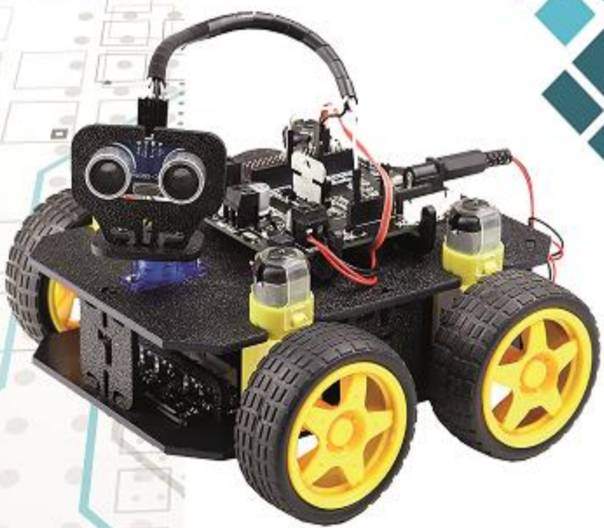
First Published in 2022
ISBN : 978-605-70651-1-7

"Bandrol Uygulamasına İlişkin Usul ve Esaslar Hakkında Yönetmeliğin 5 inci maddesinin ikinci fıkrası çerçevesinde bandrol taşıması zorunlu değildir."

Co-funded by the
Erasmus+ Programme
of the European Union

MOBILE ROBOTICS

TÜRKİYE ULUSAL AJANSI
TURKISH NATIONAL AGENCY



MOBILE ROBOTICS

Editors
- José Alberto Lencastre
- Paulina Spânu
- Gülden İlin
- Panos Milios
- Marco Bento

A low-cost mobile robot for STEM subjects

Rolando BARRADAS, Salviano SOARES & António VALENTE
University of Trás-os-Montes and Alto Douro, Portugal

José Alberto LENCASTRE
University of Minho, Portugal

1 Introduction

STEM areas (Science, Technology, Engineering and Math) are continuously growing but the number of technical workers do not accompany that growth. As the 21st century brings new challenges, students should be prepared for an increasingly complex life and work environments that will privilege proficiency in Learning and Innovation Skills that include Creativity and Innovation, Critical Thinking and Problem Solving, Communication and Collaboration [1]. Also, the need to continuously explore new pedagogical practices in teaching and learning creates an opportunity to build new contents by balancing a stable and tested curriculum with new tools that stimulate creativity, allowing students to better understand the world they live in. This article describes the development of an educational robotics kit, aimed at children and teens from 8 to 18 years old, meant to work as an interdisciplinary teaching tool that can be applied directly in a curriculum, promoting students' technical competences and allowing them to develop new skills like Computational Thinking and Problem Solving, driven by the motivation created by a Robotics Competition.

2 Background

2.1 Computational thinking

Computational thinking is defined as a mental activity carried when formulat- ing a problem to admit a computational solution that can be carried out by a human or a machine [2]. Nowadays, it influences not only science and engineer- ing methods but also other fields of study like medicine, economics, finance and even journalism, requiring that everyone working in those areas have to learn how to think computationally. Computational thinking involves solving prob- lems, designing systems, and understanding human behavior, using concepts fundamental to computer science [3].

2.2 Problem solving

One of the 21st century most wanted skills is Problem Solving and, in what concerns students, it's the most relevant learning activity they can engage in because the knowledge constructed while solving problems is better comprehended and retained [4]. According to P21, The Partnership for 21st Century Learning [5], the ability to solve different kinds of non-familiar problems in both conventional and innovative ways and to identify and ask significant questions that clarify various points of view and lead to better solutions are essential to prepare students for the future.

2.3 Micromouse Portuguese Contest

The Micromouse Portuguese Contest [6] is an international competition, held in Portugal since 2011. The main challenge is to have a full autonomous micro controlled robot vehicle, explore an unknown maze and find out the optimum route for the shortest travel time from start to end [7]. Competition is one of the key factors for motivation and getting physical results contributes to the formation of students independence, developing their leadership skills and promoting a positive educational process [8]. Also, robot competitions encourage students to apply their knowledge to real-world problems and motivates them to learn new concepts for themselves [9], making them a good vehicle to the development of both Computational Thinking and Problem Solving capabilities.

2.4 Visual programming languages

Visual programming languages (VPL) provide a way for children to begin programming, reducing the level of abstraction by using graphical program elements rather than text.

Scratch Scratch is a VPL created by the Lifelong Kindergarten group at the MIT Media Lab. It was originally thought as an approach to programming to be used by people who never thought they would ever write a single line of code. It was meant to be easy for everyone, of all ages, backgrounds, and interests, to program their own interactive stories, games, animations, and simulations, and share their creations [10].

Scratch was made with a simple grammar, based on graphical programming blocks that are put together to create programs. To make it even easier, the blocks have connectors that suggest how they can connect to each other, allowing only to create code that makes sense [11].

Fig. 1. *Scratch* key ideas

mBlock A few years ago, mBlock made its appearance in the VPL world, as a graphical programming environment based on Scratch 2.0 Open Source Code, thus maintaining all its features, and adding some others that make it possible to program Arduino projects within the same interface [12].

Fig. 2. Robots/Arduino category of blocks in mBlock

3 Method

To develop our product, we decided to follow an Instructional System Design model [13], which we will refer to as ADDIE, the acronym of its five phases: Analysis, Design, Development, Implementation and Evaluation (see Figure 3). As we haven't yet implemented the prototype, only three phases (Analysis, De- sign and Evaluation) will be described in this article. The Evaluation phase is fundamental and should be a part of the process from the beginning because it supplies information that feeds all the cyclic process of design and development and is very useful when as a part of the spiral of analysis, design, evaluation, etc., by contributing to the continuous improvement of the prototype [14].

Fig. 3. The ADDIE Model

3.1 Analysis

Analysis - Study the environment in order to understand it and describe the goals and objectives required to correct performance deficiencies (performance gap) that will improve the organization's performance [13].

The analysis phase is the foundation of a learning or training process [13] and allowed us to study the target audience of our educational product. By knowing their previous experience, education level, age, computer experience, among others, it is possible to anticipate learning difficulties and create boundaries to the complexity of the product we are to develop [15]. Through documentary analysis and classroom observations we tried to create a profile for the target audience of our product.

As we are targeting both Primary and Secondary education students, the first thing we have to consider is the age gap between the younger and the older students. In our analysis, the average age of our students is 11.3 years old. Also, the concepts and academic level differences are an important fact to consider. A relevant information is the fact that some of the students in our study already have some basic knowledge of robotics and programming in Scratch, due to the fact that Introductory Programming classes are a part of their curriculum. Also, to consider are the latest government recommendations stating that every child from Primary to Upper Secondary education should have Programming and Robotics classes.

3.2 Design

Design - Define the learning objectives - what the learners need to do to learn the new performance (activities), and what will motivate them to learn and perform. This becomes your blueprint [13].

The results obtained led us to idealize Kid Grígora (Fig. 4), an educational robotic platform that can be used as an interdisciplinary teaching tool to be integrated in the curriculum. Besides that, primary objective, Kid Grígora has dimensions that allow children to use it in the Micromouse Robotics Compe- tition. With this alpha version of the prototype some heuristic evaluation is needed.

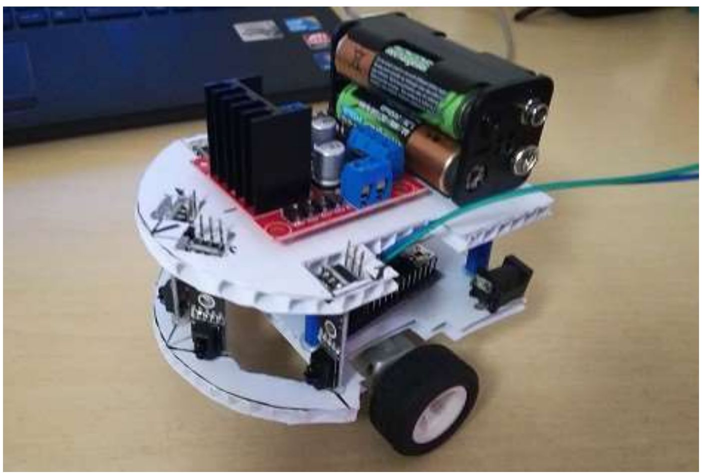


Fig. 4. Alpha version of Kid Grígora

3.2.1. Heuristic evaluation of the alpha version

The alpha version of the prototype was tested in a heuristic evaluation by experts, with the objective of appraising both usability and potential design problems, and also to gather suggestions from the experts on how to solve the problems they found, before performing usability tests with representative users.

To test the prototype, we chose double experts [15] experienced not only in usability but also with specific expertise in the kind of interface under evaluation as they potentially find about 1.5x more problems than simple usability special- ists. We used three experts, with ages ranging from 40 to 48 years old, with a degree in areas related to computing, electronics and robotics, with an average teaching experience of 15 years and with 9 years of average business experience in developing software and electronics.

The evaluations were carried out between 9th and 12th October 2017, with an average duration of 90 minutes, and started with a simple explanation of the expected use of the robot by end users, in particular on its use as an educational tool, but also on its possible use in a robotics contest. Then, the evaluators were given the robot's parts, a set of tools and assembly instructions and were asked to assemble the robot.

During the tests, each expert was asked to answer a questionnaire of heuristic evaluation and to report possible problems found using a 0 to 4 Nielsen's severity rating scale [15] in which 0 means "I don't agree that this is a usability problem at all" and 4 means a "Usability catastrophe: imperative to fix this before product can be released".

Talking about the strong points of the heuristic evaluation, all the experts mentioned that the robot was very easy to build, mostly because of its small number of components. They also referred the physical similarity to professional built Micromouse robots. Two experts referred that because it has almost no soldering parts, it should be suitable for all target users, eventually with the help of an adult. All experts referred the use of standard components as a strong point as it is easy to assemble and also due to the low price they usually have, making it an educational tool, potentially for everyone. The weakest points in the heuristic evaluation (ratings 3 and 4) are summarized in Table 1.

Table 1. Related severe and catastrophic errors, according to Nielsen's heuristics

Nielsen's heuristics		
Interface (IN)		Degree
IN1	Visibility of system status	4
IN3	User control and freedom	3
IN4	Consistency and standards	4
IN7	Flexibility and efficiency of use	3
IN8	Aesthetic and minimalist design	3
IN9	Help users recognize, diagnose, and,recover from errors	4
IN10	Help and documentation	3

Regarding IN1, two experts mentioned that the robot had no information on the status. Related with IN3, all of the experts stated that the robot needed to have an ON-OFF switch and one of them referred that as older students may require a little more control over the robot, it should be useful to have it equipped with encoders and gyros so that more elaborated algorithms could be implemented. One of the experts, referring to IN4, mentioned that the Traction system would not work at very high speeds as the motor connected directly to wheel brings speed but almost no torque. The difficulty on perceiving the robots' movements, when working with youngest students, was mentioned by one of the experts as being potentially a problem, related to IN7. All experts

3.3 Development

Development - Elaborate and build the products called for in the blueprint (the finished product is often called courseware or learning activities).[13].

3.3.1. Building the beta version

Although only Major and Catastrophic problems (ratings 3 and 4) were de- scribed, before building the beta version, all reported problems and suggestions of the experts were solved and implemented, as summarized below.

Table 2. Solutions for usability problems found by experts

Heuristic	Problem found	Solution
IN1	No information on the status	Add a Status LED
IN3	The robot needs to have an ON-OFF switch	Change the electrical connections and add a power switch
IN3	Equip the robot with encoders and gyroscope	Create a *Semi-Pro* version of the robot that uses motors with encoders, Gyro-scope and accelerometer
IN4	Traction system would not work	Use motors with reduction (Figure 6)
IN7	The type of battery used could be lighter	Change the type of battery from 4xAA 1.5v to a 9V battery
IN7	The use of IR Sensors might be too difficult to program and understand by young students	Use simpler Ultrasonic sensor types in Kid Grígora *Rookie*, but keep the IRsensors in Kid Grígora *Semi-Pro*

IN7	It may be difficulty to perceive the robot's movements, when working with youngest students	We created a Pen add-on to the Kid Grígora *Rookie* for the students to visualize the trajectories of the robot, by making it *write its way* as it moves (Figure 7)
IN8	The battery positioned on the top of the robot would create a very high gravity center.	As we changed the type of battery, we were able to position differently, low-ering the height and center of gravity
IN9	No error messages	Use a LED to display Error codes
IN10	Need for more detailed help on the electrical connections assembly	Created new electrical schematics, with different wire colors

The results of the heuristics analysis led to the idealization of two models of our robotic platform, mainly due to the age difference and academic levels between our target audience.

Kid Grígora Rookie is the simpler of the two models. Aimed to students with ages from 8 to 15 years old, this robot allows younger students to make their first steps in robotics and programming. The price and the ease of build have been taken in consideration, to make it affordable and easy to assemble.

Kid Grígora Semi-Pro is the most complex, having more powerful specifications, allowing students, from 15 to 18 years old, to apply knowledge from other areas like Mathematics or Physics. With a more powerful processor, motors with encoders, a three Axis Gyroscope + Accelerometer and four Infra-Red distance sensors, this model allows a much more accurate control of movements.

The first beta version had a different Traction System and unified platforms, allowing us to use, both the simple DC Motors in Rookie and the DC Motors with encoders needed to Semi-Pro.

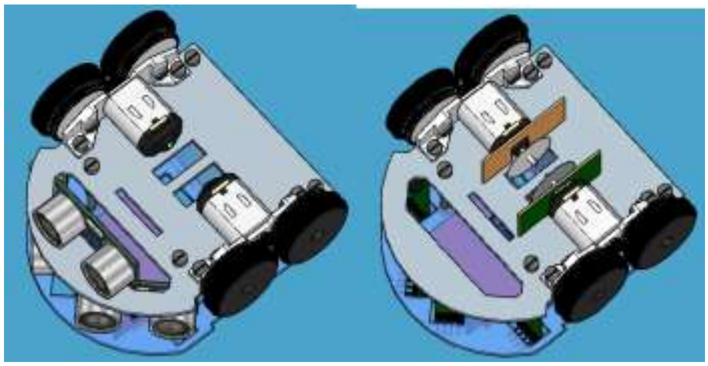


Fig. 5. Proposed design of the Traction System in Kid Grígora Rookie and Semi-Pro in Beta version

However, as one of our objectives was to build a low cost and easy to mount robot, this approach led to a solution that used a large number of 3D printed parts, making it a lot more expensive than what we anticipated. The solution was to change the Traction system once again and to use a single pair of wheels, a caster ball and two geared DC motor with reduction and a pair of pulley wheels, as seen in Figure 6. With this solution, we were able to reduce substantially the cost of the solution.

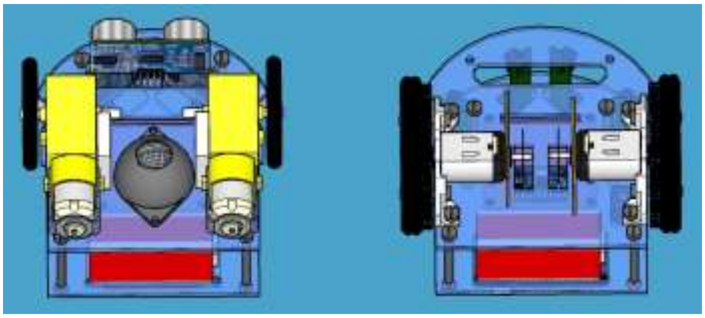


Fig. 6. Final design of the Traction System in the Beta version of Kid Grígora *Rookie* and *Semi-Pro*

In order to solve the possible difficulty to perceive the robot's movements, when working with youngest students, we designed a Pen Add-on to Kid Grígora Rookie, using a standard Servo Motor and a custom-made pen holder. By placing it on the top of the robot, it will be possible for the students to visualize the trajectories of the robot, by making it write its way as it moves.

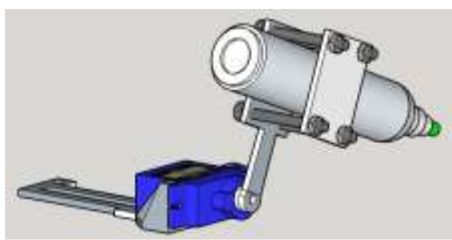


Fig. 7. Pen Add-On to Kid Grígora *Rookie*

3.3.2. Usability tests with representative users

The usability tests with representative users were carried out between the 18th and 22nd December 2017. As, according to Nielsen [16], "after the fifth user, you are wasting your time by observing the same findings repeatedly but not learning much new", we chose five representative users in different age ranges and programming and robotics knowledge to evaluate our prototype.

Although we developed and built both models of Kid Grígora, as in our analysis, our target medium range was 11.3 years old, in this article we will focus on the tests performed with Kid Grígora Rookie.

Fig. 8. Representative user performing
Usability test

The tests were carried by 5 students, aged from 11 to 17, 2 boys and 3 girls, and had an average duration of 127 minutes, with a 15-minute pause for the users to rest and then regain their focus on the tasks. As for background on robotics, only two users were already engaged in robotics activities at school. The other three had never been in close contact with robotics. Starting with a simple explanation on the basics of the assembly and best practices to do it, the users were given the robot's parts, a set of tools and the assembly instructions and were asked to assemble the robot. In all tests, we used the think-aloud protocol, letting users verbalize their thoughts as they move through the interface [15], and audio-recording to gather data.

At the end of the tests, the users were asked to fill a SUS [17] satisfaction questionnaire, whose average satisfaction results were given a meaning by using the adjective scale by Bangor, Staff, Kortum e Miller [18]. The obtained results are summarized in Table 3.

Table 3. Summary of usability tests by representative users

	Student 1	Student 2	Student 3	Student 4	Student 5	Avg
Sex:	F	M	M	F	F	
Age	16	13	11	11	17	13.6
	Student 1	Student 2	Student 3	Student 4	Student 5	Avg
Previous robotics:	No	No	No	Yes	Yes	
Length (min)	131	134	147	120	103	127
Rating	92.5	85	90	95	100	92.5
Meaning	Best Imagi n-able	Excellen t	Excelle nt	Best Imagi n-able	Best Imagi n-able	Best Imag-inable

The mean result of the five tests was 92.5 points, Best Imaginable, meaning that there were almost no usability problems detected with the prototype. The analysis of the results show that the representative users were unanimous giving the Strongly agree score to the question "I think that I would like to use this robotics kit frequently" and the to Strongly disagree to the question "I found the robotics kit unnecessarily complex" which shows the good acceptance of this robotics kit. The analysis of the think-aloud showed that most of the difficulties lied in the part of the wiring, particularly in those users who have never had contact with robotics. This led us to think that perhaps an introductory session on the concepts of electronics and wiring will be necessary before end users start building the kit.

4 Kid Grígora hardware components

To build both models of Kid Grígora, we chose the following standard compo- nents:

4.1 Kid Grígora Rookie

Arduino Nano The Arduino Nano microcontroller is a small board based on the ATmega328, clocked at 16MHz and operating at 5v. It's 32Kb of Flash memory will be sufficient for the simple programs that are meant to be built in this version of Kid Grígora.

L298N Motor Controller Because of the limited power output of the micro controller outputs, they cannot be used to drive the motors directly. Motor controllers act as an intermediate device between the micro controller and the motors, allowing to set both speed and direction. L298N is a low-cost solution that enables the control of two DC motors with voltages between 5 and 35V DC.

Geared DC Motors Because regular DC motors run too fast and cannot be used to drive the robot, we decided to use geared DC motors with a gear assembly attached to the motor, reducing the speed of the motor while increasing its torque. This approach to the design allowed to attach the wheels directly to the motor assembly thus reducing the total cost of the product.

Ultrasonic Sensors We decided to use three HC-SR04 ultrasonic sensors (front, diagonal left and diagonal right) because of it's low cost and simplic- ity. Ultrasonic sensors can measure the distance to an object by using sound waves. This particular sensor provides 2cm to 400cm of non-contact measure- ment with an accuracy up to 3mm, allowing the robot to detect the walls of the maze and navigate accordingly.

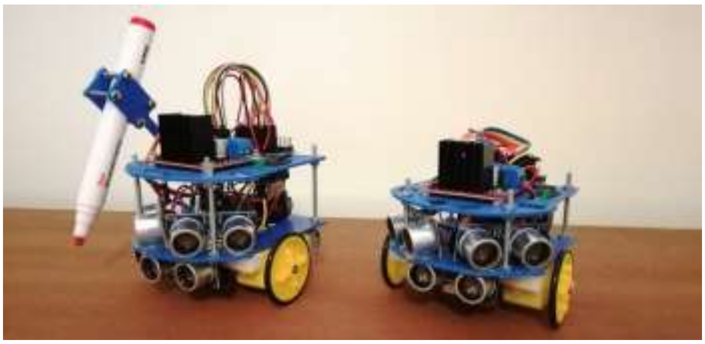


Fig. 9. Assembled Kid Grígora *Rookie*

Aimed to be a low budget robotics kit, one of our main concerns was to keep it as cheap as possible, without sacrificing its main purposes. The following table resumes all the components used in Kid Grígora. Prices are based on eBay and local stores.

Table 4. Costs of Kid Grígora *Rookie*, in EUR

Kid Grígora *Rookie*			
Parts	Qty	Unit Price	Total
Welded Arduino Nano Clone + USB cable	1	3.45	2.45
Acrylic Chassis Kit	1	1.23	1.23
M3 10mm Screws	4	0.08	0.32
M3 20mm Screws	2	0.05	0.10
M3 40mm Screws	4	0.03	0.12
M3 Hex Screw Nut	24	0.02	0.48
DC Motor with reduction	2	0.84	1.68

TT DC Geared Motor Bracket Holder (with screws)	2	1.06	2.12
Pulley Wheels 36mm	2	0.21	0.42
Rubber tyres	2	0.10	0.20
Ultrasonic sensor HC-SR04	3	0.88	2.64
L298N DC Motor Driver Module Dual H Bridge	1	1.54	1.54
9volt Battery Holder Clip	1	0.07	0.07
5mm Green LED	1	0.12	0.12
Mini ON-OFF Switch	1	0.17	0.17
Jumper cables pack	1	1,04	1,04
Mini Breadboard	1	0.97	0.97
Metal Caster Ball	1	1.19	1.19
Velcro tape (20cm)	1	1.10	1.10
Double sided tape (20 cm)	1	0.10	0.10

Total: 18.58

Table 5. Costs of the Pen Add-On to Kid Grígora *Rookie*, in EUR

Pen add-on to *Rookie*			
SG90 9G Micro Servo Motor	1	1.10	1.10
M3 20mm Screws	4	0.05	0.20
M3 Hex Screw Nut	4	0.02	0.08
Pen holder (3D printed)	1	8.58	8.86

4.2 Kid Grígora Semi-Pro

Still on its final phase of development, this version of Kid Grígora will used the following components:

Adafruit Feather 0 Powered by ATSAMD21G18 ARM Cortex M0+ processor, clocked at 48 MHz and at 3.3V logic, this microcontroller will be able to handle all the information retrieved from the

Gyroscope, Accelerometer, IR Sensors and Encoders. The 256K of FLASH memory and 32K of RAM allows students to create more complex maze solving algorithms that require lots of memory space.

Infrared Distance Sensors Because of the need for small size, we decided to create custom distance sensors using four infrared light emitting diodes in three directions (front, diagonal left and diagonal right) that detect the intensity of the reflected light, used to determine wall information and correct the robot navigation.

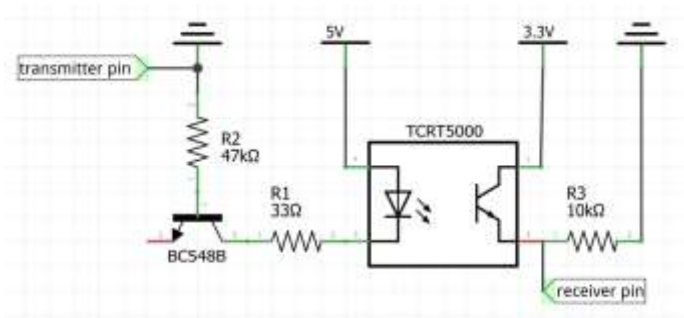


Fig. 10. Schematic of the custom-made IR sensors

DC Motors with Encoders Motor encoders are used for a precise speed control. Built in with a 334 line disc and an AB-phase Encoder, the motor's encoder dimensions are ideal to fit in the platforms we built, and provide both position and direction of rotation.
3 Axis Gyroscope+Accelerometer To give our Kid Grígora Semi-Pro a way of knowing exactly where it is in the maze, we chose the MPU-6050 3 Axis Gyro- scope+Accelerometer. By monitoring the

angular velocity and the acceleration speed this chip provides the information needed for the robot to know it's exact position in the maze.

The change of some of the components to improve performance, lead to some important changes in the costs of the robot. All the prices are based on eBay and local stores.

Table 6. Costs of Kid Grígora *Semi-Pro*, in EUR

Kid Grigora *Semi-Pro*			
Parts	Qty	Unit Price	Total
Adafruit Feather 0	1	17.60	17.60
Acrylic Chassis Kit	1	1.23	1.23
M3 10mm Screws	4	0.08	0.32
M3 20mm Screws	2	0.05	0.10
M3 40mm Screws	4	0.03	0.12
M3 Hex Screw Nut	24	0.02	0.48
DC Motor with 334-line AB-phase Encoding	2	2.34	4.68
Wheels 30mmx5mm (3D printed)	4		
Rubber tyres	8	0.10	0.80
16 Tooth Wheel for motor (3D printed)	2	2	
Wheels and motor holder (3D printed)	2		
IR distance Sensors (Custom built)	4		
L298N DC Motor Driver Module Dual H Bridge	1	1.54	1.54
MPU-6050 6DOF 3 Axis Gyroscope+Accelerometer	1	1.01	1.01
9volt Battery Holder Clip	1	0.07	0.07
Mini ON-OFF Switch	1	0.17	0.17

Jumper cables pack	1.04	1.04	
Mini Breadboard	1	0.97	0.97
Velcro tape (20cm)	1	1.10	1.10
Double sided tape (20 cm)	1	0.10	0.10

5 Software interfaces

5.1 mBlock

Currently under development, the mBlock extensions (see Figure 11), will be one of the core components of this project.

Simple KidG The Simple KidG extension will have a basic set of blocks to move the robot, like Move Forward, Turn Right and Turn Left, and will be used, typically by students from 8 to 12 years old.

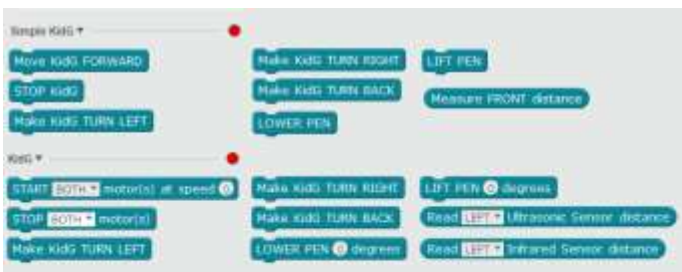


Fig. 11. KidG proposed mBlock extensions

KidG The KidG extension was planned to provide students, from 12 to 15 years old, with a greater level of control over the robot, providing, for example, different left and right motor speeds and different sensor distance measuring, allowing different kinds of interactions.

5.2 Android apps

In order to reach our younger audience, we've planned the development of two type of Android Apps, typically to be used by students from 8 to 12 years old.

KidG Remote Control The simplest App of the package works as a remote control, allowing young students to explore all the movement possibilities of the robot and the Pen Add-on.

KidG Step by Step After exploring the robot's possible movements, it's time to make it execute sequential tasks. Using a graphical interface, students can create simple algorithms, send them to the robot and watch it execute them.

5.3 Virtual maze

Virtual Maze is a representation of a real-world maze, developed in Scratch and planned to allow students from 12 to 15 years old, a first contact with the Micromouse Contest. Together with the mBlock extensions and the developed Firmware, it allows the simulation of Maze Solving algorithms and the transpo- sition of the movements on screen to the real robot, in a real-life maze.

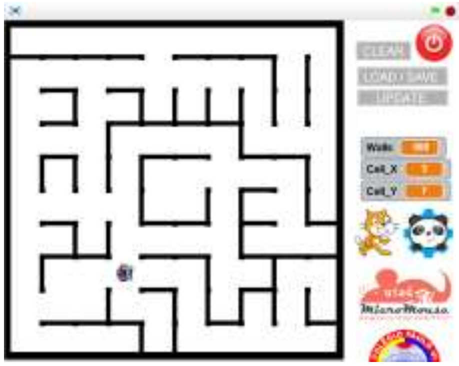


Fig. 12. VirtualMaze

5.4 Firmware development

The custom Firmware will allow interaction with both the mBlock extensions and the Android Apps. Through a Bluetooth connection to the computer or Android device, the firmware will receive commands and make the robot execute them.

5.5 C++ and the Arduino IDE

Also implemented as Firmware, aimed to be used with Kid Grígora Semi-Pro, and typically used by older students, from 15 to 18 years old, the planned Ar- duino libraries will allow them to program the Kid Grígora Semi-Pro with C++ while providing high levels of abstraction to interact with the hardware. Planned functions include movement, like MoveForward, TurnLeft, TurnRight, TurnBack, and sensing, like ReadDisplacement, isWallLeft and isWallFront.

6 Educational uses

6.1 Primary education

For this range of ages, 8 to 12 years old, our main objective will be creating activities aimed to develop Computational Thinking. Because Computational Thinking is more about conceptualizing than programming [3], we plan on using Kid Grígora Rookie with the Pen Add-on attached, together with the Android Apps and the Simple KidG mBlock extension. Using real-life problems and scenarios and interacting with virtual environments, created in mBlock, children can take their first steps in robotics and programming, by creating simple algo- rithms and watching them come to life.

6.2 Lower Secondary education

Using the motivation created by a Robotics Competition and the knowledge previously created, simulating in the Virtual Maze allows students, from 12 to 15 years old, to further develop their Problem-Solving skills by placing them on the control of a robot that needs to find the centre of a maze, in the fastest possible way. By creating Maze Solving algorithms, students can, at first, see their robot in action on screen. Later, students can assemble their own robot and, using the KidG mBlock extension connected to Kid Grígora Rookie by Bluetooth, they can debug their algorithms in both Virtual Maze and real life. Using the same mBlock extension, they can develop a program to work autonomously and participate in the Micromouse Portuguese Contest.

6.3 Upper Secondary education

Also aiming the participation in a Robotics Competition, Kid Grígora Semi- Pro allows a deeper level of control but also requires a lot more knowledge to work with. Using the custom firmware created in the form of Arduino libraries and with all the sensing hardware, students from 15 to 18 years old, are no longer limited to making their robot sense their way in the track and react. Now they can create real autonomous navigation systems for the robot to find all possible ways to the center of the maze, return to the starting point, backtrack the optimal route [19] and run to the center the fastest it can.

7 Conclusion

We strongly believe that Problem solving and Computational Thinking are two of the most needed skills for 21st century students. Following an Instructional System Design cite clarkd we created a prototype of an educational robotics kit, aimed at children and teens aged from 8 to 18, to be used in scholar activities with the objective of developing those skills. In the Analysis phase, we gathered enough information to idealize the alpha version of the product, later tested by experts. The test results revealed some interface usability issues, corrected in the development phase which was also used to implement some suggestions of the experts and create the beta version, tested by representative users. In the satisfaction test, the prototype of Kid Grígora Rookie obtained 92.5 points, Best Imaginable, that show a very stable and satisfactory robotic platform, with almost no usability problems detected, which serves as an incentive to the next development phases.

References

[1] Partnership for 21st Century Learning. (2016). Framework for 21st Cen- tury Learning. The Partnership for 21st Century Learning, 37(4), 589. https://doi.org/10.2307/3588214

[2] Wing, J.M. (2017). Computational thinking's influence on research and education for all. Italian Journal of Educational Technology, 25(2), 7-14. doi: 10.17471/2499- 4324/922

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

[4] Jonassen, D. (2011). Learning to solve problems. A Handbook for Designing Problem-Solving Learning Environments. New York: Routledge.

[5] Partnership for 21st Century Learning. (2015). P21 Partnership for 21st Century Learning. Partnership for 21st Century Learning, 9. Retrieved from http://www.p21.org/documents/P21_Framework_Definitions.pdf

[6] Valente, A. (n.d.). Micromouse PortugueseContest. http://www.micromouse.utad.pt.

[7] Silva, S.; Soares, Salviano; Valente, António; Barradas, Rolando; Bartolomeu, Paulo. 2015. "Enhancing STEM courses through a robotic innovative project", Pre- sented at the 3rd International Conference, In Proceedings of the 3rd International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM '15,Porto, Portugal. doi: 10.1145/2808580.2808668

[8] Bazylev, D., Margun, A., Zimenko, K., Kremlev, A., & Rukujzha, E. (2014). Par- ticipation in Robotics Competition as Motivation for Learning. Procedia Social and Behavioral Sciences, 152, 835-840. https://doi.org/10.1016/j.sbspro.2014.09.330

[9] D. Pack, R. Avanzato, D. Ahlgren, and I. Verner, "Fire-fighting mobile robotics and interdisciplinary design-comparative

perspectives", IEEE Transactions on education, vol. 47, pp. 369-376, 2004.

[10] Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Bren- nan, K., Kafai, Y. (2009). Scratch: Programming for All. Communications of the ACM, 52, 60-67. https://doi.org/10.1145/1592761.1592779

[11] Resnick, M. (2012). Point of View: Reviving Papert's Dream. Educational Tech- nology, 52(4), 42-46.

[12] Makeblock Co., Ltd (2017). mBlock - Program Robots/Arduino Without writing a single line of code.http://http://www.mblock.cc/.

[13] Clark, D. (2000). Instructional System Design. Retrieved on 2017-09-11 from http://www.nwlink.com/ donclark/hrd/sat1.html

[14] Lencastre, J. A. (2012). Educação on-line: análise e estratégia para criação de um protótipo. In João Batista Bottentuit Junior & Clara Pereira Coutinho (org.), Educação on-line: Conceitos, metodologias, ferramentas e aplicações (pp. 127-136). Maranhão: Editora CRV.

[15] Nielsen, J. (1993). Usability Engineering. San Francisco: Morgan Kaufmann.

[16] Nielsen, J. (2000). Why You Only Need to Test with 5 Users. Retrieved on 2017-09- 12 from http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/

[17] Brooke, J. (1986). SUS-A quick and dirty usability scale. In P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), Usability evaluation in indus- try. London: Taylor & Francis. Retrieved from hell.meiert.org/core/pdf/sus.pdf

[18] Bangor, A., Staff, T., Kortum, P., & Miller, J. (2009). Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. Journal of Usability Studies, 4(3), 114-123.

[19] Silva, S., Duarte, D., Barradas, R., Soares, S., Valente, A., & Reis, M. (2017). Arduino recursibe backtracking implementation for a robotic contest. In M. Silva et al. (Eds.), Proceedings of Clawar

2017: 20th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines (pp. 171-181), Porto, Portugal.