Educational Policy Framework to promote Computational Thinking towards STEAM in Public Schools in Boyacá - Colombia

Eduardo A. Fernández, PhD[1], Nancy C. Roa Martín, MsC[1]
[1]Universidad Pedagógica y Tecnológica de Colombia, Colombia, eduardo.avendano@uptc.edu.co, nancy.roa@uptc.edu.co

Abstract*– This paper proposes a framework of Education Policy fostering the development of computational thinking in the path of the STEAM approach for public elementary schools. This exploratory analysis concludes that it is required to perform a deep diagnosis of the teachers' knowledge degree in new pedagogic models and methodologies such as problem-based learning, project-based learning (PBL), game-based learning, etc., and the infrastructure and technology available at the schools (including connectivity), and also, eventual inversion in the public schools of Boyacá to support an adequate teaching and learning process. Besides, it is necessary to train the teachers of elementary schools in digital skills, didactics, and new learning models to develop an integral and transversal curricular design as a path to strengthen the skills of both professors and students in the context of programming, data science, robotics, IoT, arts, and soft skills, that will deliver a higher level of skills developing learning competencies centered on the student during your studies towards university.

Keywords—Computational Thinking, STEAM, language programing, PBL, game-based learning.

I. INTRODUCTION

The vision of new school curricula must include computer science using the IT (Information Technology) capacities of digital society to introduce the concepts and appropriate information that enable problem-solving, effective communication, teamworking, and critical thinking in kids and teenagers. Computational thinking (CT) was introduced by Seymour Papert in 1980 [1], promoting the "*constructionist idea that only a social and effective involvement of students into technical content will make programming and interdisciplinary tool for learning (also) other disciplines*", besides, introduces the issue of "*a possible transfer of skills or meta-skills from CT to other knowledge areas"* [2]. Papert builds their theory from Piaget's constructivism using mental models to understand the world and consider that learning will happen in a constructivist environment, but their real thesis explains that CT is not that "learning to program will in itself have consequences on how children learn and think," but that ability to program a computer can help re-empowering pupils, and bring them powerful ideas about mathematics, physics, and probability, etc. Then, in 2006 Jeannette Wing [3], declares that digital technology plays an important role in daily-life activities, requiring individuals' education, knowledge, and skills to understand IT critically, by enabling the capacity to solve problems by using computers. In [4], Wing declares that "*the teaching of computational thinking, as a basic skill across the school curriculum, will enable K-12 students to learn abstract, algorithmic and logical thinking, and be prepared to solve complex and open-ended problems*". Wing also defines "*CT involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science*". With these two related CT approaches, the question is ¿how do we promote student-centered learning to develop critical thinking and knowledge to solve problems that face everyday challenges during their lives?

It is demonstrated [4], [5] that the integration of computer science in high-school curriculum uses the logic, algorithm design, abstraction, and precision that empowers individuals to create new artifacts allowing them to go from consumers to producers of technology, and also, impact other disciplines by adopting the CS and their tools to solve complex and open-ended problems. Then, introducing the CT skills to pupils will apply those aids in different knowledge areas such as reading, math, physics, science, social studies, arts, etc., in their everyday lives. But most importantly, they will practice the communication of ideas so others can understand their thinking and collaborate in teamwork with them.

Different challenges to the adoption of CT in the curriculum are faced due to lack of resources available, lack of competence in the way to teach, and the integration among professors [5]. The study reveals that professional development, personalized support, time constraints, and alignment with assessment are all factors that influence their readiness for teaching the new curriculum content. Also in [6], the authors identify that the design and development of activities to foster the CT concepts and motivate boys and girls has proven to be critical for the teaching and learning of CT skills. From [7], the authors perform an instrumental case study to understand elementary teachers' planning of STEAM. An important conclusion was that teachers learned heavily on stories to inspire their lessons to stimulate interdisciplinary curricular development; also, teachers used more and better formative assessment prompts to promote learning of science and mathematics content and practices. In the same direction, a small scale-empirical study in Swedish elementary school shows the importance of that children had their first experience of CT and they have a positive attitude towards

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

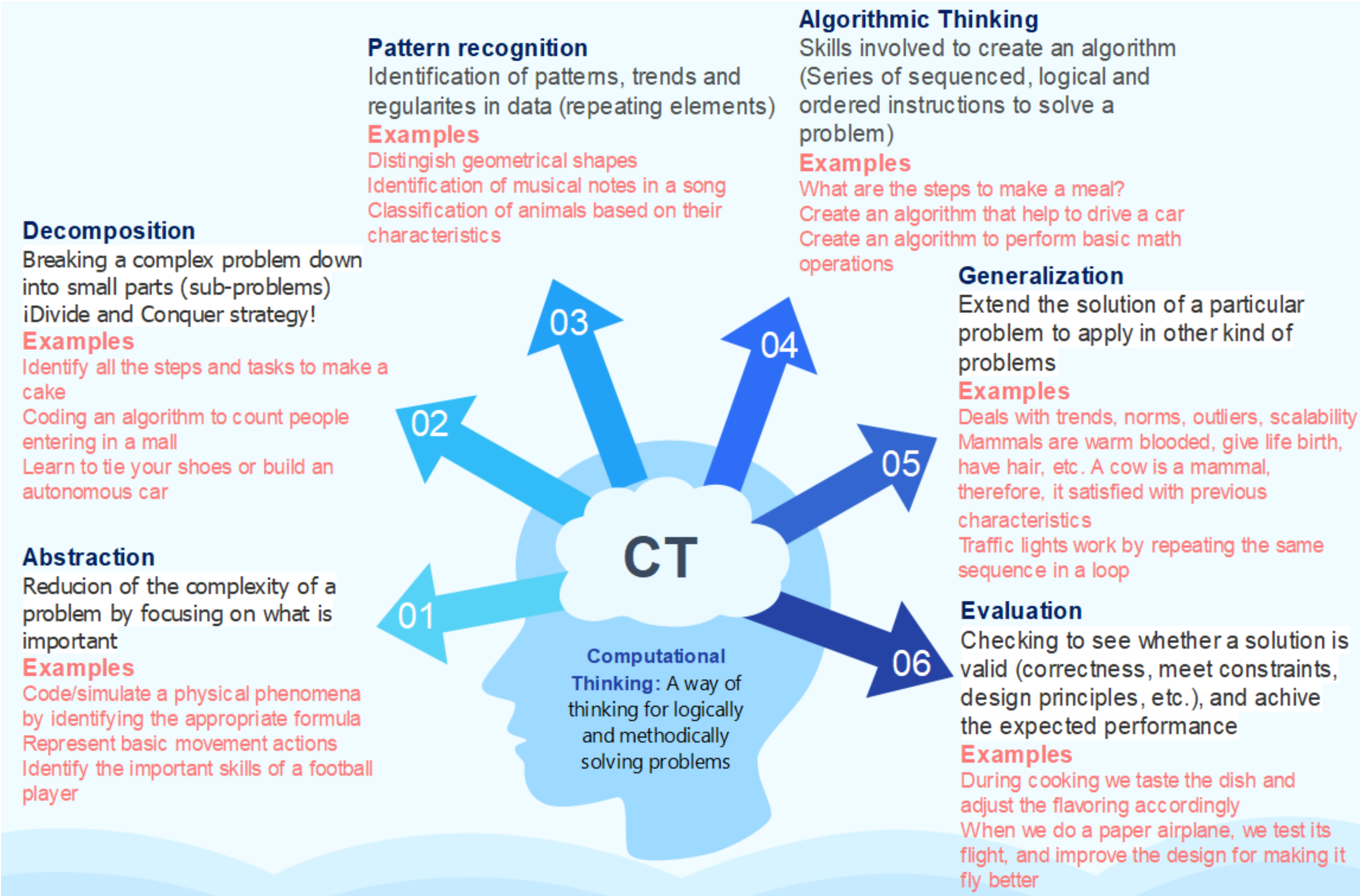

Fig. 1 Main components of CT The authors built from [3] and [5].

programming and problem solving, also, their knowledge is more visible during social interaction with the other kids and with the teachers. The authors of [8] explored the TangibleK Robotics Program as an option for classroom teachers to integrate developmentally appropriate technology education into the early childhood classroom. During the course, children spend their time building a robot, planning its actions (exploring options), using physical wooden blocks (developing mechanical reasoning) or the computer screen to construct programs, and iteratively improving the robot's functionality according to initial goals and subsequent discoveries. In conclusion, when given age-appropriate technologies, curriculum, and pedagogical models, young children can actively engage in learning from computer programming as applied to the field of robotics, being their first step in developing computational thinking. Another important approach in CT is game-based learning [9] which uses "*game construction*" pedagogy as a vehicle that demonstrates the use of Papert's constructionist learning model improves the children's capacity of solving problems by playing, also directly engages foundational literacy and numeracy, and connects to wider STEM-oriented learning outcomes and objectives. Hsu [10] affirms that to help students correctly understand and integrate into the information society, it is not enough to cultivate their creativity and improve their digital literacy; they also need to enhance their CT capability and learn how to utilize new technological tools. The key point is designing CT teaching and combining the proper learning strategies with disciplines to achieve competence. Besides, the training must be focused on learning how to develop CT in the context by using reverse engineering to understand and solve problems from the structured methodology, building on the professor's expertise, and integrating the different knowledge fields in a transversal way.

This paper proposes a framework of Education policy fostering to development of CT looking to connect with STEAM in elementary and middle school public education. The big challenge for deploying the policy is the diagnosis of teacher's competencies, and the cooperative and articulated work of teachers to construct the curriculum in the different learning stages, using the new learning models to achieve teachers the understanding of the context of problem-solving, fostering teamwork, effective communication, and interrelation among disciplines to create opportunities and promote learning by focusing the attention of pupils on challenges, problems, projects, and even games, that stimulates the creativity, imagination, critical thinking, active listening, and disposition, within others for long-life learning.

II. Computational Thinking

A. Principles of CT

Wing [3] proposed a classification of CT in 11 thinking processes. Fig. 1 synthesizes all of them into only 6 categories, which cover abstraction, decomposition, algorithm thinking, generalization, and evaluation; each class is defined conceptually and is included with some basic examples. However, these six computational practices (skills) may include data collection, and analysis; referred to as the gathering of information in a methodological way, for tagging and documenting; and then the data is organized depending if it is quantitative or qualitative, to perform the extraction, an appropriate analysis, and interpretation of the results. Typically, this information is stored in a database that can be represented in a direct communicative format such as a table, chart, figure, infographic, etc. The evaluation and testing include debugging, as a process of the correctness of the code, and when errors are found, it requires "debugging" to fix them. The errors in coding are known as bugs. The testing evaluates the output for a particular input, and the evaluation supports critical thinking and judgment, to meet the specifications of the program.

On the other hand, CT requires some attitudes: to be confident, communicative, and flexible. These features empower the individual to believe in their capacity to solve problems; communicate clearly and effectively with others, and be able to adapt with flexibility to the change and open-ended problems. Besides, the components of CT exploit five approaches to achieve the goal, which is: tinkering, creating, debugging, persevering, and collaborating.

B. Computational Thinking Steps

The CT is an interactive process that follows three steps:

1) Problem Specification: this refers the analyzing the problem from abstraction, decomposition, and pattern recognition. Besides, this step defines the criteria for at least one solution. It requires problem analysis and specification, problem requirements or specifications document, and UML diagrams.
2) Algorithm expression: applies the algorithm design thinking to find an optimal computational solution. It is supported by data representation and covers modularity, flow control, recursion, encapsulation, and even parallel computing. Also is supported by flowcharts, pseudo-code, state diagrams, class diagrams, and use cases for all the possible outputs.
3) Solution implementation and evaluation: develop and implement the solution and perform systematic testing and evaluation before generalization.

C. Learning strategies of CT

Due to the new models and methodologies for learning focused on the student, there are up to 16 approaches according to [10]. From the knowledge about the qualification of professors, here are explained some of the most used empirically involving context and teamworking (please review the T. C. Hsu paper for higher details in each approach):

1) **Project-based learning (PBL)**: focus the learning process of the student around projects, encouraging them to work autonomously, using decomposition, involving the students in the design, problem-solving, decision making, or researching about questions to reach the goal by dividing and conquering the complex tasks of the project.
2) **Game-based learning**: this strategy look-for solving specific problem scenarios using a game platform. The challenge game may include typical features of problem-solving such as an unknown outcome, several paths to the goal, requires a context to understand the problem, teamwork, and support when participating in multiple players.
3) **Storytelling**: the teachers tell stories to engage the learners, by capturing their interest, because a story places ideas in a broader context, creating a link between theory and application. Besides, the effort to connect concepts the knowledge with new ideas is essential for the learning process.
4) **Collaborative learning**: it can be collaborative when the group members need to complete a series of tasks together, they need to discuss, negotiate, and share concepts or meanings important for problem-solving tasks; it also can be cooperative learning, where the partners split the work, perform subtasks individually and make partial contributions to solve the problem.
5) **Scaffolding**: is a theory that focuses on a student's ability to learn information through the help of a more informed individual.
6) **Problem-based learning**: it is an instructional method centered on practical and ***active learning*** to explore and investigate the learning solution by themselves, also, the learner will acquire the skills to communicate in an effective way (all in the context of real-world problems resolution).
7) **Problem-solving system:** it is related to learning systematically and methodologically to solve problems by developing adequate skills from understanding the problem.
8) **Systematic computational strategies:** focuses on the formulation and addressing of questions to evaluate the performance of different learning algorithms.
9) **Concept-based learning**: from the conceptualization the students try to define attributes or features among

different concepts that can be generalized to multiple instances.

10) **Aesthetic experience**: can be defined as a special state of mind that is qualitatively different from the everyday experience. It is related to feelings, and how can be expressed and understood to learn and tolerate the ambiguity by focusing on details.
11) **Human-computeruter interaction (HCI) teaching**: a discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use that studies the major phenomena surrounding them.
12) **Design-based learning**: is an inquiry-based form of learning, that is based on the integration of ***design thinking*** and the design process into the classroom.
13) **Embodied learning**: refers to pedagogical approaches that focus on the non-mental factors involved in learning, and that highlight the importance of the body and feelings.
14) **Teacher-centered learning**: the student pay attention to lectures and guiding of all the activities without collaborative learning activites.
15) **Critical computational literacy**: emphasizes on how to use the computational method, and what can be done by understanding the social, technical, and cultural dyamics of programming.
16) **Universal desing for learning**: It arouses the learners' interest through multiple methods of communication and expression.

Of course, actually there are other learning approaches such as evidence-based learning (where the education practices should be based on the best available scientific evidence, rather than tradition, personal judgement, or other subjective influences), or experience-based learning (where the process whereby knowledge is created through the transformation of experience). All of them, introduce their own characteristics and methods, but finally is the professor who defines the best strategy according to the subjects and opportunities of the context in the learning outcomes for problem solving skills.

III. Educational Policy Framework

To understand the context and scope of the proposal, it is necessary to perform a general diagnosis of public education in Colombia and then in Boyacá. The study performed by Bonilla Mejía *et al.*, [11] shows that exist important differences between urban and rural educative institutions in Colombia, mainly in those cities not certified or outlier states that face difficulties to attract and hold the best professors. There, the effects are the low academic skills and the higher temporal employment relationship. These features on the teachers introduce higher correlations with learning on pupils at all levels, reinforcing the hard gaps of poverty in regions with high lagging. Despite the different calls for study a master's or a doctoral degree (scholarships such as Maestra, Bicentenario, or Minciencias), the expected wage, unfortunately, does not compensate for all the effort and development of the professor. In addition, within the same state, there is a big heterogeneity of the institutional and financial capacity of the territorial entities, which results in deficient conditions of infrastructure and lab capacity.

Going to the territory, in Boyacá from the 123 cities in the state, 109 cities had at least one technical public school. Sogamoso has 16, Duitama 13, and Tunja 12 schools, being the main cities in their different regions. For intermediate cities, Moniquirá 6, Paipa, and Chiquinquirá have 5 technical schools; those institutions in the rural area suffer similar difficulties as exposed before. However, the Saber Pro test positions Boyacá as one of the best elementary and high school public education states. But, the orography of the state limits internet access (as a basic source of information); the lab capacity of the schools is precarious, and the distance among cities does not enable the opportunity to access an appropriate education. Nevertheless, despite the teachers making their best effort to offer education of quality, their skills in using technology, their knowledge about the STEAM model, or knowledge and mastering of the different approaches of project/problem-based learning, game-based learning, concept-based learning, storytelling, etc., is low or even null. By reviewing the state of the art and understanding the importance of defining a path toward curriculums centered on the student, this proposal considers using the technological devices to support the path hat fostering the CT among the progressors, but creating collaborative networks and teams working with the teachers, using new agile methodologies to identify the projects, problems, challenges, etc., that allow designing curriculum, but more meaningful training the professors through an educational policy for postgraduate in Education, Didactics and Pedagogy, and of course, in STEAM as the possibility to grow as colleagues for this important trade-off.

In that context, Fig. 2, shows a proposal of an education framework, that fosters the CT for pupils within 6-8 years in the first three courses at elementary school. There the curriculum must be designed especially for young kids, the point of start is curiosity and fun, focused on the colors, graphics, block-based programming, and visual and sound effects to capture their attention and imagination (Scratch may be an excellent tool). They will implement simple codes oriented to physical movement (vertical and horizontal displacements) and logic, looking for boosting the clear and effective communication of ideas for problem-solving. Then in the second stage, the kids from 9-11 years, with higher knowledge will work with 3D blocks and take advantage of the approach of game-based learning (Roblox is an ideal platform oriented to learning by coding games). Labyrinths, construction of thematical environments, moving from a block-based language programming towards a textual (scripts) and oriented to basic algorithms. The knowledge will be accumulative to reinforce the basic concepts of the previous years but increase the complexity accordingly to the age.

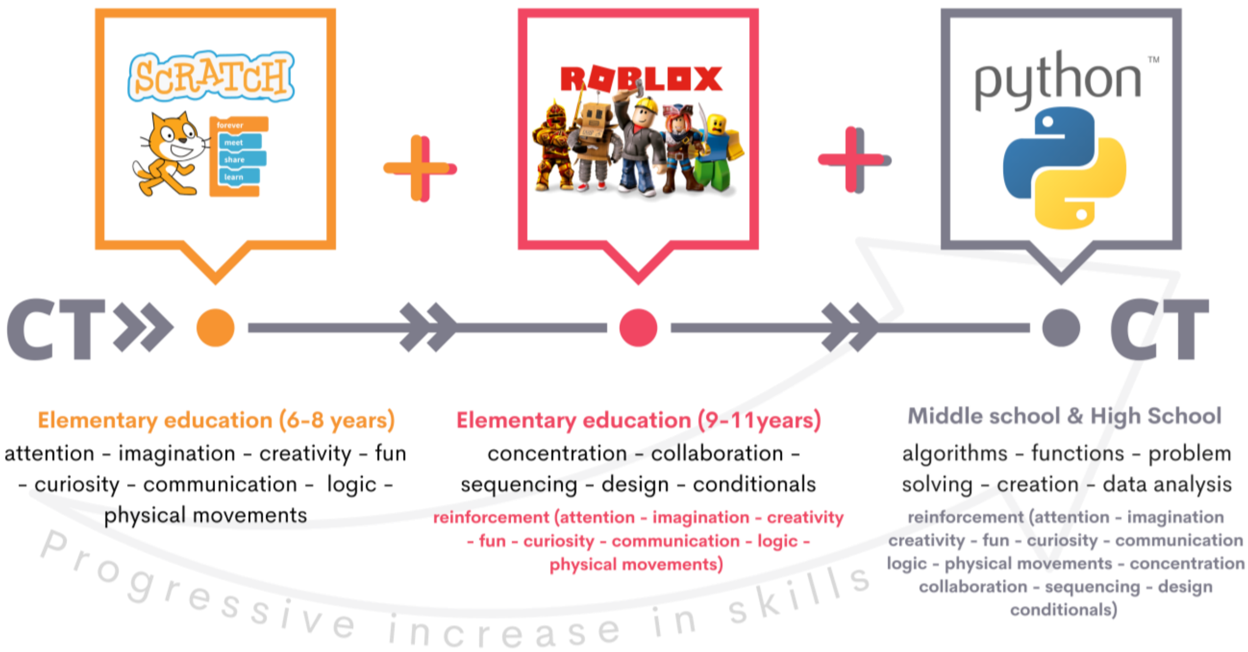

Fig. 2 Framework in the path towards STEAM (Elementary to High School). The authors.

Besides, some control structures, sequences, and loops will be learned with simple challenges or games that engage the pupil to an intermediate level, where Python will integrate the previous work to increase the CT skills and competence that can be used in other subjects' transversal in a STEAM curriculum. An additional and interesting idea consists of involving the retired with newer professors to recover the traditional methodologies and learning models, understanding that 30 years ago the technology was not considered the medium, and probably activities more creative, didactic, and ludic were used to teach some contents.

Finally, one of the main challenges is the assessment of CT considering the plugged or unplugged activities, that under the regional context (considering the internet availability) requires curricula design from the professor's expertise to formulate challenging activities involving the subjects according to age and grades in pre-school and primary education. According to Wolf & Brandt – 1998: "*Educational researchers have found that a curriculum that is focused on problem-solving around real-world problems can result in greater intellectual curiosity, motivation, improved attitude toward schooling, and higher achievement in college*". From this perspective, the curriculum has to be designed looking for real-life problems (transdisciplinary) that promote the interest and attention of students, involving them in developing the soft skills and appropriating the knowledge for founding the best solution using the tools as a way to develop the computational thinking.

IV. CONCLUSIONS

This paper proposes an educational policy framework to implement in elementary public education. The path was introduced in two stages fostering the acquisition of CT skills for the digital native kids, in rural and urban schools of Boyacá. Some difficulties in teaching experience and knowledge, lack of resources and IT infrastructure, can limit the policy, but the University may assist the curriculum design from their academic postgraduate offer with specialization and masters in the STEAM that could receive a scholarship from the governorate or Minciencias as a route to improve the qualification of teachers and impulse the CT in elementary and high school public education.

ACKNOWLEDGMENT

This work has been supported by the School of Electronic Engineering of the Universidad Pedagógica y Tecnológica de Colombia (UPTC) within the telecommunications research group GINTEL.

REFERENCES

[1] S. Papert, *Mindstorms: Children, computers, and powerful ideas*, 2nd ed. New York, NY, USA: Basic Books, Inc., Publishers, 1980.

[2] M. Lodi and S. Martini, "Computational Thinking, Between Papert and Wing," *Sci. Educ.*, vol. 30, no. 4, pp. 883–908, 2021, doi: 10.1007/s11191-021-00202-5.

[3] J. M. Wing, "Computational Thinking," *Commun. ACM*, vol. 49, no. 3, pp. 33–35, Mar. 2006, doi: 10.1145/1118178.1118215.

[4] C. Angeli *et al.*, "A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge," *J. Educ. Technol. Soc.*, vol. 19, no. 3, pp. 47–57, Feb. 2016, [Online]. Available: http://www.jstor.org/stable/jeductechsoci.19.3.47.

[5] M. Margaret I., Jeanette M. and Hartnett, "Computational Thinking in Junior Classrooms in New Zealand," *J. Open, Flex. Distance*

Learn., vol. 24, no. 2, pp. 28–48, 2020.

[6] C. Angeli and M. Giannakos, "Computational thinking education: Issues and challenges," *Comput. Human Behav.*, vol. 105, p. 106185, 2020, doi: https://doi.org/10.1016/j.chb.2019.106185.

[7] K. Cook, S. Bush, R. Cox Jr, and D. Edelen, "Development of elementary teachers' science, technology, engineering, arts, and mathematics planning practices," *Sch. Sci. Math.*, vol. 120, no. 4, pp. 197–208, Apr. 2020, doi: https://doi.org/10.1111/ssm.12400.

[8] 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, 2014, doi: https://doi.org/10.1016/j.compedu.2013.10.020.

[9] M. Jenson, Jennifer; Droumeva, "Exploring Media Literacy and Computational Thinking: A Game Maker Curriculum Study," *Electron. J. e-Learning*, vol. 14, no. 2, pp. 111–121, 2016.

[10] T.-C. Hsu, S.-C. Chang, and Y.-T. Hung, "How to learn and how to teach computational thinking: Suggestions based on a review of the literature," *Comput. Educ.*, vol. 126, pp. 296–310, 2018, doi: https://doi.org/10.1016/j.compedu.2018.07.004.

[11] L. et al. Bonilla-Mejía, "¿Quiénes son los docentes en Colombia? Características generales y brechas regionales," *Cent. Estud. Económicos Reg. Cart. - Banco la Respública*, vol. 276, 2018.