# A Proposal of a Scenario to Integrate Active Pedagogical Approaches to Teach Scratch in Primary School

https://doi.org/10.3991/ijim.v17i09.36797

Mohammed Chekour[1](✉), Yassine Zaoui Seghroucheni[2], Amina Aboulkacem[1], Moulay Mustapha Hafid[1]
[1] Higher School of Education and Training, Ibn Tofail University, Kenitra, Morocco
[2] Faculty of Sciences, Mohammed V University in Rabat, Rabat, Morocco
Mohamed.chekour@uit.ac.ma

**Abstract—**In today's digital age, it is common for people, including elementary school students, to use technology in their daily lives and show a keen interest in the digital world, particularly in games. Scratch offers these students the chance to create their own games rather than just consuming them. Additionally, it is a software that fosters the development of computational thinking. The goal of this study is to propose a teaching scenario using Scratch in primary schools, based on active pedagogical methods that prioritize the student's role in the teaching-learning process. Through this scenario, students will work in groups of three to solve problems and propose solutions using the "trial and error" method. As they try to detect and correct errors in their programs, they will be able to develop their problem-solving and investigative skills in a collaborative and motivating environment. Overall, the proposed scenario combines active teaching approaches with the nature of the subject matter, empowering students to take an active role in their own learning.

**Keywords—**scratch, teaching scenario, collaborative learning, trial and error method, investigation, problem solving

## 1 Introduction

Information and communication technologies are vital in today's society, as the 21st century is heavily reliant on connectivity. It is essential for individuals to have varying levels of computer skills [1]. Children as young as 3 or 4 years old, who may not yet be able to read, can easily use and manipulate computing devices such as tablets and smartphones [2]. By the time they reach elementary school, computer science has become a crucial discipline for understanding the world and creating a foundation for future knowledge in middle and high school [3], [4]. In education, computational thinking (CT) and programming learning are taking on a central role, with many educators promoting the integration of programming learning, particularly using the Scratch software, into curricula [5]–[7]. Scratch is an educational programming language with a user-friendly and engaging multimedia programming framework, and is just one of several programming tools that are seen as a way to develop 21st century skills like digital

literacy and CT [8]. The use of Scratch is not arbitrary, as it is suitable for programming beginners and has a positive impact on the development of CT skills. The incorporation of developmentally appropriate technologies allows for the fostering of coding skills, potentially leading to the advancement of CT fluency or at least familiarity in young children [9]. However, integrating Scratch into K-12 education presents numerous challenges, including preparing elementary school teachers to effectively teach CT. This article presents a pilot approach to this issue at Ibn Tofail University, involving the design of a pedagogical scenario for teaching Scratch in primary school. The proposed scenario is based on active learner-centered pedagogical approaches, combining various pedagogical methods that prioritize effective learning of CT and coding using the Scratch programming language. We have adopted a philosophy of social constructivism, which puts the learner at the center of the teaching-learning process and favors collaborative learning through investigation, problem solving, and error correction.

## 2 Literature review

Computer skills offer the opportunity to introduce young children to essential CT, coding, and problem-solving skills. However, little is known about the educational value of these apps. The fast pace at which developers produce these apps and the breadth of the available apps have gone beyond what it is reasonable for researchers and experts in the domain to evaluate [10]. Although the pupils are placed in the center of the teaching-learning process, the teacher plays a very important role. The teacher is a facilitator and is called upon to evaluate the entire learning process [11].

### 2.1 Computational thinking

Computational Thinking (CT) is a fundamental and cross-cutting skill that is applicable in all knowledge areas [12]. According to Tikva and Tambouris [13], it involves various concepts and capacities, including data collection, analysis, representation, abstraction, problem decomposition, algorithms, and automation. Some view CT as an essential skill that has been made possible by the proliferation of computers and can help young people unleash their creativity through the development of higher order thinking skills, such as critical thinking and problem-solving [9].

Many studies aim to address the universal learning questions of "What should be taught?" and "What can we learn?" [14]. However, there is no clear consensus on the answers to these questions when it comes to teaching CT to primary school students. CT is often an "equivocal" and poorly defined concept [15], and the way in which students are taught and evaluated in this field requires further study [16].

### 2.2 Scratch and computer thinking

Coding is an effective way to develop CT skills, particularly in early childhood when there is a great opportunity to spark an interest in computing [9]. There are various programming tools available for supporting CT, such as Logo, Arduino, robotics kits,

and Scratch. Scratch is a visual programming language that offers a rich learning environment, allowing for the creation of media-rich interactive projects such as animated stories, science projects, simulations, and games [17]. It also promotes problem-based learning, providing immediate feedback on errors in a user-friendly way and helping to refine the way of thinking through its visual structure [18]. Scratch makes CT accessible, engaging, and motivating [19], fostering inquiry-based learning and allowing learners to use their creativity and imagination [20]. Additionally, it provides an opportunity to learn the concepts of CT and programming [21] and can serve as a source of extrinsic motivation to develop useful products in a supportive environment that encourages experimentation and learning from mistakes [22].

### 2.3 Adequate pedagogical approaches to teach scratch

Human beings are naturally inclined to solve problems from birth. Sometimes, when a task becomes too complex, more competent individuals may offer their assistance to enable a novice to solve problems or achieve goals that are beyond their capabilities [22]. A problem is defined as the absence of an obvious pathway that can bridge the gap between the current situation and the desired outcome [23]. The problem-solving approach is often used in situations where it is not feasible for students to fully master a subject within a limited time frame [24]. This approach involves the crucial stage of problematization, in which students take part in constructing various problems in the classroom [25]. The teacher's role during this process is to guide the students in their learning journey, while remaining in the background [26].

The knowledge-based society strives to transform information into knowledge that can be applied to solve problems [27]. The problem-solving principle is centered on using existing knowledge to solve new problems and learn new things [24]. In the educational setting, competencies are frequently discussed from a theoretical perspective [28]. However, in a professional setting, the importance of practical problem-solving skills in daily life is emphasized [28]. For example, computer programming involves using computer equipment to model and automate abstract thought processes in order to solve problems [29].

Critical thinking and problem-solving skills can be applied in a variety of fields and through various approaches. Teaching Scratch, a programming language, is no exception. In fact, learning computational thinking concepts using Scratch is being promoted in the education system [30]. Researchers have conducted e-learning courses using the Moodle platform to test and evaluate this method, and the results have shown that it is an effective way to learn computational thinking and Scratch programming [30].

## 3 The Intersection of problem-solving approach with other active pedagogical approaches

Despite the benefits of the problem-solving approach, we also aim to incorporate the added value of other active pedagogical approaches and methods for teaching Scratch at the primary level (see Figure 1). Collaborative work, inquiry-based learning, and

trial-and-error learning can all enhance the learning of computational thinking using Scratch. In this article, we propose a scenario that integrates active pedagogical approaches to teach Scratch in primary school.


**Fig. 1.** Teaching Scratch with active pedagogical approaches

## 4 Proposal of a pedagogical scenario based on active pedagogical approaches to teach Scratch

The purpose of this article is to create a pedagogical scenario for teaching Scratch using learner-centered active pedagogical approaches. The teacher, acting as a facilitator, will encourage collaborative work in teams of three learners. At the beginning of the session, the teacher will encourage learners to suggest games to create with Scratch. Once the suggested games have been decided upon, the learners will begin proposing scenarios and solutions to the problems that arise during the game creation process. These proposed solutions may vary and may also be incomplete. The facilitator will encourage the learners to concretely implement their proposed solutions using Scratch. When the class moves on to the programming phase, there may be two possible outcomes:

- If the final product (game) meets the class's expectations, they will move on to the next problem or game.
- If the game does not work, the entire class will work together to identify the nature of the error. If the error is related to the analysis, the learners will be asked to review

their computational thinking. If it is a programming error, the learners will need to detect and correct it.

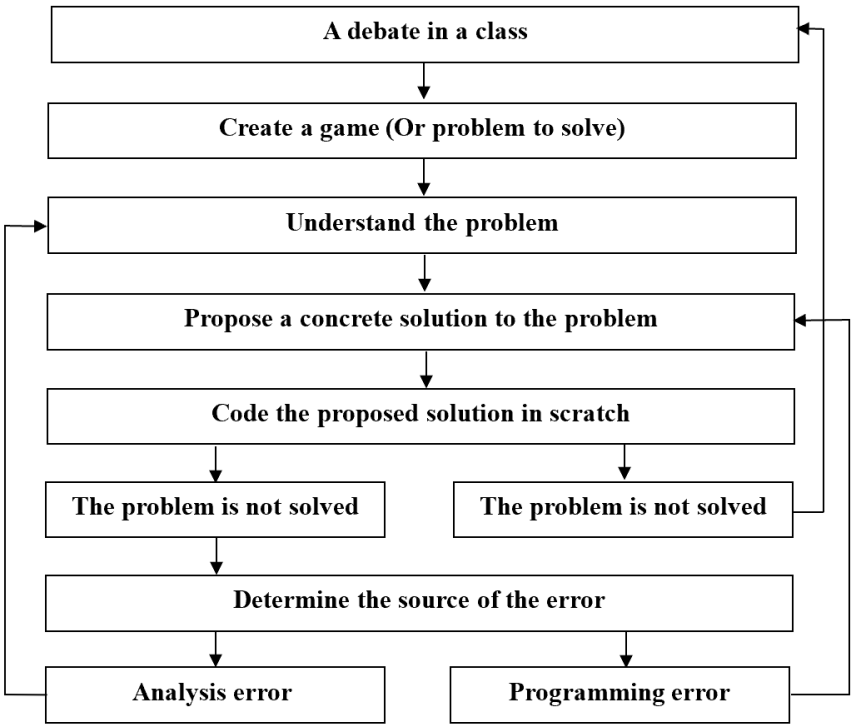


**Fig. 2.** Pedagogical scenario to teach Scratch in primary school

One unique aspect of this research is the proposal of a pedagogical scenario for teaching Scratch to primary school students that puts the student at the center of the teaching and learning process, using various active pedagogical approaches. These approaches were chosen based on previous research and the needs and expectations of the students. The proposed scenario outlines in detail the process of constructing new knowledge in the students, including anticipated obstacles and recommendations for improving their mastery of computational thinking skills.

## 5 Conclusion

Evidence that young children can learn and develop computational thinking skills has led governments to integrate CT and Scratch into their curricula, starting in the earliest grades [31]. However, successfully integrating this computer tool into education also requires qualified teachers and effective pedagogical approaches that can help young children build new skills. The goal-oriented approach is common in education [23], as goals are often presented in the form of expected student behaviors, which bears

some similarities to behaviorism. However, behavioral changes in students can be properly explained, unlike conceptual changes[32]. In contrast to behaviorists, social constructivists place the student at the center of the teaching and learning process, believing that each student constructs and reconstructs their own knowledge based on their experiences and interactions with their peers [33]. This requires a motivating classroom climate that encourages initiative, the ability to make mistakes, problem-solving, and inquiry [34]. In this way, the transmission of knowledge is more effective and sustainable because students are actively engaged in the learning process [35]. In this article, we have attempted to adopt a social constructivist philosophy in the development of a pedagogical scenario for teaching computational thinking and the Scratch programming language in primary school. Our scenario also promotes collaborative learning and teamwork, which are key factors in the constructivist perspective. The facilitator stimulates debate in the class and allows students to have a voice in determining the games to be designed. This helps to increase student motivation as they formulate their own questions (problems) and work to find answers (create games). Finally, inquiry and the pedagogy of error are present throughout all stages of the scenario. Investigating the nature of mistakes made and correcting them provides a valuable opportunity for students to develop problem-solving, computational thinking, and coding skills using Scratch.

In summary, cultivating computational thinking skills in elementary school children is becoming increasingly important in educational systems worldwide. Given the complexity of pedagogical approaches and the heterogeneity of students, teachers are encouraged to use multiple teaching approaches and methods to make learning about CT and coding more enjoyable. Therefore, we propose to help teachers design their courses using our scenario, which gives students the chance to be autonomous, make decisions, collaborate, and learn from their mistakes.

## 6 References

[1] M. Amraouy, M. Bellafkih, A. Bennane, and M. M. Himmi, 'Online Competency-Based Assessment (OCBA): From Conceptual Model to Operational Authoring System', iJIM, vol. 16, no. 04, p. 47, 2022. https://doi.org/10.3991/ijim.v16i04.28373

[2] M. Chekour, M. Laafou, and R. Janati-Idrissi, 'Distance training for physics teachers in Pspice simulator', Mediterr. J. Soc. Sci., vol. 6, no. 3 S1, pp. 232–232, 2015. https://doi.org/10.5901/mjss.2015.v6n3s1p232

[3] E. Yunus and S. B. Zaibon, 'Connecting Computational Thinking (CT) Concept with the Game-Based Learning (GBL) Elements.', Int. J. Interact. Mob. Technol., vol. 16, no. 20, 2021. https://doi.org/10.3991/ijim.v15i20.23739

[4] F. A. Salem, I. W. Damaj, L. Hamandi, and R. N. Zantout, 'Effective Assessment of Computer Science Capstone Projects and Student Outcomes.', Int J Eng Pedagog, vol. 10, no. 2, pp. 72–93, 2020. https://doi.org/10.3991/ijep.v10i2.11855

[5] A. M. Ortiz-Colon and J. L. M. Romo, 'Teaching with Scratch in compulsory secondary education', Int. J. Emerg. Technol. Learn. IJET, vol. 11, no. 02, pp. 67–70, 2016. https://doi.org/10.3991/ijet.v11i02.5094

[6] A. Sarasa-Cabezuelo, 'Use of Scratch for the teaching of second languages', Int. J. Emerg. Technol. Learn. IJET, vol. 14, no. 21, pp. 80–95, 2019. https://doi.org/10.3991/ijet.v14i21.11217
[7] Y. Lu, 'Scratch teaching mode of a course for college students', Int. J. Emerg. Technol. Learn. IJET, vol. 16, no. 5, pp. 186–200, 2021. https://doi.org/10.3991/ijet.v16i05.21269
[8] J. Nouri, L. Zhang, L. Mannila, and E. Norén, 'Development of computational thinking, digital competence and 21st century skills when learning programming in K-9', Educ. Inq., vol. 11, no. 1, pp. 1–17, 2020. https://doi.org/10.1080/20004508.2019.1627844
[9] S. Papadakis, 'Apps to promote computational thinking concepts and coding skills in children of preschool and pre-primary school age', in Research Anthology on Computational Thinking, Programming, and Robotics in the Classroom, IGI Global, 2022, pp. 610–630. http://dx.doi.org/10.4018/978-1-7998-1486-3.ch006
[10] P. Stamatios, 'Can Preschoolers Learn Computational Thinking and Coding Skills with ScratchJr? A Systematic Literature Review', Int. J. Educ. Reform, p. 10567879221076076, 2022. https://doi.org/10.1177/10567879221076077
[11] A.-I. Zourmpakis, S. Papadakis, and M. Kalogiannakis, 'Education of preschool and elementary teachers on the use of adaptive gamification in science education', Int. J. Technol. Enhanc. Learn., vol. 14, no. 1, pp. 1–16, 2022. https://doi.org/10.1504/IJTEL.2022.120556
[12] K. N. Tang, 'The importance of soft skills acquisition by teachers in higher education institutions', Kasetsart J. Soc. Sci., vol. 41, no. 1, pp. 22–27, 2020. https://doi.org/10.1016/j.kjss.2018.01.002
[13] C. Tikva and E. Tambouris, 'Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature Review', Comput. Educ., vol. 162, p. 104083, 2021. https://doi.org/10.1016/j.compedu.2020.104083
[14] S. Grover and R. Pea, 'Computational thinking in K–12: A review of the state of the field', Educ. Res., vol. 42, no. 1, pp. 38–43, 2013. https://doi.org/10.3102/0013189X12463051
[15] M. Wang and Y. feng Wang, 'A Study on Computer Teaching Based on Computational Thinking', Int. J. Emerg. Technol. Learn. IJET, vol. 11, no. 12, pp. 72–74, 2016. https://doi.org/10.3991/ijet.v11i12.6069
[16] B. H. Majeed, L. F. Jawad, and H. T. ALRikabi, 'Computational Thinking (CT) Among University Students.', Int. J. Interact. Mob. Technol., vol. 16, no. 10, 2022. https://doi.org/10.3991/ijim.v16i10.30043
[17] F. Minichiello, 'L'enseignement du code à l'école', Rev. Int. D'éducation Sèvres, no. 67, pp. 12–16, 2014. https://doi.org/10.4000/ries.4078
[18] J. Fagerlund, P. Häkkinen, M. Vesisenaho, and J. Viiri, 'Computational thinking in programming with Scratch in primary schools: A systematic review', Comput. Appl. Eng. Educ., vol. 29, no. 1, pp. 12–28, 2021. https://doi.org/10.1002/cae.22255
[19] I. Moraiti, A. Fotoglou, and A. Drigas, 'Coding with Block Programming Languages in Educational Robotics and Mobiles, Improve Problem Solving, Creativity & Critical Thinking Skills.', Int. J. Interact. Mob. Technol., vol. 16, no. 20, 2022. https://doi.org/10.3991/ijim.v16i20.34247
[20] H. Yildiz Durak, 'The effects of using different tools in programming teaching of secondary school students on engagement, computational thinking and reflective thinking skills for problem solving', Technol. Knowl. Learn., vol. 25, no. 1, pp. 179–195, 2020. https://doi.org/10.1007/s10758-018-9391-y
[21] N. Parsazadeh, P.-Y. Cheng, T.-T. Wu, and Y.-M. Huang, 'Integrating computational thinking concept into digital storytelling to improve learners' motivation and performance', J. Educ. Comput. Res., vol. 59, no. 3, pp. 470–495, 2021. https://doi.org/10.1177/0735633–120967315

[22] J. S. Bruner, 'Organization of early skilled action', Child Dev., pp. 1–11, 1973. https://doi.org/10.2307/1127671
[23] M. Chekour, M. Laafou, and R. Janati-Idrissi, 'What are the adequate pedagogical approaches for teaching scientific disciplines? Physics as a case study', J. Educ. Soc. Res., vol. 8, no. 2, p. 141, 2018. https://doi.org/10.2478/jesr-2018-0025
[24] H. Habiddin, M. Ashar, A. Hamdan, and K. R. Nasir, 'Digital Comic Media for Teaching Secondary School Science', iJIM, vol. 16, no. 03, p. 159, 2022. https://doi.org/10.3991/ijim.v16i03.28967
[25] O. D. Triswidrananta, A. N. Pramudhita, and I. D. Wijaya, 'Learning Management System Based on Assessment for Learning to Improve Computational Thinking', iJIM, vol. 16, no. 04, p. 151, 2022. https://doi.org/10.3991/ijim.v16i04.28979
[26] M. Chekour, 'The impact perception of the resonance phenomenon simulation on the learning of physics concepts', Phys. Educ., vol. 53, no. 5, p. 055004, 2018. https://doi.org/10.1088/1361-6552/aac984
[27] C. Lertyosbordin, S. Maneewan, and D. Srikaew, 'Components and Indicators of Problem-solving Skills in Robot Programming Activities', Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 9, 2021. https://doi.org/10.14569/IJACSA.2021.0120917
[28] J. Silva and I. Silveira, 'A Systematic Review on Open Educational Games for Programming Learning and Teaching', Int. J. Emerg. Technol. Learn. IJET, vol. 15, no. 9, pp. 156–172, 2020. https://doi.org/10.3991/ijet.v15i09.12437
[29] M. A. Tadlaoui and M. Chekour, 'A blended learning approach for teaching python programming language: towards a post pandemic pedagogy', Int. J. Adv. Comput. Res., vol. 11, no. 52, p. 13, 2021. http://dx.doi.org/10.19101/IJACR.2020.1048120
[30] M. J. Marcelino, T. Pessoa, C. Vieira, T. Salvador, and A. J. Mendes, 'Learning computational thinking and scratch at distance', Comput. Hum. Behav., vol. 80, pp. 470–477, 2018. https://doi.org/10.1016/j.chb.2017.09.025
[31] S. Papadakis, 'The impact of coding apps to support young children in computational thinking and computational fluency. A literature review', in Frontiers in Education, 2021, p. 183. https://doi.org/10.3389/feduc.2021.657895
[32] G. Sedrakyan, J. Malmberg, K. Verbert, S. Järvelä, and P. A. Kirschner, 'Linking learning behavior analytics and learning science concepts: Designing a learning analytics dashboard for feedback to support learning regulation', Comput. Hum. Behav., vol. 107, p. 105512, 2020. https://doi.org/10.1016/j.chb.2018.05.004
[33] P. A. Ertmer and T. J. Newby, 'Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective', Perform. Improv. Q., vol. 26, no. 2, pp. 43–71, 2013. https://doi.org/10.1002/piq.21143
[34] A. Olewnik, R. Yerrick, A. Simmons, Y. Lee, and B. Stuhlmiller, 'Defining open-ended problem solving through problem typology framework', Int. J. Eng. Pedagogy IJEP, vol. 10, no. 1, 2020. https://doi.org/10.3991/ijep.v8i4.8281
[35] Y. Kubota, A. Takahashi, Y. Hayakawa, Y. Kashiwaba, and K. Yajima, 'Analysis of Active Learning Suitability of Subjects in Information and Electronics.', Int. J. Eng. Pedagogy, vol. 7, no. 3, 2017. https://doi.org/10.3991/ijep.v7i3.6968

## 7 Authors

**Mohammed Chekour** has received his PHD in ICT and education in 2019 from the University of Abdelmalek Essaadi of Tetouan in Morocco. He is a professor assistant at Higher School of Education and Training, Ibn Tofail University, Morocco since 2021. He is the author of several publications in the field of educational technology and educational systems (mohamed.chekour@uit.ac.ma).

**Yassine Zaoui Seghroucheni** has received his PHD in computer science in 2017 from the University of Abdelmalek Essaadi of Tetouan in Morocco. He is a Professor Assistant at the Intelligent Processing and Security of Systems, Faculty of sciences, Mohammed V University in Rabat, Morocco since 2020. He is the author of several publications namely in the field of Adaptive learning systems, educational science, Mobile learning and Machine learning. He is the author of multiple publications in several international journals and has been a reviewer for several papers in international conferences (y.zaoui@um5r.ac.ma).

**Amina Aboulkacem** an assistant professor at ESESFK, The Higher School of Education and Training in Kenitra that is affiliated to the campus of, University Ibn Toufail Kenitra, Morocco. She has a PhD in language, culture and communication from Moulay Ismail University, Meknes in 2014. She has participated in the first international conference on Quality Assurance and Language development, Meknes 2022. She has attended the QU 7th Annual International Conference on English Language Teaching Doha, State of Qatar in 2022(amina.aboulkacem@uit.ac.ma).

**Moulay Mustapha Hafid**, a professor of higher education, he is a PhD holder in Physics Sciences, Graduated from University Paule Verlein Supelec-Metz in France. He is the Director of the Higher school of Education and Training in Kenitra that is affiliated to the campus of, University Ibn Toufail Kenitra, Morocco. He is the author and co-author of a number of articles and books in his field of interest. He has participated in several national and international conferences in various domains and fields (moulaymustapha.hafid@uit.ac.ma).

Article submitted 2022-11-12. Resubmitted 2023-01-10. Final acceptance 2023-01-10. Final version published as submitted by the authors.