CURRICULAR INTEGRATION OF COMPUTATIONAL THINKING, PROGRAMMING AND ROBOTICS IN BASIC EDUCATION: A PROPOSAL FOR TEACHER TRAINING

A.F. Monteiro[1], M. Miranda-Pinto[2], A. Osório[1], C.L. Araújo[3]

[1]Research Centre on Education (CIEd), University of Minho (PORTUGAL)
[2]ESEViseu - School of Education; Research Centre on Education (CIEd), University of Minho (PORTUGAL)
[3]Research Centre on Education (CIEd), University of Minho; ISCTE-University Institute of Lisbon, CIS (PORTUGAL)

Abstract

Today's children are growing up in a complex technological reality, which, in turn, is pushing for the integration of digital technologies in educational contexts. Hence, it becomes necessary to explore how more recent educational approaches to technology can be inclusively integrated into education. Among these innovative approaches are the integration of computational thinking, programming and robotics both in preschool and basic education. Considering the referential of key competences for the current 21th century [1], early training in these areas will contribute to the development of transversal competences [2]. Thus, it is crucial to provide education professionals with the skills and resources for an adequate development of programming and robotics activities in educational contexts. In this paper we present a training action developed within the scope of the project "KML II - Laboratory of technologies and learning of programming and robotics for preschool and primary school".

This training action aims to work with early education professionals towards the development of activities with children, using the tools foreseen in this research project. From the work done during the course, trainees should develop an activity plan using programming and robotics technologies, to be implemented in their respective educational contexts. Within this action, trainees are expected to: reflect on the concept of computational thinking and its development in preschool and basic education [3], [4]; collaboratively develop competencies associated with digital literacy and the use of robots and programming languages developed for children [5]; know programming and robotics resources that can be used in preschool and basic education; learn programming basics through applications such as ScratchJr, or others that can support learning development; plan activities according to the curricular contents of the respective level of education, using programming and robotics. Implemented through b-learning, this initiative will also enable educators and teachers to explore and develop distance learning and collaboration skills as well as the use of various support tools and work time management in synchronous and asynchronous sessions.

This training is one of the first actions through which KML II project plans to study how to integrate programming and robotics in preschool and basic education, transversally to all areas of knowledge. Within this project, case studies will be carried out at a Portuguese national wide level. This work has two main objectives: a) to propose a training framework for curricular units of technology, in the courses for teacher training in higher education and for in-service training; b) to design a profile of childhood educator and primary school teacher as mediator in the integration of programming and robotics learning in their educational contexts.

Keywords: Computational thinking, programming, robotics, early education, basic education.

1 INTRODUCTION

This paper presents the design of an in-service teacher training developed under project 'KML II - Laboratory of technologies and learning of programming and robotics for preschool and primary school'. The main objective of this training, delivered between July and October 2019, is to develop in-service teachers' knowledge and competences to integrate computational thinking, programming and robotics on pre-elementary and elementary school curriculum. Furthermore, it aims to develop the expertise required for educators and teachers to participate in the research to be conducted under KML II, namely case-studies on kindergartens and elementary schools.

This work will contribute to KML II main objectives, notably: survey of training needs in technology, programming and robotics of elementary school teachers and early childhood educators; build a training framework that leads to the integration of programming and robotics in undergraduate and postgraduate education, as well as continuing education; design a competency profile of educators and teachers as mediators in the use of programming and robotics in an educational context; contribute to a framework for the integration of programming and robotics in the curriculum. Furthermore, it constitutes the first stage for the development of a bilingual (Portuguese and English) MOOC for undergraduate, postgraduate and in-service teacher training.

The KML II project encompasses the creation of laboratories in higher education institutions, for the learning of programming and robotics by students of initial, postgraduate and continuing education. It further includes a mobile lab for schools participating in the project. Thus, it intends to distribute at a national level the resources provided by the project. Partners involved will also be consultants in the development of the framework.

2 CONTEXT

There is now an extensive specialized literature on the relevance of introducing children to computational thinking and coding from early ages [2], [3], [6]–[8]. Throughout the last decade, several theoretical perspectives have provided grounds for this claim. Probably, one of the most popular arguments is that early educational approaches to STEM disciplines are central in the effort to prepare a future workforce that is digitally skilled [9]. A quick glance into initiatives promoted by Eramus+ (ec.europa.eu/programmes/erasmus-plus/) or OECD's work on education (www.oecd.org/education/), to cite some emblematic examples, shows the weight of the economic argument. An interesting exercise would be to analyze, in official political documents, how frequently terms such as “21st century skills” or “21st century learning” are linked to worries about future jobs and economy. Regardless, in Europe, the political effort to push forward a “new skills agenda” seems manifest and very much connected to the promotion of a digital economy [9].

Stemming from a distinct theoretical tradition is the idea of “coding as literacy” (CAL) [10][2]. Coined by Professor Marina Umaschi Bers and members of her DevTech Research Group (http://sites.tufts.edu/devtech/) at Tufts University, this broader and more inclusive perspective understands coding as a new language. While STEM enthusiasts direct particular attention to the learning of computation concepts (e.g. abstraction, sequencing, representation, problem solving), the CAL approach “is grounded on the central principle that learning to program involves learning how to use a new language (a symbolic system of representation) for communicative and expressive functions” [2]. In this sense, the main concerns are with participation and inclusion. Here, computational concepts and skills fit in a more holistic viewpoint.

It is worth recalling that advocates of this later approach rely on Papert’s work, namely his theory of constructionism. From this perspective, technology fashions appropriate, multifaceted settings to promote learning by making. It’s about constructing knowledge rather than transmitting knowledge. Or, in Papert’s own words, the development of ‘powerful ideas’ [11]. Also rooted in this constructionist tradition is Resnick’s work, which sets the integration of ICT in education in a slightly different manner. His lifelong kindergarten framework states all educational environments should promote children’s learning and development as kindergartens do: “As kindergartners playfully create stories, castles, and paintings with one another, they develop and refine their abilities to think creatively and work collaboratively, precisely the abilities most needed to achieve success and satisfaction in the 21st century” [12]. Resnick is currently leading the Scratch project, following this philosophy (https://scratch.mit.edu/).

Behind the recognition of these competences as a social, rather than a merely technical asset, are also Wing’s seminal works on computational thinking [13], [14]. While it remains a debatable concept, it is somewhat consensual that, following Wing’s proposal, computational thinking involves resorting to computer science concepts to solve problems, understand and design systems, while also developing critical thinking, collaboration and creativity skills [15]. In a technological society, where communication, learning, participation, among other social dynamics, are increasingly mediated by computers, understanding ‘their languages’ seems crucial to comprehend and act on how technologies change or setup the social contexts we live in. This is the concern beneath what seems an exponential attention given to computer science by educational policies and practices [16].

Furthermore, national and international educational policies have increasingly valued the inclusion and participation of all, evident both in the implementation of inclusive education for all students (Decree Law No. 54/2008) and the required Student Profile when leaving School [17]. Thus, the integration of ICT in pedagogical activities has enabled the development of children's skills and diversified pedagogical practices, modifying the curriculum and the way students learn (e. g. access to materials in electronic formats; use of videos, sounds or images, use of text reading software), adjusting them to the characteristics and needs of children and their current and future social contexts.

These tools also allow educators to make learning environments flexible (where one type or format of teaching is not privileged over others), designed for all people to develop critical thinking and a willingness to challenge themselves [18], [19] and are particularly important in overcoming the barriers imposed on their learning (e. g. the use of assistive technologies by students with special educational needs) and in using compensatory strategies that would otherwise prevent certain type of learning (e. g. sensory). For example, ICTs have been found to be effective for the learning process of children with learning disabilities. Currently, the use of models such as Universal Design Learning (UDL) [20] has allowed for the optimization of the teaching-learning process by mobilizing multiple scientific evidences on how learning should unfold so that positive knowledge, skills and attitudes are achieved.

2.1 Teacher training

Ramos and Espadeiro [15] specifically reflect on the importance of integrating these skills in the initial training of teachers while empowering future teachers to make the best use of ICT as a learning means and taking into account their pedagogical objectives. In this context, effective educators and teachers must be creative and resourceful in creating flexible learning environments that address student variability, using a range of high-tech and low-tech solutions [21]. However, the use of ICTs in teaching and learning is still limited [22]. Education professionals highlight, among other reasons, the lack of preparation or training as a barrier to using these resources in the classroom [23], [24].

In the European context, great attention has been paid to the evaluation of digital competences in teaching-learning processes, resulting in the creation of a Digital Competence Framework for Educators. This framework proposes three key areas that must be considered in the context of digital competences: (1) Educators' professional competences; (2) Educators' pedagogic competences; and (3) Learners' competences [25], [26]. Also, in Portugal, the Dynamic Framework of Digital Competence References is an example of the recognition of the need for improving the skills of education professionals in this area.

However, increases in funding for the development of European projects promoting robotic and pedagogical use of ICT by young children and educational professionals are visible (e. g. EU Kids Online [27], [28]. Also, national, and local entities (e.g., municipalities, associations) have increased funding for the purchase of ICT resources in schools [29], [30].

3 METHODOLOGY

The training plan designed within KML II comprises three modules: computational thinking, robotics and programming. For this action, a virtual learning environment (VLE) was designed and built (see figure 1). Hosted by project partner Universidade Aberta (https://vle.uab.pt/colaboracao/), this is organized in 5 topics: presentation, modules 1 to 3 and final project. Each topic contains (see figure 2): session guides, where objectives, planned actions and resources indicated for each module session are described; module forum, dedicated to interactions between trainees and/or trainer to clarify doubts and make comments regarding the theme and activities specific of each module; module resources, with a repository of texts, videos, presentations, among others; activities to be performed in the module, and necessary resources for its accomplishment (see figure 3).

Figure 1. Print screen VLE header

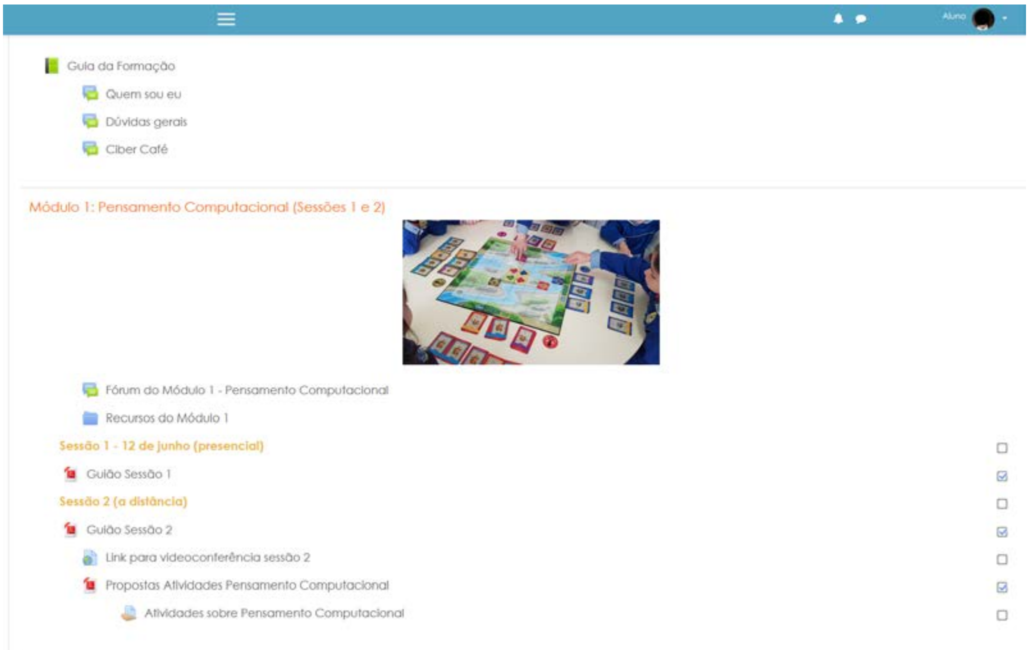

Figure 2. Print screen VLE module example

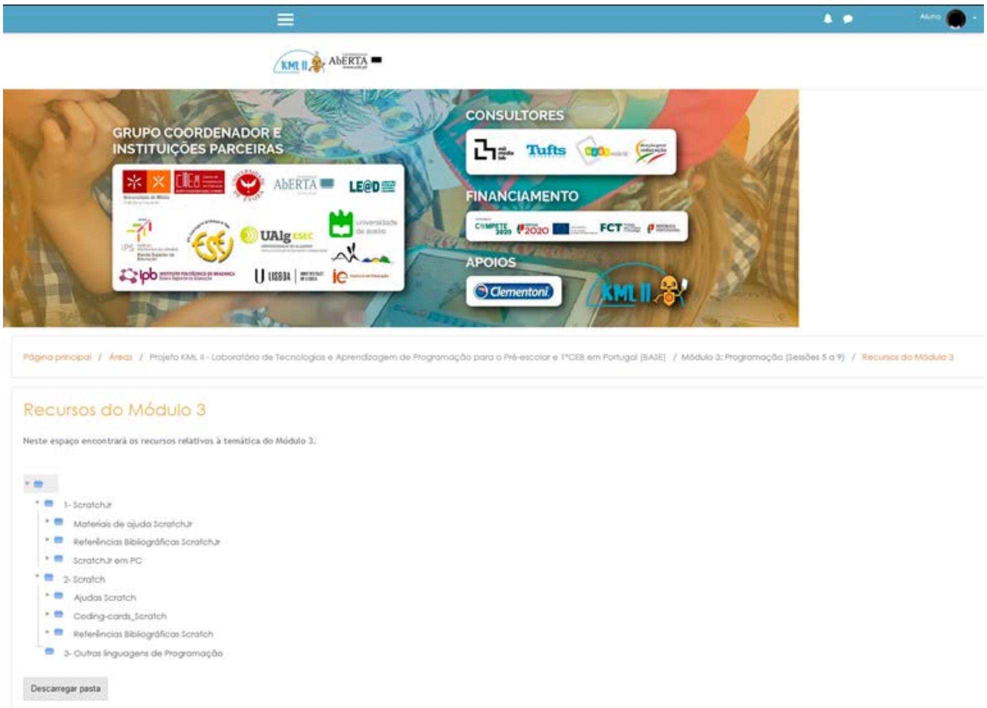

Figure 3. Print screen VLE module resources example

The training has a total workload of 50 hours, organized in classroom activities (25 hours) and autonomous activities (25 hours). The 25 face-to-face hours, in addition to face-to-face meetings at training sites, include synchronous virtual meetings (videoconferences via the Zoom platform). The 25 hours of autonomous work are carried out at a distance and intended for trainees to study reference materials available in the VLE and build the planning of computational thinking, programming and robotics activities, to be implemented in an educational context.

Activities are delivered throughout 11 sessions. For each of the sessions, trainees are provided with a session guide, bibliographic resources and activity proposals. A training guide (see figure 4) is also designed, explaining: how the VLE is organized, objectives, methodology, schedule, assessment, certification, bibliography and information about the KML II project.

Figure 4. Print screen VLE training guide

At the end of the training, participants are expected to be able to plan activities according to the curriculum of the respective level of education, using programming and robotics. Within this action, trainees are expected to: reflect on the concept of computational thinking and its development in preschool and basic education [3], [4]; collaboratively develop competencies associated with digital literacy and the use of robots and programming languages developed for children [5]; know programming and robotics resources that can be used in preschool and basic education; learn programming basics through applications such as ScratchJr, or others that can support learning development; plan activities according to the curricular contents of the respective level of education, using programming and robotics. Implemented through b-learning, this initiative will also enable educators and teachers to explore and develop distance learning and collaboration skills as well as the use of various support tools and work time management in synchronous and asynchronous sessions.

This action adopts a continuous assessment procedure, according to the following criteria: a) commitment and participation in the action; b) evaluation of the work produced, individually or in groups; c) evaluation of the participants' final reports (individual critical reflection).

4 EXPECTED RESULTS

As mentioned above, the work presented in this paper is part of KML II, thus contributing to its main goals, as described. More specifically, designing and implementing the training presented here is an initial stage for two of the project's main activities: i) a multiple case study of pre-school and elementary school classrooms that integrate computational thinking, programming and robotics in the curriculum; ii) the development of a bilingual (Portuguese and English) MOOC for undergraduate, postgraduate and in-service teacher training. A selected group of trainees will, during a school year, implement in educational contexts the activities planned throughout the training. These will constitute the case studies. Results from this study will inform the development of a framework for the integration of these approaches and resources in the curriculum.

Regarding the MOOC, an evaluation of the training by trainees and trainers involved in its implementation will assess the content, learning activities, virtual learning environment, methodologies and resources. Furthermore, participants' opinions regarding the development of a MOOC for this purpose will be collected. Our paper on this topic [31] presents some aspects of this evaluation, particularly the development and validation of a questionnaire designed for this purpose. Overall, this analysis aims to understand if trainees consider having acquired the knowledge and competences necessary to integrate computational thinking, programming and robotics in kindergarten and elementary school classrooms.

5 FINAL REMARKS

This paper presents a training program designed to develop in-service teachers' and early childhood educators' awareness and competences for the integration of programming and robotics in preschool and elementary school. This work is part of the 'KML II - Laboratory of technologies and learning of programming and robotics for preschool' project, whose main objective is to develop a framework for the integration of programming and robotics in the curriculum as well as undergraduate, postgraduate and in-service teacher training. In order to properly achieve this goal, the role of early childhood educators and teachers is paramount. Their participation is crucial for the development of guidelines effectively capable of guiding pedagogical practice. Within this context, the training presented here, and delivered between July and October 2019 to 120 teachers and childhood educators, sought to empower trainees to develop activities that integrate programming and robotics into the curriculum. Implementing these activities in educational contexts will, between 2019 and 2020, constitute the case studies that will, in turn, inform the proposed frameworks. Furthermore, once evaluated [31], this training will inform the development of a bilingual (Portuguese and English) MOOC for undergraduate, postgraduate and in-service teacher training.

ACKNOWLEDGEMENTS

Project KML II is co-financed by FEDER through the COMPETE 2020 - Operational Thematic Program for Competitiveness and Internationalization (POCI) and national funds through FCT - Portuguese Foundation for Science and Technology under project reference number PTDC/CED-EDG/28710/2017.

REFERENCES

[1] European Commission, "Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions - Improving competences for the 21st Century: an Agenda for European Cooperation on Schools," Brussels, 2008.

[2] M. U. Bers, "Coding as another language: a pedagogical approach for teaching computer science in early childhood," *J. Comput. Educ.*, pp. 1–30, Sep. 2019.

[3] M. Resnick, *Lifelong kindergarten: cultivating creativity through projects, passion, peers, and play*. Cambdrige, MA: The MIT Press, 2017.

[4] M. Miranda-Pinto and A. J. Osório, "As TIC em contexto de educação de infância: atividades sobre pensamento computacional e programação," in *XIII CONGRESSO SPCE - Fronteiras, diálogos e transições na educação*, 2016.

[5] M. U. Bers, L. Flannery, E. R. Kazakoff, and A. Sullivan, "Computational thinking and tinkering: Exploration of an early childhood robotics curriculum," *Comput. Educ.*, vol. 72, pp. 145–157, Mar. 2014.

[6] A. Sullivan and M. U. Bers, "Investigating the use of robotics to increase girls' interest in engineering during early elementary school," *Int. J. Technol. Des. Educ.*, Nov. 2018.

[7] L. Zhang and J. Nouri, "A systematic review of learning by computational thinking through Scratch in K-9," *Comput. Educ.*, p. 103607, Jun. 2019.

[8] S. Popat and L. Starkey, "Learning to code or coding to learn? A systematic review," *Comput. Educ.*, vol. 128, pp. 365–376, Jan. 2019.

[9] European Commission, "COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS A NEW SKILLS AGENDA FOR EUROPE Working together to strengthen human capital, employability and competitiveness," Brussels, 2016.

[10] M. U. Bers, *Coding as a playground : programming and computational thinking in the early childhood classroom*. New York: Routledge, 2018.

[11] S. Papert, *Mindstorms : children, computers, and powerful ideas*. Basic Books, 1993.

[12] Mitchel Resnick, "Kindergarten Is the Model for Lifelong Learning," *Edutopia*, 2009. [Online]. Available: https://www.edutopia.org/kindergarten-creativity-collaboration-lifelong-learning. [Accessed: 04-Oct-2019].

[13] J. Wing, *Computational thinking benefits society. Social issues in computing*. New York: Academic Press, 2014.

[14] J. M. Wing, "Computational thinking," *Commun. ACM*, vol. 49, no. 3, p. 33, Mar. 2006.

[15] J. L. Ramos and R. G. Espadeiro, "Os futuros professores e os professores do futuro. Os desafios da introdução ao pensamento computacional na escola, no currículo e na aprendizagem," *Educ. Formação e Tecnol.*, vol. 7, no. 2, pp. 4–25, 2014.

[16] J. Lockwood and A. Mooney, "Computational Thinking in Education: Where does it Fit? A systematic literary review," Mar. 2017.

[17] Ministério da Educação, *Perfil dos Alunos à Saída da Escolaridade Obrigatória*. Lisboa: Direção Geral da Educação, 2017, p. Despacho n.º 6478/2017, de 26 de julho.

[18] CAST, "CAST Timeline: One mission, many innovations, 1984-2010," 2011. [Online]. Available: http://www.cast.org/about/timeline/.

[19] A. Meyer, D. H. Rose, and D. Gordon, *Universal design for learning: Theory and Practice*. Wakefield: : CAST Professional Publishing, 2014.

[20] CAST, "Universal Design for Learning Guidelines version 2.0," 2011. [Online]. Available: http://www.udlcenter.org/aboutudl/udlguidelines.

[21] Seymour Papert, "Papert on Piaget," *Time magazine special issue on "The Centurys Greatest Minds", 29*, 1999.

[22] OECD, “Innovative learning environments,” 2013. [Online]. Available: http://dx.doi.org/10.1787/9789264203488-en.

[23] B. E. C. and T. A. (BECTA), “A review of the research literature on barriers to the uptake of ITC by teachers,” 2014.

[24] F. Costa, H. Peralta, and S. Viseu, *As TIC na Educação em Portugal - Concepções e Práticas*. Porto: Porto Editora, 2007.

[25] M. Lucas and A. Moreira, “Quadro Europeu de Competência Digital para Cidadãos.” 2017.

[26] R. Vuorikari, Y. Punie, S. Gomez, G. Brande, S. Gomez, and G. D. Brande, “DigComp 2.0: The digital competence framework for citizens. Update phase 1: The conceptual reference model.” 2016.

[27] EU Kids Online, “EU Kids Online: findings, methods, recommendations (deliverable D1.6),” EU Kids Online, LSE, London, 2014.

[28] Kultusministerkonferenz, “Kompetenzorientiertes Qualifikationsprofil für die Ausbildung von Erzieherinnen und Erziehern an Fachschulen und Fachakademien.” 2011.

[29] DG Communications Networks, “Survey of schools: ICT in education benchmarking access, use and attitudes to technology in Europe’s schools,” 2013.

[30] H. Shapiro, S. F. Østergaard, and M. Roccaro, “Survey on policies and practices of digital and online learning in Europe,” 2016.

[31] A. F. Monteiro, M. Miranda-Pinto, A. Osório, C. L. Araújo, L. Amante, and A. Quintas-Mendes, “Computational thinking, programming and robotics in basic education: evaluation of an in-service teacher’s training b-learning experience,” in *ICERI2019 - 12th annual International Conference of Education, Research and Innovation*.