technologies

MDPI

Article

A Dynamic STEM-Driven Approach through Mobile Robotics to Enhance Critical Thinking and Interdisciplinary Skills for Empowering Industry 4.0 Competencies

Rodrigo Antunes [1,2], Martim Lima Aguiar [1,2] and Pedro Dinis Gaspar [1,2,*]

1 Department of Electromechanical Engineering, University of Beira Interior, Rua Marquês d'Ávila e Bolama, 6201-001 Covilhã, Portugal; rodrigo.antunes@ubi.pt (R.A.); martim.aguiar@ubi.pt (M.L.A.)
2 C-MAST—Centre for Mechanical and Aerospace Science and Technologies, 6201-001 Covilhã, Portugal
* Correspondence: dinis@ubi.pt

Abstract: This study presents an innovative pedagogical approach aimed at enhancing the teaching of robotics within the broader context of STEM (science, technology, engineering, and mathematics) education across diverse academic levels. The integration of mobile robotics kits into a dynamic STEM-focused curriculum offers students an immersive and hands-on learning experience, fostering programming skills, advanced problem-solving, critical thinking, and spatial awareness. The motivation behind this research lies in improving the effectiveness of robotics education by addressing existing gaps in current strategies. It aims to better prepare students for this rapidly evolving field's dynamic challenges and opportunities. To achieve this, detailed protocols were formulated that not only facilitate student learning but also cater to teacher training and involvement. These protocols encompass code documentation and examples, providing tangible representations of the practical outcomes of the course. In addition to the presented curriculum, this paper introduces the developed methodology that strategically leverages 3D-printing technology. The primary focus of this approach is to create captivating add-ons and establish a versatile workspace, actively promoting heightened engagement and facilitating the acquisition of knowledge among students. The research involves the development of tailored laboratory protocols suited to various academic levels, employing a systematic methodology aimed at deepening students' comprehension of STEM concepts. Furthermore, an adaptable infrastructure for laboratory protocols and in-class testing was developed. The efficacy of this teaching/learning methodology is evaluated through student surveys, ensuring its continuous improvement. These protocols are to be integrated into both the robotics courses and teacher-training initiatives. This study aims to contribute to the field by using a dynamic STEM-driven approach based on mobile robotics. It outlines a strategic vision for better-preparing students and educators in the ever-evolving landscape of robotics education demanded by Industry 4.0 technologies.

Keywords: integrated educational resources; robotics; mobile robotics kits; STEM education; Industry 4.0

check for updates

Citation: Antunes, R.; Aguiar, M.L.; Gaspar, P.D. A Dynamic STEM-Driven Approach through Mobile Robotics to Enhance Critical Thinking and Interdisciplinary Skills for Empowering Industry 4.0 Competencies. *Technologies* **2023**, *11*, 170. https://doi.org/10.3390/technologies11060170

Academic Editor: Ahmad Lotfi

Received: 1 November 2023
Revised: 27 November 2023
Accepted: 27 November 2023
Published: 1 December 2023

1. Introduction

The term "robotics" first appeared in the short story "Runaround" by Isaac Asimov, published in 1942. Leonardo Da Vinci designed a knight that was supposed to move automatically, but the construction of an automaton with such characteristics only emerged in 1962 by Joseph Engelberger and George Devol, named "Ultimate" [1]. Robotics is defined as the set of techniques aimed at designing systems capable of replacing humans in their motor, sensory, and intellectual functions.

According to the World Robotics Report 2022 by the International Federation of Robotics, robotics is experiencing expansion, with a particular emphasis on industrial robotics. In 2021, a total of 517,385 industrial robots were installed in factories worldwide, representing a year-on-year growth rate of 31% and surpassing the previous peak in robot installations before the pandemic in 2018, which was around 11%. Currently, the global stock of operational robots has reached a new record of approximately 3.5 million units [2].

The increasing interest in robotics within the industry entails a need to prepare a qualified workforce to deal with technological innovations. Therefore, implementing robotics in education is imperative for the future of engineering.

Robotic educational resources are a powerful and flexible tool as they enable students to discover things on their own, learn new programming languages, and develop working methodologies and critical thinking. Students can apply theoretical concepts in practical robotics projects, which helps to solidify their knowledge and develop technical skills such as programming, mechanics, and electronics. Numerous researchers argue that activities involving robot programming increase participants' interest in the fields of STEM (science, technology, engineering, and mathematics) [3].

In the literature, there are several studies on the effect of robotic resources on students at different academic levels [1,4,5]. Different authors highlight the advantages and potential of robotics in education as a learning tool, including its support in teaching subjects that are not necessarily directly related to the field of robotics.

In a study conducted by Xia and Zhong [6], 22 articles describing experimental, quasi-experimental, and non-experimental studies with educational robotic resources in K-12 children were analysed. Overall, the findings are positive, with participants benefiting from a better understanding of concepts (sensor programming, etc.), attitude change (motivation, efficacy), and skill development. However, other studies report in their respective works that despite the mentioned benefits, most students are unable to provide detailed explanations of the mechanical concepts underlying the activity [7,8].

Despite the progress made in this regard, the implementation of robotics in education is still met with some reluctance, possibly due to limited school budgets, and a shortage of protocols and methodologies that enable activities with educational robots (ER). The lack of an adequate methodology can lead to a lack of interest and motivation on the part of students. Another important aspect is the progression of the complexity of these activities based on the academic level at which children are situated [6].

This study describes a methodology developed and implemented for teaching robotics, motivating students to acquire fundamental knowledge for the future and the development of engineering.

To achieve the overall objective, the following specific objectives have been defined:

- Development of laboratory protocols for different academic levels, sequentially deepening STEM knowledge, aimed at teaching robotics through mobile robotics kits;
- Development of code and add-ons for new laboratory protocols;
- Testing of the protocols;
- Collection of data necessary to identify opportunities for improving the laboratory protocols;
- Discussion of results and evaluation of the progress of students subjected to the methodology developed in the protocols.

This approach aims to promote knowledge acquisition on topics that are fundamental for the future automotive engineering, which will be closely related to robotics.

Thus, this introduction describes the need to use educational robotics within the context of STEM (science, technology, engineering, and mathematics) education and provides a comprehensive overview of the background, objectives, and scope of the case study. The remainder of this paper is organized as follows. Section 2 includes a literature review and fundamentals, overviewing educational robotics, strategies, and methodologies in programming education, as well as a review of the main educational robotics kits that are available. Materials, both hardware and software, and methods are described in Section 3. This section includes the training approaches. Section 4 discusses the results of the questionnaires applied to the students and the influence of this novel approach on STEM skills development. Its performance and added value are also assessed. Section 5 includes a conclusion of the research study, highlighting the contributions, and provides guidelines for future work.

2. Literature Review and Fundamentals

This section provides a comprehensive analysis of various approaches and their corresponding educational robotics kits. The primary aim is to uncover opportunities to enhance current robotics teaching methods. This detailed examination explores the distinctive characteristics, functionalities, and overall effectiveness of diverse approaches and kits available in the market. Furthermore, the literature review highlights the constructive impact of educational robotics on students' computational thinking, problem-solving abilities, and creative prowess, supporting the proposed pedagogical approach. Additionally, a comparative assessment of educational robots is provided, such as Bee-Bot, WeDo 2.0, Lego Mindstorms NXT, and mBot, which serves as a valuable resource for educators in making well-informed decisions regarding the most suitable robotics kits to support their teaching objectives. The goal is to identify methods that enhance the teaching and learning dynamics in robotics, aligning with both students' needs and the prerequisites of the educational curriculum, while also addressing the demands of the current and future job market. This research provides best practices and key components, resulting in the creation of an innovative and highly effective methodology designed to kindle students' interest and improve skills acquisition that is relevant to the fields of robotics and engineering.

2.1. Computational Thinking

Computational thinking (CT) is the systematic thinking process learners employ while "solving problems, designing systems, and understanding human behaviour by drawing on fundamental concepts of computer science (CS)". Ideas involving CT emerged in the 1950s [9]. Papert [10] was the first to describe CT in his work related to programming in Logo and the Logo turtle, an educational robot. In the early 2000s, CT was revitalized by Wing (2006) as she refined the definition and emphasized the importance of CT as part of every child's skill set. However, in the field of education, there is still no consensus on the definition of CT [11]. Some definitions of CT remain linked to disciplines in the field of computing, specifically computer science [12]. Other definitions have been created in the context of other non-CS curriculum units. For instance, Weintrop et al. [13] conducted a literature review on CT and interviewed experts in the fields of mathematics and science to develop a definition based on four categories: data essays, modelling and simulation essays, computational problem-solving essays, and systems thinking essays. Others relate computational thinking to engineering, and there are still those who define CT from a multidisciplinary approach. Shute et al. [14] assert that CT is a necessary conceptual foundation for solving problems effectively and efficiently.

The National Research Council (NRC) conducted a series of workshops focused on CT and subsequently released a report on its educational and cognitive implications. The participants in the NRC workshop agreed that it was necessary to take the next step in conducting similar activities with a greater focus on the pedagogical aspects of CT [15]. To implement CT activities in K-12 classrooms, the Computer Science Teachers Association (CSTA) and the International Society for Technology in Education (ISTE) formed a team of education and industry leaders to develop a framework that integrates computer science and computational thinking [16].

Various tools have been used to teach these components, some of which are related to the work carried out by Papert [10] in the field of educational programming language, including educational toys and applications designed for children. Currently, a wide range of robotic kits can be found on the market [12].

2.2. Strategies and Methodologies in Teaching Programming/Robotics

There are various teaching philosophies. The main recommendation that emerges from the literature is that teaching should focus not only on learning the characteristics of a particular programming language but also on combining them and particularly on the related problem of designing basic programs. One way to achieve this could be through the introduction of numerous examples as programs are developed, discussing the strategies used as part of this process [17].

According to Coll et al. [18], the teacher should: gradually present the content, with moments of recapitulation, summary, and synthesis; make analogies, using students' prior knowledge; be explanatory regarding the proposed activities and what is to be taught; provide opportunities for students to execute procedures voluntarily, consciously, and innovatively; and make improvements. Students should be motivated to learn the procedures and be able to self-evaluate, knowing that the construction of knowledge depends on their effort.

Students should develop concrete and real projects, and it is necessary to make some simplifications through a method of gradual development [19].

Some authors emphasize the importance of a trial-and-error approach for students to find programming errors, using reflection, understanding, analysis, and hypothesis testing [20].

According to Roumani [21], the curriculum should be taught in an inverted manner, meaning that after students are comfortable with the behaviour and applications of the main data structures, they should learn how to implement them.

Teachers should adopt strategies and activities that motivate students to engage in their learning and allow them to develop autonomy. When engaging in challenges, an increasing level of complexity should be emphasized, encouraging the integration of knowledge from various disciplines, and students should cooperate in small groups to solve them [22].

An interesting field to be explored is robot football (soccer), since it can embrace a large number of disciplines such as computer vision, intelligence artificial, computer science, physics, mathematics, mechanical, and general engineering. Apart from being a field that connects different topics of engineering, it is attractive for all kinds of people due to football being one of the most popular sports and the idea of seeing robots playing soccer is fascinating for children, adolescents, and adults [23].

2.3. Educational Robotics

As mentioned earlier, there is a wide range of robots available for all levels of education, serving different purposes [12]. Several studies have shown that educational activities involving robotics can be highly effective in developing skills such as critical thinking, creativity, problem-solving, teamwork, and decision-making, among others [24].

Robotics has generally been applied in education for students ranging from 3 to 18 years old, from preschool to secondary education [6]. According to Xia and Zhong [6], the majority of applications are found in elementary school students (57%), followed by secondary school students (24%), and kindergarten children (19%). More than half of the studies conducted used samples with fewer than 80 participants and a duration of less than 2 months. The dominant type of robot used in the studies was the LEGO brand (67%).

Educational robots are programmed by their users to act based on specific information obtained from the environment in which they are placed. They are equipped with a set of sensors that enable them to measure various conditions and transmit this information to the robot's controller. There is a wide range of sensors available, including light sensors, touch sensors, temperature sensors, humidity sensors, rotation sensors, sound sensors, colour sensors, and distance sensors. At the same time, the robot has actuators, which, as the name suggests, allow it to interact with the environment. Typically, these are motors that enable the addition of various mechanisms such as robotic arms, wheels, and transmission systems (gearboxes) [22]. Thus, through the analysis of scientific references and studies, it can be concluded that there is a wide range of educational robots available. Some options even allow for the construction of robots using low-cost or recyclable materials. Alongside the growth of this field of robotics, various block-based programming environments have emerged, designed for use by children. These environments facilitate programming and interaction, making initial encounters with programming more accessible and contributing to educational development. Given the diversity of educational robot offerings, this section will address some solutions available on the market and experimental studies that explore the influence of their use on learning (Table 1).

Table 1. Detailed overview of each educational robotics platform, emphasizing additional features and characteristics.

Platform	Description	Key Features	Study/Reference
Bee-Bot	Prominent floor robot in elementary education, controlled through physical buttons for directional programming.	• Resembles a bee; Controlled via physical buttons for turning, and moving forward/backward; Supports the development of programming skills, cognitive abilities, and spatial awareness.	Diago et al. [1] Schina et al. [25] Kazakoff et al. [26]
WeDo 2.0	Robotics kits by LEGO Education, designed for interactive teaching of basic concepts.	• LEGO pieces, motors, sensors; Interactive programming software (WeDo 2.0); Widely used in classrooms and robotics clubs; Fosters hands-on learning of robotics and programming.	Çakır et al. [5]
Lego Mindstorms NXT	Versatile robotics kit using LEGO building blocks with touch, colour, and ultrasonic sensors.	• Programmable NXT controller for precise control; Touch, colour, and ultrasonic sensors; Flexible building with LEGO blocks; Intuitive visual programming with blocks (similar to Scratch).	Atmatzidou and Demetriadis [4]
mBot	Educational robot by Makeblock, designed for computer science and STEM learning.	• CyberPi processor, sensors, and motors; Supports Scratch (block-based programming) and Python; Versatile for add-ons like temperature sensors, gas sensors, and accelerometers.	Voštinár [27]

2.4. Conclusive Note

In this section, a comprehensive exploration of various technologies and methodologies for teaching robotics was conducted. Building upon the existing body of work, the primary objective of this study is to make a valuable contribution to the expansion of knowledge in the field of educational robotics. The present work intends to study novel approaches and methodologies that effectively foster student engagement and inspire their pursuit of robotics.

There is an attempt in this research to improve the effectiveness of robotics education by addressing existing gaps in current strategies. The research objective is to better prepare students for the dynamic challenges and opportunities in this rapidly evolving field. Thus, detailed protocols were formulated that not only promote student learning but also cater to teacher training and involvement. These protocols encompass code documentation and examples, providing tangible representations of the practical outcomes of the course. These protocols can be integrated into both the robotics course and teacher-training initiatives. The paper introduces a methodology that leverages 3D-printing technology. The primary focus of this approach is to create captivating add-ons and establish a versatile workspace, actively promoting heightened engagement and facilitating a profound acquisition of knowledge among students.

3. Materials and Methods

3.1. Selection of a Robotics Kit for the Activities

For the implementation of the protocols and associated activities, a more recent version of the robots mentioned in the previous chapter was chosen. The selection of the mBot2 from Makeblock for this purpose is attributed to its capability to accommodate add-ons and utilize mechanical components that closely resemble those used in the field of robotics and Industry 4.0. This section will delve into all the technical aspects of the mBot2, as well as the programming environment, mBlock. Subsequently, the methodology employed in a robotics course with secondary school students will be elucidated.

3.1.1. mBot2

The mBot2 is an educational robot designed with a STEM approach [28,29]. Its extensive capabilities make it a good choice for introducing robotics at the primary academic levels, but it can also be explored in secondary and even university-level education.

The microcontroller used in the robot is the CyberPi, programmed through the mBlock 5 software, compatible with programming languages such as Scratch and Python. The CyberPi offers great versatility as it includes a set of sensors and actuators such as a microphone, speakers, an inertial measurement unit with a gyroscope and accelerometer, a light sensor, operation buttons (including a joystick), and a colour display. Complementing the CyberPi, the mBot kit includes an ultrasonic sensor and a line follower sensor with four RGB sensors to detect different colours. In addition to this range of sensors and actuators, the mBot2 allows communication via Wi-Fi. The mBots can be connected to form a local network where robots can communicate with each other wirelessly. The mBot2 can be connected to the Internet and perform functions such as voice recognition or access libraries that contain various functionalities like machine learning. The different components comprising the mBot2 are shown in Figure 1.

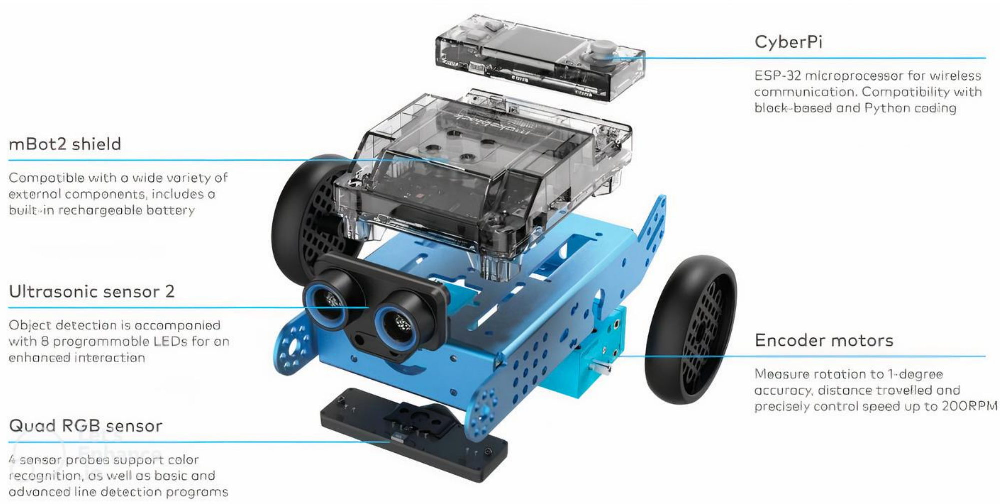

Figure 1. Exploded view of the mBot2, showcasing the different components in its structure [29]. Makeblock is their respective owners' trademark and has not sponsored, authorized, or endorsed this work.

The locomotion of the mBot2 is ensured by encoder motors. With this type of equipment, students can precisely control the rotation, speed, and position of the wheels and the robot. Additionally, the motors can be used as servos and even as knobs, returning data to the system as if they were sensors. The major advantage of encoder motors is that they allow for a greater integration of mathematical concepts in program development, such as moving precise distances, calculating curves, and mapping a path to navigate a maze, transferring the results to the computer.

The environmental recognition is performed through the ultrasonic sensor (Figure 2 on the right) and the quad RGB line follower sensor (Figure 3). The ultrasonic sensor provides more accurate and consistent readings compared to previous versions. It is located at the front of the mBot2, in the small cylindrical structures resembling "eyes". This sensor emits small amounts of ultrasound and receives the echo. Based on the time it takes for the sound to be returned to the receiver, mBot2 calculates the distance to the object.

Figure 2. Front view of the mBot2, highlighting the ultrasonic sensor [29]. Makeblock is their respective owners' trademark and has not sponsored, authorized, or endorsed this work.

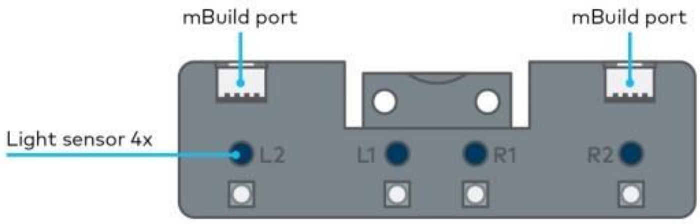

Figure 3. Integrated Quad RGB Sensor in mBot2: there are four RGB sensors incorporated into a single sensor along with a set of two ports compatible with CyberPi [29]. Makeblock is their respective owners' trademark and has not sponsored, authorized, or endorsed this work.

The line follower sensor also works as a colour-detection sensor, featuring four RGB sensors integrated into a single unit (Figure 3).

The RBG sensors are defined as L1, L2, R1, and R2 (L for the left side, R for the right side). They automatically detect the RGB (red, green, blue) values of the reflected colour and internally compare the mixture of values with different preset colors. The programming process becomes simpler because the sensor can transmit the detected colour, eliminating the need for the user to check the RGB colour codes. The sensor can detect six different colors, in addition to black and white. It also contains a button on the top of the sensor that is used to calibrate readings according to the ambient light [29].

3.1.2. Programming Software mBlock 5

mBlock 5 (example of the programming window in Figure 4) was designed for education in STEAM. Inspired by Scratch 3.0, it supports both graphical and textual programming languages. It allows the creation of custom projects, games, animations, and programming of devices such as Makeblock robots and microbit. mBlock 5 provides two editors, namely the block-based editor and the Python editor. Additionally, it integrates technologies such as artificial intelligence (AI) and the Internet of Things (IoT) [30].

The block-based editor (Scratch, Google/MIT, Cambridge, MA, USA) is the default editor in mBlock 5. It is divided into three workspace areas: stage area, blocks area, and scripts area. The stage area is where backgrounds are selected, devices are connected, and sprites are defined (two-dimensional graphical objects that move on the screen based on the program), and their respective scenes. In the blocks area, blocks are organized into categories and colors. Lastly, the scripts area is where the program can be built by dragging and arranging the blocks.

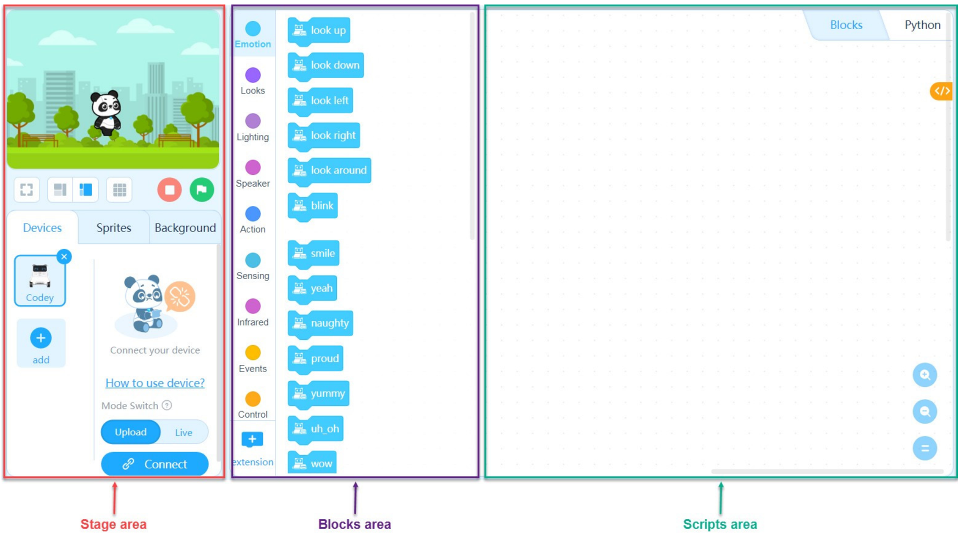

Figure 4. Scratch editor, available in the mBlock 5 software, is divided into three main work areas: stage area; blocks area, and scripts area [30]. The Scratch Foundation has not sponsored, authorized, or endorsed this work.

3.2. Procedure

The planning process aimed at teaching robotics through mobile robotics kits to students from a vocational school is presented. In this training, which lasted for 3 classes of 2 h each for each group, we opted for the application of an ABRP methodology, where students had to solve various problems, increasing the level of difficulty, always with a group work dynamic, promoting critical thinking, collaborative work, and socialization among them. During the taught classes, the students had to be the agent of their intellectual development, developing critical analysis, reasoning, and creativity. Guidance was provided only when necessary. Support to students was provided through suggestive questions and useful indications in clarifying the challenges or restating the objective to achieve the expected outcome. Students had to seek a solution taking into account previous experiences and be able to detect errors, isolate them, and correct them. Challenging problems were designed to promote reflection and foster cooperation.

In the planning of the training, the total number of classes was divided into 3 classes of 2 h each for each group, starting with a class in which the robotic kit was introduced and the basic aspects of its programming were explained. After the first contact, challenges were presented in increasing order of complexity to prepare the students for the final challenge.

Table 2 presents the protocols that were developed along with their corresponding resources for implementation. All of the protocols were applied in teaching and learning initiatives with the students, except for the “Football” protocol, which was implemented at a later stage. Figure 5 provides a summary of the three classes discussed in this section, which will be further described in the following subsections.

Table 2. Summary of the protocols developed and their respective resources for implementation.

Act.	Details	
1st steps	**Programming Language**: Scratch	**Materials required**: Computer, mBot2, ultrasonic sensor.
	Goal: Explain how to install the application and set up the connection with mBot2. Provide a brief tutorial on how to create a program that utilizes the values returned by the ultrasonic sensor to control the robot's movements.	
Line-following	**Programming Language**: Scratch	**Materials required**: Computer, mBot2, black tape, Quad RGB sensor.
	Goal: Introduce students to the operation of the line following and Quad RGB colour-detection sensors. It follows the logic sequence of the previous activity. All steps are described in detail to enhance students' understanding. The objective is to follow a path using the line follower sensor. A step-by-step list is given for the preparation and execution of the activity.	
Maze	**Programming Language**: Scratch or Python	**Materials required**: Computer, mBot2, black tape, Quad RGB sensor, ultrasonic sensor, maze scenario.
	Goal: Apply path planning algorithms, taught in the robotics course, to navigate the mBot2 through a random maze. The strategy employed will be at the discretion of each group.	
Foot-ball	**Programming Language**: N/A	**Materials required**: Smartphone or tablet, mBot2, bumpers, ball, multi-purpose worktable.
	Goal: This protocol aims to provide an interactive, playful, and engaging activity that does not require programming skills and is quick to prepare and explain. It is ideal for short visits by students of all ages to the robotics laboratory. mBots are controlled using the Makeblock application to play football. The objective is to provide a fun and immersive experience that allows visitors to directly interact with the robots, thereby stimulating interest and curiosity in the field of robotics through a competitive activity.	

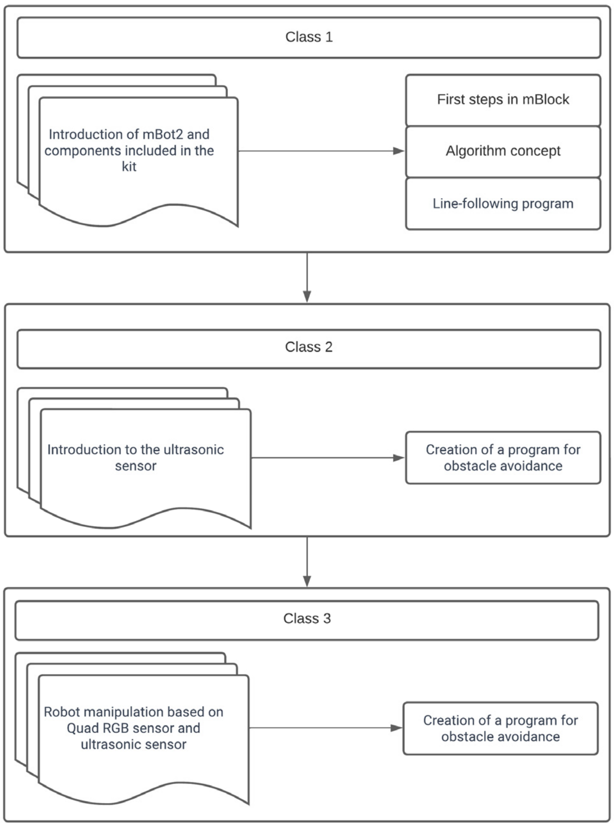

Figure 5. Outline of the lessons taught to students from a vocational school using educational robotics kits (mBot2).

3.2.1. Class 1—Introduction to mBot2

The first class began with an introduction to robotics, explaining how robots can assist humans in performing tasks and the contexts in which they are used. The students completed the first questionnaire, a set of questions aimed at understanding if they had any previous contact with robotics, programming, sensors, and other elements constituting a robot. The mBot2 was introduced with the protocol "Initial configuration and first steps", an overview of this protocol is presented in Table 2, including its mechanical components, the types of sensors it contains, and a brief explanation of their functioning principles. The wide range of possible extensions that can be acquired and implemented on the mBot2 was also discussed. While the participants installed the mBlock application on their computers, an overview of what an algorithm is, the types of commands available in Scratch, and their functionalities were provided. During this phase, the focus was mainly on the commands necessary for the planned set of activities in the first class. The objective of the first activity was to provide a set of instructions to the mBot2 that would allow it to navigate a path without using any sensors. The students had to figure out which set of blocks would enable them to reach the goal by perceiving the space, estimating distances, and adjusting the rotation value in curved sections of the track. Whenever students had difficulty understanding the functionality of a particular block, a review of the command's content was conducted, followed by using that instruction with the robot so that the students could associate a physical movement with the previous explanation. This process was repeated in all classes. After the groups achieved the goal, they were asked to think about possible improvements, and the role of sensors in robotics was emphasized. The activity was repeated, but this time with the assistance of a line-following sensor installed on the robot. This change required the students to rework the entire program, as all the decisions the robot had to make were based on the values transmitted by the sensor. This last activity is part of "Line-following", an overview of the protocol is presented in Table 2.

3.2.2. Class 2—Sensor Conjugation

In the second class, a general introduction to the functioning of an autonomous robot was given, emphasizing the imperative role of sensors in the successful performance of tasks, with particular emphasis on the ultrasonic sensor in the case of the mBot2. Examples of living organisms that use ultrasound, as well as examples in the healthcare field, were presented. The content taught in the previous class was reviewed, and students had the opportunity to optimize the program to make the robot faster in completing the course, creating some competition among groups while consolidating concepts.

After completing the introductory activity of the class, programming blocks provided by the ultrasonic sensor extension were introduced, and students were taught how to create a simple program to stop the robot's locomotion when it detected an object. Here, students had some time to explore the functionalities on their own, test, make mistakes, and try to understand the cause of the errors.

Finally, students were challenged to make the robot follow the course from the first class, but this time there were obstacles on the track that the robot had to detect and navigate around. The method of navigating the obstacles was defined by the group, encouraging their creativity.

3.2.3. Class 3—Maze and Creation of Variables

A summary of the content covered in the previous class was presented, highlighting the programming blocks that worked best for different groups. Subsequently, the students resumed the previous activity, which involved navigating around objects placed on the track. This provided an opportunity to implement improvements and explore new ways of robot navigation. Next, the maze scenario (Figure 6 on the left) was introduced as the final project of the training: an overview of this protocol is presented in Table 2. This activity was originally intended to be solved using the Python language. However, due to the limited amount of time available with the students, Scratch was used instead. The students were required to build a program that would navigate the maze from a starting position to a red tape strip on the floor, symbolizing the finish line. Since the robot only has a front-facing ultrasonic sensor, it is unable to detect walls on its sides. This posed a major challenge for the students who initially attempted to always turn the robot in the same direction, resulting in infinite loops within the maze. They started to develop the program for a line follower (see Figure 6 on the right).

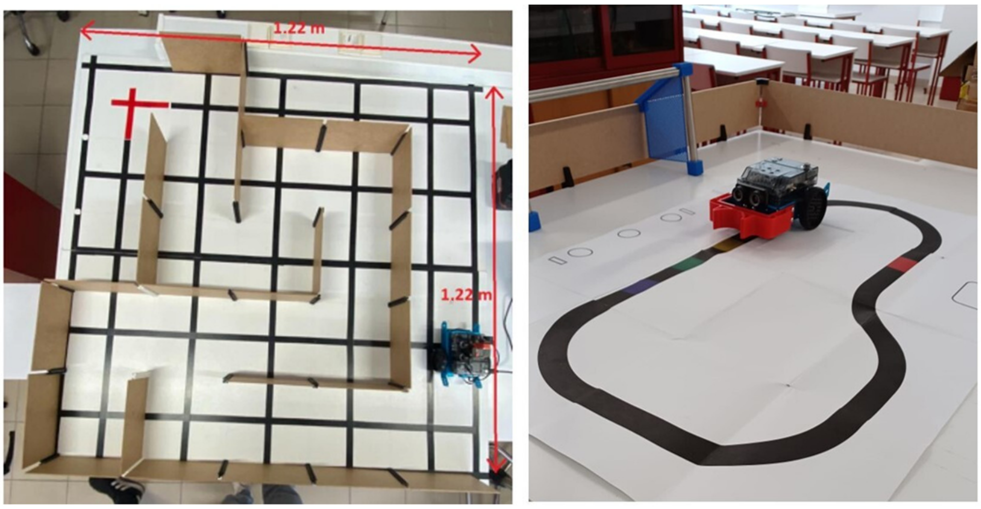

Figure 6. A maze scenario was created for the last activity from class 3 (**left**). The default track is available in the mBot2 kit (**right**).

Eventually, one of the groups suggested creating a variable that would count the number of turns, allowing them to adjust the rotation direction based on the number of turns the robot had previously made.

3.3. Short STEM Courses

Within the scope of short STEM courses developed in the university, the students participated in a mini-robot football tournament. This activity does not require programming skills. Suitable for quick visits of students of all ages to the robotics laboratory, Futebot consists of a simple and fast game where mBots are controlled through the Makeblock application on a mobile phone, acting as players in a mini-soccer game. The objective is to provide a fun and immersive experience that allows visitors to interact directly with the robots, thereby sparking interest in robotics through a competitive activity.

Makeblock provides the option to freely control the robot using the controller shown in Figure 7. Each game had four participants simultaneously, divided into two teams (2×2), with one student per mBot. The matches ended when one of the teams scored 2 goals in the opponent's goal or when the game duration reached 5 min. The implementation of Futebot required the creation of a scenario, which included two goals and a midfield line where the ball would be placed at the beginning of each match.

Figure 8 shows some images taken during the activities. Some 3D-printed bumpers were designed to protect the robots while also giving a more captivating appearance. These bumpers also had the additional function of facilitating ball control, thanks to a customized socket specifically designed for the ball used. Moreover, as additive manufacturing is also a current topic of research and development, besides being of interest to the students, the possibility of an activity where students develop add-on parts made by 3D printing for their robotic solutions is also a procedure to extend the motivation and the knowledge acquired. Although additive manufacturing is not the objective, it can be used to increase the motivation of the students while basic concepts of this technology are provided.

Figure 7. The "Drive" mode display of the Makeblock application is shown above. On the left side, there is an analogue stick that allows the control of the robot's movement. The buttons on the right side perform specific movements or emit predefined sounds.

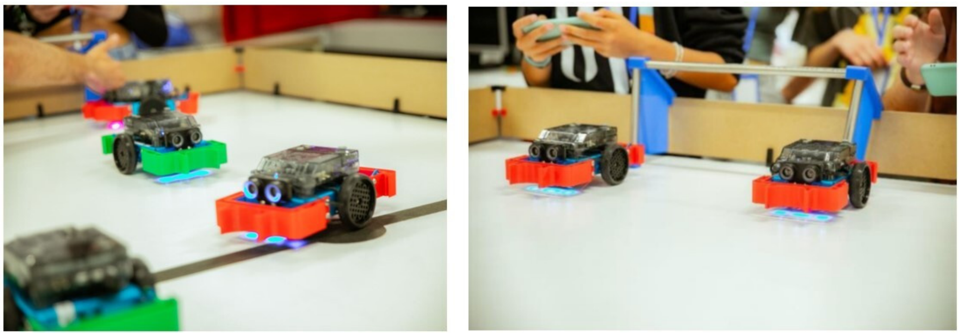

Figure 8. Photographs captured during the football tournament. Some 3D-printed bumpers were designed to protect the robots while also giving a more captivating appearance.

3.4. Multi-Purpose Workbench

All the research described in this paper involved the creation of structures or scenarios to enable robots to complete the proposed challenges, ranging from maze solving to soccer playing.

One of the initial priorities was to allocate a dedicated space within the robotics laboratory for mobile robotics, ensuring a safe environment for robot navigation without the risk of collisions or falls that could result in irreparable consequences.

The designated area for mobile robotics consisted of a table with a working surface measuring 2300 × 1200 mm. To prototype this workstation, the plan included the installation of wooden panels around the table to serve as side barriers, preventing the robots from leaving the designated area. The goal was to establish a robust structure that allowed for the addition or removal of components depending on the specific activities performed.

Furthermore, a new project was initiated, focusing on the development of a workstation specifically tailored for projects and activities involving mobile robotics. This initiative aimed to integrate projects from related subjects such as industrial automation, industrial robotics, and robotic systems.

To implement the maze walls, it was necessary to equip the base of the workstation with a mechanism that facilitated the addition or removal of supports. The solution devised was a snap-fit mechanism incorporating springs; 3D printing was utilized to produce the supports, and the snap-fit mechanisms comprised two distinct components. Nevertheless, it was essential to ensure that the piece accommodating the column remained securely fixed to the table. To accomplish this objective, a support structure was designed to be embedded within the 16 mm-thick MDF board serving as the workstation's base. The piece featured different sections to prevent its removal when the support was pulled. Later, a flat design of the panels that make up the workstation was created, and assembly was carried out to verify compatibility between the parts. The cutouts at the edges facilitate the attachment of the side panels. The side panels have cutouts that align with those on the base. There are also slots spaced 300 mm apart, where pieces will be inserted to support the internal walls of the maze.

The MDF panels were cut using CNC machining. The side panels were glued and screwed onto the base. Careful attention was given to the final finishing of the table. After meticulous preparations, the cut surfaces were finished, and all the parts were glued and screwed. Subsequently, the sanding and levelling of the surface were performed. As for the coating, a pore filler was applied. This crucial step helped to fill in small imperfections and create a smooth surface for subsequent painting applications. Figure 9 displays the completed rendition of the table, featuring the incorporation of several maze walls.

Figure 9. The final iteration of the table, including the integration of maze walls.

4. Results

During the training, the performance and progress of the entire group were evaluated to implement improvements in the teaching approach.

The construction of assessment instruments is generally complex and requires consideration of goal definition, inventory of available resources, identification of individuals, selection of the representative sample, development of questionnaire outline, conducting a pre-test, finalizing the questionnaire, data collection, response coding, data analysis, and processing, and preparation of the final report [22,31].

Since each class had only three sessions, each lasting two hours, the assessment instruments could not be overly complex.

The research plan for this study is qualitative, as some data were collected through direct observation and recorded using descriptive notes and/or observation documents. To complement the information gathered through observation, the analysis of student responses to two anonymous questionnaires was employed, as well as questions posed directly during the classes.

An initial questionnaire served as a diagnostic tool for assessing the student's progress in the field of robotics throughout their academic path. This diagnostic, conducted in the form of a questionnaire, enabled us to adapt and better plan the subsequent classes. It allowed for a more precise allocation of time for each activity and consideration of the student's autonomy levels in completing them. The final questionnaire was distributed after the last class, enabling students to self-assess and evaluate the teaching methodology used throughout the training. The primary purpose of this questionnaire, along with observations and dialogues with the students, was to identify potential areas for improvement in the approach followed in the protocols developed in this work.

The process of questioning students, referred to as an interview, is more flexible than other techniques, enabling participants to discuss specific subjects with a certain openness, thereby allowing for more in-depth coverage. The first questionnaire (Questionnaire 1) consisted of a set of eight questions requiring a binary response (Yes or No):

No.	Question
1	Have you had any contact with a robot before?
2	Have you programmed a robot before?
3	Have you used a block-based programming language before?
4	Have you used any type of sensor during your academic journey?
5	Are you familiar with the programming language "Scratch"?
6	Have you heard of Boolean variables?
7	Are you familiar with the concept of an algorithm?
8	Have you worked with encoder motors before?

Out of a total of 56 participants, Figure 10 presents the percentage of affirmative and negative answers, thus providing a better understanding of the prior knowledge acquired by the students and allowing for minor adjustments to be made to the curriculum taught in subsequent classes.

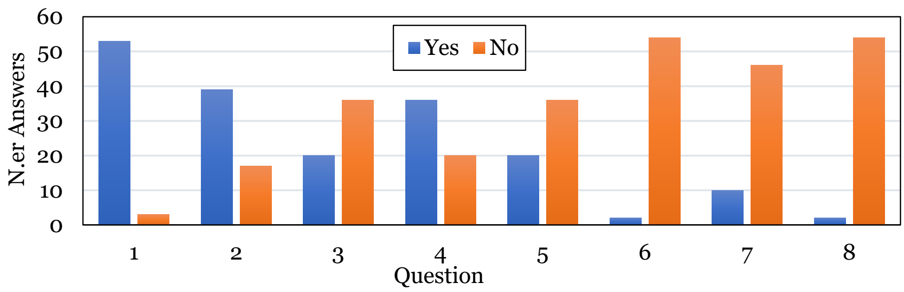

Figure 10. Answers from the first questionnaire applied to participants in the training focused on teaching robotics through mobile robotics kits (mBot2).

The last questionnaire (Questionnaire 2) focused on assessing the interests and opinions of the participants regarding the classes:

No.	Question
1	Did you find the course challenging?
2	Did you enjoy working in a team?
3	Did you feel more comfortable with programming as the classes progressed?
4	Was it challenging to program with mBlock?
5	Were the proposed problems important for the development of reasoning skills?
6	Did the instructor support students in achieving success in solving activities?
7	Do you feel that this training will be useful for your future?
8	Would you like to continue programming mBots?

The number of participants fluctuated throughout the classes, with one of the groups not attending the last session, resulting in a total of 44 responses to the final questionnaire.

Figure 11 presents the percentage of affirmative and negative answers to each of the questions, with the discussion of results conducted in the following section.

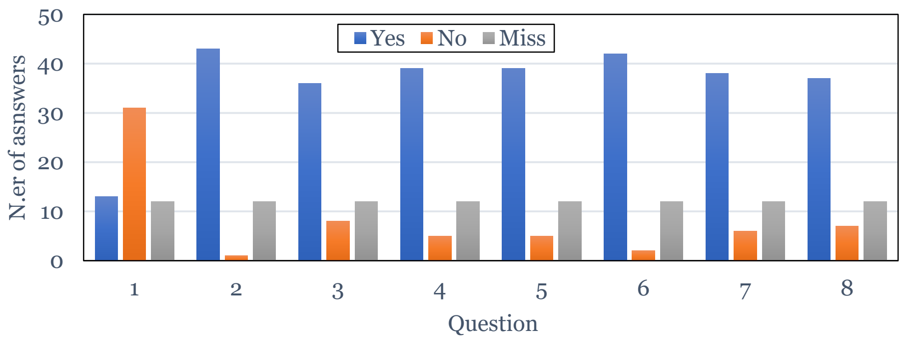

Figure 11. Answers from the last questionnaire applied to participants in the training focused on teaching robotics through mobile robotics kits (mBot 2).

5. Discussion

Regarding the intervention with the students from the vocational school, based on the results of the first questionnaire (Figure 10), it was found that most students have had previous contact with robots. About 70% claimed to have programmed a robot, and 64.3% had worked with sensors during their academic journey. However, only a few students reported working with encoder motors (less than 4%). The aforementioned finding suggests that the participants' familiarity with advanced robotics components was very superficial. While many of them possessed knowledge regarding the sensors present in the mBot2 kit and had a general understanding of their functioning, comprehending the concept of an encoder motor proved to be more challenging due to its concealed nature. The limited depth of their knowledge was consistently observed throughout the classes.

A minority of students had previous experience with Scratch or any block-based programming language. Through dialogue with the students, it became evident that they had not explored control structures such as loops and conditions (if statements) in their programming endeavours, which demonstrates a lack of familiarization with the concept of the algorithm.

Although the first questionnaire consisted of direct yes/no questions, the complementary dialogue with the students helped to understand that they had a very basic level of programming knowledge.

The focus of the second questionnaire was not to assess the student's knowledge, as the training duration did not allow for continuous progress evaluation. The objective was solely to understand how the methodology used inspired the students to learn robotics and whether there is an interest in exploring this field in the future. The majority of students (66.1%) expressed a desire to continue programming mBots 2 if given the opportunity. Even the teachers showed interest in acquiring the mobile robotics kits. Regarding difficulty, 23.2% admitted to experiencing difficulty in completing the activities, but 64.3% acknowledged that the difficulties decreased as the classes progressed.

Regarding the methodology, 69.6% believed that the proposed problems were important for the development of reasoning skills, and 75% agreed that the support provided during the classes contributed to the successful completion of the activities, stimulating participation and allowing students time to reflect on the problems. This compilation of results concludes with a large majority of participants (67.9%) recognizing that the course was useful for their future, whether it be in academia or their professional careers. One of the main objectives of the training, besides promoting critical thinking and interest in robotics, was to foster teamwork. This objective was relatively successful, with 76.8% of students stating that they enjoyed solving challenges in groups.

From the short STEM courses, although there are no measured results for students' performance in this activity, there are noteworthy observations. Namely, the high level of student engagement during the activity, visible enjoyment in their facial expressions, and, finally, a common difficulty that emerged: the challenge of adopting the robot's perspective when attempting to move backward on the terrain.

6. Conclusions

Many students have experience as users in interacting with new technologies. Technology has the potential to foster discovery and learning both inside and outside the classroom. There are a set of computational thinking skills that can benefit from this approach to technology, and the purpose of a robotics discipline is to empower students to solve problems.

In the classes taught, the aim was not to follow traditional teaching methods, where the teacher is seen as the holder of knowledge and the student as the receiver. Instead of using expository teaching and one-way communication, there was constant sharing and clarification of doubts, making the pace of the class dynamic and preventing students from getting distracted from the proposed tasks [22].

During the classes, the importance of developing various problems of increasing complexity was highlighted. In solving the final problem, students applied the knowledge acquired during the training and expanded their horizons, promoting autonomy.

The choice of educational robotics was made to reinforce skills, promote interdisciplinary learning, foster computational thinking, reasoning, creativity, and persistence, and engage students in developing other necessary competencies in the development of automotive engineering.

In the first two classes, there was some difficulty in implementing loops in general, as well as in combining the line sensor with the ultrasonic sensor, as students had to prioritize the information received by one of the sensors so that the robot could make decisions without conflicting with other instructions. In the last class, perhaps due to the complexity of the final challenge, competitiveness diminished, and a cooperative environment emerged among some groups with the common goal of finding a strategy capable of navigating the entire maze regardless of its configuration.

Although some groups were able to navigate the maze with the robot, the defined strategy was not infallible. Some configurations were impossible to solve, so students would need more time and classes to achieve a solution capable of resolving any scenario. Based on the previous observation, the importance of time in this type of activity is emphasized. Students need many classes to gradually acquire knowledge, as programming is a discipline perceived as difficult not only by secondary school students but also by university students. The innovative use of 3D-printing technology empowers educators to craft fascinating add-ons that capture students' imagination and encourage active involvement in the learning process. These engaging additions spark curiosity and enthusiasm, effectively igniting a passion for exploration and discovery. Moreover, the inclusion of a well-equipped workspace further augments the educational experience. It allows for the construction of dynamic scenarios, enabling educators to design challenging activities that test students' critical thinking skills and creative problem-solving abilities. The synergy between 3D-printing technology and the adaptable workspace creates a favourable environment for fostering a deeper understanding of complex concepts and promotes active student participation.

Overall, the described methodology represents a forward-thinking approach to education, connecting the potential of cutting-edge technology to create an immersive and enriching learning experience. By leveraging 3D printing and a versatile workspace, educators can effectively teach a new generation of engaged and knowledgeable learners.

Still, it is fundamental to include diverse data collection methods to comprehensively assess the impact of educational robotics. The constraints posed by the second questionnaire, resembling the format of the first one with only affirmative/negative responses, limit the ability to conclude students' advancements in critical thinking or reasoning abilities. To address this, conducting real-world testing of all protocols within a classroom setting becomes imperative. Employing weighted criteria, such as resolution time, teamwork quantification, and evaluation of algorithmic skills and program construction, is essential for the effective monitoring and measurement of students' progress.

It would be interesting to take football to another level by developing an autonomous tournament. This approach would not only enhance the complexity of the game but also provide an opportunity for students to engage in various disciplines and further their understanding of subjects such as mathematics, physics, electronics, CAD (computer-aided design), programming, collaborative robots, control systems, artificial vision, and artificial intelligence.

Author Contributions: Conceptualization, P.D.G.; methodology, P.D.G., R.A. and M.L.A.; software, R.A.; validation, P.D.G.; formal analysis, R.A., P.D.G. and M.L.A.; investigation, R.A.; resources, P.D.G.; data curation, R.A.; writing—original draft preparation, R.A.; writing—review and editing, R.A., P.D.G. and M.L.A.; supervision, P.D.G. and M.L.A.; funding acquisition, P.D.G. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to express their gratitude to Fundação para a Ciência e Tecnologia (FCT) and C-MAST (Centre for Mechanical and Aerospace Science and Technologies) for their support in the form of funding, under the project UIDB/00151/2020.

Institutional Review Board Statement: Not applicable.

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

Data Availability Statement: The data are not publicly available due to confidentiality reasons concerning the individuals involved in the study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Diago, P.D.; González-Calero, J.A.; Yáñez, D.F. Exploring the development of mental rotation and computational skills in elementary students through educational robotics. *Int. J. Child-Comput. Interact.* **2022**, *32*, 100388. [CrossRef]
2. IFR. IFR Presents World Robotics Report 2022, Automation. 2022. Available online: https://www.automation.com/en-us/articles/october-2022/ifr-presents-world-robotics-report-2022 (accessed on 11 July 2023).
3. Mead, R.A.; Thomas, S.L.; Weinberg, J.B. From grade school to grad school: An integrated STEM pipeline model through robotics. In *Robots in K-12 Education: A New Technology for Learning*; IGI Global: Hershey, PA, USA, 2012; pp. 302–325.
4. Atmatzidou, S.; Demetriadis, S. Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robot. Auton. Syst.* **2016**, *75*, 661–670. [CrossRef]
5. Çakır, R.; Korkmaz, Ö.; İdil, Ö.; Erdoğmuş, F.U. The effect of robotic coding education on preschoolers' problem solving and creative thinking skills. *Think. Ski. Creat.* **2021**, *40*, 100812. [CrossRef]
6. Xia, L.; Zhong, B. A systematic review on teaching and learning robotics content knowledge in K-12. *Comput. Educ.* **2018**, *127*, 267–282. [CrossRef]
7. Chambers, J.M.; Carbonaro, M.; Murray, H. Developing a conceptual understanding of mechanical advantage through the use of Lego robotic technology. *Australas. J. Educ. Technol.* **2008**, *24*, 387–401. [CrossRef]
8. Leonard, J.; Buss, A.; Gamboa, R.; Mitchell, M.; Fashola, O.S.; Hubert, T.; Almughyirah, S. Using robotics and game design to enhance children's self-efficacy, STEM attitudes, and computational thinking skills. *J. Sci. Educ. Technol.* **2016**, *25*, 860–876. [CrossRef]
9. Tedre, M.; Denning, P.J. The long quest for computational thinking. In Proceedings of the 16th Koli Calling International Conference on Computing Education Research, Koli, Finland, 24–27 November 2016; pp. 120–129.
10. Papert, S. *Mindstorms: Children, Learning and Powerful Ideas*; Amazon: Bellevue, WA, USA, 1980.
11. Denning, P.J. Remaining trouble spots with computational thinking. *Commun. ACM* **2017**, *60*, 33–39. [CrossRef]
12. Hamilton, M.; Clarke-Midura, J.; Shumway, J.F.; Lee, V.R. An emerging technology report on computational toys in early childhood. *Technol. Knowl. Learn.* **2020**, *25*, 213–224. [CrossRef]
13. Weintrop, D.; Beheshti, E.; Horn, M.; Orton, K.; Jona, K.; Trouille, L.; Wilensky, U. Defining computational thinking for mathematics and science classrooms. *J. Sci. Educ. Technol.* **2016**, *25*, 127–147. [CrossRef]
14. Shute, V.J.; Sun, C.; Asbell-Clarke, J. Demystifying computational thinking. *Educ. Res. Rev.* **2017**, *22*, 142–158. [CrossRef]
15. National Research Council. *Report of a Workshop on the Scope and Nature of Computational Thinking*; National Academies Press: Washington, DC, USA, 2010.
16. Computer Science Teachers Association (CSTA) Standards Task Force. CSTA K-12 Computer Science Standards. 2011. Available online: http://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/Docs/Standards/CSTA_K-12_CSS.pdf (accessed on 11 July 2023).
17. Robins, A.; Rountree, J.; Rountree, N. Learning and teaching programming: A review and discussion. *Comput. Sci. Educ.* **2003**, *13*, 137–172. [CrossRef]

18. Coll, C.; Solé, I. Os professores e a conceção construtivista. In *O Construtivismo na Sala de Aula: Novas Perspetivas Para a Ação Pedagógica*; Coll, C., Martín, E., Mauri, T., Miras, M., Onrubia, J., Solé, I., Zabala, A., Eds.; Edições Asa: Porto, Portugal, 2001; pp. 8–27.
19. Pattis, R.E. A philosophy and example of CS-1 programming projects. *ACM SIGCSE Bull.* **1990**, *22*, 34–39. [CrossRef]
20. Janzen, D.S.; Saiedian, H. Test-driven learning: Intrinsic integration of testing into the CS/SE curriculum. *ACM SIGCSE Bull.* **2006**, *38*, 254–258. [CrossRef]
21. Roumani, H. Practice what you preach: Full separation of concerns in CS1/CS2. *SIGCSE Bull.* **2006**, *38*, 58–64. [CrossRef]
22. Rodrigues, A.M.G. Aplicação da Robótica na Resolução de Problemas: Reflexões Para a Aprendizagem Inicial de Programação no Ensino Básico. Ph.D. Thesis, Universidade de Lisboa, Lisboa, Portugal, 2020.
23. Calderon, J.M.; Rojas, E.R.; Rodriguez, S.; Baez, H.R.; Lopez, J.A. A Robot soccer team as a strategy to develop educational initiatives. In Proceedings of the Latin American and Caribbean Conference for Engineering and Technology, Panama City, Panama, 23–25 July 2012.
24. Benitti, F.B.V. Exploring the educational potential of robotics in schools: A systematic review. *Comput. Educ.* **2012**, *58*, 978–988. [CrossRef]
25. Schina, D.; Esteve-Gonzalez, V.; Usart, M. Teachers' perceptions of bee-bot robotic toy and their ability to integrate it in their teaching. In *Robotics in Education: Methodologies and Technologies*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 121–132.
26. Kazakoff, E.R.; Sullivan, A.; Bers, M.U. The Effect of a Classroom-Based Intensive Robotics and Programming Workshop on Sequencing Ability in Early Childhood. *Early Child. Educ. J.* **2013**, *41*, 245–255. [CrossRef]
27. Voštinár, P. Using mBot robots for the motivation of studying computer science. In Proceedings of the 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 28 September–2 October 2020; pp. 653–657.
28. Pisarov, J.; Mester, G. Programming the mbot robot in school. In Proceedings of the International Conference and Workshop Mechatronics in Practice and Education, MechEdu, Subotica, Serbia, 12 December 2019; pp. 45–48.
29. Makeblock. *mBot2 Getting Started Activities V1.1*; MakeBlock Education: Irwindale, CA, USA, 2022.
30. Makeblock. mBlock Block-Based Editor- about mBlock5- What Is mBlock 5 Block-Based Editor. 2022. Available online: https://education.makeblock.com/help/about-mblock-5-block-based-editor/ (accessed on 11 July 2023).
31. Vilelas, J. *Investigação—O Processo de Construção do Conhecimento*, 2nd ed.; Edições Sílabo: Lisboa, Portugal, 2017.