, helping them to persevere with specific support to overcome obstacles and encouraging collaboration between peers.

[1] http://www.istruzione.it/allegati/2015/prot2187.pdf
[2] https://scratch.mit.edu

A learning path was defined, including the following elements:

- A set of 10 lessons, about one hour long, aiming at the introduction to some of the basic elements of programming and to the Scratch online programming environment;
- A role-play with the students working in groups as game developers creating their own digital game, planned in 10 working sessions of about one hour and a half each;

After the definition of the learning path, a single exploratory Case Study was organized with the aim of testing if the planned activities would have worked in a real classroom environment and with what results.

The choice of testing the learning path through a case study was due to the fact that the evaluation had to happen in a typical classroom, involving directly the class teacher, and the influence of the specific context could not be separated from the object of the study (Yin, 2003).

Data was collected during the whole case study through observations, evaluation of the output of the coding activities, students' diaries and final questionnaires. The collected data is still under examination.

Finally, guidelines for teachers wanting to organize a similar intervention were defined.

2.2 Participants and location

One grade 5 class of an Italian primary school (10-11 years old students) took part to the project for the whole school year. The class included twenty-four students, with a high number of Spanish speakers.

The class teacher was an active part of the project, organizing the students, managing the class and introducing each activity to them. Two researchers and four trainee teachers from the University of Genova (two in the first part of the project and another two in the second part) were present at all the meetings. Researchers supported the class teacher in the organization of the meetings and offered technical help to the students while trainee teachers were mostly involved in observing the work done by the students, their behaviour, group dynamics, problems that may arise and their solutions, etc.

The activities took place in the school computer laboratory and in the classroom, and were organized during school hours, once a week starting from November 2017 to April 2018.

2.3 Introduction to Scratch programming

The first 10 meetings were devoted to an introduction to the Scratch programming environment. Several different activities were suggested to the students, working in pairs, trying to encourage their curiosity and free exploration of the tool.

The project started from the implementation of a simple "Pacman" game, aiming at giving the students a general idea of Scratch. Students were given detailed instructions, ready-made graphics and parts of the needed routines.

The following meetings were devoted to exploring Scratch: specific hints were given to the students and then they were left free to explore. For example, one graphic effect was shown, and then students were free to try all the other available effects.

Some exercises were then presented, asking the students to implement simple routines with an explicit purpose. For example, the image of a cat and a cushion were given and students were asked to write the code to allow the player to move the cat to the cushion with the mouse. When the cat reached the cushion it had to say "Thank you!". Furthermore, small projects either incomplete or containing errors were then given, asking students to modify or complete the code to make them work properly. Special attention was paid to the use of variables, which were introduced as a way to record the player's score, and further detailed as the students' coding activities became more complex.

Towards the end, two simple games were implemented: a labyrinth and a simple game in which falling apples had to be collected with a basket. In these cases, instructions were given only on paper, and the students had to develop the whole game in Scratch.

2.4 The role play

In the second phase of the project, students were asked to develop a complete digital game. A role-play was organized: the class was divided into six groups of four students playing role of game developers, while a researcher acted as the client, asking each group to develop a game. In a Skype meeting with the class, the client showed some examples of games to choose from and fixed some basic requirements. They were all simple well-known games, and a working sample of each was shown to give students a clear idea of how the finished game could be.

After the games were presented, each group chose the one they wished to develop, and a very brief verbal description of the game, along with specific requirements was given to them.

The chosen games were:

- Labyrinth – a typical labyrinth with some coins that had to be collected before reaching the exit.
- Memory – a set of cards containing two copies of each card. The cards were placed back up and the player had to find the two cards that were the same (two groups implemented a memory game).
- A simple jigsaw with no rotation – an anime image was broken into pieces and the player had to put it back together by moving the pieces to their correct place.
- Another jigsaw with rotation only – a football field was broken and the player had to fix it by rotating the pieces to make them match.
- Educational Balloons – some balloons raised in the air, when the player clicked a balloon, it burst and a question on a school subject was asked. If the answer was correct, the player got a point.

All the games had to have:

- Short and clear instructions for the player;
- Both written and spoken instructions;
- All texts in Italian and Spanish;
- A bright and coloured interface;
- A beginning and an end.

While other elements were optional:

- Different levels of difficulty;
- A score;
- A time limit;
- Some other optional characteristics depending on the single games.

At the end of the project, a meeting with the client, the head of the school and some students from other classes was organized to present the games.

2.5 Types of scaffolding

A complete scaffolding structure was defined to support the students, based on a theoretical analysis of the envisaged difficulties in the game making activity. A list of the competences needed for CT, the specific programming skills involved, and the foreseen difficulties related to the single games gave the basis for the definition of the needed scaffolding.

Distributed scaffolding was employed, including both unstructured help from peer and experts and structured support (printed explanations and other printed supports, ready-made Scratch routines, game samples, etc.). These include both techniques aimed at simplifying and structuring students' work and other techniques that problematize some aspects of the work by focusing students on specific aspects of the task to be performed (Reiser, 2004). In the following a detailed description is given.

2.5.1 Students' motivation

In the role-play, the presence of an external "real" client was highly motivating. The Skype meeting at the beginning of the case study obtained a high participation from the whole class. The client sent regularly his personal comments to the groups and took part to the final project demonstration, in which all the groups presented their work.

2.5.2 Complete game examples and brief description

Before starting the coding phase, a complete working version of all the games to choose from were shown and students had the possibility to play with them for a short time. The main objective of this activity was to help them create a mental representation of the games and define their personal version. Along with this, a brief description of the games and the characteristics they had to have were given to each group.

2.5.3 Specific incremental support for each game

For each of the chosen games, in order to offer the students the correct amount of support needed, several different levels of help were created in advance:

- A general description of the game;
- A list of the game elements;
- A detailed description of each part of the game in Italian;
- A detailed description of each part of the game enriched with some Scratch programming blocks;
- Some of the most difficult routines were given ready-made to the students. For example, the measurement of the distance between the actual position of an object and the correct one.

At the first coding lesson, the general description of the game was given to all the groups, pushing the students to try to design their own version of the game. The teacher and the researchers continuously monitored the ongoing work, intervening as they noticed that a group had problems.

Starting from the second meeting, incremental scaffolding was offered, trying to reach the level in which the group would understand the task and take over actively. At first, the help offered was only relative to the initial part of the games: player instructions, switch between the Italian and the Spanish version of the game, and start button. In the following meetings, the quantity of help offered was evaluated on the single groups' achievements in the coding activities and, when possible, the level of help was decreased. At the end of the project, all the groups were still using some kind of help to enable them to create all the parts of their game.

2.5.4 Structured sheet for game analysis

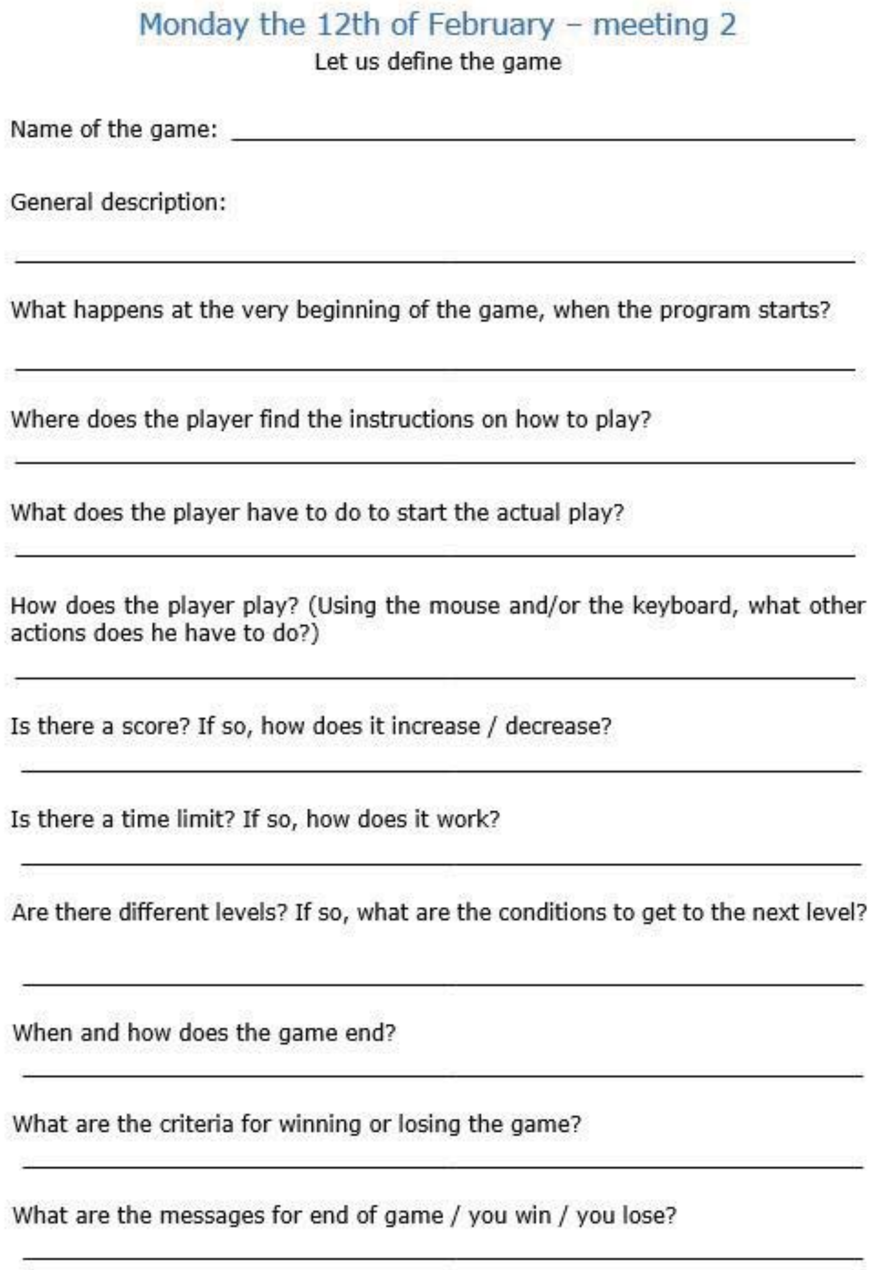
At the beginning of the case study, after each group of students had chosen the game to develop, pre-defined sheets were used to help students build a clearer idea of the game and identify its main elements.

Students had to name and describe the following:

- Backgrounds;
- Objects (their looks, behaviours, sounds);
- Texts in Italian and Spanish;
- Music and/or sounds;
- Variables (e.g. to keep the score).

Not all the questions were fully answered at this stage: it is not easy to imagine the whole game on paper. Nevertheless, the activity was very useful to help the groups focus on their objective, try to imagine how the game should work and identify the main elements needed.

Figure 1 shows an English translation of the structured sheet used.



Monday the 12th of February – meeting 2

Let us define the game

Name of the game: ______________________________

General description:

____

What happens at the very beginning of the game, when the program starts?

____

Where does the player find the instructions on how to play?

____

What does the player have to do to start the actual play?

____

How does the player play? (Using the mouse and/or the keyboard, what other actions does he have to do?)

____

Is there a score? If so, how does it increase / decrease?

____

Is there a time limit? If so, how does it work?

____

Are there different levels? If so, what are the conditions to get to the next level?

____

When and how does the game end?

____

What are the criteria for winning or losing the game?

____

What are the messages for end of game / you win / you lose?

____

Figure 1: Example of a structured sheet.

2.5.5 Whole class meeting

As the students started coding with Scratch, some doubts were solved, but some other problems arose. A list of the main issues from all the groups was made and a special meeting was devoted to devising possible solutions involving the whole class together.

Even though the questions regarded a specific game to be developed, they could be of interest also for the other groups; therefore, it was decided to involve all the students in the discussion. After an issue was presented, a student tried to find a solution using Scratch on the interactive whiteboard with the active participation of the whole class.

For example, in jigsaw and memory games, it is needed to know when the last pieces have found their place or the memory cards are finished. To address this issue, the problem was isolated and students were asked to write a simple Scratch project in which objects on the screen disappeared as they were clicked. When the last object disappeared, an end of game message had to be shown.

2.5.6 Real life performance

Sometimes, a real life representation was needed to explain students the details of a problem to be solved. For example, to address the problem of understanding when the chosen cards in a memory game were the same, a small performance was organized.

A researcher played the role of the computer, using a piece of paper as a "variable" to store data temporarily. Four cards were put on the table, identified with a "type-of-card" information. The "computer" was not allowed to see the cards, so, when the player turned a card, he had to tell the "computer" the type of card that was selected. The "computer" used the "variable" either to memorize the type of card (when the first card was turned), or to compare it to the previous card and decide if the two chosen cards were the same or not. The "variable" was cleared every time a couple of cards had been processed.

Students were asked to take note of what was happening in the performance and then to "translate" their notes into Scratch programming blocks.

2.5.7 Tutor support and monitoring

The role of the expert in scaffolding activities is to encourage and support the students' work, as well as diagnose the level reached in order to be able to decide the right amount of help needed. In the project, the participation of the two researchers and the trainee teachers allowed to follow closely each group and tailor the help to their needs.

Observation sheets were defined to guide both data collection for later analysis and the definition of the scaffolding to be delivered. The observation sheet focused on the ability of each student to organize the work, solve problems, use the tool, ask for help, be autonomous, as well as being active and participating to group work.

The class teacher, during the last two meetings, focused on one group only, asking the most skilled member to explain her what they were doing. By explaining, the skilled member managed to gain a deeper understanding of the game and, at the same time, helped the rest of the group in understanding the task.

Another group was in difficulty due to lack of internal organization and the presence of a couple of weaker students with respect to the others. One of the trainee teachers concentrated on this group, working with them at every meeting. The very tight monitoring was effective: the group managed to develop a simple, yet complete labyrinth game.

The researchers and the other trainee teachers did not focus specifically on any group, but concentrated on understanding the right kind of scaffolding needed by each group and promoting a peer-to-peer support both within and between groups.

2.5.8 Other support

Since the time available for project development was rather short, the researchers decided to offer specific help for some activities:

- Students actively searched the web for images to be used in their games (for the memory cards, the football field, the image to be put back together, etc.), while researchers took care of image processing: making all the pictures of the same size, splitting up images, etc.
- Students defined the text for the player's instructions in both Italian and Spanish. Researchers helped in the voice acquisition phase: while students read the instructions, the researcher managed the acquisition and editing of the sound.

2.6 Guidelines

Based on the collected data and the problems that arose, guidelines for primary teachers willing to design an intervention aimed at game making for their classes have been produced. These address the organization of the groups and of their work, indications for the observations as well as the level and type of scaffolding to be offered according to the planned activities and the students' needs.

The following main aspects give an overview of the guidelines:

- The lessons to be planned should be tailored to fit the students' previous experience with the chosen coding tools and their programming skills. Some lessons may have to focus on the acquisition of the basic knowledge needed for later development. What we found works best, is to list the needed concepts, focus on one concept at a time and start in a guided manner, leaving students free to explore only after they have been introduced to the main concepts.
- Class organization will have to be tailored to the specific case. Working in groups allows enhancing each person's specific ability and encourages peer support. Nevertheless, negotiations and communications within the group have not always been very smooth. Furthermore, there are often difficulties due to the limited number of computers available.
- Students are used to playing with all sorts of digital games, with very complex game plays and fine graphics. When defining their own game, they tend to overestimate their possibilities and invent very complex games. For this reason, they have to be guided towards reachable objectives.
- In the experience, the role-play approach was highly motivating. As Riser states, one of the characteristics of scaffolding is the student's motivation, "… eliciting students' commitment of attention and resources to reasoning about an aspect of a problem" (Reiser, 2004, p.287). It is important for the teacher to be able to keep the students' interest high.
- When designing the intervention, it is suggested to device scaffolding in support of different levels of proficiency, possibly tailored to the organization of the class (i.e. students working individually or in groups). Defining different modes for giving support is also advisable: printed instructions, ready-made routines, examples, tutoring, peer support can all be mixed in a synergic distributed scaffolding. Often asking students to rehearse a routine by acting it in the real world can help in an active and concrete manner.
- Observation sheets specifically defined may be very helpful in directing the attention towards various elements of the students' work and attitude in class. This is fundamental to be able to decide, at each meeting and for each group involved, the level and the kind of help needed.

3. Conclusions

A single exploratory case study was organized to evaluate a project oriented learning path aiming at fostering CT skills in primary students through a game making activity. The project was divided into two parts: an introduction to the Scratch online visual programming environment, followed by a group activity aiming at the development of a simple digital game.

Before starting the case study, a complete outline of the introductory lessons, the possible games to be implemented and a complete scaffolding structure had been defined. The case study aimed at assessing how the learning path would have been accepted and integrated into other learning activities of a typical Italian primary school. A result of the project are guidelines for teachers supporting the design of similar interventions. These guidelines stem from the experience of the case study and aim at bringing to awareness possible issues while giving suggestions on the preparation of a complete intervention. Data collected during the case study, in terms of observations, students' diaries, final questionnaires and interviews with students and with the teacher is currently under examination; therefore, guidelines may be enriched in future.

Unfortunately, due to time limits, it has not been possible to reach a complete fading of the scaffolding required. Furthermore, transfer of learning for autonomous performance has not been assessed. This is a limit of this experience since the real objective is to foster students' CT abilities, not simply being able to program a simple game in Scratch. Experience shows that, in some cases, it may not be possible for scaffolding to fade completely since some students demonstrated a weak comprehension of the approach and difficulty in using logic. Probably these students may need a much longer intervention, guiding them towards a more autonomous approach to problem solving. Further and longer projects will be needed in future in order to give students the chance to apply the newly acquired knowledge to other subject areas, consolidating it while demonstrating learning transfer.

Acknowledgements

The authors would like to thank the trainee teachers: Luca Nervi, Serena Proietto, Elena Novella and Fabiana

Pastorino for valuable support during all the meetings, the class teacher Laura Benedetti for her availability and support and all the students of the 5A class from the "Scuola primaria Cantore" in Genova (Italy) for their enthusiastic participation, and finally Giovanni Caruso for playing the part of the client in the role-play.

References

Bottino, R.M., and Ott, M. (2006) "Mind games, reasoning skills, and the primary school curriculum: hints from a field experiment", *Learning Media & Technology*, vol. 31, No. 4, 359-375.

Brennan, K., and Resnick, M. (2012) "New frameworks for studying and assessing the development of computational thinking", *Proceedings of the 2012 annual meeting of the American Educational Research Association*, Vancouver, Canada, pp. 1-25.

Curzon, P., Dorling, M., Ng, T., Selby, C., and Woollard, J. (2014) *Developing computational thinking in the classroom: a framework*

Davis, E. A., and Miyake, N. (2004) "Explorations of Scaffolding in Complex Classroom Systems", *The Journal of the Learning Sciences*, Vol. *13, No.* 3, pp. 265-272.

Freina, L., and Bottino, R. (2018) "Visuospatial Abilities Training with Digital Games in a Primary School", To be published in *International Journal of Serious Games*, due in September 2018.

Freina, L., Bottino, R., Ferlino, L., and Tavella, M. (2017) "Training of Spatial Abilities with Digital Games: Impact on Mathematics Performance of Primary School Students", *Proceedings of the Game and Learning Alliance International Conference (GALA*) (December 5-7, 2017, Lisbon, Portugal).

Lockwood, J., and Mooney, A. (2017) *Computational Thinking in Education: Where does it fit? A systematic literary review*.

Lye, S.Y., and Koh, J.H.L. (2014) "Review on teaching and learning of computational thinking through programming: What is next for K-12?", *Computers in Human Behavior*, Vol. *41*, pp. 51-61.

Papert, S. (1980) *Mindstorms. Children, computers and powerful ideas*, New York: Basic Books.

Pea, R.D. (2004) "The Social and Technological Dimensions of Scaffolding and Related Theoretical Concepts for Learning, Education, and Human Activity", *The Journal of the Learning Sciences, Vol. 13, No. 3*, pp. 423-451.

PNSD (2015) *Piano Nazionale Scuola Digitale* (PNSD), [online], http://www.istruzione.it/scuola_digitale/allegati/Materiali/pnsd-layout-30.10-WEB.pdf

Reiser, B.J. (2004) "Scaffolding Complex Learning: The Mechanisms of Structuring and Problematizing Student Work", *The Journal of the Learning Sciences*, Vol. 13, No. 3, pp. 273-304.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... and Kafai, Y. (2009) "Scratch: programming for all", *Communications of the ACM*, Vol. 52, No. 11, pp. 60-67.

Román-González, M., Pérez-González, J. C., and Jiménez-Fernández, C. (2017) "Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test", *Computers in Human Behavior*, Vol. 72, pp. 678-691.

Vygotsky, L. S. (1978) *Mind and Society: The Development of Higher Mental Processes*.

Wing, J. M. (2006) "Computational thinking", *Communications of the ACM*, Vol. 49, No. 3, pp. 33-35.

Wood, D., Bruner, J. S., and Ross, G. (1976) "The Role of Tutoring in Problem Solving", *Journal of Child Psychology and Psychiatry*, Vol. 17, No. 2, pp. 89-100.

Yin, R.K. (2003) *Case study research: Design and methods*, Thousands Oaks, Sage.