

Article

Using Educational Robotics in Pre-Service Teacher Training: Orchestration between an Exploration Guide and Teacher Role

Ricardo Silva ^{1,2,3,*} , Fernando Martins ^{3,4} , José Cravino ^{1,2} , Paulo Martins ^{1,5} , Cecília Costa ^{1,2} 
and J. Bernardino Lopes ^{1,2} 

¹ Escola de Ciências e Tecnologia, Universidade de Trás-os-Montes e Alto Douro, 5000-801 Vila Real, Portugal

² CIDTFF—Centro de Investigação em Didática e Tecnologia na Formação de Formadores, Universidade de Aveiro, 3810-193 Aveiro, Portugal

³ Instituto Politécnico de Coimbra, ESEC, IIA, NIEFI, 3030-329 Coimbra, Portugal

⁴ Instituto de Telecomunicações, Delegação da Covilhã, 6201-001 Covilhã, Portugal

⁵ INESC TEC-Institute for Systems and Computer Engineering, Technology and Science, 4200-465 Porto, Portugal

* Correspondence: rjpsilva@esec.pt

Abstract: The proper integration of technology in teaching and learning processes must consider the role of teachers and students, as well as the design of tasks and the context in which they are implemented. Teachers' perceived self-efficacy significantly influences their willingness to integrate educational robotics (ER) into their practice, so initial teacher training should provide opportunities for teachers to participate in structured activities that integrate ER. In this study, a class of pre-service teachers from an initial teacher training programme were provided with their first contact with an ER platform through the use of a simulator. We present the design process of a student exploration guide and teacher guide, developed over three iterative cycles of implementation, assessment and redesign. The analysis of the data collected allowed for improvements in the design of the tasks, the graphic component of the student exploration guide, and more precise indications for the teacher's actions. The main contribution of this study is the chain orchestration between the simulator, student exploration guide and teacher guide, which allowed pre-service teachers to solve a set of challenges of increasing complexity, thereby progressively decreasing their difficulties and contributing to an adequate integration of ER in their future teaching practices.

Keywords: educational robotics; virtual simulator; elementary school pre-service teacher; synchronous remote learning; exploration guide



Citation: Silva, R.; Martins, F.; Cravino, J.; Martins, P.; Costa, C.; Lopes, J.B. Using Educational Robotics in Pre-Service Teacher Training: Orchestration between an Exploration Guide and Teacher Role. *Educ. Sci.* **2023**, *13*, 210. <https://doi.org/10.3390/educsci13020210>

Academic Editors: Sapounidis Theodosios, Michail Kalogiannakis, Nikolaos Fachantidis and Dimitrios Stamovlasis

Received: 28 September 2022

Revised: 15 January 2023

Accepted: 14 February 2023

Published: 16 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

There are different understandings of what constitutes an Open Educational Resource (OER), depending on the focus given to different aspects of copyright or motivations for sharing these resources [1]. No matter what definition one adopts, using an OER is no guarantee of improved teaching or meaningful learning [2]. Global access to OERs brings with it the importance of considering linguistic or contextual aspects for their adaptation to the classroom. For this, we need tools that facilitate the teacher's appropriation of the potential of these resources [3]. In the case of OERs with digital components, integrating technology into teaching and learning processes is an additional problem.

Some researchers looking at the conceptualisation of technology-use have sought to bring education research closer to teaching practice [4–6]. To integrate technology into teaching and learning processes to contribute to meaningful learning, it is necessary to pay particular attention to the role of teachers and students, as well as the design of tasks and the context in which they are implemented [7]. Paiva and Costa [8] suggest using exploration guides for integrating educational software into classroom practice. The need to structure the teaching process in technological environments, taking into account the interaction

between people and artefacts, is the fundamental concern that led to the development of instrumental orchestration [9]. Drijvers et al. [10] propose three constituent elements of instrumental orchestration: a didactic configuration, an exploitation mode and a didactical performance. “A didactical configuration is an arrangement of artefacts in the environment, or, in other words, a configuration of the teaching setting and the artefacts involved in it. These artefacts can be technological tools, but the tasks students work on can be seen as artefacts as well” [11].

The importance of the teacher’s role in integrating technology into the teaching and learning processes is indisputable, creating conditions for students to interact with mathematical concepts [12]. To this end, teachers must be able to mobilize a set of skills that allow them to choose, adapt or create resources appropriate to the defined learning objectives [13]. It is not enough to be proficient in the manipulation of such technological resources to be able to create learning-promoting didactical situations [14]; teachers need to have a thorough knowledge of those resources to transform them into educational tools that serve the learning objectives [6].

The teacher’s role in mediating the construction of students’ knowledge is also important [15]. Inquiry-based learning, known to contribute favourably to students’ learning outcomes [16], is a complex process (Lederman, 2009) strongly influenced by the level of guidance provided by the teacher [15,16]. By providing guidance during tasks, the teacher’s actions create learning environments that contribute to increase students’ competence in solving tasks and gathering information through their investigative practices [16]. For this, two instructional principles must be considered: “generating student ways of reasoning and building on student contributions” [15], valid both for the performance of the teacher and the design of inquiry-oriented instructional materials. The simple inclusion of digital resources in the tasks proposed by the teacher does not imply the creation of learning environments, for such there must be an orchestration that encompasses the digital resource, exploration guide, and teacher’s mediation [17]. Creating a chain of orchestration of artefacts generates favourable conditions for the production of knowledge [5].

The contributions of educational robotics (ER) to learning are recognised, presenting different potentialities in the physical and simulated versions [18]. At the same time, certain characteristics that affect teachers’ acceptance towards integrating ER in their teaching practices have been identified [19,20]. Without teachers’ acceptance of this technology, proper integration into their teaching practices is not possible [21]. Despite recognising the potential benefits of integrating ER into their teaching practices, many teachers still feel uncomfortable using ER [22]. Thus, there is a need to understand how to promote the professional development of teachers through teacher training [22,23]. To this end, the teacher should be proficient in manipulating technological tools and be able to mobilise the specialised knowledge necessary to create conditions that promote learning [14]. Since the perception of self-efficacy strongly conditions the willingness of pre-service teachers to integrate ER into their teaching practices, contact with ER platforms should be included during training [24]. The inclusion of ER in initiatives for teacher professional development serves two purposes: (1) to promote first contact with ER platforms and the development of teachers’ technological knowledge and self-efficacy regarding the use of ER, facilitating future integration into their teaching practices; and (2) enable teachers to participate with their peers in tasks that integrate ER [23]. The possibility to experience the integration of technology in practical and contextualised situations [25] contributes to an improved understanding of the potential and constraints of the artefact and its relationship with the curriculum.

The research problem arises: how to promote the first contact of pre-service teachers with an ER platform, allowing them to understand and experience its functions, so that later they can take advantage of its potential and constraints to create didactical situations that foster learning?

In this paper, we present the process of developing an artefact [26], an inquiry-oriented instructional material [15], more specifically an educational exploration guide for a set of

tasks to be implemented with the ER platform KEIRO, taking into account the characteristics of an efficient exploration guide [8]:

- Maintain a balance between constructivist freedom and minimal guidance;
- Offer suggestions for action and reflection;
- Include screenshots of the software, especially if the guide is aimed at younger learners;
- Include questions that provoke discussion;
- Gradually increase the level of complexity of the questions and tasks throughout the guide;
- Make guides available in printed and digital format;
- Make guides adaptable to different contexts and users;
- Suggest that guide users keep written records of their exploration of the tasks;
- Include instructions that allow users to start exploring the software;
- Include descriptions of relevant aspects of the software;
- Include instructions on how to run the simulation;
- Promote discussion of the results;
- Create conditions to propose and test hypotheses.

The guide was designed to promote pre-service elementary school teachers' first contact with ER in a synchronous remote learning format [27], to enable students to understand some of the functionalities of the chosen ER platform with a high degree of autonomy, supported by inquiry-based teaching practice [16]. We thus seek to establish a starting point for these pre-service teachers to integrate ER into their future practices, in order to promote student learning.

2. Educational Robotics

Starting with Papert [4] as one of its earliest advocates, there is a strong and growing interest in ER within the scientific community [23,28,29]. The literature refers to the integration of ER as a promoter of meaningful learning [30] and interdisciplinarity, facilitating connections between the various STEAM disciplines and students' other knowledge [19,31]. Kim et al. [32] suggest that participation in tasks integrating ER may have a positive impact on perceptions of associated gender stereotypes.

At the start of the past decade, Benitti [33] noted that a large part of ER applications focused on the technological component (robotics, mechatronics and programming) or sought curriculum links close to these areas, a trend that persists to the present day [34,35]. As argued by Alimisis [20] and Jung and Won [28], it is important that research in ER should also assume pedagogical and didactic concerns. In this vein, some researchers have worked on the development of didactic knowledge necessary for pre-service teachers to design lesson plans that integrate ER (e.g., [36,37]) and have sought to understand how mathematics pre-service teachers establish mathematical connections in tasks that integrate ER into educational contexts [38]. However, there are still obstacles to the implementation of ER in teaching practices: specificity of the technical knowledge involved; scarcity of pedagogical and didactic material that facilitates articulation with the curriculum; absence of curriculum guidelines; lack of specific training for teachers' development of didactic knowledge; and high cost of most of the existing platforms [19,20]. The use of Lego robotic kits in interventions with ER is predominant [28,29,39]: students' motivation and the quality of these resources are undisputed strengths, but the high cost prevents widespread access, which is why there is a growing movement towards low-cost and maker-based platforms [35,40].

The constraints imposed by the COVID-19 pandemic brought new challenges to research in ER [41]. Gomes et al. [42] suggest resorting to synchronous remote learning formats, also listing several virtual ER platforms available for smartphone and desktop computers. Birk et al. [43] present a robotics course that took place in an asynchronous online format, supported by slides (PDF) and videos (mp4) made available through a website, reporting an improvement in the students' grades when compared to previous editions. Associating augmented reality with ER, Peralidou and Fachantidis [44] developed

an e-learning environment, presenting as part of their results the g student’s involvement in the proposed tasks. The discussion around physical and virtual ER platforms is not new [18,45] and has been given new impetus by the constraints imposed by COVID-19 [44,46]. Olary et al. [47] used a virtual ER platform to explore the introduction to machine learning with 24 children. Ketterl et al. [48] reported on the potential identified over the years by the German initiative Roberta—Learning With Robots, an online platform for visual programming of educational robots. Ángel-Díaz et al. [49] presented a new, free virtual ER platform, created to facilitate the development of programming skills, computational thinking, and robotics teaching.

3. Materials and Methods

The envisaged intervention plan was implemented in a pre-service teacher training programme, with the design of the learning environment focused on the use of a technological tool (virtual educational robotics platform). Furthermore, the intervention plan foresees multiple iterations, seeks improvements in teaching practice, and investigates relationships between theory and practice to optimise an exploration guide [50], in close collaboration between the researcher and teacher responsible for the curricular unit [51]—which in our study was Didactics of Mathematics I. We, therefore, chose design-based research (DBR) [52] as our research methodology. Reeves [53] suggests that DBR comprises four phases: (1) analysis of practical problems by researchers and practitioners in collaboration; (2) development of solutions informed by existing design principles and technological innovations; (3) iterative cycles of testing and refinement of solutions in practice; and (4) reflection to produce design principles and enhance solution implementation. This study comprises two iterative cycles of testing and refinement of the exploration guide and a third incomplete iterative cycle that does not include intervention and evaluation, described schematically in Figure 1.

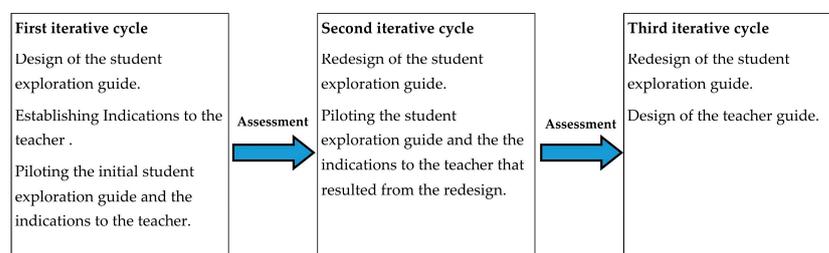


Figure 1. Iterative cycles of testing and refining the exploration guide.

The iterative process of designing the student exploration guide and teacher’s guide was supported by data collected in each intervention. Besides the teacher’s field notes and students’ productions (written answers and algorithms created), video recordings with screen recording software—a common practice in the class routine—were collected. We collected students’ opinions regarding the difficulties they experienced in solving the tasks, as well as suggestions for improving the student exploration guide through unstructured interviews [54].

3.1. Participants

The multidisciplinary team conducting this study was composed of the first author (the teacher-researcher, hereafter referred to as teacher or teacher-researcher), two specialists in mathematics education, and three researchers in the field of STEM teaching and learning with experience in developing and researching educational resources.

The process of designing and developing the exploration guide took place in a first-year class of an elementary school pre-service teacher training programme in central Portugal. The 12 students, all female, enrolled in the curricular unit (Didactics of Mathematics I) participated in the study. Students worked in dyads forming 6 groups, maintaining

the usual pedagogical pairs, and adopting the work routine established in the curricular unit. The right to privacy and anonymity was ensured during all stages, as well as the right to withdraw or refuse participation. Most of the students had some previous experience with programming, as they had already attended a Scratch curricular unit, but had no experience with ER.

3.2. Study Phases

3.2.1. First Iterative Cycle

Starting Point

The starting point for this study was how to create conditions for future teachers to understand and experience the functioning and constraints of an ER platform so that they can later take advantage of its potential to integrate ER in their future teaching practices.

The study aims to answer the following research question: How can the orchestration between ER, scripting, and the teacher's role in solving tasks with a robot be improved in order to minimise prospective teachers' difficulties in using an ER platform?

Design of the First Version of the Student Exploration Guide

Due to the constraints imposed by the COVID-19 pandemic, the interventions took place in a synchronous remote learning format [27], supported by the Zoom platform, creating conditions that allow communication between all participants [23]. The teacher's performance respected the principles of minimal guidance [55] in a collaborative learning environment [56], proposed by Lederman [57] for level 2 of inquiry-based learning (Direct Inquiry), since the problem and the procedures are prescribed by the teacher, and it is up to the students to obtain their conclusions based on an analysis of the information gathered during the investigative practices.

Given the research question, it was necessary to choose an ER platform to be used. There are numerous robotics simulation platforms [18] with different affordances and constraints. For this work, we opted for the KEIRO platform (<https://enginoeducation.com/downloads/>, accessed on 24 May 2021). This choice was grounded on a set of considerations relating to the Open Educational Resource that we should use: besides the simulator, KEIRO also has a physical version of the robots; it also offers additional resources, such as task proposals with instructions for teachers and students, an ER curriculum, and a Youtube channel (<https://www.youtube.com/channel/UCX8mnms5dmyIHtyeKQhRU8A>, accessed on 24 May 2021) dedicated to different aspects of the simulator and physical robots. We considered that this diversity of resources could encourage the pre-service teachers to continue the work initiated with the intervention. This versatility was only considered as an added value.

The KEIRO simulator has a simple, appealing layout and is partially in Portuguese, an important factor, considering that the future pupils of the participants will have rather low reading and writing skills in the early years. KEIRO uses a block-based programming language similar to Scratch [58], a programming language that is already known to the participants through a previous curricular unit. The algorithms created can be saved for later use, as well as exported in image format, making it easier to use in the exploration guide answers. The 3D simulation module is visually appealing and has templates available and minimalist configuration options. These characteristics were key in choosing this OER since they were deemed suitable for the context of the group of participants and duration of the intervention.

The design of the exploration guide sought to include Paiva and Costa's [8] suggestions for the characteristics of an effective exploration guide, as well as respect the characteristics of inquiry-oriented instructional material [15]. In the initial design, two objectives were set for use in the exploration guide, aiming to create conditions for participants to:

- (O1) Learn to programme the Engino Mini robot to move in different directions.
- (O2) Learn to programme the Engino Mini robot so that it can travel in a closed circuit without colliding with the limiters.

The exploration guide is divided into two parts: the first aims to familiarise students with the software and its use [23,24], enabling a greater degree of autonomy in exploring the tasks proposed in the second part. The second part is composed of three tasks (Table 1), with an increasing degree of complexity. The sequence has been designed to allow students to build on what they discover in one task to complete the next one [16].

Table 1. Task sequence and purpose.

	Task	Purpose
1. Programming simple movements	1.1. Programme the robot to keep moving forward for 7 s.	Programme a simple instruction and run its simulation.
	1.2. Find a solution to keep the robot moving for 10 s.	Modify a simple instruction to respect imposed conditions.
	1.3. Create and run an algorithm in the simulator that contains the four available movements (forward; backward; turn left and turn right) with a duration of 15 s.	Create an algorithm with a sequence of instructions.
2. Motion and proximity sensors	2.1. Programme the robot to move forward for 10 s at a speed of 50. Open the Simulator and, before running the simulation, activate Text Feedback and Visual Feedback.	Understand that it is possible to receive information from the robot sensors.
	2.2. In the EnViRo tab, select Back to Home, then the Race Track scenario, with the same robot. 2.2.1. Programme the robot to move forward forever at a speed of 100. Open the Simulator and, before running the simulation, activate Visual Feedback. Briefly describe what you observed. 2.2.2. Find out how to programme the robot to avoid the first collision and test your algorithm in the Simulator.	Create an algorithm that enables them to programme the robot to react according to the information returned by the sensors, achieving the intended objective.
3. Programme the robot to run the track without hitting the limits and test your algorithm in the Simulator.		Create a recursive algorithm that allows the robot to act autonomously according to the information received from the sensors; to experience the process of debugging, optimization of algorithms, and segmentation of problems into parts, simplifying their resolution.

The sequence of tasks proposed includes moments of individual, whole class group and small group work, as well as sharing and discussing the products obtained:

- The software exploration foreseen for the first part of the guide should be undertaken individually, with indications from the teacher for the whole class group;
- For tasks 1, 2 and 3, virtual rooms are created for each group, where they carry out the tasks. After completing the first task, they return to the main room to share and discuss the solutions and their efficiency. It is hoped that this allows students to understand that it is possible to create different algorithms that fulfil the same objective. This procedure is repeated for the remaining tasks.

First Intervention

Seek improvements in the design and implementation of the exploration guide (Appendix A); an intervention was carried out within the mentioned curricular unit, Didactics of Mathematics I.

Before the date of the first intervention, participants were sent an installation link for the KEIRO software and an explanatory document containing information on the objectives and format of the intervention, rights to privacy and anonymity during all stages, as well as the right to withdraw or refuse participation, and established channels for technical support (synchronous and asynchronous) and clarification of participants' questions. We

conducted a pre-session, which served to inform participants about the functioning of the intervention session for the student exploration guide. In particular, the session imparted details concerning the synchronous remote learning format and aimed to help with eventual problems in the installation of the KEIRO simulator. The synchronous session lasted 1 h through the Zoom Colibri platform, and was attended by the teacher and students.

The exploration guide was sent to the students through the institutional platform on the day before the first intervention. This was followed by a 2 h synchronous session via the Zoom Colibri platform, attended by the teacher and students. The tasks proposed in the exploration guide were solved using the Keiro simulator by the groups of students that had previously been set up as described in Section 3.1. They were solved first in small groups and then with the whole class group, as indicated in the student exploration script (Appendix A). The teacher-researcher managed the simultaneous zoom rooms and returned to the main room for the whole class group discussion [59], ensuring communication between all participants [23]. During the solving of the tasks, in both the small groups and whole class group, the teacher-researcher followed the principles of minimal guidance [55].

Assessment

After the intervention, unstructured interviews were conducted during a one-hour synchronous session via the Zoom Colibri platform, attended by the teacher-researcher and students. We collected the students' opinions on the difficulties they experienced in solving the tasks concerning the written and visual indications in the student exploration guide, manipulation of the Keiro simulator, and teacher's actions, as well as suggestions for improving the student exploration guide.

Based on the analysis of the collected data (field notes, analysis of recordings, and contributions of students in the unstructured interviews), the research team mapped the students' difficulties during the solving of the tasks, needs for redesigning the student's guide and teacher's actions.

3.2.2. Second Iterative Cycle

First Redesign of the Student Exploration Guide and Teacher Actions

Based on the evaluation of the first iterative cycle, the research team took some decisions for the redesign of the student exploration guide. The following changes were made: an alteration of the description of task 2.2 adding new and improving the existing images of the guide, which served as graphic indications for the use of the Keiro simulator, and the suppression of task 3.1.

As for the teacher's actions, we recognized the importance for the teacher to provide oral indications regarding the manipulation of the simulator tabs, as well as demonstrate how to export the algorithms created by the students during the resolution of the tasks.

Second Intervention

After the problems in installing the software in the first cycle, a new clarification and technical support session was scheduled to ensure that all participants were able to install and run the software programme. The teacher and students attended this one-hour synchronous session via the Zoom Colibri platform.

The new version of the exploration guide was sent to the students via the institutional platform on the day before the intervention. This intervention, as in the previous iterative cycle, took place in a synchronous remote teaching format lasting two hours and was attended by the teacher and students. Similarly, as described in "First Intervention" for the first iterative cycle, the tasks proposed in the exploration guide were solved using the Keiro simulator by the previously formed groups of students.

Assessment

As in the first iterative cycle, we conducted unstructured interviews after the intervention in a one-hour synchronous session via the Zoom Colibri platform attended by the teacher and students.

Based on the analysis of the collected data [53] (field notes, analysis of recordings, and opinions of students in the unstructured interviews [54]), we mapped the students' difficulties during the solving of the tasks and needs for redesigning the student guide. We considered that adjusting the teacher's actions had been insufficient, and felt the need for an artefact that would allow establishing a relationship between the Keiro simulator and the student's exploration guide [5].

3.2.3. Third Iterative Cycle

Second Redesign of the Student Exploration Guide

Based on the assessment of the second intervention, a prompt was added to the description of Task 2 for students to activate Text Feedback in the Enviro-Tab.

Design of the Teacher's Guide

We proceeded to design the teacher's guide (Appendix B), formalising the indications for the teacher's actions already established, creating a chain of orchestration of artefacts [5]. This design took into account two distinct types of concerns: (1) the management of a class in a synchronous remote teaching format [27] supported by a virtual ER platform [42]; and (2) mediation by the teacher [15,17].

We included indications for management of the class and the small group workrooms on the Zoom Colibri platform, aiming to create conditions that enable communication between all participants [23] in a collaborative learning environment [56]. A balance was sought between minimal guidance [55], prescription of procedures [57], and monitoring during tasks [16]. Our aim was to promote students' autonomy in solving the tasks proposed in the student exploration guide [8], highlighting features of the learning environment that allowed students to reach their conclusions based on the information collected [57] and finding solutions to the sequence of challenges based on previous discoveries [15].

4. Results

4.1. First Iterative Cycle

During the pre-intervention meeting to provide support in installing the virtual ER platform, all participants claimed they had installed the software and had it running without incidents. However, at the start of the intervention only three students had installed the programme. This forced changes not foreseen in the exploration guide, namely regarding group formation and task management: the first 30 min of the intervention were used in group formation as some participants had delays and connection problems.

A summary of the results from the first iterative cycle concerning the student's performance in solving the tasks, difficulties experienced by them and changes made to the student exploration guide, as well as the indications governing the teacher's actions is presented below (Table 2). The difficulties identified during the intervention were mapped based on the analysis of the field notes, whereas the mapping of difficulties during the data analysis resulted from the analysis of the recordings of the unstructured interviews, students' written productions, and screen recordings.

Task 1 was performed without difficulty. The teacher passed through each of the virtual rooms, and participants managed to overcome anticipated obstacles. The questions prepared in advance by the teacher, as well as the chosen everyday examples (Appendix B), allowed the participants to find solutions in a natural language, which they then translated into code blocks and algorithms.

Table 2. Summary of results of the first iterative cycle.

Task	Students' Performance	Students' Difficulties		Amendments to the Exploration Guide	Changes in Teachers' Actions
		Identified During the Intervention	Identified During Data Analysis		
Introductory task	Completed the task	Not all students had the exploration guide with them, which was corrected in the next intervention cycle.		None	None
1	Completed the task without difficulty		Export the created code, validating the solution of the task.	None	Oral instructions during the introductory task and demonstration on how to export the codes in screenshot format.
2	Completed the task without difficulty		Export the created code, validating the solution of the task. Exit the EnViRo tab and return to the programming environment.	Correction to the statement in Task 2.2, adding another image of the work environment and more details to the instructions for opening tabs.	None
3	Only solved Task 3	There was no time to solve Task 3.2.		Task 3.1 was eliminated.	None

However, only Group 1 submitted their task resolutions, managing to keep the robot moving for 10 s (Figure 2) and presenting a solution to keep the robot moving for 15 s through an algorithm that included the four simple movements available (Figure 3). The analysis of the recordings allowed us to realize that the remaining groups also solved the task but had difficulties exporting the code created. This was corrected in the second intervention cycle, with the teacher exemplifying the mechanics of exporting the code in the introductory task, also alerting to the importance of proper file identification to avoid overwriting the previous one.



Figure 2. Resolution of task 1.2 by Group 1.

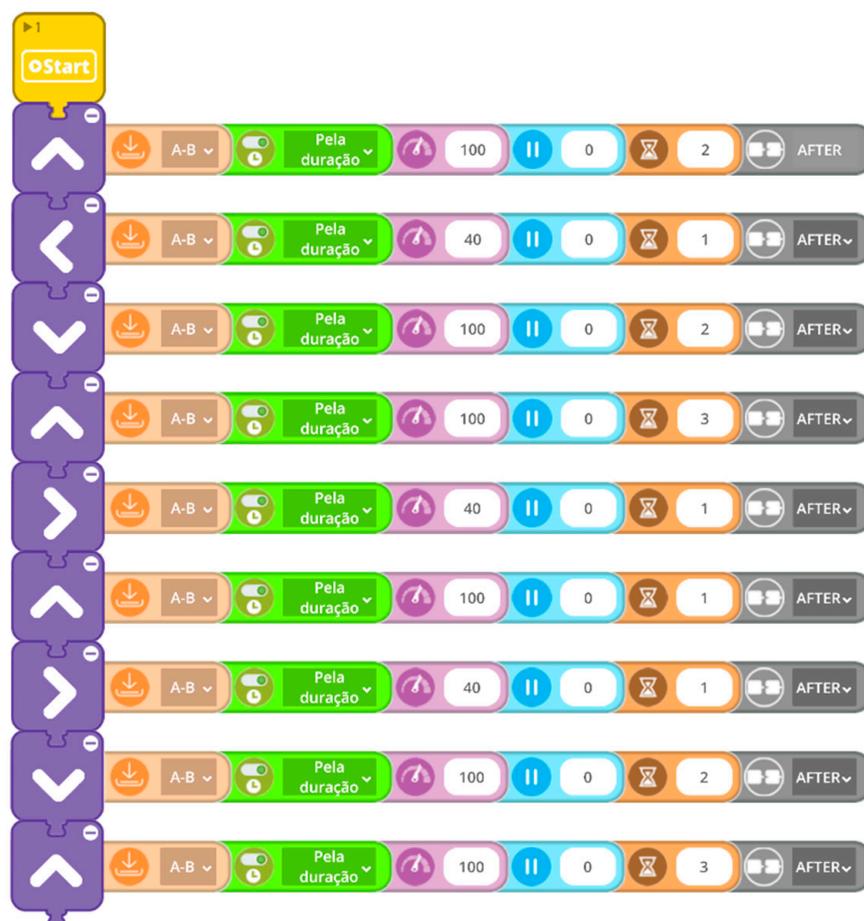


Figure 3. Resolution of task 1.3 by Group 1.

Task 2 was similarly performed by all groups. However, due to difficulties in exporting the algorithms created, it was necessary to use the recordings to find evidence of the participants' productions. Group 6 completed Task 2.1 by programming the robot to move forward for 10 s at a speed of 50 and activating Text Feedback and Visual Feedback in the Enviro-Tab (Figure 4). In the recording we can see that they were able to programme the robot to avoid the first collision by using the information returned by the sensors to trigger a change of direction (Figure 5), building on their discovery in the previous task. Through analysis of the recordings, we could see that the students experienced great difficulties in leaving the EnViRo tab and returning to the programming environment, with some groups choosing to close the programme and start the task again. We decided to correct the description of Task 2.2 by adding the indication to select Back to Home to exit the EnViRo tab, reinforcing it with a new image showing how to close the tab (Figure A5).

Task 3 was performed in a whole class group, with screen sharing by the teacher and collaborative creation of an algorithm, with suggestions from participants (Figure 6). This version of the exploration guide proved to be unsuitable for the length of the intervention, so tasks had to be eliminated. We decided to remove a part of Task 3 as discoveries made by students in Tasks 1 and 2 were necessary to complete Task 3. Task 3.1 corresponded to an optimisation of the solution obtained in Task 3, and therefore did not imply a significant challenge, nor did it require the students to discover new features or programming blocks. Thus, Task 3.1 was removed due to class time management.



Figure 4. Resolution of task 2.1 by Group 6.



Figure 5. Resolution of task 2.2.2 by Group 6.

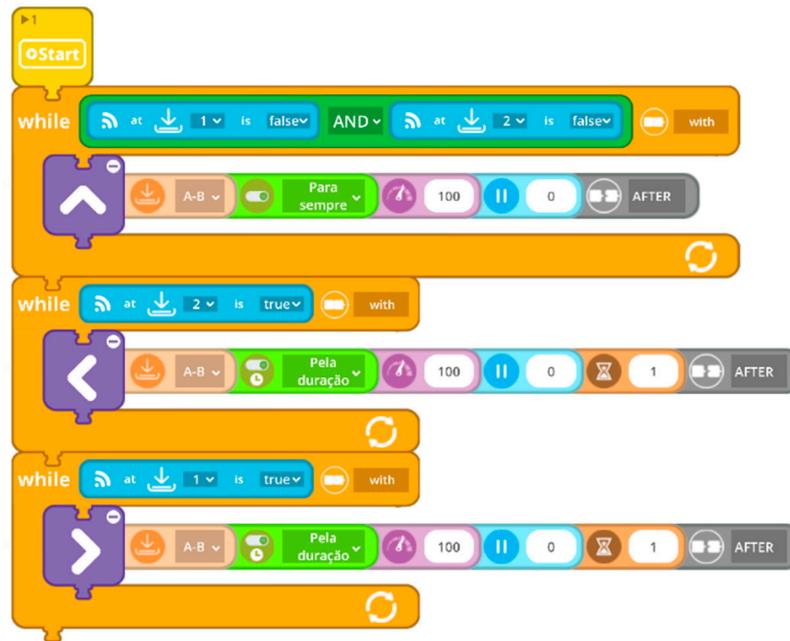


Figure 6. Algorithm created collaboratively for Task 3.

4.2. Second Iterative Cycle

A summary of the results from the second iterative cycle concerning the student’s performance in solving the tasks, difficulties experienced, and changes made to the student exploration guide, as well as the indications governing the teacher’s actions is presented below (Table 3). The difficulties identified during the intervention were mapped based on the analysis of the field notes, whereas the mapping of difficulties during the data analysis resulted from the analysis of the recordings of the unstructured interviews, students’ written productions, and screen recordings.

Table 3. Summary of results of the 2nd iterative cycle.

Task	Students’ Performance	Students’ Difficulties		Amendments Made to the Exploration Guide	Changes in Teachers’ Actions
		Identified During the Intervention	Identified During Data Analysis		
Introductory task	Completed the task	None.	None	None	None
1	Completed the task without difficulty	Manipulation of the display options in the EnViRo tab.	Relate ports A and B to their function in the code blocks.	None	Indications for demonstrating manipulation in the introductory task, and monitoring this difficulty in group follow-up.
2	Completed the task without difficulty in programming		Relate correctly the information returned by the sensors to the robot programming.	Include an indication in the phrasing to activate Text Feedback	Prompts to ask students during task monitoring whether the information returned by the sensors should be included in the written answer.

Table 3. Cont.

Task	Students' Performance	Students' Difficulties		Amendments Made to the Exploration Guide	Changes in Teachers' Actions
		Identified During the Intervention	Identified During Data Analysis		
3	Only one group was able to solve the task		Relate correctly the information returned by the sensors to the programming.	None.	Indications to ensure during monitoring that students understand the relationship between the information returned by the sensors and programming for the required change of direction.

The software exploration task went smoothly. The teacher's demonstration of exporting algorithms in image format was successful, as all groups included the code in their answers to the exploration guide.

During the monitoring of Task 1, a difficulty in manipulating the visualization of the simulation tab was identified. Before sharing and discussing the solutions found, the teacher shared the screen and demonstrated manipulating the visualisation of the simulation tab, and how to return to the default view. In the post-intervention interviews, participants mentioned that this indication should be given in the exploration part of the software, as well as the warning that manipulation is only possible with a mouse, because the touchpad of computers is not interpreted correctly by the software. These procedures were added to the teacher's guide (Appendix B).

The solution found for Task 1.1 was to change the robot's speed to keep the robot moving forward for 7 s. The different solutions found by the groups differ only in the value assigned to the speed parameter, as can be seen in the example of the solution found by group 5 (Figure 7).



Figure 7. Resolution of task 1.1 by Group 5.

For Task 1.2 each group created a different algorithm, ranging from changes to the speed parameter, to the sequence of movements, or the duration of each movement, in order to keep the robot moving forward for 10 s. The solution found by group 3 was to reduce the robot's speed parameter for the previous task (Figure 8), while group 6 chose to maintain the speed, but include rotation movements in their reformulation of the previous programming (Figure 9).



Figure 8. Resolution of task 1.2 by Group 3.



Figure 9. Resolution of task 1.2 by Group 6.

For Task 1.3, each group again found different solutions, with the diversity of solutions again found in the number, sequence or duration of movements. To create an algorithm that includes the four simple movements available and keep the robot moving for 15 s, group 2 opted for an algorithm with a block corresponding to each of the movements, managing the time allocated for each one to make up the 15 s and making the necessary adjustments to the duration parameters (Figure 10). Group 5 decided to add two extra go-forward blocks to the minimum requirements, adjusting the speed and duration parameters (Figure 11).



Figure 10. Resolution of task 1.3 by Group 2.

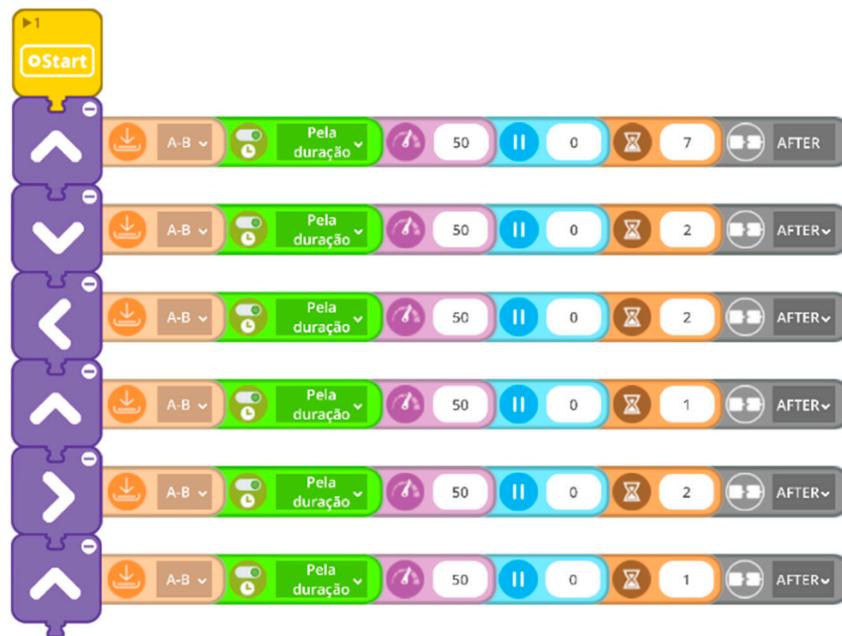


Figure 11. Resolution of task 1.3 by Group 5.

Analysis of the recordings suggested that during the monitoring of Task 1, the teacher should ensure that all students understand the meaning of ports A, B, 1, and 2 (Figure 12), as it was evident that some students struggled to understand their relationship to the code blocks. This was added to the teacher's actions indications, as seen in Section 4.3.2.

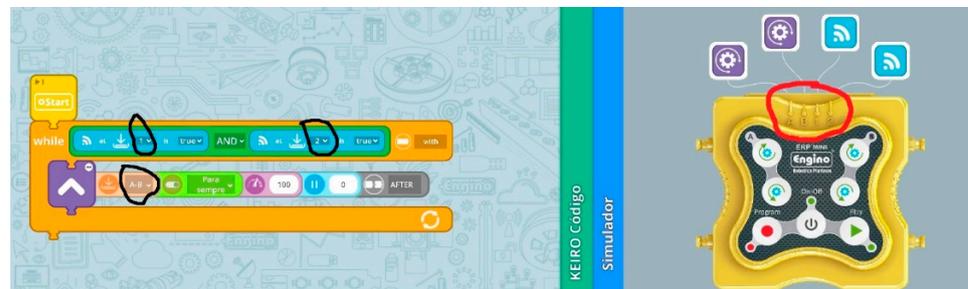


Figure 12. Marked ports and code blocks.

Although no difficulties were evident in the creation of the algorithms for task 2, the analysis of the written answers shows that not all groups were able to relate the information returned by the sensors with the movement of the robot, as can be seen in the answer of Group 2:

"2.1 When you activate Visual Feedback, the sensors become active, displaying a beam of light. Also, in Text Feedback, when the robot hits the wall, the speed goes to 0.

2.2.1 The robot moves forward for about 7 s, stopping when it hits the wall (although time continues to count)."

With a different approach, group 5 tried to describe the relationships between the robot's movement and the information returned by the sensors:

"2.1 The robot started its march and moved for 10 s, and at the ninth second, it hit the wall. When the robot started moving the sensor state was false because it didn't detect any obstacle in its radius. When the robot got closer to the wall, the sensor state became true. We also noticed that when the sensor does not detect any obstacle, its colour is red; and when it detects, it is green.

2.2.1. After we have programmed the robot, it starts its forward march until the collision. At this point, the sensors change from a false to a true state."

Analysis of the written responses suggests that not all groups considered it important to include the feedback of the sensors in their description. This aspect needed to be reinforced in the teacher's monitoring of the task, and an extra suggestion was included in the indications for teacher's actions, as seen in Section 4.3.2.

Task 3 was much more challenging than the previous two, with only one group being able to present a functional solution to the problem (Figure 13). The remaining groups, despite being very close to achieving a solution, had difficulty in relating the correct sensor to the intended change of direction or other small details, as in the case of group 4, which did not realise that the last 3 while-loop blocks were not suitably integrated (Figure 14). This failure in the positioning of the blocks led to a simulation that was quite different from the students' expectations, confusing their line of reasoning. Consequently, they were unable to find a logical explanation for what they observed, which conditioned the process of finding a solution to the problem. A simple reminder from the teacher about the positioning of the blocks could have given this group a completely different perspective on the problem, perhaps enabling them to find a solution.

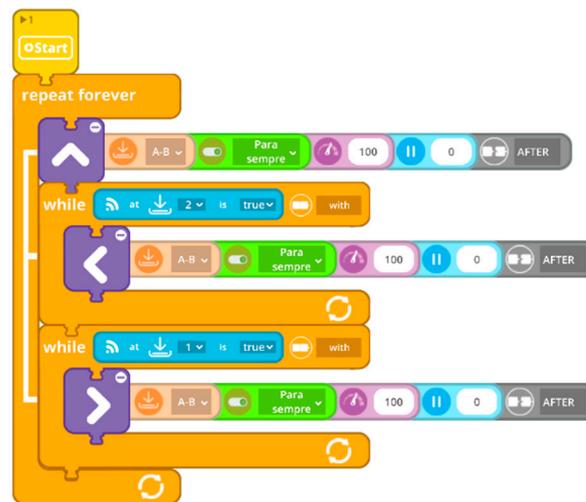


Figure 13. Resolution of Task 3 by Group 2.



Figure 14. Attempted resolution of Task 3 by Group 4.

In the first iterative cycle, this task was solved collaboratively by the whole class, which may have removed some of the difficulties experienced by some groups or students during the second iterative cycle. The teacher's time management made it impossible for all groups to find a solution to this task, as it was not possible to allocate time for the necessary follow-up to each group.

As a corrective measure, it is suggested that during the monitoring of task 2, the teacher ensures students take measurements from each sensor and keep a written record (e.g., sensor 2 is the one on the right side of the train).

Despite only one group being able to successfully solve task 3, analysis of the recordings led us to conclude that no direct changes were needed to the exploration guide, but instead to the teacher's monitoring of the task. Besides the situation already mentioned with group 4, the remaining groups (1, 3, 5, and 6) were unable to establish a correct relationship between sensors 1 and 2 and its position on the robot, which made it impossible to understand which change of direction was required given the information returned by each of the sensors.

Analysis of the recordings showed that group 2 managed to translate the solution found in natural language into an algorithm in the programming language. It seems that the indications provided for the teacher's actions contributed to this discovery by the students, since they suggested that the students first try to write a solution in natural language and only then try to translate this solution into the creation of an algorithm with the available blocks of code.

The achievement of group 2 in managing to find a solution that is a clear optimization of the algorithm created collaboratively in the previous intervention (Figure 6) is considered relevant. The higher efficiency of the algorithm is achieved through a smaller number of programming blocks, made possible by the higher complexity of the recursive solution, also distancing itself from the simulator's pre-defined algorithm so that the robot can travel along the track without crashing into the side limiters.

4.3. Third Iterative Cycle

The results of the third iterative cycle—the third moment of redesign, based on the assessment of the previous iterative cycles—correspond to the current student exploration guide presented in Appendix A and teacher's guide presented in Appendix B.

4.3.1. Third Redesign of the Student Exploration Guide

The wording of Task 2.1 (Appendix A) was changed, including a prompt for students to activate the Text Feedback option: "Programme the robot to move forward for 10 s at a speed of 50. Open the Simulator and, before running the simulation, activate Text Feedback and Visual Feedback (Figure A4)".

4.3.2. Design of the Teacher's Guide

The table below (Table 4) shows the indications created for the teacher's guide (Appendix B) based on the assessment of the first two iterative cycles.

Table 4. Indications created for the teacher's guide.

Placement	Indications to the Teacher
Next to Figure A8	Demonstrate how to manipulate the robot's display options.
Next to Figure A9	Demonstrate how to export an algorithm.
After Task 1.1	Explain how the rooms created for each group work and ask students to manipulate the simulator; make suggestions that allow students to relate the indications given to the servo motor to the movement of the robot; suggest that the students think about how they would solve the problem with their body, hoping that they can transpose the solution into the robot's programming.

Table 4. *Cont.*

Placement	Indications to the Teacher
After Task 1.2	Challenge the students to look for solutions that include different movements, creating conditions to facilitate the solving of the next task.
After Task 1.3	Suggest that students think about the movement of their body to understand that rotation does not imply displacement; ensure that they relate the operation of the different ports to the code blocks for moving the robot.
Next to Figure A10	Encourage students to repeatedly observe the movements of the robot and make them aware that there are several sources of feedback from the robot.
After Task 2.2.2	Suggest that they think about how they would solve the problem with their body, hopefully figuring out that the robot needs to swerve when it “sees” an obstacle; suggest that they use the information returned by the sensors, just as they do with a car’s parking sensors; ensure that they correctly relate the positioning of each sensor and information returned to the direction the robot should take to avoid the obstacle.
After Task 3.3	Suggest that they try to write the solution in plain language and then try to translate it into an algorithm that allows the robot to travel along the track without hitting the side limiters; if difficulties persist, suggest they try to remember what they have learned about Scratch and, in particular, the function of the loop blocks.

5. Discussion of Results

The main conclusion from this study is that an orchestrated framework between the ER, student exploration guide, and teacher actions decreases the difficulties of pre-service teachers using an ER. This is crucial for increasing pre-service teachers’ acceptance [19,20] and self-efficacy [23,24], contributing to an appropriate integration of ERs into their future teaching practices [21].

We discuss the orchestration between a student exploration guide designed for a group of students in a master’s degree course in initial teacher training for elementary school, the teacher’s guide for lesson management, and the Keiro ER simulator. The design of both guides was guided as one of its objectives to create conditions for students to establish first contact with the ER platform Keiro—in synchronous remote teaching [27,42], participating with their peers in tasks that integrate ER [23,24]. Contact with the ER platform should enable the participants to understand some of the functionalities of the Keiro platform and increase their proficiency in its manipulation [14], within a collaborative learning environment [56] in which the teacher’s guidance and monitoring during the tasks would allow students to reach their conclusions based on the information they have gathered [57], finding solutions to the proposed sequence of challenges and building on previous findings [15].

The initial design of the student exploration guide was intended to be consistent with the features of an effective exploration guide as proposed by Paiva and Costa [8]. Nonetheless, the results show that we needed three iterative redesign cycles to improve the script—although the third redesign did not constitute a complete cycle as there was no intervention or assessment. Concerning form: the student exploration guide was made available in digital and printed format, but only in the second intervention was it ensured that all students had it with them; it was found to be necessary to improve the images and, in some cases, add screenshots of the software that would increase the students’ autonomy; the initial instructions for exploration of the software and use of the simulation were not sufficient, an aspect that is considered to have been mostly corrected in the redesign; the

need to keep written records during some of the exploration tasks had not initially been foreseen, a shortcoming that was evident in the relation between the information returned by the sensors and the programming necessary to avoid collisions.

Other features proposed by Paiva and Costa [8] were embedded in the initial design, and there was no evidence to suggest the need to redesign them: the overall complexity level of the challenges presented remained unchanged, with the first task requiring only a simple movement of the robot and culminating with the last challenge of autonomously driving along the track without colliding with the limiters; we sought a balance between students' autonomous solution-finding and minimal guidance by the teacher [55], which was envisaged through everyday examples, translation from natural language into blocks of code, and questioning by the teacher; by choosing tasks that allowed for different solutions and including in the design moments for sharing and discussion in both small groups and whole class groups [59] in a collaborative learning environment [56], we attempted to create conditions for discussing results and proposing and testing different hypotheses and communicating amongst all participants [23]. Finally, we hope that the student exploration guide may be adapted to different contexts and users; to ascertain the adaptability of the guide further research should be conducted aiming to test the exploration guide in different contexts.

The student exploration guide resulting from the redesign process consists of a set of challenges and prescribed procedures, allowing students space to reach their conclusions based on the information collected [57] while creating conditions for students to find solutions for the following tasks based on what they discover in the previous ones [15]. As an example, this was seen in the solutions found by students to keep the robot moving during predetermined time intervals, building on previously discovered solutions [16].

The design of a teacher's guide was not initially planned. However, the results of the first two iterative cycles highlighted the need to formalise the indications for teachers' actions, as well as their evolution throughout the redesign process [53]. We claim that, by establishing a link between the Keiro ER simulator and student's exploration guide [17], the teacher's guide allowed the creation of a chain of artefact orchestration [5]. By including indications about task mediation and monitoring in the teacher's guide [16,55], the orchestration facilitates the balance between constructivist freedom and minimal guidance envisaged in the student exploration guide [8].

The indications in the teacher's guide for lesson management in synchronous remote teaching [27] created favourable conditions to facilitate communication amongst all participants [23] and discussion of different solutions [8,59]. This, in turn, contributed to students being able to find solutions to the proposed tasks based on the contributions of peers [15], and overcoming the challenges of increasing complexity presented in the student exploration guide [8]. The prescription of procedures [57] and demonstration of some aspects of simulator manipulation by the teacher [55] provided in the teacher's guide complemented the instructions for simulator manipulation present in the student's exploration guide [8], allowing students greater autonomy in solving the tasks [8,16].

We hope that participation in the proposed set of tasks contributed to an improvement in students' perceptions of self-efficacy and predisposition to integrate ER into their future teaching practices [24]. Future studies could aim to assess such changes following participation in a similar exercise.

6. Conclusions

Answering the research question based on the discussion of results, we argue that the chain of artefact orchestration presented supported the students in solving the proposed set of challenges of increasing complexity, progressively decreasing the difficulties identified in the first two iterative cycles. The teacher's guide established the connection between the Keiro simulator and student's exploration guide, complementing its written and graphical features. This facilitated the resolution of the proposed tasks and increased degree of autonomy of the students. Ultimately, we claim the proposed orchestration supported

by the teacher's mediation promoted a positive first contact of the students with an ER platform that may encourage future uptake in teaching practice.

Regarding the objectives proposed in the exploration guide, the first one—programme the Engino Mini robot to move in different directions—was fully achieved in the two first iterative cycles. The second goal—programme the Engino Mini robot to travel in a closed circuit without colliding with the limiters—was fully achieved in the first iterative cycle and partially achieved in the second. It is important to highlight that the participants of the study are students at an elementary school pre-service teacher training programme. As such, they were not intended to develop fully functional and efficient algorithms, but rather to solve tasks involving programming a robot with which they had no previous contact. It is hoped that the suggested changes to the teacher's monitoring of the task may improve these results. The chain orchestration of the guides and the Keiro Platform is expected to facilitate the adaptation and replication of the tasks, even in other contexts such as within elementary school. The design process was oriented to allowing the pre-service teachers to adapt—with due care—the proposed tasks to their students in elementary school, particularly regarding technological resources.

Since the participants in this study are future teachers, it is argued that the teacher's guide—as well as an eventual process of adaptation to a new context—can contribute to their professional development, in particular considering the importance of the role of teacher mediation in the construction of students' knowledge and the complex relationship between teacher mediation and the level of guidance during tasks.

The fact that we could not test the implementation of the student exploration guide resulting from the redesign process together with the teacher's guide is considered the main limitation of the current study. In addition to the aspects identified in the second iterative cycle, it would be important to test the student exploration guide and the teacher guide (preferably with another class) to be able to adjust the time allotted for each task.

Scarcity of tasks to use/adapt and lack of training in educational robotics are identified in systematic reviews on the subject as weaknesses in initial teacher training. It is hoped that our results can contribute to this problem.

Author Contributions: Conceptualization, R.S., F.M., C.C., J.C. and J.B.L.; validation J.C., J.B.L. and P.M.; resources, F.M.; writing—original draft preparation, R.S.; writing—review and editing, R.S., C.C., F.M., J.C., J.B.L. and P.M.; supervision, C.C. and F.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work received support from the Applied Research Institute (i2A) of the Polytechnic of Coimbra within the scope of the Exemption for Applied Research (Order no. 7333/2020). This work is funded by FCT/MCTES through national funds and when applicable cofunded EU funds under the project UIDB/50008/2020. This work is also financed by national funds through FCT—Fundação para a Ciência e a Tecnologia, I.P., under the doctoral scholarship 2020.06821.BD.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Ethics Committee of Instituto Politécnico de Coimbra (protocol code 106_CEPC2/2021).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; collection, analyses, or interpretation of data; writing of the manuscript, or decision to publish the results.

Appendix A. Student Exploration Guide

Key: Indications created for the first iterative cycle; Indications created for the second iterative cycle; Indications created during the third iterative cycle.

Objectives:

- Learn to programme the Engino Mini robot to move in different directions;

- Learn to programme the Engino Mini robot so that it can travel in a closed loop without colliding with the limiters.

Introductory activity:

Open the KEIRO application and choose STEM & Robotics Mini, then Open the EnViRo tab (bottom of the window), select the Train, and Plain options, and then click Start (Figure A1).



Figure A1. Initial settings.

Minimise the previous tab and open the Simulator side Table. Explore the different programming blocks available in both side tabs (Figure A2).

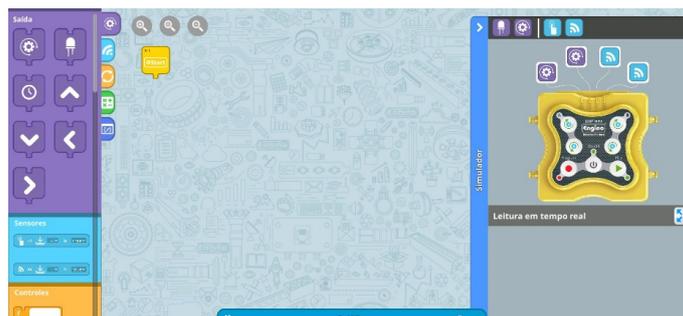


Figure A2. Keiro desktop.

In the following activities, you should always save your algorithm (Figure A3), identifying the task and the group (e.g., Group 1_task1_1).



Figure A3. Export algorithm.

1. Programming simple movements
 - 1.1. Programme the robot to keep moving forward for 7 s.
(After creating your algorithm, open the EnViRo tab and run the simulation).
 - 1.2. Find a solution to keep the robot moving for 10 s.
 - 1.3. Create and run an algorithm in the simulator that contains the four available movements (go forward; go backward; turn left and turn right) and has a duration of 15 s.
2. Motion and proximity sensors
 - 2.1. Programme the robot to move forward for 10 s at a speed of 50. Open the Simulator and, before running the simulation, activate Text Feedback and Visual Feedback (Figure A4).



Figure A4. Text Feedback and Visual Feedback.

Briefly describe what you observe.

- 2.2. In the EnViRo tab, select Back to Home, then the Race Track scenario, with the same robot (Figure A5).



Figure A5. Back to Home and EnViRo settings.

- 2.2.1. Programme the robot to move forward forever at a speed of 100. Open the Simulator and, before running the simulation, activate Visual Feedback.

Briefly describe what you observed.

- 2.2.2. Find out how to programme the robot to avoid the first collision and test your algorithm in the Simulator.

3. Programme the robot to run the track without hitting the limits and test your algorithm in the Simulator.

3.1. Compare your algorithm with the example suggested by the software (Figure A6) and discuss whether yours needs optimisation (excluded at the end of the 1st intervention cycle).

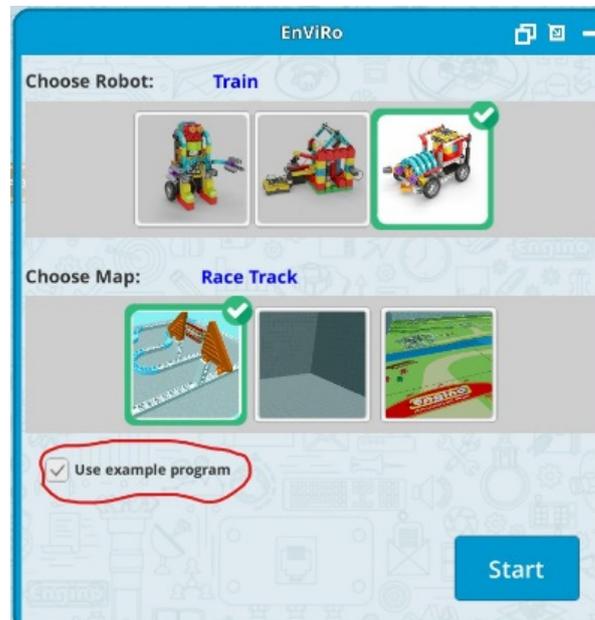


Figure A6. Select example programme.

Appendix B. Teacher's Guide

Key: Indications created for the first iterative cycle; Indications created for the second iterative cycle; Indications created during the third iterative cycle.

Estimated duration: 2 h

Introductory activity:

Open the KEIRO application and choose STEM & Robotics Mini, then Open the EnViRo tab (bottom of the window), select the Train and Plain options, and then click Start (Figure A7).



Figure A7. Initial settings.

Minimise the previous tab and open the Simulator side table. Explore the different programming blocks available in both side tabs (Figure A8).

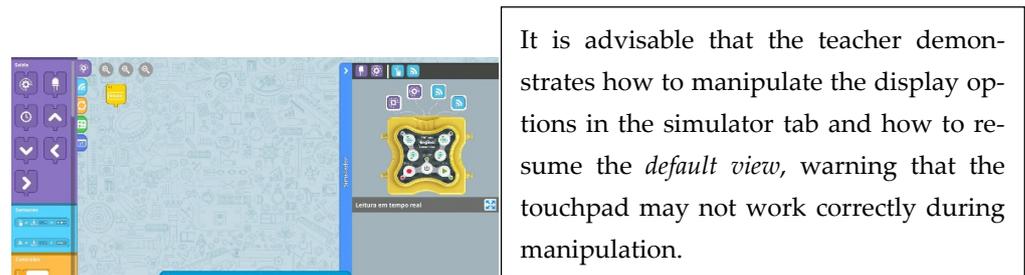


Figure A8. Keiro desktop.

In the following activities, you should always save your algorithm (Figure A9), identifying the task and group (e.g., Group 1_task1_1).

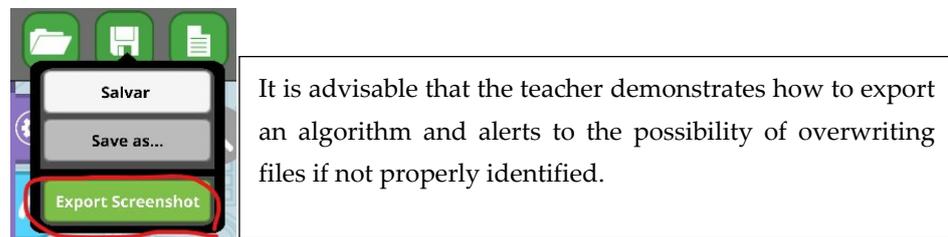


Figure A9. Export algorithm.

1. Programming simple movements

1.1. Programme the robot to keep moving forward for 7 s.

(After creating your algorithm, open the EnViRo tab and run the simulation.)

Before assigning each group to their designated virtual room, the teacher explains how to streamline the process: one person shares the simulator screen and does the manipulation, and the others must collaborate in solving the activities (the group members must take turns in this role so that everyone can manipulate the simulator).

Students are expected to encounter obstacles in solving the challenge. As a suggestion, the teacher should use questioning to help unblock students' difficulties, such as asking what makes the robot move—helping students to associate the instructions for the servo motor to the robot's movement.

If students cannot find a solution to keep the robot moving for 7 s, the teacher may use a hyperbolic example involving body movement (such as walking straight ahead in the kitchen for two minutes). Students may mention walking slower, relating it to the speed parameter in the code block.

1.2. Find a solution to keep the robot moving for 10 s.

The teacher should challenge students to look for solutions that do not exclusively involve modifying the speed parameter. Including other movements in this task that set up the creation of the algorithm needed to solve the next activity.

1.3. Create and run an algorithm in the simulator that contains the four available movements (go forward; go backward; turn left and turn right) and has a duration of 15 s.

Students may not immediately understand the meaning of the instruction "turn right" or "turn left". The teacher may again use examples involving body movement so students understand that turning (left or right) is an instruction that involves rotation.

During the monitoring of activity 1, the teacher should ensure that students understand the meaning of ports A, B, 1, and 2 and their function in code blocks. For example, for the robot to move forward, it is necessary to give simultaneous instructions to both servo motors (one for each wheel), so it is necessary to choose ports "A, B".

When the groups have completed the task, they return to the main room to share and discuss their solutions.

2. Motion and proximity sensors

2.1. Programme the robot to move forward for 10 s at a speed of 50. Open the Simulator and, before running the simulation, activate Text Feedback and Visual Feedback (Figure A10).

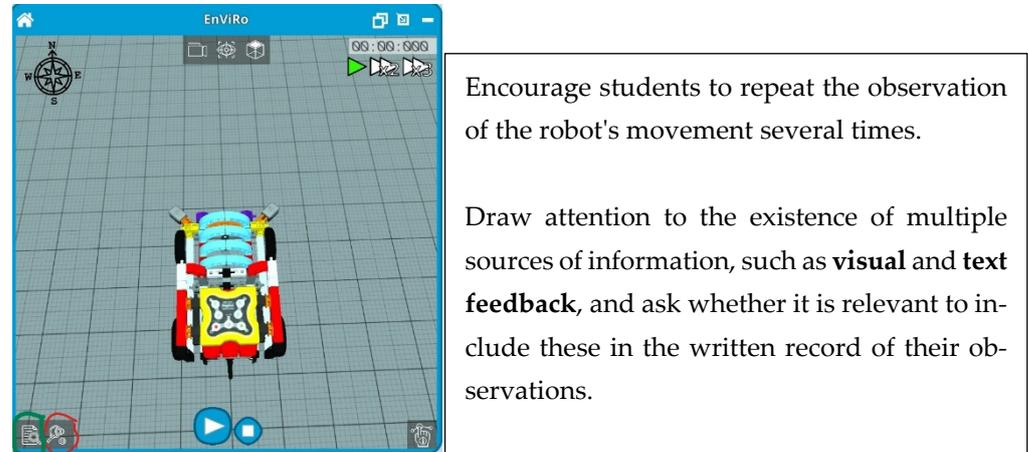


Figure A10. Text Feedback and Visual Feedback.

Briefly describe what you observe.

2.2. In the EnViRo tab, select Back to Home, then the Race Track scenario, with the same robot (Figure A11).



Figure A11. Back to Home and Enviro settings.

2.2.1. Programme the robot to move forward forever at a speed of 100. Open the Simulator and, before running the simulation, activate Visual Feedback.

Briefly describe what you observed.

2.2.2. Find out how to programme the robot to avoid the first collision and test your algorithm in the Simulator.

If students struggle to find a solution to the problem, the teacher may again use examples of body movement (If they are running straight ahead, how do they avoid bumping into an obstacle? The expected answer is to swerve, avoiding the obstacle while running).

To encourage students to use the information provided by the sensors, the teacher may use an example that refers to the use of the senses to decide how to act (such as removing the hand from the fire when we receive information that it is too hot).

As a well-known technological example, the teacher could mention that some cars have parking sensors.

During the follow-up of this activity, the teacher should ensure that students can correctly identify each sensor and its position on the train (Figure A12), advising them to take measurements and make a written record. Sensor 2 is located on the right front side of

the train. This is easily verified by activating text and visual feedback in the simulator tab (Figure A12) and then clicking on the sensor with the mouse pointer. The state changes from False to True, and the colour from red to green (Figure A13). It is expected that with this information students will be able to understand that when sensor two is in the True state, there is an obstacle to the right, and they will be able to imagine a solution to avoid it.



Figure A12. Text Feedback and Visual Feedback.



Figure A13. Activated sensor.

Once this activity is completed, the groups return to the main room to share and discuss their solutions.

3. Programme the robot to run the track without hitting the limits and test your algorithm in the Simulator.

For this challenge, the teacher should start by suggesting to the students to imagine a solution in their natural language and only then try to translate it into the programming language. If students encounter difficulties, the teacher should resort to questioning so that students can conclude that what is intended is that the robot should always move forward and avoid obstacles. They should be able to relate this conclusion to what they know from the Scratch programming language about recursive instructions and loops. If students have no knowledge of programming or cannot remember it, the teacher may promote a discussion about the different programming blocks available (Figure A14) and whether any of them translates to the solution found in a natural language.

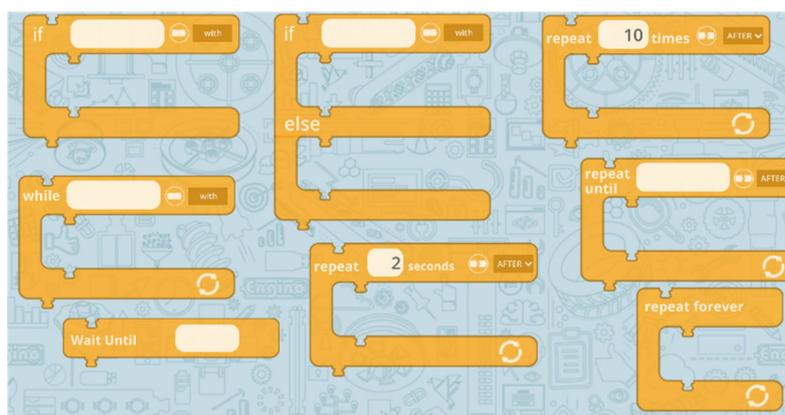


Figure A14. Loop blocks available.

Once activity 3 is completed, they return to the main room to share and discuss their solutions.

References

1. Wiley, D.; Bliss, T.J.; McEwen, M. Open Educational Resources: A Review of the Literature. In *Handbook of Research on Educational Communications and Technology*; Spector, J.M., Merrill, M.D., Elen, J., Bishop, M.J., Eds.; Springer New York: New York, NY, USA, 2014; pp. 781–789. ISBN 978-1-4614-3185-5.
2. Mishra, S. Open Educational Resources: Removing Barriers from within. *Distance Educ.* **2017**, *38*, 369–380. [[CrossRef](#)]
3. Richter, T.; McPherson, M. Open Educational Resources: Education for the World? *Distance Educ.* **2012**, *33*, 201–219. [[CrossRef](#)]
4. Papert, S. *Mindstorms Children, Computers, and Powerful Ideas*, 2nd ed.; Basic Books, Inc.: New York, NY, USA, 1980; Volume 1, ISBN 978-0-465-04627-0.
5. Lopes, J.B.; Costa, C. Digital Resources in Science, Mathematics and Technology Teaching—How to Convert Them into Tools to Learn. In *Proceedings of the Technology and Innovation in Learning, Teaching and Education*; Tsitouridou, M., Diniz, J.A., Mikropoulos, T.A., Eds.; Springer: Cham, Switzerland, 2019; pp. 243–255.
6. Koehler, M.J.; Mishra, P. What Is Technological Pedagogical Content Knowledge? Contemporary Issues in Technology and Teacher Education. *Contemp. Issues Technol. Teach. Educ.* **2009**, *9*, 60–70.
7. Herrington, J.; Reeves, T.C.; Oliver, R. Authentic Learning Environments. In *Handbook of Research on Educational Communications and Technology*; Spector, J.M., Merrill, M.D., Elen, J., Bishop, M.J., Eds.; Springer: New York, NY, USA, 2014; pp. 401–412.; ISBN 978-1-4614-3185-5.
8. Paiva, J.C.; Da Costa, L.A. Exploration Guides as a Strategy to Improve the Effectiveness of Educational Software in Chemistry. *J. Chem. Educ.* **2010**, *87*, 589–591. [[CrossRef](#)]
9. Drijvers, P.; Grauwin, S.; Trouche, L. When Bibliometrics Met Mathematics Education Research: The Case of Instrumental Orchestration. *ZDM Math. Educ.* **2020**, *52*, 1455–1469. [[CrossRef](#)]
10. Drijvers, P.; Doorman, M.; Boon, P.; Reed, H.; Gravemeijer, K. The Teacher and the Tool: Instrumental Orchestration in the Technology-Rich Mathematics Classroom. *Educ. Stud. Math.* **2010**, *75*, 213–234. [[CrossRef](#)]
11. Drijvers, P.; Doorman, M.; Boon, P.; van Gisbergen, S. Instrumental orchestration: Theory and practice. In *Proceedings of the Proceedings of the Sixth Congress of the European Society for Research in Mathematics Education*; INRP: Lyon, France, 2009; p. 10.
12. NCTM. *Principles to Actions: Ensuring Mathematical Success for All*; National Council of Teachers of Mathematics, Inc.: Reston, VA, USA, 2014; ISBN 0873537742.
13. Hegedus, S.; Laborde, C.; Brady, C.; Dalton, S.; Siller, H.-S.; Tabach, M.; Trgalova, J.; Moreno-Armella, L. *Uses of Technology in Upper Secondary Mathematics Education*; ICME-13 Topical Surveys; Springer: Cham, Switzerland, 2017; Volume 2018, ISBN 978-3-319-42610-5.
14. Tabach, M.; Trgalová, J. The Knowledge and Skills That Mathematics Teachers Need for ICT Integration: The Issue of Standards. In *Technology in Mathematics Teaching: Selected Papers of the 13th ICTMT Conference*; Springer: Cham, Switzerland, 2019; pp. 183–203.
15. Kuster, G.; Johnson, E.; Keene, K.; Andrews-Larson, C. Inquiry-Oriented Instruction: A Conceptualization of the Instructional Principles. *Primus* **2018**, *28*, 13–30. [[CrossRef](#)]
16. Lazonder, A.W.; Harmsen, R. Meta-Analysis of Inquiry-Based Learning: Effects of Guidance. *Rev. Educ. Res.* **2016**, *86*, 681–718. [[CrossRef](#)]
17. Lopes, J.B.; Costa, C. Converting Digital Resources into Epistemic Tools Enhancing STEM Learning. In *Proceedings of the Technology and Innovation in Learning, Teaching and Education*; Reis, A., Barroso, J., Lopes, J.B., Mikropoulos, T., Fan, C.-W., Eds.; Springer: Cham, Switzerland, 2021; pp. 3–20.
18. Tselegkaridis, S.; Sapounidis, T. Simulators in Educational Robotics: A Review. *Educ. Sci.* **2021**, *11*, 11. [[CrossRef](#)]

19. Kuhl, P.K.; Lim, S.-S.; Guerriero, S.; van Damme, D. *Developing Minds in the Digital Age*; Educational Research and Innovation; OECD Publishing: Paris, France, 2019. [[CrossRef](#)]
20. Alimisis, D. Robotics in Education & Education in Robotics: Shifting Focus from Technology to Pedagogy. In *Proceedings of the 3rd International Conference on Robotics in Education*; Charles University in Prague: Prague, Czech Republic, 2012; pp. 7–14.
21. Keren, G.; Fridin, M. Kindergarten Social Assistive Robot (KindSAR) for Children’s Geometric Thinking and Metacognitive Development in Preschool Education: A Pilot Study. *Comput. Hum. Behav.* **2014**, *35*, 400–412. [[CrossRef](#)]
22. Zhong, B.; Xia, L. A Systematic Review on Exploring the Potential of Educational Robotics in Mathematics Education. *Int. J. Sci. Math. Educ.* **2020**, *18*, 79–101. [[CrossRef](#)]
23. Anwar, S.; Bascou, N.A.; Menekse, M.; Kardgar, A. A Systematic Review of Studies on Educational Robotics. *J. Pre-Coll. Eng. Educ. Res.* **2019**, *9*, 19–42. [[CrossRef](#)]
24. Schina, D.; Valls-Bautista, C.; Borrull-Riera, A.; Usart, M.; Esteve-González, V. An Associational Study: Preschool Teachers’ Acceptance and Self-Efficacy towards Educational Robotics in a Pre-Service Teacher Training Program. *Int. J. Educ. Technol. High Educ.* **2021**, *18*, 28. [[CrossRef](#)] [[PubMed](#)]
25. Huang, R.; Zbiek, R.M. Prospective Secondary Mathematics Teacher Preparation and Technology. In *The Mathematics Education of Prospective Secondary Teachers around the World*; Springer: Cham, Switzerland, 2017; pp. 17–23; ISBN 978-3-319-38965-3.
26. Guin, D.; Trouche, L. Mastering by the Teacher of the Instrumental Genesis in CAS Environments: Necessity of Instrumental Orchestrations. *Zent. Für Didakt. Der Math.* **2002**, *34*, 204–211. [[CrossRef](#)]
27. Pedro, N.; Monteiro, J.; Fonte, M. Inovação Pedagógica Em E-Learning: Proposta de Um Framework de (Auto) Avaliação de Práticas No Ensino Superior. In *Experiências de Inovação Didática no Ensino Superior*; Ministério da Educação e Ciência: Lisboa, Portugal, 2015; pp. 257–270.
28. Jung, S.E.; Won, E.S. Systematic Review of Research Trends in Robotics Education for Young Children. *Sustainability* **2018**, *10*, 905. [[CrossRef](#)]
29. Toh, L.P.E.; Causo, A.; Tzuo, P.W.; Chen, I.M.; Yeo, S.H. A Review on the Use of Robots in Education and Young Children. *Educ. Technol. Soc.* **2016**, *19*, 148–163.
30. Athanasiou, L.; Mikropoulos, T.A.; Mavridis, D. Robotics Interventions for Improving Educational Outcomes—A Meta-Analysis. *Commun. Comput. Inf. Sci.* **2019**, *993*, 91–102.
31. Miller, D.P.; Nourbakhsh, I. Robotics for Education. In *Springer Handbook of Robotics*; Springer: Cham, Switzerland, 2016; pp. 2115–2134.
32. Kim, C.; Yuan, J.; Gleasman, C.; Shin, M.; Hill, R.B. Preparing Pre-Service Early Childhood Teachers to Teach Mathematics with Robots. *Comput. Support. Collab. Learn. Conf. CSCL* **2017**, *2*, 617–620.
33. Benitti, F.B.V. Exploring the Educational Potential of Robotics in Schools: A Systematic Review. *Comput. Educ.* **2012**, *58*, 978–988. [[CrossRef](#)]
34. Angeli, C.; Jaipal-Jamani, K. Preparing Pre-Service Teachers to Promote Computational Thinking in School Classrooms. In *Computational Thinking in the STEM Disciplines*; Springer: Cham, Switzerland, 2018; pp. 127–150.
35. Giacomassi Luciano, A.P.; Altoé Fusinato, P.; Carvalhais Gomes, L.; Luciano, A.; Takai, H. The Educational Robotics and Arduino Platform: Constructionist Learning Strategies to the Teaching of Physics. *J. Phys. Conf. Ser.* **2019**, *1286*, 012044. [[CrossRef](#)]
36. Kim, C.; Kim, D.; Yuan, J.; Hill, R.B.; Doshi, P.; Thai, C.N. Robotics to Promote Elementary Education Pre-Service Teachers’ STEM Engagement, Learning, and Teaching. *Comput. Educ.* **2015**, *91*, 14–31. [[CrossRef](#)]
37. Kucuk, S.; Sisman, B. Behavioral Patterns of Elementary Students and Teachers in One-to-One Robotics Instruction. *Comput. Educ.* **2017**, *111*, 31–43. [[CrossRef](#)]
38. Souza, C.D.F.; Júnior, A.S.; Barbosa, F. A Interseção Da Robótica Educacional e a Matemática Na Formação Inicial de Professores: Reflexões Acerca Das Conexões Matemáticas. In *Proceedings of the Livro de Atas do EIEM 2019, Encontro de Investigação em Educação Matemática*; Amado, N., Canavarro, A.P., Carreira, S., Ferreira, R.S., Vale, I., Eds.; SPIEM: Loulé, Portugal, 2019; pp. 387–400.
39. Pedersen, B.K.M.K.; Larsen, J.C.; Nielsen, J. The Effect of Commercially Available Educational Robotics: A Systematic Review. In *Proceedings of the Robotics in Education*; Merdan, M., Lepuschitz, W., Koppensteiner, G., Balogh, R., Obdržálek, D., Eds.; Springer: Cham, Switzerland, 2020; pp. 14–27.
40. Stager, G.S. Papert’s Prison Fab Lab. In *Proceedings of the 12th International Conference on Interaction Design and Children*; ACM: New York, NY, USA, 2013; pp. 487–490.
41. Younis, H.; Mohamed, A.S.A.; Jamaludin, R.; Wahab, M.N.A. Survey of Robotics in Education, Taxonomy, Applications, and Platforms during COVID-9. *Comput. Mater. Contin.* **2021**, *67*, 687–707. [[CrossRef](#)]
42. Alves Gomes, A.S.; Da Silva, J.F.; De Lima Teixeira, L.R. Educational Robotics in Times of Pandemic: Challenges and Possibilities. In *Proceedings of the 2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE), Natal, Brazil, 9–13 November 2020*; IEEE: Natal, Brazil, 2020; pp. 1–5.
43. Birk, A.; Dineva, E.; Maurelli, F.; Nabor, A. A Robotics Course during COVID-19: Lessons Learned and Best Practices for Online Teaching beyond the Pandemic. *Robotics* **2020**, *10*, 5. [[CrossRef](#)]
44. Pasalidou, C.; Fachantidis, N. Distance Learning in the Era of COVID-19: Supporting Educational Robotics with Augmented Reality. In *Proceedings of the Education in & with Robotics to Foster 21st-Century Skills*; Malvezzi, M., Alimisis, D., Moro, M., Eds.; Springer: Cham, Switzerland, 2021; pp. 39–51.

45. Samuels, P.; Haapasalo, L. Real and Virtual Robotics in Mathematics Education at the School-University Transition. *Int. J. Math. Educ. Sci. Technol.* **2012**, *43*, 285–301. [CrossRef]
46. Nikolos, D.; Misirli, A.; Komis, V. Children’s Debugging Processes and Strategies with a Simulated Robot: A Case Study. In *Proceedings of the Education in & with Robotics to Foster 21st-Century Skills*; Malvezzi, M., Alimisis, D., Moro, M., Eds.; Springer: Cham, Switzerland, 2021; pp. 64–74.
47. Olari, V.; Cvejok, K.; Eide, Ø. Introduction to Machine Learning with Robots and Playful Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, online, 2–9 February 2021; pp. 15630–15639. [CrossRef]
48. Ketterl, M.; Jost, B.; Leimbach, T.; Budde, R. Open Roberta—A Web Based Approach to Visually Program Real Educational Robots. *Tidsskr. Læring Og Medier* **2015**, 1–22. [CrossRef]
49. Ángel-Díaz, C.M.; Segredo, E.; Arnay, R.; León, C. Simulador de Robótica Educativa Para La Promoción Del Pensamiento Computacional. *Rev. De Educ. A Distancia* **2020**, *20*, 63. [CrossRef]
50. Barab, S.; Squire, K. Design-Based Research: Putting a Stake in the Ground. *J. Learn. Sci.* **2004**, *13*, 1–14. [CrossRef]
51. Gravemeijer, K.; Cobb, P. Design Research from a Learning Design Perspective. In *Educational Design Research*; Routledge: Abingdon, UK, 2006; pp. 29–63.
52. Anderson, T.; Shattuck, J. Design-Based Research: A Decade of Progress in Education Research? *Educ. Res.* **2012**, *41*, 16–25. [CrossRef]
53. Reeves, T. Design Research from a Technology Perspective. In *Educational Design Research*; Akker, J.V.D., Gravemeijer, K., McKenney, S., Nieveen, N., Eds.; Routledge: New York, NY, USA, 2006; pp. 64–78.
54. Cohen, L.; Manion, L.; Morrison, K. *Research Methods in Education*, 8th ed.; Routledge: London, UK, 2018; ISBN 978-1-315-45653-9.
55. Kaiser, G. Mathematical Modelling and Applications in Education. In *Encyclopedia of Mathematics Education*; Lerman, S., Ed.; Springer: Cham, Switzerland, 2020; pp. 553–561.
56. Chan, C.K.K. Co-Regulation of Learning in Computer-Supported Collaborative Learning Environments: A Discussion. *Metacognition Learn.* **2012**, *7*, 63–73. [CrossRef]
57. Lederman, D.J.S. Teaching Scientific Inquiry: Exploration, Directed, Guided, and Opened-Ended Levels. *Natl. Geogr. Sci. Best Pract. Res. Base* **2009**, *8*, 20.
58. Malan, D.J.; Leitner, H.H. Scratch for Budding Computer Scientists. *ACM SIGCSE Bull.* **2007**, *39*, 223–227. [CrossRef]
59. Stein, M.K.; Engle, R.A.; Smith, M.S.; Hughes, E.K. Orchestrating Productive Mathematical Discussions: Helping Teachers Learn to Better Incorporate Student Thinking. In *Socializing Intelligence Through Academic Talk and Dialogue*; American Educational Research Association: 2015; pp. 375–388. Available online: <http://www.jstor.org/stable/j.ctt1s474m1.32> (accessed on 11 August 2022).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.