Collaborative Game-Based Environment and Assessment Tool for Learning Computational Thinking in Primary School: A Case Study

María Zapata-Cáceres, Estefanía Martín-Barroso, and Marcos Román-González

***Abstract*—Computational thinking (CT) can be considered a 21st century core skill and, accordingly, it should be taught to students at an early age. Nevertheless, the implementation of CT in school curricula is still in an experimental stage, given that different performance metrics remain unclear, including the appropriate age for learning each skill, the significant and achievable computational concepts for each school year, teaching strategies, learning effectiveness, and strategies for assessing development. Based upon constructivist problem-solving learning strategies and supported by a three-dimensional framework, a game-based environment with both individual and collaborative playing modes has been developed as a learning and an assessment tool via learning analytics. Moreover, an exploratory case study has been carried out involving 176 primary school students. Results suggest that this environment is suitable as a learning and assessment tool for CT skills in primary school, providing enduring learning, particularly when it engages the collaborative playing mode. It seems to be better adapted to early primary school stage students, who showed greater improvements and who were able to assimilate more computational concepts than expected. Likewise, special needs or low percentile students benefit even more greatly from the learning tool and especially from the collaborative playing mode. The combination of different assessment methodologies—including CT pre- and posttests, data-driven analytics, direct observation, and questionnaires—provides assessment for each of the framework computational dimensions.**

***Index Terms*—Assessment, collaborative learning, computational thinking (CT), early childhood education, educational games, learning environments, programming.**

I. Introduction

COMPUTATIONAL thinking (CT) is a core skill that is not just limited to computer scientists' activities but that can be extensively applied in daily life and is essential to adapt to the future [1]. CT was first defined as a "human problem-solving process that uses decomposition and requires thinking at multiple levels of abstraction" [2]. However, many definitions of CT have been suggested since then, and can be categorized into three [3]: 1) generic definitions that focus on the skills involved in the problem-solving (e.g., Aho definition [4]); 2) operational definitions that provide different CT breakdowns (e.g., Shute *et al.* breakdown [5]); and 3) educational definitions or frameworks for the development of CT in the classroom (e.g., 3-D framework [6]).

Nowadays, in a high-tech, constantly changing world, students must be able to think critically and solve complex problems [7]. Accordingly, initiatives are being undertaken in various parts of the world to include CT concepts in compulsory education curricula at schools [8]. However, before introducing curricula for CT development in primary school, it is essential to determine what CT skills children should learn and how the acquisition of such skills should be assessed [9], considering the varying cognitive ability of students depending on their age [1].

As a contribution to research on this area, a theoretical background has been established based upon social constructivism [10] and cognitive development theories [11]; frameworks such as activity theory [12]; and on established learning strategies.

Although CT in the classroom can be developed through different activities, it is mainly taught through programming [13], moreover, visual game-based environments are powerful instruments for this learning [9]. Based on the theoretical background, a constructionism-based problem-solving game-based environment: Blue Ant Code (BAC), has been developed specifically for this research project as a learning and an assessment tool. It includes individual and collaborative playing modes. For categorization of CT dimensions, we will refer to the 3-D framework [6] since its context is similar to that of this article.

An exploratory case study in primary school drawn upon a sample of 176 students, has been carried out using the following instruments:

1) the beginners CT test (BCTt), for the assessment of CT skills at an early age [14];
2) the developed environment (BAC) as a learning and assessing tool;
3) several questionnaires to supplement the assessment.

Adequacy of learning and assessing strategies, as well as contents for the implementation of CT in primary school education, has been evaluated accordingly. Implementation of the three instruments in the same case study aims to cover the three computational dimensions (see detailed in Sections II-B and II-C) of the 3-D framework, both in terms of learning and

Manuscript received August 6, 2020; revised March 19, 2021 and August 25, 2021; accepted September 5, 2021. Date of publication September 13, 2021; date of current version January 17, 2022. This work was supported in part by the Madrid Regional Government through the e-Madrid-CM Project under Grant S2018/TCS-4307, and in part by the two European Structural and Investment Funds, namely the European Social Fund and the European Regional Development Fund. (*Corresponding author: María Zapata-Cáceres.*)

María Zapata-Cáceres and Estefanía Martín-Barroso are with the Computer Science Department, Universidad Rey Juan Carlos, 28933 Móstoles, Spain (e-mail: maria.zapata@urjc.es; estefania.martin@urjc.es).

Marcos Román-González is with the Faculty of Education, Universidad Nacional de Educación a Distancia, 28040 Madrid, Spain. (e-mail: mroman@edu.uned.es).

Digital Object Identifier 10.1109/TLT.2021.3111108

assessing approaches. This article specifically aims to answer the following research questions.

RQ1: Is a game-based strategy suitable as a programming learning methodology and assessment tool (via data-driven analytics) for CT skills throughout primary school?

RQ2: What CT skills can start being learned at each stage of primary school?

RQ3: What are the differences between collaborative and individual game-based learning approaches in primary school and which strategy is appropriate for each age?

II. Theoretical Perspectives and Rationale

A. Computational Thinking: Definition and Framework

There is a growing international movement that focuses on the 21st-century skills required for students to succeed within a constantly changing digital society. These refer to a wide range of skills such as learning and innovation skills, information, media, and technology skills [15]. Of these, learning and innovation skills are usually classified into four blocks known as the 4Cs: 1) critical thinking; 2) creativity; 3) collaboration, and 4) communication [16]. These have been increasingly acknowledged as essential ingredients of school curricula [17].

In addition, CT is a basic skill to be placed alongside other analytical skills, together with reading, writing, and arithmetical processes, in order to adapt to the near future [18]. Shute *et al.* [5] defined CT as "the conceptual foundation required to solve problems effectively and efficiently with solutions that are reusable in different contexts". CT can be considered another core skill or even the "5th C" of the 21st century and, therefore, should be taught to all students [5], [19], [20].

Wing first defined CT [2] and since then many other definitions, breakdowns and frameworks of CT have been suggested which has encompassed broad debates [6], [9]. Shute *et al.* [5] categorized CT skills within the four categories that appear most often in the literature (abstraction, decomposition, algorithms, and debugging) and add iteration and generalization. Zapata-Ros and Pérez-Paredes [21] included numerous components, such as creativity and divergent thinking. Within this last CT decomposition approach, CT elements overlap 21st-century skills to some extent.

The existence of multiple: 1) generic definitions; 2) operational definitions or breakdowns; and 3) educational definitions or frameworks, and the consequent difficulty in precisely identifying exactly what CT is, makes it difficult for educators and researchers to decide what should be taught or evaluated [22], [23]. From them, the 3-D framework educational definition covers a broad spectrum of CT [24], and many curricula emphasize the learning of basic CT through its CT concepts [25]. The 3-D framework emerged from studied by Brennan and Resnick [6] on the activities of young programmers from around the world using Scratch, the most popular visual block-based environment used in K–12 for designing CT learning activities and as a learning tool [2], [26]. Scratch is based on visual, and process thinking, emphasizes the social attribute of CT and has strong operability in educational practices [27]. The 3-D framework categorizes CT according to three dimensions (see Table I): 1) computational concepts (concepts that programmers use); 2) computational practices (problem-solving practices that occur in the process of programming); and 3) computational perspectives (perspectives designers form about themselves and the world around them).

TABLE I
Key Dimensions of the 3-D Framework as Adapted to a Beginner's Level

Dimensions	Items	Beginner's level
Computational concepts	Sequences	Yes
	Loops	Yes
	Events	No
	Parallelism	No
	Conditionals	Yes
	Operators	No
	Data	No
Computational practices	Being incremental and iterative	Yes
	Testing and debugging	Yes
	Reusing and remixing	No
	Abstracting and modularizing	Yes
Computational perspectives	Expressing	Yes
	Connecting	Yes
	Questioning	Yes

The 3-D framework seems appropriate for contextualizing CT in a collaborative visual programming learning environment designed for early ages, such as the one developed (BAC). It will, therefore, be adopted here. Nevertheless, as this research aims at learners who are younger than those involved in the Brennan and Resnick study and the environment is not Scratch, the 3-D framework has been adapted to a beginner's level and to the BAC environment (see Table I).

Although there is a progression of concept learning in Scratch or Scratch Jr for different age groups [28], the computational concepts selected are those that have been proven to be achievable by primary school students [14], [28]–[30], and, therefore, can be assessed by the BCTt [14]. The computational practices and perspectives selected are those that are related to the concepts addressed [29]—for example, although abstracting is known to be difficult for young students, this practice might be observed while students interact with BAC and with their peers in the collaborative mode [6], [29]. However, *reusing and remixing* has not been included as BAC is not a game creation environment, as Scratch is, so there is no code reuse option.

A review of the literature reveals that CT is an important subject in the national education curricula in many countries. Indeed, some of them have structured educational programs centered around CT [1]. The promotion of CT in education has made great progress over the last decade, starting as early as kindergarten in some countries, such as Australia or Japan [8]. Children are being trained in CT, independent thinking and problem solving so that they can then apply these skills to different disciplines or to daily life. However, there remains some hesitation regarding how best to teach CT, when to begin such instruction or what to teach [7] and there is especially scant research on computational practices in early grades [31].

B. Learning Strategy

Programming exposes students to CT and, therefore, to problem-solving through computer science concepts such as

TABLE II
MOST APPLIED LEARNING STRATEGIES

Learning strategy	Definition
Problem-based learning	Problems are the stimulus and focus of student activity: a problem scenario is presented to the students, so they should explore the solution by themselves, setting their own learning goals [33]
Collaborative learning	Concerns the following items: situation, interactions, processes and effects. The relations between the four items are key, as group members complete a task or situation together, negotiate, and share opinions and knowledge to solve the problem [34], [35].
Project-based learning	Is focused on teaching by engaging students in the investigation of a complex task based on a challenging question or a problem [36].
Game-based learning	Is a problem-solving play framework where a challenge or a problem-solving scenario is created, so students search for solutions using game mechanics with a sense of achievement, while enhancing knowledge and skills acquisitions [17], [37], [38].

abstraction and decomposition [32]. Over the past decade, scholars have attempted different learning strategies to help students learn to program to improve CT-related skills; most recent research focuses on problem-based learning, collaborative learning, project-based learning, and game-based learning [1] (see Table II).

The project-based strategy has been discarded as students need to be cognitively engaged with the subject matter over a long period of time [36], which exceeds the scope of this research. In BAC, the problem-based strategy has been the main approach adopted, the game-based strategy has been applied as a scaffolding vehicle to guide and ease the solving process through game mechanics, and the collaborative strategy has been selected as a transversal instrument to enhance the learning process.

Most research focuses on computational concepts 3-D framework dimension [28], [32] and there is recent research on learning and assessment of computational practices and perspectives dimensions [25], [28]–[30], [39], however, more research is needed, especially beyond the Scratch environment, as CT is an interdisciplinary set of skills and there should be operational methods of learning and assessing CT, and through diverse learning strategies [29], [32]. Computational practices and perspectives are even more pertinent for K–12 settings, since the rationale for introducing CT through programming is to equip students with problem-solving skills that they can later transfer to nonprogramming domains [40]. In this article, three learning strategies are combined.

Shown in Fig. 1 is the expected impact of each learning strategy within the key dimensions of the 3-D framework considered in this article: computational dimensions are located at the triangle vertices so that the closer the strategy appears near a vertex, the greater likelihood this dimension will be impacted by the learning strategy.

In contrast to problem-based learning strategies, in game-based learning strategies the student does not necessarily receive a prior definition of concepts, but guides his or her own learning freely, motivated by a sense of achievement (see Table II), so concept acquisition may not be complete. In

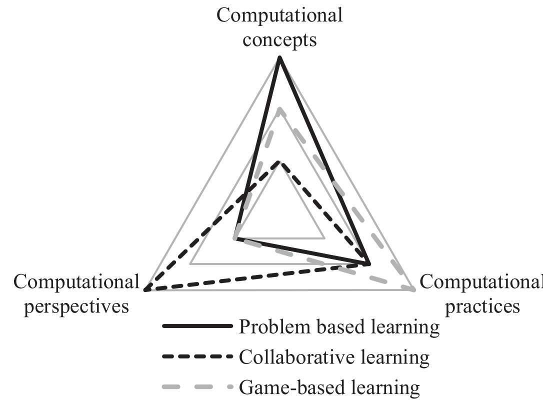

Fig. 1. Expected impact of learning strategy to the key dimensions.

contrast, computational practices are particularly enhanced, as its mastery is essential to progress in the game [32]—for example, students may not know that they are using *sequences* but have completed a BAC level by figuring out the optimal sequence among different alternatives, so the computational practice of *testing and debugging* has also been mastered. Additionally, it has been shown that computational perspectives are enhanced by collaborative strategies (see Section II-D).

C. Learning Theory

Based on Vygotsky's social constructivism [10] and Piaget's theory of cognitive development [11], constructivism in education postulates a dynamic, collaborative, and interactive learning process by means of which knowledge is actively created by the student himself. Diverse theoretical frameworks have emerged in response to technological and pedagogical innovations. Papert́s constructivist learning framework helps students learn social interaction, cognition, high-level thinking, and CT [41]. Activity theory [12] provides an appropriate framework for analyzing needs, tasks and outcomes when designing constructivist learning environments, since its assumptions are consonant with those of constructivism, situated learning, case-based reasoning, social cognition, and everyday cognition; furthermore, activity theory has been widely used to provide a clear operational framework for designing these environments.

In this article, the main approach has been a problem-based learning strategy along with game based and collaborative learning strategies. Problem-solving environments are consistent with the principles of constructivism since they focus on learners as constructors of their own knowledge. Students are expected to think creatively and critically, and monitor their own understanding. Additionally, the social negotiation of meaning is an important part of collaborative problem-solving environments [42], [43].

Positive outcomes for game-based learning predominate studies based on learning theory [44] since "designing educational games is an interdisciplinary process, which requires a deep understanding of game design theory, knowledge on the academic topic, and foundation in relevant learning theories" [45]. Along with flow theory, constructivism (e.g., the sociocultural theory of learning), which holds that learning takes place in social, active, and situated environments [10], [17], is one of the major theoretical foundations used by game-based learning researchers in science education [20], [46], [47]. In line with Papert's constructivist learning framework [41], in virtual game environments students solve learning challenges through interaction and collaboration [1]. Moreover, flow theory is key to fostering motivation in games and learning since the player activity is driven by pleasure rather than external rewards [17], [48], [49].

D. Learning Environment

In this case study, BAC has been developed as a collaborative scaffolded problem-solving visual environment and a CT learning and assessment tool. BAC is based on the learning strategies and theories discussed previously, supported by the 3-D framework and is in line with Lye and Koh's proposal [32] to create a constructionism-based problem-solving learning environment in order to support all 3-D framework computational dimensions.

Research by Grover and Pea on CT environments and tools for K–12 education concluded that visual game-like programming tools enhance student creativity and problem-solving skills [9]. Research by Lye and Koh [32] found that CT was mainly taught through programming and concluded that the scaffolding process is a key element in CT development. Buitrago Flórez, *et al.* [50] claimed that CT-related skills should be taught in primary school to foster students' cognitive development at an early age. They also highlight the importance of peer-based collaborative environments for meaningful programming learning experiences.

Collaborative learning promotes critical thinking skills much more effectively than individualistic environments [51]. This results in greater levels of achievement and higher productivity, as well as greater psychological health, social competence, and self-esteem than individual learning [52]. Hainey *et al.* [53] presented a systematic literature review of research on game-based learning, concluding that further studies comparing single and collaborative play are needed to identify their pedagogical benefits in primary school. There is also evidence that cooperative teams achieve at higher levels of thought and retain information longer than learners who work quietly as individuals [54], [55]. BAC can switch between individual and collaborative play mode to compare the two learning strategies.

Reflection is a strategy often used in studies involving programming experiences and may enhance both computational practices and computational perspectives dimensions, since students review and think about their programming experience thereby improving their own learning process [32]. In a collaborative environment, students reflect not only on their programming experience but also review and think about their peer's programming performance. Moreover, thinking aloud, as well as discussing peer concerns, may also enhance these two dimensions of the 3-D framework CT, especially for computational perspectives [27], [32], [56].

According to Qian and Clark, the effectiveness of a game-based learning environment depends on game designs that are founded on established learning theories and on game elements that have proven successful [17]. Many studies claim that CT skills can most effectively be taught using programming languages [1]. Furthermore, visual programming environments may improve beginner programming skills and strategies, as well as foster greater engagement with programming tasks. Additionally, they may assist students in modeling, simulation, and problem solving, which are key in CT development [32]. Thus, it is worthwhile to integrate game elements into visual environments and explore their influence on learner experience and motivation to solve computational problems. Therefore, game-based programming environments, such as BAC, can impact all 3-D framework dimensions and particularly enhance computational practices (see Fig. 1).

Finally, digital tablets appear to be suitable devices for learning environments at early ages. The integration of multiple features within one device and the touch interfaces are key facets making tablets suitable as learning tools in schools [57]. Peer collaboration enhances, given that digital tablets provide a more natural and direct interaction with the content [58], [59]. Moreover, Lin *et al.* [60] concluded that the use of tablets improves learning outcomes, especially in the cases of many-to-one groups, as more peer collaboration and higher quality artifacts are produced.

E. Assessment

Several researchers emphasize the importance of student assessment for pedagogical purposes, as measurement and evaluation are essential for the introduction of CT into the curriculum [22]. Though again, there is even less agreement on which strategies are best suited to assess the development of CT at early ages. Tang *et al.* [61] identified four types of CT assessment: traditional test, portfolio, survey, and interview. Most studies employed traditional test and portfolio assessments, and few of them reported reliability and validity evidence of their assessment. Attempts to measure and assess CT include TechCheck CT unplugged assessment for very young students [62]; Chen *et al.* [63] pre- and postinstrument aimed at fifth-grade students, the CTP-Quiz instrument [64], Fairy assessment in Alice as a portfolio assessment based in a specific environment [65], and Garneli and Chorianopoulos [66] evaluation based on Scratch projects. In addition, the test for measuring basic programming abilities [67] and commutative assessment [68] are both validated instruments under a psychometric approach but aim at middle and high school students. There are several studies and instruments involving assessments of CT through Scratch [29], such as Dr, Scratch [39] where computational practices are evaluated along with concepts.

Román-González *et al.* [3] developed a CT test (CTt), which stands out as an instrument for the assessment of CT, as evidence of reliability and criterion validity were provided under a psychometric approach [69], [70]. It is consistent with the Mühling *et al.* [67] and with Weintrop and Wilensky [68] tested and aligned with the international standards for psychological and educational testing [71]. Even though CTt is aimed at students between 10 to 16 years old, it was a consolidated and firm basis for the BCTt [14]. This BCTt is aimed at primary school and has been validated in terms of content and reliability and, thus, has been applied in this article.

There is a need for "systems of assessments," to evaluate deeper learning combining different data measures [72], [73] In terms of the 3-D framework, the BCTt focuses on computational concepts and, partially, on computational practices, whereas ignoring computational perspectives. To cover a broader area of the 3-D framework dimensions, as a system of assessments, data-driven real-time learning analytics have been combined with the BCTt, in line with recent research outcomes that demonstrate the adequacy of assessing CT through dynamic information, which can reveal learner's abilities and progression over time, and particularly using internal analytics to collect data on game-based environments. Through the analysis of the actions or code proposed by the student, computational practices used could be identified [74] and, through the actions performed (e.g., analyzing students' play style in the collaborative mode), computational perspectives might be assessed to some extent. The assessment of CT through internal analytics is aligned with flow theory since learning via game play can continue smoothly while assessments are unobtrusively handled so that flow is maintained [75]–[77].

To assess 3-D framework computational perspectives, qualitative data were collected via direct observation of the students while they interacted with the environment and their mates. Finally, questionnaires were made to complete and compare the data collected.

The expected impact of the assessment approaches used in this study within the 3-D framework is shown in Fig. 2: computational dimensions are located at the triangle vertices so that the closer the assessment approach is represented to a vertex, the greater the likelihood that this dimension will be impacted by the approach.

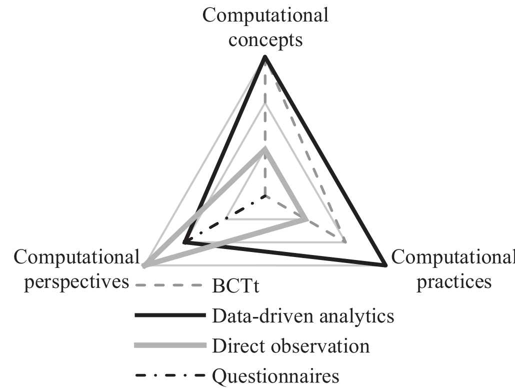

Fig. 2. Assessment approaches expected impact on the key dimensions.

III. Method

A. Instruments

In keeping with the theoretical background, three main instruments have been applied as a system of assessments in order to carry out an exploratory case study in primary school and answer the research questions: the BCTt, aimed at assessing CT skills, a collaborative game-based environment: BAC, as a learning and assessment tool, and several questionnaires to supplement the assessment.

1) Beginners Computational Thinking Test: The BCTt (see Section II-E) has provided evidence of content validity and reliability for the assessment of CT in primary school students [14]. The authors report high internal consistency reliability (Cronbach's alpha $\alpha = .824$) of the BCTt. Therefore, the BCTt seems to be a proper instrument that can be used in research requiring assessment of computational concepts and (partially) computational practices throughout primary education.

Consequently, in our present case study, the BCTt has been used as a pretest and posttest quantitative assessment data collection tool (see Fig. 2). BCTt is 25 items long, with three alternative responses per item, and has an estimated time of completion of 40 min. It is divided into six sets, each related to one basic computational concept (1–6: sequences; 7–11: simple loop; 12–18: nested loop; 19–20: if-then; 21–22: if-then-else; 23–25: while). [14].

2) Game-Based Learning Environment and Assessment Tool: A visual game-based problem-solving environment: BAC, was specifically designed and developed in correspondence with the theoretical framework (see Section II). BAC is designed to be both a learning instrument as well as an assessment tool via data-driven real-time analytics.

Special attention was paid to game design elements since the effectiveness of game-based learning seems to depend on them [17]. Thus, collaboration, complexity, competition, strategy, adapted challenges to maintain flow, clear goals, communication, interactivity, and scaffolds are the main game design elements implemented within the environment. Likewise, user experience precepts have been taken into careful consideration and adapted for each target age (e.g., transitions in puzzle layouts, big draggable blocks, colorful, and animated sets and characters).

The game is divided into six problem-solving levels with increasing degrees of difficulty. Each level is set in a different maze puzzle represented by an anthill where several chambers are connected so the main character can explore them. There are four game items: blue ant (the main character), red ant, leaf, and a pile of leaves. At each level, the student is challenged to solve a computational problem using visual block-based instructions that can be dragged to assemble a piece of sequenced code, in order to guide the blue ant through several chambers of the anthill, picking up one or more leaves, eventually avoiding the red ant along the way and, finally, reaching the pile of leaves. Each increasing difficulty level is focused on one computational concept simple sequences, complex sequences, simple loops, nested loops, simple while, and complex while).

The game interface is divided vertically into two. The puzzle (anthill) appears on the left-hand side of the screen; programming blocks with which to build the piece of code and solve the puzzle are available on the right-hand side. BAC can switch between two play modes: individual and collaborative. In individual mode, the four game items are randomly placed in the anthill chambers, and the student must build the code to solve the challenge. The collaborative play mode is based on Dillenbourg's set of four conditions for setting up an active collaborative context in which learning mechanisms are likely to be triggered [34]: to set up initial conditions, to over-specify the "collaboration" contract with a scenario based on roles, to scaffold productive interactions by encompassing interaction rules in the medium, and to monitor and regulate the interactions. With respect to role assignments in the collaborative mode, the first student sets a challenge, placing the game items; and the second one takes on the role of solving the challenge (see Fig. 3). The hypothesis is that two students, regardless of their role, learn from each other, that this is not a

Fig. 3. Level 3 collaborative mode screenshot: The second student has built some code in order to solve the challenge posed by the first student.

competitive but a collaborative game since they both interact and help each other.

To succeed and complete the game level, students must test various solutions and find the most effective challenge solving strategy. Skillful players can adopt strategies they have already used to solve new problems, working on 3-D framework computational practices, such as problem decomposition, testing, abstraction, and iteration (see Fig. 1). As an engaging game element, a scoring system has been established so the student who sets the challenge earns more points the harder the challenge is, while the student who solves the challenge earns more points the more optimal the solution is.

BAC has a student registration system whereby players are identified in a database so that their progress over time can be tracked. A unique identification number is given to each student. Several data may be registered (birthdate, school, educational stage and grade, gender, previous coding experience, experience with devices such as digital tablets and applications, special educational needs, and additional comments). As students are playing, the monitoring system records each game step, collecting real-time data in a remote database, so that computational concepts of the 3-D framework, computational practices, and, to some extent, computational insights can subsequently be assessed (see Fig. 2). The data collected includes information about the time it has taken to set and to solve problems, the degree of optimization of the path posed in the solution, or the number of games won by a player.

Finally, to assess computational perspectives (see Fig. 2), qualitative data can be collected via direct observation while students interact with both the environment and their mates, particularly in the collaborative play mode. To collect additional data on motivation, the students are asked via the user interface about their mood before and after the playing session.

3) Questionnaires: Finally, through several questionnaires, data related to students' motivation, characteristics, skills, and interests were collected (see Table III). These questionnaires were targeted at both students and teachers and were used at different times during the experience. In Section IV-C, results that came from questionnaires are descriptively reported just for single and independent items (not for sets of items that supposedly compose a scale). In this sense, there is no need to analyze the reliability of the questionnaires as a whole.

TABLE III
SUMMARY OF THE CONTENT OF THE QUESTIONNAIRES

Quest. number	Target	Time point	Sub-samples	Descriptive title
1	Students	0	A, B, C	Student data (e.g., age, sex) and previous experience with games, digital devices and applications, and programming
2	Students	0	A, B, C	School previous performance, preferences, interests for the different subjects and motivational aspects before playing period
3	Students	2	A, B	Motivational aspects and preferences after the playing period
4	Teachers	0	A, B, C	Students' characteristics, special needs, school performance and others
5	Teachers	0	A, B, C	School characteristics, methodologies, and programming related subjects
6	Teachers	0	A, B, C	Teachers concerns about programming and game-based methodologies (before playing)
7	Teachers	2	A, B	Teachers concerns and opinions about BAC after the playing period

B. Participants and Procedure

The participants in this article were a sample of 176 primary school students from three Spanish public schools. As is shown in Table IV, the research focused on one educational stage in each school. Depending on the reasons for sampling the different subjects these can be divided, as shown in Table V, as the sampling procedure is intentional.

To ensure that the research is carried out under the same conditions in each school, an action protocol was developed. As shown in Table VI, at time 0 questionnaires (see Table III) were filled out by students and teachers. Consequently, at time 0, the BCTt, was administered as a pretest concurrently to each student following the action protocol. To ensure that students' skills or previous experience in the use of computer devices do not interfere with the test results, BCTts were printed on paper.

Next, A and B subsamples subjects played with BAC for five weeks using digital tablets. Each week, every subject in each subsample played for ten minutes, regardless of the playing mode. Subsample A played in the individual mode; subsample B played in the collaborative mode, where two randomly paired students play at the same time on the same digital tablet, one taking on the role of setting the challenge while the other attempts to solve the problem posed. The roles in the collaborative mode were swapped between students so that every student assumes both roles equitably. During the playtime (time 1), quantitative data were collected from the gameplay of each student, without altering the flow of the play, and saved in a remote database. Qualitative data were also collected during the playtime via direct observation.

At the end of the five-week playing period (time 2), the BCTt was readministered to subsamples A and B subjects, as a posttest, as well as to subsample C as the control group. Additionally, questionnaires were filled out by students and teachers of both subsamples A and B.

TABLE IV
PRIMARY SCHOOL EDUCATIONAL STAGE IN EACH SCHOOL

School	Educational stage	Grades	Student ages
Colegio Público Carlos Ruiz	1st	1st and 2nd	5–8
Colegio Los Escolapios	2nd	3rd and 4th	7–10
CEIP León Felipe	3rd	5th and 6th	9–12

TABLE V
NUMBER OF STUDENTS (*N*) IN EACH SUBSAMPLE DIVISION

Description	Id	Educational Stage		
		1st	2nd	3rd
Individual mode	A	A1: $n = 23$	A2: $n = 28$	A3: $n = 25$
Collaborative mode	B	B1: $n = 23$	B2: $n = 26$	B3: $n = 23$
Control group	C		C2: $n = 28$	

Finally, ten weeks after the end of the game period, at time 3, the BCTt was administered to A and B subsamples subjects, to check the aspects related to the learning retention.

At the end of week 16, the research team carefully analyzed all the quantitative and qualitative data collected.

IV. RESULTS AND DISCUSSION

A. *Beginners Computational Thinking Test*

The BCTt was administered in times 0, 2, and 3 (see Table VI). As expected, the reliability as the internal consistency associated with BCTt scores is high (Cronbach's alpha, subsamples A and B: time 0, $\alpha = .803$; time 2, $\alpha = .801$; time 3, $\alpha = .837$).

Next, BCTt data collected were analyzed to determine if the expected learning had taken place. Considering the BCTt score as the sum of all correct answers throughout the 25 items of the test, an initial analysis of BCTt pretest score results was performed. Since the sample population appears to be normally distributed, the subjects were intentionally divided into subsamples (see Table V). Next, the Student's *t*-test was applied to score results, separately at each educational stage, for the A and B subsamples (subjects who would play with BAC at a later stage: A: individual play mode; B: collaborative play mode), to assess the statistical significance of the difference between the two population means. Results show no significant difference in the test scores ($p > .05$) between groups from the same grade, so subsamples A and B are balanced and can be subsequently compared at a later stage (see Table VII).

After the five-week playing period, a primary analysis of the results was conducted: The Student's *t*-test was applied between BCTt pretest and posttest score results, obtaining overall better mean score results in the A and B subsamples posttest (see Table VIII). Since $p < .05$, there is a statistical significance in the difference between the pretest and posttest scores in nearly every A and B subsample, compared to results obtained between pre- and posttest on C subsample (control group) in which there was no significant score gain ($p > .05$). These results lead us to conclude that effective learning occurred due to the playing period, considering global scores from each student both in individual and collaborative modes; however, in the fifth grade collaborative mode (B3

TABLE VI
SUBSAMPLES CONSIDERED AND INSTRUMENT USED AT DIFFERENT TIME POINTS IN RESEARCH (BY WEEKS)

Instrument	Subsamples				
Questionnaires	A, B, C	…	A, B	…	…
BCTt	A, B, C	…	A, B, C	…	A, B
BAC	…	A, B	…	…	…
Week number	0	1 to 5	6	7 to 15	16
	Time 0 Pretest	Time 1	Time 2 Posttest		Time 3 Retention test

TABLE VII
STUDENTS *T*-TEST RESULTS OF BCTT PRETEST SCORE IN A AND B SUBSAMPLES, BY GRADES

Ed. stage	Grade	Sub-sample	*n*	Mean	Std. Deviation	*t*	*p*
1st	1st	A1	23	17.00	3.03	0.24	0.81
		B1	23	16.78	3.03		
2nd	4th	A2	28	21.79	2.99	0.53	0.60
		B2	26	21.35	3.15		
3rd	5th	A3	25	21.40	2.47	-1.60	0.12
		B3	23	22.43	1.95		

subsample), no significant gain was observed ($p = .878 > .05$). This was expected since the pretest results in this subsample were already high, leaving little room for improvement and, besides, students overall played less time in the collaborative mode; however, an alternative hypothesis is that older students take less advantage of collaborative mode than younger ones. Furthermore, the effect size was large (Cohen's $d > 0.8$) in lower grades and medium in higher grades (Cohen's $d > 0.5$), perhaps due to BCTt lower reliability in older students, except B3 subsample, which is a very favorable result, since the total playing time was only 50 min per student (ten minutes each week).

Score results in every grade were higher than expected, as a ceiling effect was observed in higher grades (see Table IX), so a second analysis was carried out, focusing exclusively on low percentiles subjects in every experimental subsample (calculated from pretest results). Results show very significant gains between pretest and posttest scores when considering the first quartiles: Q1 (see Table X), and Q2 (see Table XI), in all subsamples and play modes, considering $\alpha = .01$ (Students *t* test between pretest and posttest individual mode scores Q1: grade 1, $p = .0065$; grade 5, $p = .0015$). The only exception was in the fifth grade collaborative mode, in which there was no significant gain considering $\alpha = .01$ ($p = .0248$), but it was when considering $\alpha = .05$. Similar results were obtained in quartile Q2, which shows significant results between the pretest and posttest scores in all subsamples, considering $\alpha = .01$ (Students *t* test between pretest and posttest individual mode scores: grade 1, $p = .0042$; grade 5, $p = .0007$), except, again, in fifth grade collaborative mode ($p = .447$).

Furthermore, a third comparison was made, this time splitting BCTt items on computational concepts and counting how many students correctly answered the items in each set (i.e., if the subsample is $n = 23$, 21 points score in a set means that 21 of the 23 students answered it correctly).

TABLE VIII
COMPARISON BETWEEN PRETEST AND POSTTEST SCORE RESULTS BETWEEN DIFFERENT PLAY MODE SUBSAMPLES (STUDENT'S *T*-TEST)

Grade	BAC play mode	Subsample	*n*	BCTT mode	Mean	Std. Deviation	*t*	*p*	Effect size Cohen's *d*
1	Individual	A1	23	pretest posttest	17.00 19.74	3.03 3.97	-2.63	.012	0.903
	Collaborative	B1	23	pretest posttest	16.78 18.96	3.03 3.40	-2.29	.027	0.804
4	Individual	A2	28	pretest posttest	21.79 23.89	2.99 1.40	-3.38	.001	0.522
	Collaborative	B2	26	pretest posttest	21.35 23.92	3.15 1.35	-3.83	.000	0.615
	Control group	C2	28	pretest posttest	22.29 22.54	2.27 2.73	-0.372	.711	0.292
5	Individual	A3	25	pretest posttest	21.40 22.96	2.47 1.67	-2.62	.012	0.619
	Collaborative	B3	23	pretest posttest	22.43 22.52	1.95 1.88	-0.15	.878	0.037

TABLE IX
PERCENTILES BY GRADE

Educational stage	grade	Subsample	Minimum	Maximum	Mean	Std. Deviation	Percentiles 25	50	75
1st	1	A1	12.00	24.00	17.00	3.03	14.00	17.00	20.00
		B1	11.00	23.00	16.78	3.03	15.00	16.00	19.00
2nd	4	A2	14.00	25.00	21.79	2.99	19.00	23.00	24.00
		B2	15.00	25.00	21.35	3.15	18.00	22.50	24.00
3rd	5	A3	16.00	25.00	21.40	2.47	19.50	22.00	23.50
		B3	19.00	25.00	22.43	1.95	21.00	23.00	24.00

TABLE X
COMPARISON BETWEEN PRETEST AND POSTTEST SCORE RESULTS BETWEEN DIFFERENT PLAY MODE SUBSAMPLES (STUDENT'S *T*-TEST). Q1, PERCENTILE 25

Grade	BAC play mode	Subsample	Percentile	*n*	BCTT mode	Mean	Std. Deviation	*t*	*p*	Effect size *d* Cohen
1	Individual	A'1	< 25	6	pretest posttest	13.50 17.50	0.84 2.74	-3.42	.007	2.690
	Collaborative	B'1		8	pretest posttest	13.88 17.13	1.36 3.18	-2.66	.019	2.925
4	Individual	A'2	< 25	9	pretest posttest	17.89 23.89	1.62 1.45	-8.28	.000	3.045
	Collaborative	B'2		7	pretest posttest	16.86 23.43	1.21 1.90	-7.70	.000	5.242
5	Individual	A'3	< 25	6	pretest posttest	17.83 21.83	1.17 1.94	-4.32	.002	4.345
	Collaborative	B'3		6	pretest posttest	19.67 22.33	0.82 2.34	-2.64	.025	1.859

This new score, which is related to each computational concept, shows interesting results, since items have increasing difficulty and higher gains on the score are expected on the most difficult sets. In fact, this trend was confirmed empirically. There was a Student's *t*-test significant gain on the last four sets considering $\alpha = .01$ in every experimental subsample, so effective learning takes place particularly in relation to more difficult BCTt items. Fig. 4 shows BCTt score gains (score difference between pretest and posttest—for example, for level six, A1 subsample score mean was 11.90 in the pretest and 18.20 in the posttest, so the score gain was 6.30) for each computational concept set (A subsamples): there was a positive correlation between item difficulty and the score gain observed (coefficient of determination: A1: R2 = .90; A2: R2 = .88; A3: R2 = .69). Fig. 4 also shows that there are greater score increases for every computational concept in lower school grades (A1) and that this decreases proportionally in higher grades (A2, A3).

It was observed that considerably shorter times were needed to perform the BCTt posttest than to perform the pretest; indeed, the times were halved in all the experimental subsamples. This result is consistent with the improvement of CT skills, particularly automatization [21].

Finally, the results obtained from the third administration of the BCTt (see Table VI) were analyzed, to check aspects related to the retention of what was presumably learned (see

TABLE XI
COMPARISON BETWEEN PRETEST AND POSTTEST SCORE RESULTS BETWEEN DIFFERENT PLAY MODE SUBSAMPLES (STUDENT'S *T*-TEST). Q2, PERCENTILE 50

Grade	BAC play mode	Subsample	Percentile	*n*	BCTT mode	Mean	Std. Deviation	*t*	*p*	Effect size *d* Cohen
1	Individual	A"1	< 50	13	pretest	14.92	1.61	-3.16	.004	1.666
					posttest	18.54	3.80			
	Collaborative	B"1		13	pretest	14.69	1.49	-2.90	.008	2.053
					posttest	17.77	3.52			
4	Individual	A"2	< 50	17	pretest	20.12	2.71	-4.45	.000	0.812
					posttest	23.53	1.62			
	Collaborative	B"2		13	pretest	18.77	2.39	-6.66	.000	2.304
					posttest	24.00	1.53			
5	Individual	A"3	< 50	15	pretest	19.87	1.96	-3.82	.001	1.292
					posttest	22.47	1.77			
	Collaborative	B"3		17	pretest	21.65	1.62	-0.77	.447	0.197
					posttest	22.12	1.93			

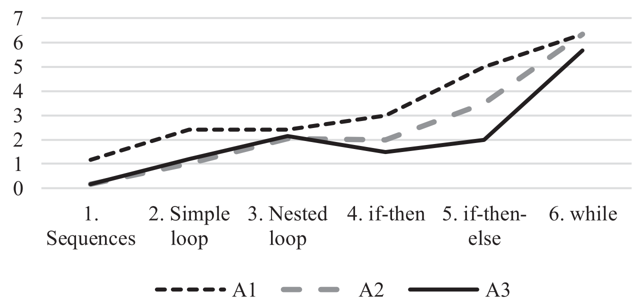

Fig. 4. BCTt score gain (ordinate axis: posttest score minus pretest score) by computational concept item set. Comparison between A subsamples (individual play mode).

Table XII). Pretest and retention test scores were compared to see if, after a ten weeks' inactivity time lapse, the learning previously acquired was maintained and still significant. Likewise, posttest and retention test scores were also compared, to see if there had been any significant learning loss. Comparisons between the A and B subsamples pretest and retention test shows significant score gains ($\alpha = .05$) in all subsamples despite the time-lapse, except in the B3 subsample, as was the case between pretest and posttest scores (Student's *t*-test between pretest and retention test scores: A1, $p = .042$; A2, $p = .011$; A3, $p = .043$; B1, $p = .040$; B2, $p = .000$; B3, $p = .889$). The effect sizes (Cohen's *d*) between pretest and retention test in all grades (except B3 subsample) where in the Hattie's (2009) zone of desired effects ($d > 0.4$); moreover, the effect sizes in lower grades collaborative subsamples (B1: $d = 0.694$; B2: $d = 0.700$) were larger than those in individual subsamples (A1: $d = 0.548$; A2: $d = 0.555$). These results reflect the retention of what had been learnt in all groups, especially in collaborative subsamples, even though there was cumulative fatigue since the retention test was performed at the end of the school year.

Additionally, comparisons between posttest and retention test in A and B subsamples (see Table XII) show no significant learning loss (considering $\alpha = .05$) in any of the subsamples (Student's *t* test between posttest and retention test scores: A1, $p = .376$; A2, $p = .278$; A3, $p = .715$; B1, $p = .667$; B2, $p = .817$; B3, $p = 1$). It must be stressed that there was less score loss between posttest and retention-test in all B subsamples than there was in A subsamples, as well as lower effect sizes, particularly in lower grades and low percentiles. This leads us to conclude that the collaborative play mode fosters more effective, enduring learning as was expected [55], although the initial pretest and posttest results might appear to be accompanied by lower score increases when compared to the individual mode.

When analyzing the low percentile subsamples results, more favorable outcomes were observed in all cases, considering $\alpha = .01$, significant score gains between pretest and retention test despite the time-lapse, which was even more significant in the collaborative mode subsamples (e.g., the difference between the B3 Q2 percentile subsample pretest and retention test was $p = .0093$, despite it being the worst case before); and there was no significant loss (considering $\alpha = .05$) between the posttest and the retention test. These results lead us to conclude that low percentile population and younger students benefit even more greatly from the learning tool, and especially from the collaborative playing mode.

B. Blue Ant Code

A preliminary analysis of data collected from BAC was conducted while students were playing. First, regardless of play mode or game level, the average time needed to solve the problem (in collaborative mode, the time the student has taken to solve the problem) throughout the playing period was analyzed for each subsample. As Fig. 5 shows, there is a negative correlation in all subsamples between the number of games played and the average time spent solving the problem (coefficient of determination: A1, B1: R2 = .935; A2, B2: R2 = .897; A3, B3: R2 = .937).

Likewise, the average time spent solving each level among school grades was compared throughout the playing period. In Fig. 6, A1 and B1 subsamples are compared to A3 and B3 subsamples by average time spent, in seconds, to solve the problem at each level at two time-points: time 1 (earliest time-point, the first day of the playing period), and end of time 1 (last time-point, after a five-weeks playing period) (see Table VI). These results lead us to conclude that effective learning took place progressively throughout the game period. It must be particularly pointed out that in the A1 and B1 subsamples (1st school grade), at time 1, students took longer to solve the problem, at all levels, than students in subsamples A3 and B3; but, at the end of time 1, when the playing period

TABLE XII
COMPARISON BETWEEN PRETEST/RETENTION-TEST AND POSTTEST/RETENTION-TEST SCORE RESULTS BETWEEN DIFFERENT PLAY MODE SUBSAMPLES

grade	BAC play mode	Subsample	*n*	BCTT mode	Mean	Std. deviation	*t*	*p*	Effect size *d* Cohen
1	Individual	A1	23	pretest retention test	17.00 18.83	3.03 2.87	-2.098	.042	0.548
				posttest retention test	19.74 18.83	3.97 2.87	0.894	.376	-0.258
	Collaborative	B1	23	pretest retention test	16.78 18.57	3.03 2.68	-2.115	.040	0.694
				posttest retention test	18.96 18.57	3.40 2.68	0.433	.667	-0.168
4	Individual	A2	28	pretest retention test	21.79 23.46	2.99 1.53	-2.648	.011	0.555
				posttest retention test	23.89 23.46	1.40 1.53	1.096	.278	-0.310
	Collaborative	B2	26	pretest retention test	21.35 23.85	3.15 1.01	-3.856	.000	0.700
				posttest retention test	23.92 23.85	1.35 1.01	0.232	.817	-0.070
5	Individual	A3	25	pretest retention test	21.40 22.76	2.47 2.15	-2.080	.043	0.435
				posttest retention test	22.96 22.76	1.67 2.15	0.368	.715	-0.101
	Collaborative	B3	23	pretest retention test	22.43 22.52	1.95 2.23	-0.141	.889	0.044
				posttest retention test	22.52 22.52	1.88 2.23	0.000	1	0.000

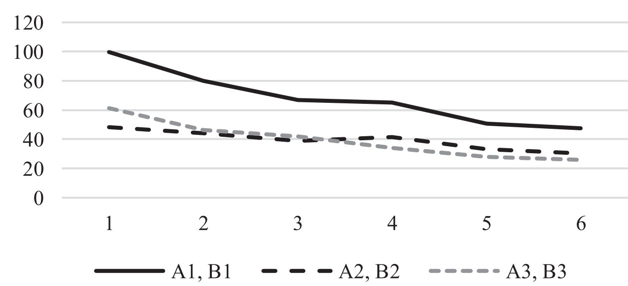

Fig. 5. Average time spent in seconds (ordinate axis) to solve the problem for each game played: 1 the first game played (abscissa axis), by subsamples.

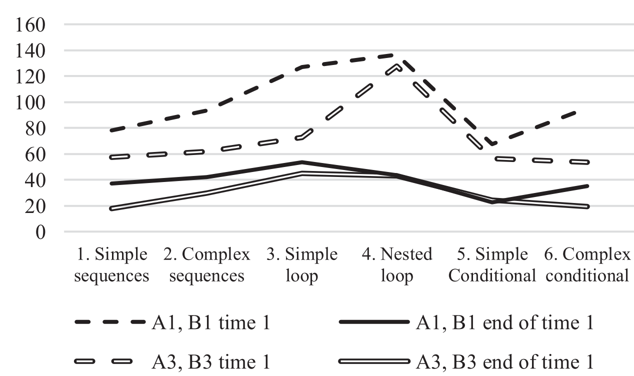

Fig. 6. Average time spent in seconds (ordinate axis) to solve the problem by computational concept (abscissa axis), in two time-points (time 1: first day of the playing period; end of time 1: after 5-weeks playing period).

ended, the subjects of subsamples A1 and B1 were able to solve the problems in nearly the same time as students in subsamples A3 and B3. Indeed, at some levels they even needed less time. This result is consistent with BCTt results, where much larger improvements were observed in younger students. It could also be added that the computational skills of first grade students are similar or even better (regarding certain concepts) than those of older students.

C. Questionnaires and Direct Observation

In order to assess 3-D framework computational perspectives (see Fig. 2), questionnaires were administered, and qualitative data were collected via direct observation while the students were playing. Computational perspectives entail students developing understandings of themselves and their relationships with others and the technological world. Thus, peer review can enhance the CT computational practices and perspectives, as Hsu *et al.* [1] suggested and is also confirmed by this article.

Data from the questionnaires show that the participating schools are moderately involved with computers and technology. Although there are digital blackboards in each classroom as well as a computer rooms, learning to program is not considered as a subject for the primary school setting. Moreover, primary school teachers report very low levels of computer proficiency (16.7%), low (16.7%), or medium levels (66.7%). Nevertheless, 100% believe that learning to program for coping with the demands of the future is either of very high (33.3%) or high importance (66.7%). Regardless of previous experience with games (76.6% of the students have played before), programming applications (83.7% of the students did not know any, and 83% have never programmed), or digital devices (95.5% of the students use a digital device on a regular basis); 70.2% of the students consider it important to learn programming and 52.2% think videogames could help them to learn this skill.

Strong motivation and interest to play with BAC were observed from the beginning, and lasted throughout the whole experience, regardless of the play mode. This is consistent with the questionnaires results, where 100% of the participants in every sample group enjoyed the experience and wanted to continue learning with game-based environments. Teachers also reported a positive and enjoyable for the students and thought that students did indeed learn with BAC (83.3%).

Regarding outputs coming from direct observation, BAC level 4 (nested loop) was the one in which, at time 1, students reported most problems in finding solutions, consistent with quantitative data indicating that it was the level for which students needed more time (see Fig. 6). While some lower grade students tried to avoid level 4 and others perceived it as a challenge and tried to overcome it; the higher grades students preferred to play challenging levels. Nevertheless, at the end of time 1, the times needed for level 4 were significantly reduced by one-third. It was found that environment game elements such as clear objectives, complexity, challenges, competition, or scaffolding were balanced and improved engagement, since students chose a higher level if they overcame the previous one, and a lower level if they could not solve the problem. In higher grades, more interest was shown in the scoring system as a motivating game element with students trying to beat their peers' or their own score throughout the game time.

Preference for the collaborative game mode was observed in all samples, as 100% of the students reported a preference for this play mode. The understanding and dynamics between peers were very positive, despite their lack of familiarity with game-based learning strategies. The hypotheses that students, regardless of their role, learn from each other and that it is not a competitive but a collaborative game since they both interact and help each other was confirmed. Most of the students preferred the problem-solving role. In higher grades, students setting the problems tried harder to make the challenges more difficult for their classmates.

All students, regardless of the play mode, reflect while playing. In individual mode, they think aloud, and in the collaborative mode they also discuss possible challenges and solutions with their partner. In the end, students reflect on the finished game; for example, the one solving the challenge typically advises the classmate on how he or she could made it more complex. Moreover, most students report reflecting on future games—for example, thinking about how to set a more difficult challenge or solve a specific puzzle next time. This need to review and think about their peers' programming process was found to encourage the review of their own learning performance, thereby engaging the students into thinking-doing, as Søndergaard and Mulder [78] concluded in their study. As Lye and Koh [32] suggested, peer review can enhance the computational practices and perspectives dimensions of CT.

In all experimental samples, students with special needs (8.3% of the participants), low academic results (15.1% of the participants), or low percentile according to pretest BCTt results, showed a great interest in the game and motivation to overcome the challenges. Additionally, these students demonstrated very significant improvements in the quantitative BCTt pre-post test results (see Section IV-A). These results are consistent with those obtained by Snodgrass *et al.* [79], who conclude that students with special needs can train their CT skills using visual programming environments. For example, a student with dyscalculia and dyslexia (B2 subsample) who had not achieved good results in the BCTt pretest was able to overcome the challenges in BAC in considerably shorter time than the B2 average. Indeed, she significantly improved her performance in the BCTt posttest and also showed a noticeable improvement in her self-esteem, being surprised at her ability to solve the challenges better than her peers with higher school performance, in line with research by Laal and Ghodsi [55].

V. Conclusion

Learning and assessing CT in schools is still in an experimental stage, especially at early ages. Most recent research focuses on the middle/high school stages and on computational concepts 3-D framework dimension; however, there is not enough research on early ages and the two other dimensions: 1) computational practices; and 2) computational perspectives [32]. In this article, three of the most used learning strategies have been combined in a collaborative game-based environment, built upon a constructivist problem-based strategy that has been tested on primary school. Results show that this approach is suitable for CT learning, especially so for early ages, under the 3-D framework, covering its three key dimensions (see Fig. 1).

The combination of different assessment methodologies, as a system of assessments [72]: BCTt, data-driven analytics, direct observation, and questionnaires, allows for a 3-D assessment (see Fig. 2). Specifically, BCTt assesses computational concepts and, partially, computational practices. Data driven analytics seem suitable for assessing computational concepts and practices and, partially, computational perspectives. Questionnaires and direct observation, especially in the collaborative play mode, cover the computational perspectives assessment. Furthermore, the CT assessment through internal analytics in game-based collaborative learning environments has been proved to be aligned with flow theory, since learning through gameplay can continue fluidly while assessments are unobtrusively handled maintaining the flow throughout.

The pre and post BCTt score gain was significant in the A1, B1, A2, B2, and A3 subsamples (see Table VIII) while the time spent performing the test was halved, suggesting that students had improved their CT skills after the playing period, in contrast with the control sample, which showed no significant gain or time reduction. These results were consistent with BAC data-driven analytics, which showed a very significant improvement over time. The similar results obtained from BCTt and BAC from each student and sample, reveal data-driven analytics as an adequate assessment tool which, in addition, can offer real-time data that may be useful for detailed personal assessment, strong low-level data analysis and the identification of unachievable computational concepts for each specific age in future research. Positive results were obtained from the retention test, suggesting that this game-based methodology fosters enduring learning. Nevertheless, BCTt and BAC seem to be better adapted to younger students since bigger improvements in CT skills were made in the lower grade samples and no significant gain was observed in the B3 subsample (which was to be expected, since in higher grades some subjects reached the maximum score in the pre-test leaving little room for improvement). As the research is aimed at early ages, this unfavorable outcome was expected, and we surmise BCTt and BAC may be suitable for even lower grades.

Furthermore, an unexpected and relevant outcome was found, as much larger improvements both in BCTt and BAC were observed in younger students as compared to older ones. Even though a significant gain was observed in most difficult concepts in every subsample, there is more overall score gain, in every computational concept, in lower school grades (A1) and proportional decreases in higher grades (A2, A3) (see Fig. 4). Furthermore, the first grade students' CT skills were similar or even better regarding certain concepts such as nested loops) than those of older students after the playing period. This leads us to suggest teaching CT skills from earlier ages and incorporate certain concepts, such as nested loops or conditionals, into the curriculum as early as first grade or even earlier, rather than waiting until the last stage of primary school (or even secondary), as it is currently the case in most schools in Spain. However, further research is needed due to possible bias in the selection of schools for each educational stage.

This article has shown that special-needs or low-percentile students in every sample benefit even more from the learning tool, and especially the collaborative playing mode, since very significant gains were achieved in these subsamples (considering $\alpha = .01$). Additionally, these students showed great interest in BAC and motivation to overcome the challenges. This leads us to conclude that game-based strategy and visual programming environments are suitable and engaging for these students, which is consistent with research by Snodgrass *et al.* [79].

It was confirmed via direct observation that all students reflect while playing in both playing modes, especially in the collaborative one. Reflection and peer collaborative playing encourage learning performance and engage students with thinking-doing; thus, peer review can enhance the CT computational practices and perspectives dimensions as Lye and Koh suggested [32]. The learning retention results were better in the collaborative mode than in the individual mode, particularly in lower grades, which leads us to the conclusion that the collaborative play mode fosters more effective and enduring learning (although from initial pretest and posttest results it would appear to result in lower score increases when compared to the individual mode). These results are in line with research done by Laal and Ghodsi [52], where they conclude that collaborative learning results not only in higher achievement and greater productivity, more caring, supportive, and committed relationships, but also in greater psychological health, social competence, and self-esteem. Results are also consistent with Johnson and Johnson [54], and Laal and Laal [55] that suggest that cooperative teams retain information longer than learners working as individuals.

Hainey *et al.* [53] systematic literature reviewed of research on game-based learning, concludes that further studies comparing single and collaborative strategies were needed to identify their pedagogical benefits in primary school. Results obtained in this article show that collaborative strategy benefits younger students, fosters reflection through peer communication, enhances computational perspectives (as it was suggested in previous research), and more effective and enduring learning is achieved. However, a collaborative strategy may not be as effective for last primary school stages, as these students seem to be capable of more efficient individual reflection. Further research is needed to prove this hypothesis.

This article shows significant learning outcomes when using game-based educational collaborative computer systems, particularly with low-percentile students, students with special needs, and in early educational stages. Moreover, combining game-based and collaborative learning strategies covers all three CT dimensions. Computational perspectives can be assessed with these systems, alongside with direct observation. Thus, when designing CT educational game-based computer systems targeted at first primary school stages, it is recommended to include explicit collaborative strategies, always taking into account flow theory and game design elements for younger students. When the environment is targeted at the final primary school stages, it is recommended to complement them with activities focused on individual reflection.

Finally, some limitations have been detected in this article. It has been concluded that first and second grade students benefit more from the learning strategy than older students do, but additional research on third grade students may be necessary to determine precisely at what age students are more likely to take advantage of these types of strategies. Further research must focus on the preschool educational stage since the lower age limit has not been established. Additionally, it would be advisable to replicate the study including control groups in every grade. Since this study reveals high benefit for students with special needs, further research on this aspect of the topic may also be desirable. The use of an assessment system is intended to cover the three computational dimensions, but further analysis is needed to identify exactly, which computational practices and perspectives had been addressed. In addition, it would be enlightening to replicate the study in other countries and populations.

Acknowledgment

The authors would like thank the student participants, teachers, and schools involved in the learning experiences reported in this article.

References

[1] T.-C. Hsu, S.-C. Chang, and Y.-T. Hung, "How to learn and how to teach computational thinking: Suggestions based on a review of the literature," *Comput. Educ.,* vol. 126, pp. 296–310, 2018, doi: 10.1016/j.compedu.2018.07.004.

[2] J. M. Wing, "Computational thinking test," *CACM Viewpoint*, vol. 49, no. 3, pp. 33–35, 2006. [Online]. Available: http://www.cs.cmu.edu/~wing/

[3] M. Román-González, J. Pérez-González, and C. Jiménez-Fernández, "Which cognitive abilities underlie computational thinking? Criterion validity of the Computational thinking test," *Comput. Human Behav.,* vol. 72, pp. 678–691, 2017. doi: 10.1016/j.chb.2016.08.047.

[4] A. V. Aho, "Computation and computational thinking," *Comput. J.,* vol. 55, no. 7, pp. 832–835, 2012.

[5] V. J. Shute, C. Sun, and J. Asbell-Clarke, "Demystifying computational thinking," *Educ. Res. Rev.,* vol. 22, pp. 142–158, 2017, doi: 10.1016/j.edurev.2017.09.003.

[6] K. Brennan and M. Resnick, "New frameworks for studying and assessing the development of computational thinking," in *Proc. Amer. Educ. Res. Assoc. Annu. Meeting*, Vancouver, Canada, Apr. 2012. [Online]. Available: https://bit.ly/3gtKdHE

[7] E. Hunsaker, "*Computational Thinking*," in *The K–12 Educational Technology Handbook*, A. Ottenbreit-Leftwich and R. Kimmons, Eds. EdTech Books, 2020, pp. 80–95. [Online]. Available: https://bit.ly/2UP0gZi

[8] S. Bocconi, A. Chioccariello, G. Dettori, A. Ferrari, and K. Engelhardt, "Developing computational thinking in compulsory education," *Eur. Commission, JRC Sci. Policy Rep.*, EUR-28295-EN, 2016, doi: 10.2791/792158.

[9] S. Grover and R. Pea, "Computational thinking in K–12: A review of the state of the field," *Educ. Res.,* vol. 42, no. 1, pp. 38–43, Jan. 2013, doi: 10.3102/0013189X12463051.

[10] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*. Cambridge, MA, USA: Harvard Univ. Press, 1978.

[11] J. Piaget, *Psychology and Epistemology: Towards a Theory of Knowledge.* London, U.K.: Allen Lane, 1972.

[12] D. H. Jonassen and L. Rohrer-Murphy, "Activity theory as a framework for designing constructivist learning environments," *Educ. Technol. Res. Develop.,* vol. 47, no. 1, pp. 61–79, 1999, doi: 10.1007/BF02299477.

[13] S. Y. Lye and J. H. L. Koh, "Review on teaching and learning of computational thinking through programming: What is next for K–12?," *Comput. Human Behav.,* vol. 41, pp. 51–61, 2014, doi: 10.1016/j.chb.2014.09.012.

[14] M. Zapata-Cáceres, E. Martín-Barroso, and M. Román-González, "Computational thinking test for beginners: Design and content validation," in *Proc. IEEE Glob. Eng. Educ. Conf.*, Porto, Portugal, 2020, pp. 1905–1914.

[15] M. Binkley, O. Erstad, J. Herman, M. Ripley, M. Miller-Ricci, and M. Rumble, "Defining twenty-first century skills," in *Assessment Teaching of 21st Century Skills*. Dordrecht, The Netherlands: Springer, 2012, pp. 17–66.

[16] B. Trilling and C. Fadel, *21st Century Skills.* San Francisco, CA, USA: Jossey-Bass, 2009.

[17] M. Qian and K. R. Clark, "Game-based learning and 21st century skills: A review of recent research," *Comput. Human Behav.,* vol. 63, pp. 50–58, 2016, doi: 10.1016/j.chb.2016.05.023.

[18] J. M. Wing, "Computational thinking and thinking about computing," *Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.,* vol. 366, no. 1881, pp. 3717–3725, 2008, doi: 10.1098/rsta.2008.0118.

[19] S. Grover, "The 5th 'C' of 21st century skills? Try computational thinking (not coding)." Accessed: Oct. 1, 2021. [Online]. Available: http://bit.ly/346qK78

[20] K. Tang, T. Chou, and C. Tsai, "A content analysis of computational thinking research: An international publication trends and research typology," *Asia-Pacific Educ. Res.,* vol. 29, pp. 9–19, 2020, doi: 10.1007/s40299-019-00442-8.

[21] M. Zapata-Ros and P. Pérez-Paredes, "El pensamiento computacional," in *Análisis De Una Competencia Clave*. New York, NY, USA: Create Space Independent, 2018.

[22] National Research Council, *Report of a Workshop on the Pedagogical Aspects of Computational Thinking*. WA, DC, USA: National Academies, 2011.

[23] J. A. Velázquez-Iturbide, "Towards an analysis of computational thinking," in *Proc. Int. Symp. Comput. Educ.*, Sep. 2018, pp. 1–6.

[24] S. Kong, "A framework of curriculum design for computational thinking development in K–12 education," *J. Comput. Educ.,* vol. 3, no. 4, pp. 377–394, 2016.

[25] G. Falloon, "An analysis of young students' thinking when completing basic coding tasks using scratch jnr. on the iPad," *J. Comput. Assist. Learn.,* vol. 32, no. 6, pp. 576–593, 2016.

[26] D. Statter and M. Armoni, "Teaching abstract thinking in introduction to computer science for 7th graders," in *Proc. 11th Workshop Primary Secondary Comput. Educ.*, 2016, pp. 80–83.

[27] B. Zhong, Q. Wang, J. Chen, and Y. Li, "An exploration of three-dimensional integrated assessment for computational thinking," *J. Educ. Comput. Res.,* vol. 53, no. 4, pp. 562–590, 2016, doi: 10.1177/0735633115608444.

[28] L. Zhang and J. Nouri, "A systematic review of learning computational thinking through scratch in K–9," *Comput. Educ.*, vol. 141, 2019, Art. no. 103607.

[29] J. Fagerlund, P. Häkkinen, M. Vesisenaho, and J. Viiri, "Computational thinking in programming with scratch in primary schools: A systematic review," *Comput. Appl. Eng. Educ.,* vol. 29, no. 1, pp. 12–28, 2021.

[30] J. Nouri, L. Zhang, L. Mannila, and E. Norén, "Development of computational thinking, digital competence and 21st century skills when learning programming in K–9," *Educ. Inquiry,* vol. 11, no. 1, pp. 1–17, 2020.

[31] P. J. Rich, S. F. Browning, M. Perkins, T. Shoop, E. Yoshikawa, and O. M. Belikov, "Coding in K–8: International trends in teaching elementary/primary computing," *Tech. Trends,* vol. 63, pp. 1–19, 2018.

[32] S. Y. Lye and J. H. L. Koh, "Review on teaching and learning of computational thinking through programming: What is next for K–12?" *Comput. Human Behav.,* vol. 41, pp. 51–61, Dec. 2014, doi: 10.1016/j.chb.2014.09.012.

[33] D. Boud and G. E. Feletti, *The Challenge of Problem-Based Learning*. London, U.K.: Routledge, 1997, doi: 10.4324/9781315042039.

[34] P. Dillenbourg, *Collaborative Learning: Cognitive and Computational Approaches*. Oxford, U.K.: Pergamon, 1999.

[35] J. Roschelle and S. D. Teasley, "The construction of shared knowledge in collaborative problem solving," in *Computer Supported Collaborative Learning*, C. O'Malley, ed. Berlin, Germany: Springer-Verlag, 1995, pp. 69–97, doi: 10.1007/978-3-642-85098-1_5.

[36] P. C. Blumenfeld, E. Soloway, R. W. Marx, J. S. Krajcik, M. Guzdial, and A. Palincsar, "Motivating project-based learning: Sustaining the doing, supporting the learning," *Educ. Psychol.,* vol. 26, no. 3/4, pp. 369–398, 1991, doi: 10.1080/00461520.1991.9653139.

[37] J. Kirriemuir and A. Mcfarlane, *Literature Review in Games and Learning*. Bristol: Futurelab, 2004.

[38] M. Prensky, "Fun, play and games: What makes games engaging?," in *Digit. Game-Based Learn.*, McGraw-Hill, New York, NY, USA.

[39] J. Moreno-León, G. Robles, and M. Román-González, "Towards data-driven learning paths to develop computational thinking with scratch," *IEEE Trans. Emerg. Topics Comput.,* vol. 8, no. 1, pp. 193–205, Jan.–Mar. 2020.

[40] V. Barr and C. Stephenson, "Bringing computational thinking to K–12," *ACM Inroads*, vol. 2, pp. 48–54, Feb. 25, 2011.

[41] S. Papert, *Mindstorms: Children, Computers, and Powerful Ideas.* New York, NY, USA: Basic Books, 1980.

[42] J. R. Savery and T. M. Duffy, "Problem based learning: An instructional model and its constructivist framework," *Educ. Technol.,* vol. 35, no. 5, pp. 31–38, 1995.

[43] J. R. Savery, "Overview of problem-based learning: Definitions and distinctions," *Interdiscipl. J. Problem-Based Learn.*, vol. 1, no. 1, pp. 9–20, 2015, doi: 10.7771/1541- 5015.1002.

[44] W. Wu, H. Hsiao, P. Wu, C. Lin, and S. Huang, "Investigating the learning-theory foundations of game-based learning: A meta-analysis," *J. Comput. Assist. Learn.,* vol. 28, no. 3, pp. 265–279, 2012, doi: 10.1111/j.1365-2729.2011.00437.x.

[45] E. Boyle, T. M. Connolly, and T. Hainey, "The role of psychology in understanding the impact of computer games," *Entertainment Comput.,* vol. 2, no. 2, pp. 69–74, 2011, doi: 10.1016/j.entcom.2010.12.002.

[46] M. Li and C. Tsai, "Game-based learning in science education: A review of relevant research," *J. Sci. Educ. Technol.,* vol. 22, no. 6, pp. 877–898, 2013, doi: 10.1007/s10956-013-9436-x.

[47] W. Wu, W. Chiou, H. Kao, C. A. Hu, and S. Huang, "Re-exploring game-assisted learning research: The perspective of learning theoretical bases," *Comput. Educ.,* vol. 59, no. 4, pp. 1153–1161, 2012, doi: 10.1016/j.compedu.2012.05.003.

[48] M. Csikszentmihalyi, *Flow and the Psychology of Discovery and Invention*, vol. 39. New York, NY, USA: Harper Perennial, 1997.

[49] M. Csikszentmihalyi, *Finding Flow: The Psychology of Engagement With Everyday Life.* New York, NY, USA: Basic Books, 1997.

[50] F. Buitrago Flórez, R. Casallas, M. Hernández, A. Reyes, S. Restrepo, and G. Danies, "Changing a generation's way of thinking: Teaching computational thinking through programming," *Rev. Educ. Res.,* vol. 87, no. 4, pp. 834–860, 2017, doi: 10.3102/0034654317710096.

[51] R. T. Johnson and D. W. Johnson, "Active learning: Cooperation in the classroom," *Annu. Rep. Educ. Psychol. Jpn.,* vol. 47, pp. 29–30, 2008.

[52] M. Laal and S. M. Ghodsi, "Benefits of collaborative learning," *Procedia - Social Behav. Sci.,* vol. 31, pp. 486–490, 2012, doi: 10.1016/j.sbspro.2011.12.091.

[53] T. Hainey, T. M. Connolly, E. A. Boyle, A. Wilson, and A. Razak, "A systematic literature review of games-based learning empirical evidence in primary education," *Comput. Educ.,* vol. 102, pp. 202–223, 2016.

[54] D. W. Johnson and R. T. Johnson, "Effects of cooperative and individualistic learning experiences on interethnic interaction," *J. Educ. Psychol.,* vol. 73, no. 3, pp. 444–449, 1981, doi: 10.1037/0022-0663.73.3.444.

[55] M. Laal and M. Laal, "Collaborative learning: What is it?" *Procedia - Social Behav. Sci,* vol. 31, pp. 491–495, 2012, doi: 10.1016/j.sbspro.2011.12.092.

[56] D. H. Jonassen, *Learning to Solve Problems With Technology: A Constructivist Perspective.* Upper Saddle River, NJ, USA: Merrill, 2003.

[57] B. Haßler, L. Major, and S. Hennessy, "Tablet use in schools: A critical review of the evidence for learning outcomes," *J. Comput. Assist. Learn.,* vol. 32, no. 2, pp. 139–156, 2016.

[58] Y. Rogers, Y. Lim, W. R. Hazlewood, and P. Marshall, "Equal opportunities: Do shareable interfaces promote more group participation than single user displays?," *Human–Comput. Interact.,* vol. 24, no. 1/2, pp. 79–116, 2009.

[59] D. Roldán-Alvarez, E. Martín, P. A. Haya, M. Garcia-Herránz, and M. Rodríguez-González, "DEDOS: An authoring toolkit to create educational multimedia activities for multiple devices," *IEEE Trans. Learn. Technol.,* vol. 11, no. 4, pp. 493–505, Oct.–Dec. 2018, doi: 10.1109/TLT.2017.2788867.

[60] C. Lin, L. Wong, and Y. Shao, "Comparison of 1: 1 and 1: M CSCL environment for collaborative concept mapping," *J. Comput. Assist. Learn.,* vol. 28, no. 2, pp. 99–113, 2012.

[61] X. Tang, Y. Yin, Q. Lin, R. Hadad, and X. Zhai, "Assessing computational thinking: A systematic review of empirical studies," *Comput. Educ.*, vol. 148, Apr. 2020, Art. no. 103798, doi: 10.1016/j.compedu.2019.103798.

[62] E. Relkin, L. de Ruiter, and M. U. Bers, "TechCheck: Development and validation of an unplugged assessment of computational thinking in early childhood education," *J. Sci. Educ. Technol.,* vol. 29, no. 4, pp. 482–498, 2020, doi: 10.1007/s10956-020-09831-x.

[63] G. Chen, J. Shen, L. Barth-Cohen, S. Jiang, X. Huang, and M. Eltoukhy, "Assessing elementary students' computational thinking in everyday reasoning and robotics programming," *Comput. Educ.,* vol. 109, pp. 162–175, 2017, doi: 10.1016/j.compedu.2017.03.001.

[64] A. Basawapatna, K. H. Koh, A. Repenning, D. Webb, and K. Marshall, "Recognizing computational thinking patterns," in *Proc. 42nd ACM Tech. Symp. Comput. Sci. Educ.,* 2011, pp. 245–250.

[65] J. D. Werner, S. Campe, and D. Kawamoto, "The fairy performance assessment," in *Proc. 43rd ACM Tech. Symp. Comput. Sci. Educ.,* 2012, pp. 215–220.

[66] V. Garneli and K. Chorianopoulos, "Programming video games and simulations in science education: Exploring computational thinking through code analysis," *Interactive Learn. Environ.,* vol. 26, no. 3, pp. 386–401, Apr. 2018, doi: 10.1080/10494820.2017.1337036.

[67] A. Mühling, A. Ruf, and P. Hubwieser, "Design and first results of a psychometric test for measuring basic programming abilities," in *Proc. Workshop Primary Secondary Comput. Educ.*, 2015, pp. 2–10.

[68] D. Weintrop and U. Wilensky, "Using commutative assessments to compare conceptual understanding in blocks-based and text-based programs," in *Proc. Int. Comput. Educ. Res. Workshop*, Aug. 2015, pp. 101–110.

[69] M. Román-González, J. Pérez-González, J. Moreno-León, and G. Robles, "Can computational talent be detected? Predictive validity of the computational thinking test," *Int. J. Child-Comput. Interaction*, vol. 18, pp. 47–58, 2018.

[70] M. Román-González, J. Pérez-González, J. Moreno-León, and G. Robles, "Extending the nomological network of computational thinking with non-cognitive factors," *Comput. Human Behav.,* vol. 80, pp. 441–459, 2018, doi: 10.1016/j.chb.2017.09.030.

[71] AERA, APA, and NCME, *Standards For Educational and Psychological Testing.* Washington, DC, USA: Amer. Educational Res. Assoc., 2014.

[72] S. Grover and R. Pea, "Systems of assessments for deeper learning of computational thinking in K-12," in *Proc. Annu. Meeting Amer. Educ. Res. Assoc.*, 2015, pp. 15–20.

[73] M. Román-González, J. Moreno-León, and G. Robles, "Combining assessment tools for a comprehensive evaluation of computational thinking interventions," in *Computational Thinking Education*, S. Kong and H. Abelson, eds. New York, NY, USA: Springer, 2019, doi: 10.1007/978-981-13-6528-7_6.

[74] J. Moreno and G. Robles, "Automatic detection of bad programming habits in scratch: A preliminary study," in *Proc. 44th ASEE/IEEE Front. Educ. Conf.*, 2014, pp. 1–4.

[75] N. Diana, M. Eagle, J. Stamper, S. Grover, M. Bienkowski, and S. Basu, "An instructor dashboard for real-time analytics in interactive programming assignments," in *Proc. 7th Int. Learn. Analytics Knowl. Conf.*, 2017, pp. 272–279.

[76] L. Gal, A. Hershkovitz, A. Morán, M. Guenaga, and P. Garaizar, "Suggesting a log-based creativity measurement for online programming learning environment," in *Proc. 4th ACM Conf. Learn. Scale*, 2017, pp. 273–277.

[77] D. Olivares and C. Hundhausen, "Supporting learning analytics in computing education," in *Proc. 7th Int. Learn. Analytics Knowl. Conf.*, 2017, pp. 584–585.

[78] H. Søndergaard and R. A. Mulder, "Collaborative learning through formative peer review: Pedagogy, programs and potential," *Comput. Sci. Educ.,* vol. 22, no. 4, pp. 343–367, 2012.

[79] M. R. Snodgrass, M. Israel, and G. C. Reese, "Instructional supports for students with disabilities in K–5 computing: Findings from a cross-case analysis," *Comput. Educ.,* vol. 100, pp. 1–17, 2016.

María Zapata-Cáceres received the B.Sc. and M.Eng. degrees in architecture from the Polytechnic University of Madrid, Madrid, Spain, in 2002, the M.Sc. degree in virtual environments from Centro Superior de Arquitectura, Madrid, in 2005, the M.Sc. degree in videogame design and production from the European University of Madrid, Madrid, in 2005, and the B.Sc. degree in computer science from the National Distance Education University, Madrid, in 2018. She is currently working toward the Ph.D. degree in computer science from Rey Juan Carlos University, Madrid.

She is also a Researcher and visiting Professor in the B.Sc. degree program in game design and development with Rey Juan Carlos University. She has more than 15 years of professional experience in 3-D design, videogames, and teaching. Her main research interest includes videogames as learning tools for computer science.

Estefanía Martín-Barroso received the B.Sc. and M.Eng. degrees in computer science from Rey Juan Carlos University (URJC), Madrid, Spain, in 2002 and 2004, respectively, and the Ph.D. degree in computer science and telecommunications from the Autonomous University of Madrid (UAM), Madrid, in 2008.

Her Doctoral thesis focused on mobile adaptive learning systems for collaborative contexts. From 2003 to 2008, she was an Assistant Professor at UAM and URJC. Since 2008, she has been an Associate Professor with the Computer Science Department, URJC. She leads the Blue Thinking Project and the DEDOS Project, as well as the development of ClipIt, a video-based social network platform created as part of the EU project JuxtaLearn. She is a Co-Principal Investigator of the CS-Track project, focused on Citizen Science. She has authored or coauthored more than 50 articles and received awards for various projects of which she has been a part. Her research interests include learning systems, human-computer interaction, and disabilities.

Marcos Román-González received the B.Sc. degree in educational psychology, the M.Sc. degree in educational research, and the Ph.D. degree in education from National Distance Education University (UNED), Madrid, Spain, in 2006, 2013, and 2016, respectively.

His Doctoral thesis focused on code literacy and computational thinking (CT) development in primary and secondary schools. From 2012 to 2019, he was an Assistant Professor with UNED, where he has been an Associate Professor since January 2020. He is the lead author of the CT Test, which is being used worldwide for CT assessment at the primary and secondary school levels. He has authored and coauthored in top journals, such as *Computers and Education*, *Computers in Human Behavior*, and IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING.