Coding as a Playground: Promoting Positive Learning Experiences in Childhood Classrooms

Article *in* Computers & Education · April 2019

DOI: 10.1016/j.compedu.2019.04.013

Marina U. Bers
Tufts University

Carina Soledad González González
Universidad de La Laguna

Draft versión of paper published in:

Bers, M. U., González-González, C., & Armas–Torres, M. B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education*, *138*, 130-145. https://doi.org/10.1016/j.compedu.2019.04.013

Coding as a Playground: Promoting Positive Learning Experiences in Childhood Classrooms

Marina U.Bers[a], Carina S. González-González[b], Mª Belén Armas–Torres[b]

[a]**Eliot-Pearson Department of Child Study and Human Development. Computer Science Department. Tufts University, USA**

[b]**Department of Computer Engineering and Systems, University of La Laguna, Spain**

Abstract

In recent years, there has been a push to introduce coding and computational thinking in early childhood education, and robotics is an excellent tool to achieve this. However, the integration of these fundamental skills into formal and official curriculums is still a challenge and educators needs pedagogical perspectives to properly integrate robotics, coding and computational thinking concepts into their classrooms. Thus, this study evaluates a "coding as a playground" experience in keeping with the Positive Technological Development (PTD) framework with the KIBO robotics kit, especially designed for young children. The research was conducted with preschool children aged 3 to 5 years old (*N*=172) from three Spanish early childhood centers with different socio-economic characteristics and teachers of 16 classes. Results confirm that it is possible to start teaching this new literacy very early (at 3 years old). Furthermore, the results show that the strategies used promoted communication, collaboration and creativity in the classroom settings. The teachers also exhibited autonomy and confidence to integrate coding and computational thinking into their formal curricular activities, connecting concepts with art, music and social studies. Through the evidence found in this study, this research contributes with examples of effective strategies to introduce robotics, coding and computational thinking into early childhood classrooms.

Keywords: cooperative/collaborative learning; teaching/learning strategies, improving classroom teaching, elementary education.

1. Introduction

Children around the globe are being raised in environments that are saturated with smart devices. At the same time, there is a growing need for a future workforce that understands technology. Given this new reality, national educational programs and private initiatives are focusing on STEM (Science, Technology, Engineering, and Mathematics) literacy and making coding and computational thinking a priority for education (Manches & Plowman, 2017). However, research has found that educational interventions in early childhood are related with lower costs and more lasting effects than interventions that begin later on (Cunha & Heckman, 2006). Also, some studies demonstrate gender-based stereotypes involving STEM careers (Metz, 2007; Steele, 1997) and fewer obstacles to entering the workforce (Madill et al., 2007; Markert, 1996) when children are exposed to STEM in childhood (Metz, 2007; Steele, 1997)

Different studies have shown the potential of robotics education in early years (Jung & Won, 2018). Some of them have presented methods to implement a robotic curriculum (Authors, 2010), to evaluate CT skills (Authors et al., 2014; Roman-González et al., 2017; Chen et al., 2017), to develop executive functions (Di Lieto et al., 2017), attitudes toward society and science (Kandlhofer & Steinbauer, 2016), and the technological characteristics of robots and interactions (Burlson et al., 2017; Vogt et al., 2017; Serholt, 2018). However, research on robotics and computational thinking in childhood education is still in its early stages (Öztürk & Calingasan, 2018; Ching, Hsu & Baldwin, 2018; Guanhua et al, 2017; García-Peñalvo & Mendes, 2017). Several studies have focused on technological aspects or robots, interaction aspects or robotics curricula, rather than on how learners engage and learn and how teachers introduce the new skills into their classrooms and curricula (Jung & Won, 2018; Serholt, 2018; Vogt et al, 2017). This study tries to help bridge this gap in the current research by exploring the following research questions:

- R.Q.1. *How do teachers integrate coding and computational thinking into their curricular activities?*
- R.Q.2. *What programming and computational thinking skills do preschool children 3-5 years old master after being introduced to robotics (KIBO)?*
- R.Q.3. *What positive behaviors are developed by children in a learning environment of coding as a playground?*

This paper is structured as follows: first we present a review of the literature on coding, computational thinking, and robotics in childhood education; the case study and main issues of the experience are then defined after the research method is described; and finally, the results and conclusions are summarized and analyzed.

2. Literature review: coding, computational thinking and robotics in childhood education

2.1. Coding and computational thinking as new literacy

Coding is defined as a new literacy for the 21st century. However, computational thinking (CT) is defined as the skill to solve problems algorithmically and to develop a sense of technological fluency (Wing, 2006). Computational thinking is the ability to use the concepts of computer science to formulate and solve problems. CT entails a wider range of abilities (e.g. problem analysis, algorithmic thinking, etc.) usually involving the core concepts of abstraction, algorithm, automation, decomposition, debugging and generalization. It can be understood as directly linked to and as a component of "digital competence". Computational thinking represents a type of analytical thinking that shares many similarities with mathematical thinking (e.g., problem solving), engineering thinking (designing and evaluating processes), and scientific thinking (systematic analysis). Moreover, computational thinking can be viewed as an expressive process that allows for new ways to communicate ideas. Coding can be seen as a tool to teach CT. Programming is writing connected with technology. Programming is writing the code (symbolic representation in a computing language).

In this sense, we approach the concept of "coding as a playground" as a new literacy, a new language for children where they can learn to code at a young age through fun, play and creativity (Authors, 2017).

An increasing number of nations and regions have plans for introducing technology and coding in early childhood (Siu & Lam, 2005; UK Department of Education, 2013). For example, the United Kingdom published a national curriculum in 2013 that incorporates computer science in the early years. In Finland, all elementary students have been required to learn coding since 2016. Estonia, Ireland and Italy are actively modifying their curricula to include computing (Pretz, 2014).

In Europe, the academic community has led the discussion on the introduction of computational thinking skills in the curricula through several reports, such as the Royal

Society of UK (Furber, 2012), the Academie des Sciences in France (l'Académie des sciences, 2012), the Sociedad Científica Informática in Spain (Meseguer et al, 2015) and ACM (Association of Computing Machinery) Europe (Gander et al., 2013). The European Commission has assumed an active role in this subject and promotes a digital agenda in which coding is the literacy of today (Kroes & Vassiliou, 2014; Moreno-León & Robles, 2015). Sixteen European countries have included coding in their curricula, but with different approaches and at different levels (Balanskat & Engelhardt, 2015; Bocconi et al., 2016; Spanish Ministry of Education, Culture and Sports, 2018).

In the United States, new initiatives focused on 21st century skills suggest programming and tech literacy skills as a priority for early childhood education (e.g. International Society for Technology in Education, 2007; National Association for the Education of Young Children and the Fred Rogers Center for Early Learning and Children's Media at Saint Vincent College, 2012). For example, non-profit organizations, such as Code.org and the Scratch Foundation, are having a major impact in supporting these endeavors (Portelance, Strawhacker, & Authors, 2018).

In Asia, Singapore's "PlayMaker Programme" brought technology into early childhood education centers as part of a Smart Nation initiative (Digital News Asia, 2015). As part of this nationwide program, Authors and Sullivan (2016) conducted a study to evaluate children's learning outcomes after completing a seven-week KIBO robotics curriculum, which proved highly successful at teaching coding and provided a fruitful, collaborative and creative setting.

2.2. Robotics in Early Childhood

The introduction of STEM programs into childhood education has been based on the tangible aspects of working with robotics, which reinforce the development of fine motor skills, and the need to introduce young children to coding early on before stereotypes are formed (Authors, Seddighin, & Sullivan, 2013). Robotics can engage children in a playful and developmentally appropriate learning experience that includes problem solving, abstract and logical thinking (Authors, 2018).

The majority of research on robotics, coding and computational thinking has been focused on later schooling. But teaching these concepts and skills in the early childhood years can be positive in promoting STEM when combined with social science in a natural and playful way. The current generation of robotic kits for young children allows learning through manipulatives. Resnick et al (1998) show how these tools promote a robust understanding of mathematical concepts like other traditional materials (blocks, beads, balls, etc.). Furthermore, robotics does not usually involve screen time and can promote teamwork and collaboration (Sullivan & Authors, 2016).

Prior research has shown that young children aged 4 to 7 years old can create and program basic robotics projects (Authors et al., 2002; Cejka, Rogers, & Portsmore, 2006; Perlman, 1976; Wyeth, 2008; Sullivan & Authors, 2013). Furthermore, robotics allows working with other important skills for their development, like fine motor skills and hand-eye coordination (Lee et al., 2013; Authors, Seddighin, & Sullivan, 2013). Moreover, coding and robotics let children develop problem-solving, meta-cognitive and reasoning skills.

However, when introducing robotics into an early childhood context, there is a need to make the pedagogical approach developmentally appropriate. The use of different metaphors can convey this. In this sense, Resnick (2006) compared programming to a paintbrush, describing it as a medium for self-expression and creative design. Authors (2012; 2018) liken robotics to “coding as a playground” due to the way it can engage children cognitively, socially, physically, emotionally, and creatively. For that reason, in the following section we describe a case study on the introduction of effective educational strategies for teaching coding and computational thinking in childhood classrooms.

3. Case study: coding as a playground experience

The study described in this paper evaluates a "coding playground" experience in which KIBO robotics (Authors, 2018) was used in Tenerife, Spain to teach children programming and computational thinking skills in the context of an educational program that uses robotics to support positive interpersonal behaviors. These behaviors are described by the Positive Technological Development (PTD) framework (Authors, 2012) as the six C's: communication, collaboration, community building, content creation, creativity, and choices of conduct. Some of the Cs underpin behaviors that enhance the intrapersonal domain (content creation, creativity, and choices of conduct); others address the interpersonal domain and consider social aspects (communication, cooperation, and community building). These behaviors involve developmental assets that have been described by decades of research on positive youth development. PTD provides a framework that aids in understanding how technology can be designed and utilized to promote positive behaviors and how those behaviors can, in turn, yield developmental assets. The theoretical model of Positive Technological Development framework involves three components: individual assets, technology-mediated behaviors or activities, and applied practice. The diagram below (Fig. 1) shows how the C's are connected and provides examples of how they can be implemented in a classroom setting.

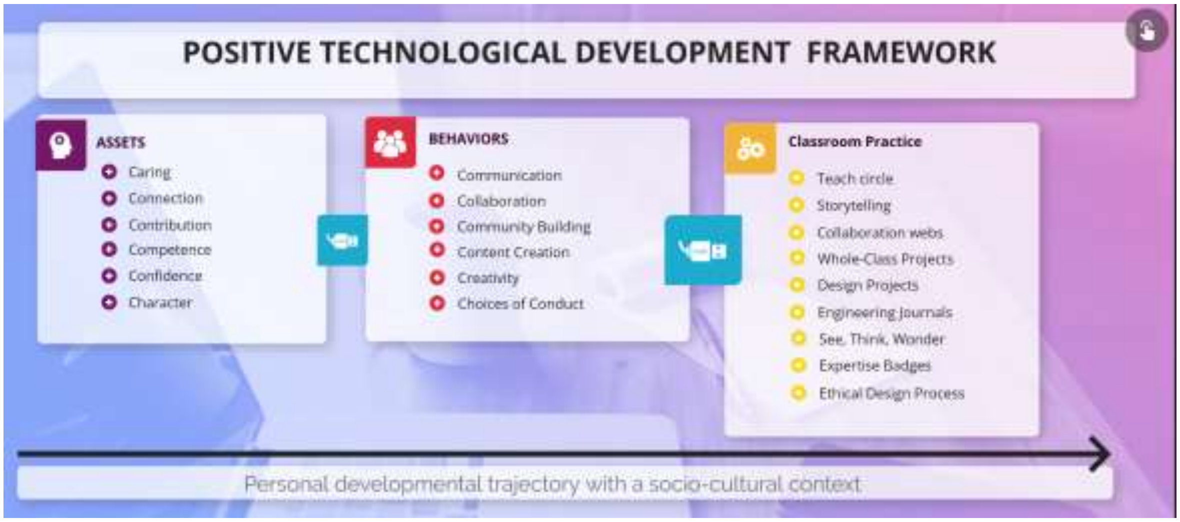

Fig. 1. Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom. Bers, M. (2018)

The PTD framework provides a method for supporting these positive behaviors through the use of new technologies (i.e. KIBO robotics) in different contexts. Many robotics activities also include "competition" (i.e. First Lego League is one of the most famous educational robotics competitions). PTD encourages collaboration instead of competition, promoting shared resources and caring about one another. Collaboration is included into the whole learning process.

PTD involves both the design of new educational technologies and technology-rich interventions, as well as their evaluation. Some activities could be sharing tools/materials, working on the same project, seeking the help of other students, making suggestions and giving feedback, etc.

In keeping with the six C's of the PTD framework, it is possible to design curriculums that integrate robotics, such as *Dances from Around the World*, which has been developed by the DevTech Research Group at Tufts University[1] to integrate music,

[1] *Dances from around the world curricular unit:* *https://sites.tufts.edu/devtech/files/2018/03/KIBOCurriculum_DancesAroundtheWorld.pdf and the version adapted to the Spanish context: https://goo.gl/6if4Y9*

dance, and culture with engineering and programming. In this study we designed a curriculum based on a PTD framework (an adaptation of Dances from Around the World) and then we evaluated the development of the six C's by using the PTD Engagement Checklist.

The robotics kit used in the curriculum has to satisfy the age-related needs of young children, as is the case with KIBO (see Fig. 2). This robotic kit is composed of hardware (the robot proper as well as wheels, motors, light output, and a selection of sensors) and software (tangible programming to program the robot's actions). Children use wooden coding blocks, with no hidden electronic parts, to program KIBO (see Fig. 1). KIBO has a scanner embedded in the robot's body that is used to scan the barcodes on the wooden blocks. Thus, devices that require "screen-time" are not part of the KIBO programming experience. This design choice was made in keeping with the American Academy of Pediatrics' guidelines (American Academy of Pediatrics, 2013). KIBO's programming language is comprised of 21 unique blocks that can be combined to form complex sequences including repeat loops, conditional statements, and nesting statements. Furthermore, to foster STEAM (Science, Technology, Engineering, Arts, and Math) integration, the KIBO kit has various art platforms that children can use to personalize their projects.

Fig. 2. KIBO robot with sensors, light output, and turntable platform attached

Fig. 3. Blocks for programming KIBO and a sample KIBO program (sequence of spin, shake, move backward, move forward, and turn on a red light).

The working memory of young children changes drastically between the time they are 3 and 5 years of age (Shonkoff, Duncan, Fisher, Magnuson, & Raver,2011), which allows them to effectively learn new content. When children enter preschool, at around the age of 3, most of them can complete tasks that involve carrying out two steps, such as throwing out a napkin and putting their lunchbox away after snack time (Rhode Island Department of Education [RIDE], 2013; Shonkoffetal.,2011). By the time children leave preschool and enter kindergarten, around the age of 5, they can follow multi-step instructions and retell stories that they know well in the correct order (RIDE, 2013). By using the KIBO robot, children can enhance their working memory skills and learn to sequence increasingly complex programs and master all of KIBO's syntax rules.

By using robotics manipulatives such as the motors, sensors, outputs, and wooden programming blocks that are used by KIBO, children are able to develop fine motor skills and hand-eye coordination. By playing in a way that requires them to manipulate physical objects with a symbolic meaning (i.e., KIBO's programming blocks, symbolizing robotic actions), children can start exploring more complex symbolic thinking (Bers, 2008). In addition to these technical manipulatives, children also work on their fine motor skills through the addition of arts, crafts, and recyclable materials. Specifically, the two art platforms provide a space for exploring the engineering design process to build sturdy creations that are personally meaningful (Sullivan et al.,2015).

Children can use KIBO to explore logical sequencing and organization by using the tangible programming blocks. They can explore making different decisions and their consequences, they can learn that computing systems need both hardware (robotic parts) and software (blocks) to operate or carry out the iterative process that is used to develop programs and tangible artifacts. These possibilities can be used to teach children the basics of computational thinking.

In the following section, the method, procedure and instruments used in the present study are described.

4. Method

When conducting this study, we applied a mixed method (Creswell, 2015), a methodology that is characterized by the process of quantitative and qualitative data, which are combined to allow for a better understanding of the research problem. Thus, the design was concurrent triangulation, in which the qualitative and quantitative data have been collected and analyzed and then, during the interpretation and discussion, the results are explained and compared. Concurrent triangulation involved the data collection throw different methods and instruments in order to achieve a more validated results (Coleman & Briggs, 2002).

The research questions relied on inductive reasoning (Twining et al, 2017). The instruments (questionnaires, PTD checklist, Solve-its, teacher journal, etc.) were validated by the DevTech research group in a similar study developed in Singapore in 2016 (Sullivan and

Authors, 2016). These instruments provide the criteria for designing and evaluating digital educational experiences with young children. The quantitative instruments applied were questionnaires (pre-workshop questionnaire, post-workshop questionnaire, post-experimentation questionnaire) and the PTD checklist. The qualitative instruments used were observations, interviews, diary journal and a focus group. The qualitative data were categorized and codified for analysis.

The characteristics/variables studied and their relationships with the instruments/methods, participants and the research questions are shown in Fig. 4.

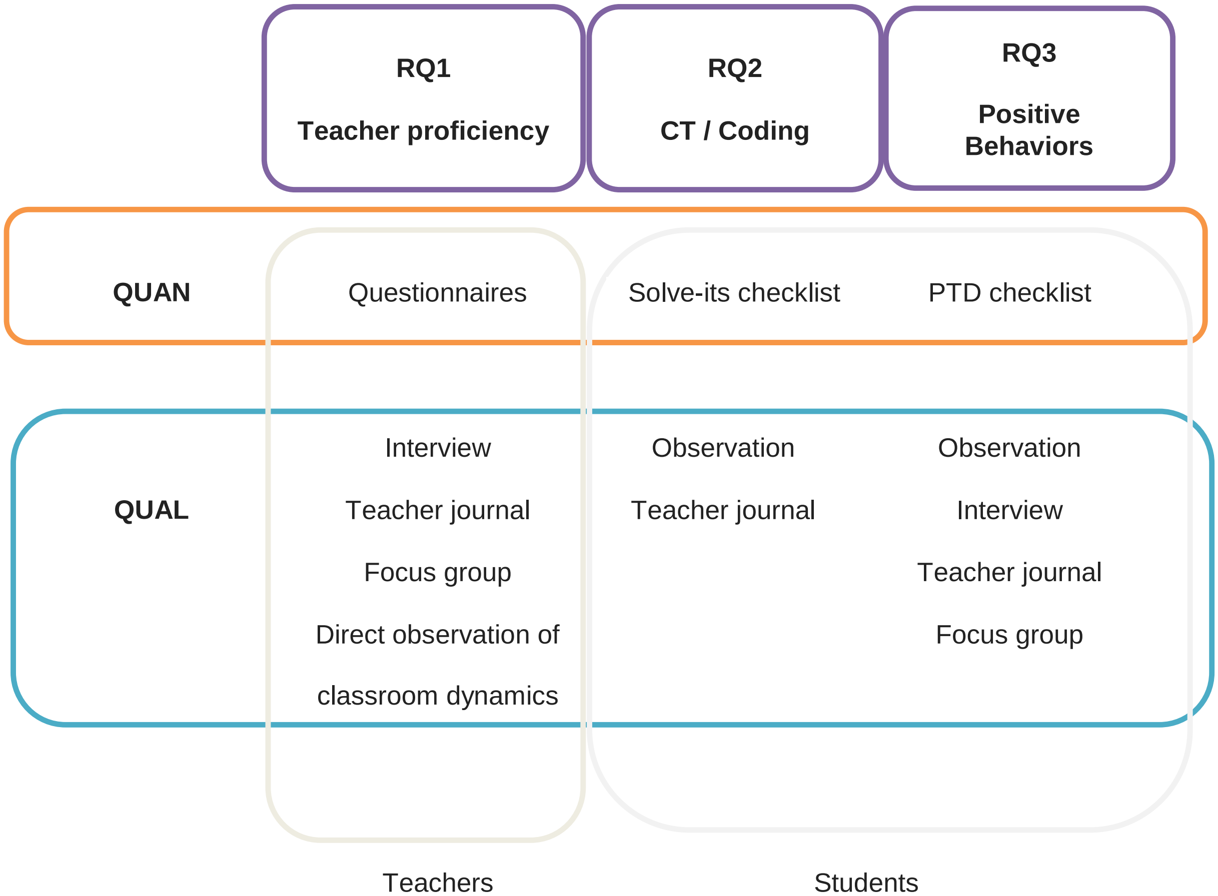

Fig. 4. Mixed qualitative and quantitative methods used in this study.

4.1. Participants

A total sample of *N*=172 young children (84 girls and 88 boys) from three childhood school centers in Tenerife, Canary Islands, Spain, participated in this research (See Table 2). The children ranged between three and five years of age at the start of this study, were divided into 16 classes of different age levels (1, 2 and 3). The centers represent different school settings in Spain: public, private, and semi-private. Sixteen teachers from each of the participating schools and their collaborators, such as school staff, participated in this study. An informed consent was provided to all research participants. In the case of children, the informed consent was signed by their parents.

Level / Age	Girls	Boys	Total
3 Childhood/5 years old	56	51	107
2 Childhood/4 years old	12	17	29
1 Childhood/3 years old	16	20	36
Total	**84**	**88**	**172**

Table 1. Sample distribution by level and gender.

The selection of the schools was by invitation of the research group and the Canary Islands Board of Education. The sample corresponds to 16 classes in three schools. The classes participating in this research were evaluated at different points in time. The objective was to study the entire group by measuring the effectiveness of the intervention through the learning of coding, computational thinking and the development of positive behaviors. Thus, there was no comparison group for this study.

As concerns the representativeness, the sample is determined by: a) the characteristics (or variables) evaluated, which are related with the problem that is being studied: b) the ability to measure these variables; and c) information on these characteristics or variables to be used as an evaluation variable (Yanow & Schwartz-Shea, 2015).

4.2.Procedure

Teachers from the three schools participated in a one-day face-to-face training session on the KIBO robotics kit and were also introduced to the *Dances from Around the World* curriculum as an example, and its adaptation to the *Canarian traditional dances* curriculum (Appendix I). The teachers then had a period to adapt the curriculum to their classrooms. This adaptation did not modify the contents related to robotics, programming and computational thinking. Some teachers had the option of combining the contents of two sessions into just one (i.e. what is robot and what is programming). Also, specific adaptations were made for the three-year-old children (i.e. repeat session is not recommended for them, or the use of conditionals). But the strategies used in every case were the same. Thus, the sessions all followed the same basic structure:

a. Preliminary games

b. Introduction of powerful ideas through a challenge

c. Individual or group work

d. Presentation and sharing of the final activity (technology circle)

e. Free exploration and play

Also, the learning goals that the children had to achieve were the following:

1. To learn and apply the engineering process to building things (and robots).
2. To learn different components of a robot and how it works.
3. To learn how the robot perceives its environment.
4. To learn how to instruct KIBO using the coded blocks.
5. To understand that KIBO sensors resemble human senses, and that they can program the robot using sound stimuli.
6. To understand the repeat instruction (only for children older than 4).
7. To understand the conditional instruction (only for children aged 5).
8. To learn the different traditions and dances of the Canary Islands and be able to program KIBO to dance them.

The teachers introduced the powerful ideas with KIBO using narrative, although the story could be different in each case. The teachers also adapted the narrative used to the children's level of development, presenting the concepts, behaviors and skills required of them in an orderly and continuous progression.

The fundamental STEAM concepts were introduced through "powerful ideas", such as the engineering design process, robotics, programming and sensors. In addition, the activities cover other aspects of the curriculum such as language, mathematics and arts. For example, when programming, children practice with sequence, order, counting, number sense and estimation. In addition to the connections between the physical environment, mathematics and different languages (verbal, audiovisual and artistic) of this didactic unit, there is also a connection between the culture and traditions of the Canary Islands.

Local researchers supported the teachers by using virtual platforms and tools. The local researchers were members of the research group and were previously trained in the research method, robotics and the curriculum to be taught. The virtual classroom included a space for coordination with forums and a video-conference tool, a calendar with the schedule of activities, a space for posting curriculum materials (i.e. the Canaries childhood curriculum, a KIBO activities guide, an Engineering Design Process poster, gamification tutorials, among others), and a space for posting research-related materials (instruments for pre- and post-workshop training, informed consent for research participants, teacher journal, PTD, solve-its, and interviews). Other tools such as Adobe Connect for videoconferences and online mobile messaging system tools (Telegram/WhatsApp) were used to support teachers while they were working in their classrooms with KIBO. Text messages from WhatsApp and Telegram were useful for answering the teachers' questions over the course of the study. Moreover, Adobe Connect and Moodle were used to train the blended teaching staff and to monitor and support of them during the course of the study. In addition, the local researchers visited each of the schools at the start and end of each week of the study to collect data.

Teachers generally taught coding and computational thinking using KIBO integrated into their curriculum during several sessions per week, over a period of two to three weeks, with each session lasting approximately 45 minutes. Teachers adapted the sessions to incorporate them into their usual class schedule. Some teachers combined two sessions on the same day. This was the case in the 3rd level of childhood education, with children older than 5 years of age.

The study lasted was from February to June of 2017. The first step was to select the centers, then contact management and the teachers, then contact the families to receive their authorization. The teacher training and the adaptation of the curriculum to their classrooms started in March. The intervention sessions with the students were then carried out from April to June of 2017. All the schools completed a minimum of three sessions per week. Some of the schools did extra activities with KIBO. Since one of the goals of the study was to observe if and how teachers adapted the curriculum to the requirements of their students, these extra activities were carefully documented.

4.3.Data collection and analysis

The first and last sessions of each class were observed (direct observation) and videotaped. Students' programming knowledge was assessed through structured observation of video recordings of their final projects in which they created a KIBO dance routine. Data regarding positive behaviors, such as collaboration, was also collected on students' engagement using the PTD checklist (Authors, 2012) (See Appendix II).

For the qualitative analysis of the results of the observational instruments, we studied the level of agreement between judges for each subjectively evaluated item in the sample. To do this, we built a table with the cases observed, a category system was set up and the joint assessments were made as previously agreed. This procedure was used to validate the level of reliability of the observers' agreement. The Kappa index was used to measure the level or inter-rater agreement for PTD and Solve-its checklists. In case of PTD checklist six categories (the 6's C) were used. The inter-rater agreement of two trained researchers was calculated. Regarding to the Solve-its instrument used in this study, the scoring rubric was developed after a pilot assessment was administered, to identify incorrect answer patterns that could demonstrate developmental level rather than programming comprehension. Inter-scorer reliability tests showed precise agreement (two items; $K = 0.902$, $p<0.001$) (Strawhacker and Author, 2015). For the qualitative analysis of the teachers' notes and the interviews, also, the codes were categorized into six categories and their frequencies were analyzed depending on the questions to be addressed.

4.3.1. Structured observation of the classroom dynamics

We observed and videotaped the first and final robotic sessions of each grade level within each of the three schools with two video cameras. The children were aware that there were cameras in the classroom; however, they carried out the activities in a natural way since the cameras were placed on tripods in different corners of the classroom from where they carried out the activities so as to capture their actions in a way that was non-invasive.

We used a direct observation method in order to study the classroom dynamics with KIBO. Some of the aspects observed included: a) c*urriculum sessions* (number and duration of each session), b) s*tudent groups* (size, organization and composition of the group), c) *tutoring* (rotation among groups, number of students per teacher/tutor), d) *materials* (types of crafts and recycled materials used, organization of robotic kits, availability, accessibility of materials in the classroom), e) *organization* (allocation of the robots in the classroom: one per group, stations, corners), and f) *didactic strategies* (how the project was introduced, the role of teachers and students).

4.3.2. Solve-Its

In order to measure the students' understanding of the programming concepts and computational thinking skills, we analyzed their final KIBO projects using indicators derived from the Solve-Its assessments, which provide a window into young children's knowledge of foundational programming concepts, from basic sequencing to complex conditional statements, using a 0-6 scale (Strawhacker, Sullivan, & Authors, 2013; Strawhacker & Authors, 2014). An adapted version of Solve its assessment has been designed and applied in this study (See Appendix III). The adaptation made in our study has been based in the observation of the checklist, but it does not modify the metric and the inter-scorer reliability test of the instrument.

4.3.3. PTD engagement checklist

As mentioned above in Section 3, we followed the PTD theoretical framework developed by Authors (2012) to assess the positive behaviors associated with the 6 C's (communication, collaboration, community building, content creation, creativity, and choice of conduct). Thus, the instrument used in this study was the "PTD Engagement Checklist"[2] for the assessment of positive behaviors (See Appendix II). The instrument is divided into six sections (each one representing a behavior described in the PTD framework) and measured using a 5-point Likert scale. The checklist is meant to evaluate a group of children or an individual child as they work in a space. Researchers had to identify the frequency observed during each robotics session using a 1-5 scale (1: never and 5: always). A total of 59 sessions was scored and analyzed. For each of the C's, a number was output consisting of the average scores per session, and a composite final

[2] https://sites.tufts.edu/devtech/ptd/

score at the end of the study.

4.3.4. Teacher journal and interview

In order to obtain more nuanced, qualitative data, after each session the educators completed an online journal (See Appendix IV) with six questions, where the teachers shared their thoughts on the effectiveness of the strategies used, problems they encountered, and other aspects of the session. Also, they reported on how they modified and tailored the given sample robotic curriculum to meet their children's needs, their own classroom environment, and the context of their schools.

In addition, the educators completed interviews (Appendix V) at the end of the experience, and participated in a discussion panel with other teachers and a focus group. These experiences were set up in a flexible way in an effort to ascertain the teachers' views of the experience.

5. **Results**

5.1.Curriculum implementation

The teachers adapted and introduced coding and computational thinking into their current curriculum by following the example in the curriculum presented during their training. In keeping with their plan, the children were first presented step-by-step activities to familiarize them with the different programming concepts and skills. Through different challenges, the children were driven to master KIBO, and later, to integrate with social sciences.

The teachers were encouraged to adapt the curriculum to their particular needs and context and to propose their own lesson plans. While one of the schools choose to strictly follow the scope and sequence of the given curricular unit, in the other two schools, each teacher adapted the unit to their own overarching curricula. For example, in the youngest class in one school, the teachers adapted the curriculum to integrate it with the learning of geometrical shapes (circle, square and triangle), numbers, graphomotor skills, and reading vowels. In other schools, the teachers integrated the use of KIBO with other digital tools (e.g. digital boards, tablets) and gamified strategies and narratives.

The students had to design, build, and program KIBO to dance to selected music in their final projects (see Fig. 5). This final activity represented the students' technical knowledge of the curriculum, and a functional robotics project. The activity finished with the presentation of the final project to the rest of the groups.

Fig. 5. Examples of decorations of KIBO, representing typical dancers from the Canary Islands

The minimum components required for every group's final project were at least two motors with wheels, light output and basic sequences of movements, though some groups used advanced programming concepts such as repeat loops with numbers and various other sensors. They were also able to integrate arts to exemplify the dance associated with the dress of their KIBO (see Fig. 6).

Fig. 6. Some children performed dances from the Canary Islands in their final projects.

5.2.Structured observation of sessions

The results of the organization and dynamics of the sessions are summarized by the aspects shown in Table 2.

Aspects observed	Findings
Curriculum sessions	Each class, regardless of the age of the students, met for three to five sessions. One school scheduled 45-minute sessions, while the other two planned sessions lasting 1 hour 15 minutes. This difference in time allocated to the project did not have an influence on the student's learning.
Student Groups	There was a variation in the number of children in each class within each school. While some classes had 15 students, others had 26. In every classroom, children were divided into mixed groups (boys and girls) of 3-5 children. Some teachers assigned children to rotate through the different activities involved in making their KIBOs (i.e. some programmed the robot while others crafted decorations).
Tutoring	Teachers and adult tutors rotated among the groups, supporting children and helping them solve problems. The student-teacher ratio ranged from 8-15 students per teacher.
Materials	Different arts and crafts materials were utilized, such as drawings, aluminum foil, cardboard, painted kitchen roll tubes, double-sided tape, recycled material (e.g. toilet paper tubes, lids), toothpicks, glitter, temperas, modeling clay, plugs, tissues, etc. To gamify the activity, some teachers created level badges using cards to assign different roles in the robotic team.
Organization and distribution of the robots in the classroom	Each group was given one KIBO robotics kit. Some classes organized groups to work at tables, and other classes alternated between the tables and the corner of the room. In other cases, the classroom was adapted to make space in the center so that children could rotate between the tables and activities on the floor. One school designated a specific corner of the classroom for KIBO.
Didactic strategies	The didactic strategies used by teachers were observed. For example, to introduce the children to KIBO concepts, teachers used storytelling as a strategy. Some teachers introduced the KIBO activities to build skills around a story about a robot that visits prehistory from the future. Another teacher used an epic mission: to save the Earth through a space mission that students will carry out with KIBO. Some teachers worked the diversity concept through storytelling.
Assessment	All the teachers strived to reach the daily goals and evaluated the students' performance with KIBO; however, not all of them utilized the assessment tools provided, instead using their own instrument based on observation.

Table 2. Organizational and dynamic aspects observed in the sessions.

5.3. Mastery of Coding and Computational Thinking

The main goal of this study is to teach children fundamental computational thinking and coding skills. Brennan and Resnick (2012) defined a Computational Thinking Framework that matches the developmental capacity of young children and includes: sequencing (ordering a sequence of steps to perform actions), repeats (performing the same sequence a number of times), conditionals (decisions related to events or actions), and debugging (finding and fixing errors in the code). To assess the mastering of the coding we used the aspects evaluated in the Solve-Its instrument (Authors, 2012). Solve-Its allow evaluating the programming's level of complexity from easy to hard. Note that Solve-Its was originally designed to be used with children 4 years old and up, and this study also contained children aged 3 years old.

Student programming sequences were labeled “easy” or “hard” depending on their complexity and the number of programming blocks used. For example, “hard” Solve-Its required the use of more programming commands and control loops through sensors, while “easy” ones targeted motion programming concepts and fewer blocks. Fig. 7 shows an example of an easy sequencing concept, and Fig. 8 a hard one.

Fig. 7. Examples of easy sequencing concepts

Fig. 8. Examples of hard sequencing concepts

Therefore, for analysis purposes, this paper presents results from an analysis of programming sequences created by the children in their final KIBO dance projects using the Solve-Its assessment checklist. The researchers scored the students’ mastery of programming concepts on a 0-6 scale, with a higher score representing a greater sequence complexity. On average, students scored highly on all the programming concepts worked in class, demonstrating they learned the fundamental computational thinking skills of sequence, repeats, conditionals and debugging during the study. More complex sensors involving the use of repeat and conditional blocks in many cases were excluded from the curriculum for the three-year-old students, and were instead replaced with “n” readings of blocks with actions or conditionals using the “wait for clap” block (see Fig. 9 and Fig. 10).

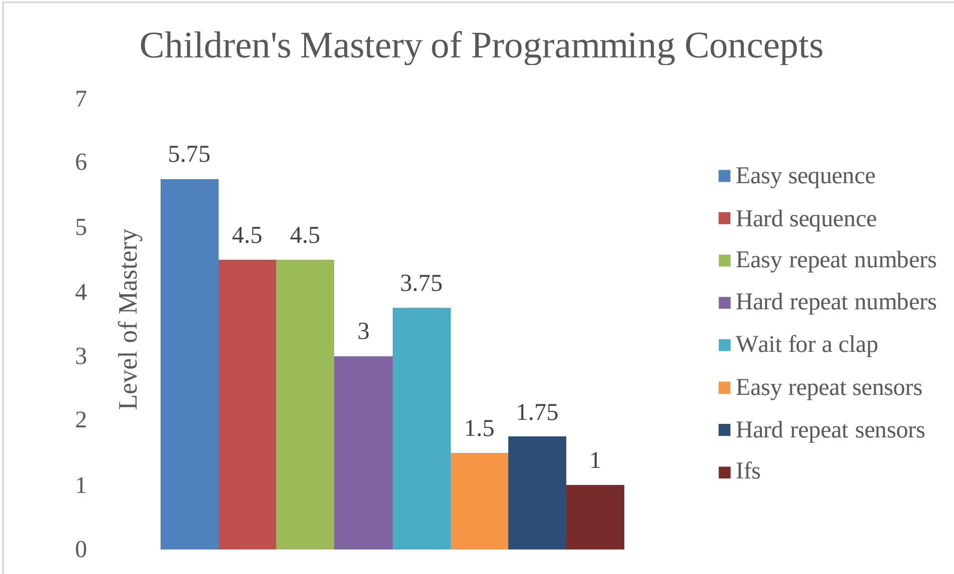

Fig. 9. Mean scores of programming sequences created by children in their final KIBO dance projects.

Fig. 10. Programming sequence created by children involving an easy sequence with a repeat number.

5.4.PTD Checklist

The researchers analyzed the data resulting from the completed PTD Checklists. The analysis provided information regarding the occurrence of each of the 6C behaviors:

communication, collaboration, community building, content creation, creativity, and choices of conduct (Authors, 2012). For instance, children traded ideas (communication), helped one another when using the materials (collaboration), shared their projects with family members (community building), programmed a KIBO dance and constructed a KIBO dancer (content creation), used materials in a divergent, unexpected manner (creativity), and showed respect to peers and teachers (choices of conduct). The 6C's were scored on a scale of 1-5, with higher scores indicating behaviors observed more regularly. This was calculated for each session, with 59 sessions scored in total.

At the end of the program with each class, an average score for each of the six C's was calculated. The results show that the program was most effective at promoting communication (M = 4.6) and collaboration (M = 4.1), with creativity and content creation also exhibiting a fairly high score (M = 3.1) (see Fig. 11).

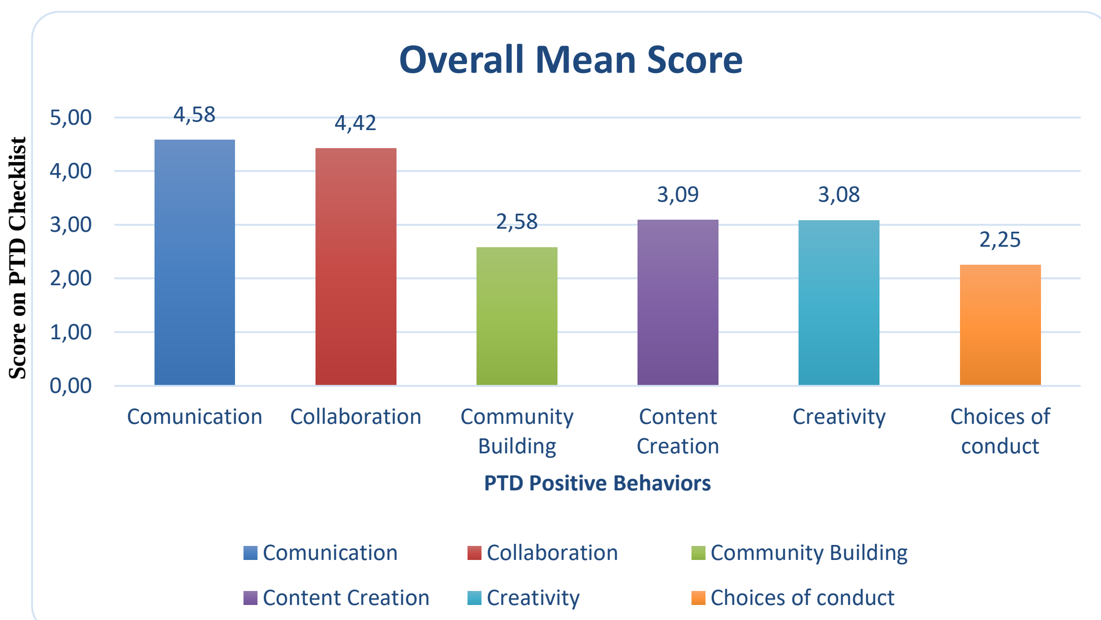

Fig. 11. Mean scores on PTD Checklist.

5.5.Teachers' experiences

An analysis of the teachers' reflection journals shows overall effectiveness in reaching their teaching goals. We analyzed 43 qualitative registers involving the robotic, coding and computational thinking teaching goals, with highly positive results in their achievement. The strategies used by educators to teach complex engineering and programming concepts and skills differed. Teachers made their own curricular adaptations based on the curriculum provided: omitting lessons/activities (i.e. the conditionals were removed, because they were complicated for some ages), additions to the curriculum (i.e. graphomotor skills with the strokes of the robot's movements, geometric shapes (circle, square and triangle), the number series 1-2-3; basic literacy skills), adapting games/activities (i.e. integrating the use of KIBO into their current "Prehistory" project), adjusting the time spent on concepts in the robotics curriculum (i.e. devoted more time to decorating their project) and cultural adaptations (i.e. programming sequences to dance Canarian folk dances). Figures 12, 13, 14 and 15 show some of the curricular adaptations created by the teachers. For example, Fig. 12 shows how KIBO can be linked to other parts of the curriculum, in that as computational thinking is being taught, so is the curriculum, while also motivating the students. In the case of vowels, KIBO is used as a motivating element through a game in which KIBO has to be programmed to travel different routes. Given the name of an object or animal, the children have to program KIBO to travel to the first letter of each name (e.g. ant for A).

An analysis of the activity journals kept by teachers shows that most of them introduced new concepts through songs, dances, games, or storytelling; engaged their students through group discussions (both small groups and the full class); and utilized metaphors from cars and other vehicles to teach the different mechanical aspects of the KIBO robot.

Fig. 12. Example of curricular adaptation to work on basic literacy skills (vowels).

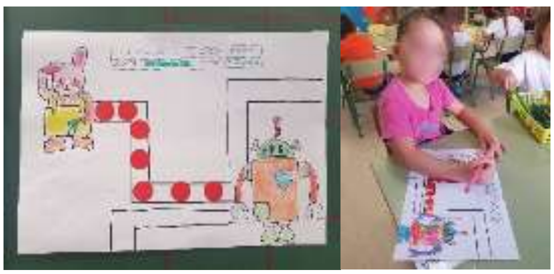

Fig. 13. Example of an adaptation to work on manipulative-graphomotor skills through the movements of the robot.

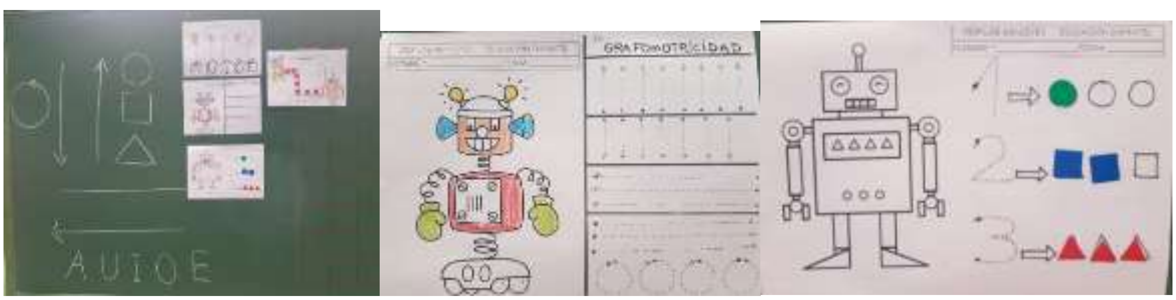

Fig. 14. Examples of several adaptations to curriculum: geometric shapes (circle, square and triangle), numbers, graphomotor skills, and reading vowels.

Fig. 15. Children showing their prehistoric projects.

Through interviews and reflection journals, the teachers shared their experiences with robotics, including some of the positive experiences and the challenging moments they encountered throughout the project. Some examples of the hermeneutic units of analysis identified using the same PTD behaviors (communication [COM], collaboration [COL], community building [CB], content creation [CC], creativity [CR], and choices of conduct [CHC]) and motivation [MOT]) are the following:

- E1. *"KIBO promoted teamwork, cooperative learning and role commitment [COL]. The promotion of values such as respect for a partner and their opinion [COM], the ability to wait, the development of responsibility and autonomy, as well as the care of materials* [CHB] *[….] The theme of the Canarian identity brought children closer to a knowledge of their traditions and culture."* [CB].
- E2. *"…The groups were discussing [COM] and reasoning together."* [COL]
- E3. *"After explaining the activity with several examples and presenting it in the form of a game, two groups were formed [COL], some programmed KIBO and others built it."* [CC]
- E4. *"Incredible motivation to experiment and find solutions."* [MOT]
- E5. *"It turned out that sick students did not want to miss school because there was KIBO time."* [MOT]
- E6. *"…KIBO has been exceptionally motivating for our students."* [MOT]

PTD behavior CODES	Goals	Challenges	Strategies	Meaningful moments	Code Frequencies
Communication [COM]	5	0	6	1	12
Collaboration [COL]	13	2	26	7	48
Community building [CB]	1	0	0	5	6
Content creation [CC]	34	0	7	1	42
Creativity [CR]	5	0	9	4	18
Choices of conduct [CHC])	2	2	5	4	13
Motivation [MOT]	3	4	5	15	27

Table 3. Frequencies in Reflection Journals and Interviews

Teachers found that the experience promoted hard work and perseverance while also allowing students to engage in PTD behaviors such as collaboration and communication (see Fig. 16). While teachers were generally novices when it came to teaching with robotics, many of them expressed that they were self-motivated to learn to use KIBO because they liked the idea of robots performing a folk dance as part of the curriculum.

Fig. 16. Examples of children engaging in PTD behaviors.

Despite the general feeling of success, many teachers did say they would have benefitted from longer training and more professional development. This was even more noticeable when trying to teach complex concepts such as repeat loops and conditional statements. Although teachers were given many online resources, they expressed that with the hands-on nature of KIBO, the virtual support was not as helpful as the in-person practice and training.

The main problems reported by teachers in their sessions with KIBO have been the following:

- The accuracy of the KIBO scanner is sometimes low. Therefore, children needed to be explained how to scan the blocks correctly. The teacher points it out as a way to improve since the children complain about it and ask for their help to sometimes scan the bar codes of the blocks. Sometimes, in the scanning of the codes the children put the blocks too close to the scanner
- Likewise, problems were found in the assembly of the wheel and their motors (backwards).
- Spite of the easiness to use the kit, if a specific objective to pursue is not given to children, they only put the blocks together without any logical sequence. Thus, children need specific goals and instructions to program the robot (i.e. that KINO arrives at a certain place, that the robot makes concrete dance steps, etc.).
- Many times children do not wait to hear the KIBO beep or to see the yellow led that confirm the scanned of a block to continue scanning blocks. So they sometimes have to re-scan the sequence.
- The sessions on Fridays at the last hours should be avoided due to the children are tired and altered at the same time by the presence of KIBO robots.

6. Conclusions

This paper evaluates an experiment carried out in three Spanish early childhood schools in Tenerife, Spain, with 172 children (3-5 years old) who learned coding and computational thinking integrated into their actual curriculum activities. This study used KIBO robotics, a developmentally appropriate robot designed for very young children that can be programmed without the use of screens or keyboards by connecting wooden blocks that give different commands to the robot, and can be decorated using craft materials.

We used qualitative and quantitative instruments and combined different research techniques to study an educational experience in order to achieve a more accurate and valid estimate of results for a particular phenomenon, in this case an educational intervention with KIBO robotics. This study focused on the following variables in the development of “coding as a playground” experience: computational thinking, coding skills and positive behaviors in children, and teacher proficiency with KIBO robotics. We claim that it is possible to develop appropriate programming learning experiences in childhood classrooms by integrating coding into different curricular areas (literacy, math, science, engineering, arts) through a project-based approach.

The results obtained allowed us to answer the research questions that guided this study. Regarding RQ1 on *how do teachers integrate coding and computational thinking into their curricular activities?*, the phenomena has been study by the triangulation of different data collected using several methods and instruments (questionnaires, interviews, teacher journal, focus group and direct observation of classroom dynamics) .

The analysis of their notes, discussions, and reflection journals shows that the educators were able to personalize the curriculum. The teachers exhibited autonomy and confidence as they integrated the coding and computational thinking into their curriculum, connecting these concepts with art, music, social studies, while at the same time teaching values and inclusiveness. Furthermore, they were able to adapt their curricular activities to use robotics to teach numbers, geometric forms, colors, literacy, and graphomotor skills. This finding is relevant because it means that coding and computational thinking can be integrated into childhood curriculums in conjunction with other subjects. It is also possible to connect STEAM and coding to their cultural contexts, further promoting significant learning. Also, although the training was the same for all participating educators, the teachers adapted the curriculum to meet their own classrooms’ needs by integrating coding and computational thinking into the formal instruction (Van den Akker, 2007; Bannan, 2009).

About RQ2 on *what programming concepts and computational thinking do preschool children master after being introduced to KIBO robotics?,* the mastery of coding and computational thinking of the students also have been study by the triangulation of different data collected using several methods and instruments (Solve-its checklist observation and teacher journal). The results showed that children achieved a high level of mastery of coding and computational thinking skills using robotics (Benitti, 2012). The students were able to use motors, platforms, and sensors with KIBO. Furthermore, they were able to effectively integrate arts, crafts, and recycled materials into their final robotics projects. Moreover, our results coincide with the current research showing that children can learn to code at early ages (Authors et al., 2002; Cejka, Rogers, & Portsmore, 2006; Perlman, 1976; Wyeth, 2008; Sullivan & Authors, 2013). Specifically, we also worked with 3-year-old children, as compared to other coding and computational thinking studies that focus on the ages of 4-7 years old (Jung & Won, 2018; Öztürk & Calingasan, 2018; Papadakis et al., 2016), confirming that it is possible to start very early (3 years old) with this new literacy (Manches & Plowman, 2017). Our study, which was based on one conducted by Sullivan & Authors (2017) in Singapore, confirms the efficacy of the robotics curriculum designed and the PTD framework for teaching programming in childhood education in different cultures and contexts. Also, our case study was implemented in three different schools with different socio-economic situations, and our findings involving both the teachers and children are highly positive.

As for RQ3 on *what positive behaviors are developed by children in a learning environment of coding as playground?* similarly to the other research questions, the positive behaviors of the students have been studied by the triangulation of different data collected using several methods and instruments (PTD checklist, observation, interviews, teacher reflection journals and focus group). The PTD scores indicate that this intervention was successful in fostering communication and collaboration. At the same time, its effect on promoting content creation and creativity was moderate, and low in terms of promoting conduct choices and community building. The teachers' notes also focus on observations regarding the high level of collaboration among their students and the frequent and varied forms of communication. Educators need pedagogical perspectives to properly integrate robotics, coding and computational thinking concepts into their classrooms (Öztürk & Calingasan, 2018). Thus, this study shows positive implications for expanding this kind of learning environment, which relies on coding and computational thinking as playground, to other early childhood contexts around the world.

The results indicate that children from preschool onwards used KIBO to learn fundamental coding concepts independently of their socio-economic situation. Although contextual factors influence teachers in the design of ICT lessons (Koh, Chai & Tay, 2014), no differences among different school types (public, semi-private or religious, and private) were found in this study, even though one of the participating schools is located in one of the region's lowest income neighborhoods.

The study had limitations, such as the derived from the developmental research in the complex nature of the educational practices (Van den Akker, 1999) and the people involved in the research. Thus, this research comprises the involvement of teachers in the investigation and reflection about their own practices and their students' learning, and one developer of KIBO robot as researcher too. So, in this study we tried to incorporated their voices and perspectives from a critical position.

Another limitation includes the difficulty in using the Solve-Its to assess computational thinking and coding as was planned initially. Even though the teachers were trained by the research team on how to implement the Solve-Its, most of them found it logistically complicated to implement in a short period of time; as a result, we evaluated the programming concepts acquired by children through the structured observation of programming sequences created by children in their final robotic projects using Solve-its checklist. However, since the children were working in groups, it was difficult to isolate each individual child's learning outcomes. Although observation is a common assessment method in early childhood education, group metrics provide limited information in comparison to individual metrics. Also, in one class there was a child with special education needs, since in Spain they are integrated with the general group. This still requires a personalized and individual adaptation of the curriculum that could not be carried out in this study.

Several problems were found in the use of KIBO, for example in the assembly of the wheel and motors (backwards) and in the scanning of the bar codes, due to the children putting the blocks too close to the scanner or sometimes children did not wait to the beep or led to confirm the read of the code to continue the reading of the sequence. Regarding the practices, if children do not have a specific goal to pursue with KIBO, they just put the blocks together but without any logical sequence, and avoid the sessions developed the last hours of Friday because children were tired and altered by the presence of robots in their classroom.

Future research should focus on individual adaptations by teachers of curriculums, including cross-subject coding and computational thinking skills. Also, comparisons of learning outcomes can provide a better understanding of the impact of the teacher's pedagogical strategies and the level of expertise acquired by the children in the area of robotics. In addition, as robotic kits for childhood education such as KIBO become increasingly popular, cross-cultural research may serve to determine best practices and successful pedagogical methods.

References

American Academy of Pediatrics. (2003). Prevention of pediatric overweight and obesity: Policy statement. Pediatrics, 112, 424–430.

Balanskat, A., & Engelhardt, K. (2015). Computing our future: Computer programming and coding. Priorities, school curricula and initiatives across Europe. Technical report, European Schoolnet.

Bannan, B. (2009). The integrative learning design framework: An illustrated example from the domain of instructional technology. An introduction to educational design research, 53-73.

Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. Computers & Education, 58(3), 978-988.

Bers, M. U. (2008). Blocks, robots and computers: Learning about technology in early childhood. New York: Teacher's College Press.

Bers, M.U. (2010). Beyond computer literacy: Supporting youth's positive development through technology. New Directions for Youth Development, 128, 13-23.

Bers, M. U. (2012). Designing digital experiences for positive youth development: From playpen to playground. Cary, NC: Oxford.

Bers, M. U. (2018). Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom. New York, NY: Routledge press.

Bers, M. U., Seddighin, S., & Sullivan, A. (2013). Ready for robotics: Bringing together the T and E of STEM in early childhood teacher education. Journal of Technology and Teacher Education, 21(3), 355-377.

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). Developing computational thinking in compulsory education - Implications for policy and Practice. JRC Science for Policy Report, edited by P. Kampylis, P. and Y. Punie. EUR 28295 EN; doi:10.2791/792158.

Brannen, J. (2017). Mixing methods: Qualitative and quantitative research. Routledge.

Brennan, K., & 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. Retrieved from http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf

Burlson, W., Harlow, D. B., Nilsen, K. J., Perlin, K., Freed, N., Jensen, C., ... & Muldner, K. (2017). Active Learning Environments with Robotic Tangibles: Children's Physical and Virtual Spatial Programming Experiences. IEEE Transactions on Learning Technologies.

Cejka, E., Rogers, C., & Portsmore, M. (2006). Kindergarten robotics: Using robotics to motivate math, science, and engineering literacy in elementary school. International Journal of Engineering Education, 22(4), 711–722

Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., & Eltoukhy, M. (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. Computers & Education, 109, 162-175.

Ching, Y. H., Hsu, Y. C., & Baldwin, S. (2018). Developing Computational Thinking with Educational Technologies for Young Learners. TechTrends, 1-11.

Coleman, M., & Briggs, A. R. (Eds.). (2002). Research methods in educational leadership and management. Sage.

Creswell, J. W., & Clark, V. L. P. (2017). Designing and conducting mixed methods research. Sage publications.

Cunha, F., & Heckman, J. (2007). The technology of skill formation. American Economic Review, 97(2), 31–47.

Di Lieto, M. C., Inguaggiato, E., Castro, E., Cecchi, F., Cioni, G., Dell'Omo, M., ... & Dario, P. (2017). Educational Robotics intervention on Executive Functions in preschool children: A pilot study. Computers in human behavior, 71, 16-23.

Digital News Asia. (2015). IDA launches S$1.5m pilot to roll out tech toys for preschoolers.

Elkin, M., Sullivan, A., & Bers, M. U. (2014). Implementing a robotics curriculum in an early childhood Montessori classroom. Journal of Information Technology Education: Innovations in Practice, 13, 153-169.

Furber S. (2012). Shut down or restart? The way forward for computing in UK schools. Technical report, The Royal Society, London.

Gander, W., Petit, A., Berry, G., Demo, G., Vahrenhold, J., McGettrick, A., Boyle, R., Mendelson, A., Stephenson, C., Ghezzi, C., et al. (2013). Informatics education: Europe cannot afford to miss the boat (2012). Technical report, ACM.

García-Peñalvo, F. J. (2017). Pensamiento computacional en los estudios preuniversitarios. El enfoque de TACCLE3.

Guanhua Chen, Ji Shen, Lauren Barth-Cohen, Shiyan Jiang, Xiaoting Huang, Moataz Eltoukhy (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. Computers & Education. Volume 109,2017, Pages 162-175, ISSN 0360-1315, https://doi.org/10.1016/j.compedu.2017.03.001.

Hill, N. R., Hanks, B. B., Wagner, H. H., & Portrie-Bethke, T. (2016). Early Childhood: Physical and Cognitive Development. Human Growth and Development Across the Lifespan: Applications for Counselors, 177.

International Society for Technology in Education. (2007). National educational technology standards for students. ISTE (Interntl Soc Tech Educ.

Jung, S. E., & Won, E. S. (2018). Systematic Review of Research Trends in Robotics Education for Young Children. Sustainability, 10(4), 905.

Kandlhofer, M., & Steinbauer, G. (2016). Evaluating the impact of educational robotics on pupils' technical- and social-skills and science related attitudes. Robotics and Autonomous Systems, 75, 679-685.

Koh, J. H. L., Chai, C. S., & Tay, L. Y. (2014). TPACK-in-Action: Unpacking the contextual influences of teachers' construction of technological pedagogical content knowledge (TPACK). Computers & Education, 78, 20-29.

l'Académie des sciences (2012). L'enseignement de l'informatique en France - Il est urgent de ne plus attendre. Technical report, Institute de France.

Lee, K., Sullivan, A., Authors, M. U. (2013). Collaboration by design: Using robotics to foster social interaction in Kindergarten. Computers in the Schools, 30(3), 271-281.

Llorens Largo F., García-Peñalvo F. J., Molero Prieto X., and E. Vendrell Vidal, "La enseñanza de la informática, la programación y el pensamiento computacional en los estudios preuniversitarios," Education in the Knowledge Society, vol. 18, no. 2, pp. 7-17, 2017.

Madill, H., Campbell, R. G., Cullen, D. M., Armour, M. A., Einsiedel, A. A., Ciccocioppo, A. L., & Coffin, W. L. (2007). Developing career commitment in STEM-related fields: Myth versus reality. In R.J. Burke, M.C. Mattis, & E. Elgar (Eds.), Women and Minorities in Science, Technology, Engineering and Mathematics: Upping the Numbers (pp. 210 – 244). Northhampton, MA: Edward Elgar Publishing.

Manches, A., & Plowman, L. (2017). Computing education in children's early years: A call for debate. British Journal of Educational Technology, 48(1), 191-201.

Markert, L. R. (1996). Gender related to success in science and technology. The Journal of Technology Studies, 22(2), 21-29.

Meseguer P., Moreno J., Moreno J., Olco, K., Pimentel E., Toro, M., Velázquez, A., & Vendrell, E. (2015). Ensenanza de la informática en primaria, secundaria y bachillerato: estado español, 2015. Technical report, Sociedad Científica Informática de España, Conferencia de Directores y Decanos de Ingeniería Informática.

Metz, S. S. (2007). Attracting the engineering of 2020 today. In R. Burke and M. Mattis (Eds.) Women and Minorities in Science, Technology, Engineering and Mathematics: Upping the Numbers (pp. 184-209). Northampton, MA: Edward Elgar Publishing.

Moreno-León, J., & Robles, G. (2015). The Europe code week (CodeEU) initiative: Shaping the skills of future engineers. In 2015 IEEE Global Engineering Education Conference (EDUCON), pp. 561–566. IEEE, 2015.

Moskal, B. M. (2000). Scoring rubrics: What, when and how? Practical Assessment, Research & Evaluation, 7(3). http://PAREonline.net/getvn.asp?v=7&n=3.

National Institute for Literacy. (2008). Developing Early Literacy: Report of the National Early Literacy Panel. A Scientific Synthesis of Early Literacy Development and Implications for Intervention. T. Shanahan, Chair. Louisville, KY: National Center for Family Literacy

Öztürk, H. T., & Calingasan, L. (2018). Robotics in Early Childhood Education: A Case Study for the Best Practices. In H. Ozcinar, G. Wong, & H. Ozturk (Eds.), Teaching Computational Thinking in Primary Education (pp. 182-200). Hershey, PA: IGI Global. doi:10.4018/978-1-5225-3200-2.ch010

Papadakis, S., Kalogiannakis, M., & Zaranis, N. (2016). Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: a case study. International Journal of Mobile Learning and Organisation, 10(3), 187-202.

Portelance, D. J., Strawhacker, A., & Authors, M. U. (2015). Constructing the ScratchJr programming language in the early childhood classroom. International Journal of Technology and Design Education. pp. 1-16. Online First.

Pretz, K. (2014). Computer science classes for kids becoming mandatory. The Institute: The IEEE News Source.

Resnick, M. (2006). Computer as paint brush: Technology, play, and the creative society. Play= learning: How play motivates and enhances children's cognitive and social-emotional growth, 192-208.

Resnick, M., Martin, F., Berg, R., Borovoy, R., Colella, V., Kramer, K., & Silverman, B. (1998). Digital Manipulatives. Proceedings of the CHI '98 conference, Los Angeles, April 1998.

Rhode Island Department of Education (RIDE). (2013). Rhode Island early learning and development standards. Retrieved from http://www.ride.ri.gov/InstructionAssessment/EarlyChildhoodEducation/EarlyLearningandDevelopmentStandards.aspx#1669797-literacy-l

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

Serholt, S. (2018). Breakdowns in children's interactions with a robotic tutor: A longitudinal study. Computers in Human Behavior, 81, 250-264.

Shonkoff, J. P., Duncan, G. J., Fisher, P. A., Magnuson, K., & Raver, C. (2011). Building the brain's "air traffic control" system: How early experiences shape the development of executive function. Contract, 11.

Siu, K. W. M., & Lam, M. S. (2005). Early childhood technology education: A sociocultural perspective. Early Childhood Education Journal, 32(6), 353-358.

Spanish Ministry of Education, Culture and Sports (2018). Programación, robótica y pensamiento computacional en el aula. Situación en España. Retrieved from: http://code.educalab.es/wp-content/uploads/2017/09/Pensamiento-Computacional-Fase-1-Informe-sobre-la-situaci%C3%B3n-en-Espa%C3%B1a.pdf

Strawhacker, A. L., & Bers, M. U. (2015). "I want my robot to look for food": Comparing children's programming comprehension using tangible, graphical, and hybrid user interfaces. International Journal of Technology and Design Education. 25(3), 293-319.

Strawhacker, A., Sullivan, A., & Bers, M. U. (2013). TUI, GUI, HUI: Is a bimodal interface truly worth the sum of its parts?. Proceedings from IDC '13: The 12th International Conference on Interaction Design and Children. New York, NY: ACM.

Sullivan, A., & Bers, M. U. (2016). Robotics in the early childhood classroom: learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade. International Journal of Technology and Design Education, 26(1), 3-20.. International Journal of Technology and Design Education. Online First.

Sullivan, A., & Bers, M. U. (2017). Dancing robots: Integrating art, music, and robotics in Singapore's early childhood centers. International Journal of Technology and Design Education. Online First. doi:10.1007/s10798-017-9397-0

Twining, P., Heller, R. S., Nussbaum, M., & Tsai, C. C. (2017). Some guidance on conducting and reporting qualitative studies. Computers & Education. Volume 106, March 2017, Pages A1-A9. DOI: https://doi.org/10.1016/j.compedu.2016.12.002

U.K. Department for Education. (2013). The National Curriculum in England: Framework document. London: The Stationery Office.

U.S. Department of Education, Office of Educational Technology. (2010). Transforming American education: Learning powered by technology. Washington, DC. Retrieved from http://www.ed.gov/technology/netp-2010

Van den Akker, J. (1999). Principles and methods of development research. In *Design approaches and tools in education and training* (pp. 1-14). Springer, Dordrecht.

Van den Akker, J. (2007). Curriculum design research. An introduction to educational design research, 37.

White House, Office of the Press Secretary. (2016). FACT SHEET: President Obama Announces Computer Science For All Initiative [Press release]. Retrieved from https://www.whitehouse.gov/the-press-office/2016/01/30/fact-sheet-president-obama- announces-computer-science-all-initiative-0

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35.

Wyeth, P. (2008). How young children learn to program with sensor, action, and logic blocks. The Journal of the learning sciences, 17(4), 517-550.

Yanow, D., & Schwartz-Shea, P. (2015). Interpretation and method: Empirical research methods and the interpretive turn. Routledge.