



Teaching Coding Using Scratch to Elementary Students: Exploration of Benefits for Students

Tira Nur Fitria

Institut Teknologi Bisnis AAS Indonesia, Sukoharjo, Central Java, Indonesia

* Corresponding Author: tiranurfitria@gmail.com

INFO ARTIKEL

Riwayat Artikel

Diterima : 03-12-2024

Disetujui : 17-06-2025

Diterbitkan : 26-06-2025

Kata Kunci: *coding, pelajaran coding, siswa SD*

Keywords:

coding, coding lesson, elementary students, primary students

Abstract

In today's rapidly advancing digital era, informatics knowledge and skills have become essential, especially for preparing the younger generation to face future challenges. One key approach is integrating technology into elementary education through coding programs that develop students' interests and talents in programming. This library research explores how coding is taught at the elementary level and the benefits derived from its implementation. In Indonesia, many elementary schools have started incorporating coding into their curricula or as extracurricular activities, recognizing its potential to foster computational thinking, creativity, and problem-solving skills. Scratch, a widely used visual programming tool, is particularly effective for beginners, enabling students to grasp core programming concepts without complex syntax. Research shows

that using Scratch enhances students' computational thinking, creativity, and understanding of programming. It also supports the development of critical thinking, problem-solving strategies, and precision. Moreover, students gain practical experience by creating animations, games, and interactive projects, which increases their engagement and confidence. The evidence suggests that Scratch makes learning coding interactive and enjoyable while helping students build foundational digital competencies. Integrating Scratch into elementary education not only enriches learning but also prepares students to be active participants in the digital world.

Abstrak

Di era digital yang terus berkembang, pengetahuan dan keterampilan di bidang informatika menjadi semakin penting. Untuk membekali generasi muda menghadapi tantangan masa depan, pendidikan harus mengintegrasikan teknologi ke dalam kurikulumnya seperti program coding untuk menumbuhkan minat dan bakat siswa dalam pemrograman dan teknologi. Penelitian ini menjelaskan bagaimana coding diajarkan di tingkat sekolah dasar dan manfaat yang diperoleh dari penerapan coding berdasarkan penelitian yang ada. Penelitian ini merupakan penelitian kepustakaan. Saat ini banyak sekolah dasar di SD Indonesia yang mulai mengajarkan coding sebagai pelajaran dalam kurikulum resmi dan sebagai kegiatan ekstrakurikuler. Seiring kemajuan teknologi dan pentingnya keterampilan digital di abad ke-21, banyak sekolah mungkin menyadari bahwa mengajarkan coding sejak dini dapat membantu siswa mengembangkan pemikiran komputasi, kreativitas, dan keterampilan pemecahan masalah yang penting untuk masa depan mereka. Temuannya menunjukkan penggunaan coding khususnya Scratch pada siswa sekolah dasar. Scratch telah menjadi alat yang populer dalam mengajarkan pemrograman kepada anak-anak di tingkat sekolah dasar. Scratch adalah alat pemrograman visual yang dirancang untuk memperkenalkan konsep pemrograman, khususnya kepada anak-anak, untuk melatih pemikiran komputasi pada pemula. Scratch memungkinkan siswa mempelajari pemrograman tanpa harus menguasai bahasa pemrograman yang rumit. Penggunaan Scratch dalam pembelajaran coding terbukti memberikan banyak manfaat bagi siswa. Scratch membantu meningkatkan keterampilan berpikir komputasi, pemahaman pemrograman, dan kreativitas. Scratch juga membantu siswa mempraktikkan strategi pemecahan masalah, akurasi, dan berpikir kritis. Selain itu, siswa dapat membuat animasi dan permainan setelah mengikuti pelatihan, dengan didukung oleh lingkungan sekitar yang penting. Scratch mendukung pembelajaran yang interaktif dan menyenangkan, memudahkan pemahaman logika pemrograman, dan meningkatkan kompetensi dasar pemrograman.

Introduction

The 21st century is commonly known as the information age, with technology advancing at a fast pace, making it challenging to keep up. We live in a digital era where nearly everything relies on computers (Mannova, 2022). Computational thinking (CT) skills are increasingly important in various areas of work and daily life, and CT has been integrated into K-12 education worldwide (Wei et al., 2021). Computational thinking (CT), regarded as a form of literacy, has become a part of the ICT curriculum in many countries at the K-12 level. As a result, there is a growing need for more evidence on both the theoretical and practical aspects of developing CT skills in K-12 students (Dağ et al., 2023). Besides, improving digital skills in elementary school students is becoming increasingly important in the era of Industrial Revolution 4.0 (Budiman et al., 2023). This means that in the era of Industrial Revolution 4.0, digital skills have become very important for students in elementary schools. This is because rapid technological developments affect almost all aspects of life, and the ability to understand and master digital technology is a basic skill that the younger generation must have to compete and adapt in the future.

In the past, computer programming might only be studied at the college or high school level. However, with the development of technology, many countries are starting to introduce coding from an early age at the elementary school level. Learning to code at an early age aims to equip students with skills that will be useful throughout their lives, not only in the world of technology but also in developing other cognitive skills such as logical, creative thinking, and problem-solving. Therefore, it is hoped that introducing coding from an early age can provide a strong foundation for the younger generation to innovate, create technology-based solutions, and participate actively in the digital world.

In the digital era that continues to develop, understanding and skills in the field of informatics becoming increasingly important. To prepare the younger generation to face the challenges of the times in the future, education must be able to integrate technology into its curriculum. One of the approaches that have been adopted by many schools is extracurricular coding programs, which aim to develop students' interests and talents in the field of programming and technology (Putro & Astuti, 2022). The digital era presents a challenge to human adaptation and growth, particularly in the education sector. Students are expected to acquire innovative values through creative thinking, develop new products and services, and adopt a perspective known as Computational Thinking (Mufidah & Majid, 2024). Programming, also known as coding, is the activity of pouring logical thinking or algorithms from solutions obtained from computational thinking into programming language code and following the rules in the language used (Lestari et al., 2023). That is why computational thinking, computers, and programming (coding) are things that are interrelated with each other.

Rapid technological developments in this digital era have had a significant impact on the way we learn and work. One important skill that is increasingly needed is the ability in the field of coding or computer programming. Coding is the process of writing instructions for a computer and is now one of the basic skills that the younger generation needs to have. Therefore, learning coding for elementary school (SD) students is very relevant and important. As technology continues to advance, the jobs of the future will rely heavily on an understanding of coding, data, and artificial intelligence. Therefore, it is hoped that introducing coding from an early age can provide a strong foundation for the younger generation to innovate, create technology-based solutions, and participate actively in the digital world. As technology continues to advance, many jobs in the future will require an understanding of coding, data, and artificial intelligence. By starting to learn coding from an early age, students are not only prepared to face future challenges but are also equipped with the ability to innovate and create technology-based solutions.

Technology has become an essential part of daily life (Pramono, 2024). As a result, elementary school (SD) students must master basic computer skills in the digital age. Students who are proficient in computers not only adapt more easily to technological advancements but also gain access to a wide range of educational opportunities, enhance their creativity, and express their imagination. With solid computer skills, students can improve their problem-solving abilities, think critically, and collaborate effectively. A strong understanding of technology also

enables students to learn more efficiently, access information, and communicate better. This prepares them to become productive members of a globally connected, technology-driven society. Therefore, computer education in elementary schools is not just an optional addition but a necessity for the future, helping to shape the next generation.

Education in the 21st century demands skills that are relevant to technological developments, and one of the increasingly important skills is the ability in the field of coding or computer programming. In many countries, learning coding has been integrated into school curricula starting from the elementary level. This is because information and communication technology (ICT) plays a very important role in everyday life. Therefore, the ability to understand and master the basics of coding is an essential competency for the younger generation. Learning coding at the elementary school level is not only relevant to prepare students for a more technology-based world of work but also to develop a mindset that can help them solve problems and create new ideas.

As technology continues to advance, the jobs of the future will rely heavily on an understanding of coding, data, and artificial intelligence. Therefore, it is hoped that introducing coding from an early age can provide a strong foundation for the younger generation to innovate, create technology-based solutions, and participate actively in the digital world. Learning coding at the elementary school level is not only relevant to prepare students for a more technology-based world of work but also to develop a mindset that can help them solve problems and create new ideas.

Learning coding for elementary school students is not only relevant to prepare them for a more technological future but also to train cognitive and social skills that can be useful in various aspects of life. Elementary school students are an age group that is at a very rapid stage of cognitive development. At this age, children can understand basic, abstract concepts and adapt to creative and fun ways of learning. Learning coding for elementary school students aims to instill the basics of programming logic, as well as improve problem-solving skills, critical thinking, and creativity. Apart from that, coding can also help students prepare themselves for the challenges of the world of work in the future, where technology and digital devices will continue to develop rapidly. The integration of coding into the elementary school curriculum is also encouraged by the existence of various tools and platforms designed specifically for children, such as Scratch, Code.org, and various other educational applications. These tools provide a simple and attractive visual interface, allowing students to learn coding in a fun and easy-to-understand way.

Children at the elementary school level are at a stage of rapid cognitive development and can understand abstract concepts. Coding learning can be presented in a fun and interactive way, using educational tools or applications designed specifically for children, such as Scratch, Blockly, or Code.org. With this platform, students can learn basic coding concepts in a visual and fun way, without having to be hampered by complicated programming languages. Apart from that, coding also hones other skills, such as teamwork, communication, and the ability to adapt to technology. The revival of computer programming in school curricula offers the potential to equip students for the future, extending beyond simply teaching them how to code (Popat & Starkey, 2019).

Recent actions by governments in various countries have resulted in the inclusion of coding in school curricula, or its promotion as part of computing, mathematics, or science programs (Falloon, 2016). A key aspect of this shift is introducing computing lessons in the early grades, with some countries starting as early as kindergarten (Vlahu-Gjorgievska et al., 2018). In England, all primary school students must write computer programs and study computational thinking (Price & Price-Mohr, 2018). Even, the instruction of computing is increasingly recognized as a crucial component of twenty-first-century education (Rich et al., 2019). Learning and understanding coding logic can be done from early childhood, especially elementary school students. The process of growing students' interest in learning coding can use simple applications in gameplay (Prasti et al., 2022).

Computational thinking (CT) and its coding component are gradually being integrated into primary school curricula around the world (Arfé et al., 2020). Teaching coding and computational

thinking has become a growing educational priority, now incorporated into the mandatory curriculum in countries such as the United States, Finland, the UK, Germany, Belgium, the Netherlands, New Zealand, and Australia (Mills et al., 2024). Many countries have integrated or are planning to integrate computational thinking into their educational curricula. This necessitates a well-structured educational strategy that involves various stakeholders, including policymakers, educators, and students. Lloyd & Chandra (2020) explain that coding and computational thinking are gaining growing significance in primary classrooms. In Australia, these subjects are being progressively incorporated into the school curriculum across all states and territories. As a result, preservice teachers in primary education programs must possess the necessary technological and pedagogical skills to design and implement classroom activities that align with the national curriculum.

Vice President Gibran Rakabuming Raka has proposed that coding lessons be included in the elementary school (SD) and junior high school (SMP) curriculum. This proposal was submitted to the Minister of Primary and Secondary Education, Abdul Mu'ti, to prepare Indonesia's young generation to face technological challenges in the future. Gibran emphasized the importance of mastering skills such as programming, artificial intelligence (AI), and machine learning to achieve the vision of a Golden Indonesia 2045. He hopes that by including coding lessons in the curriculum, Indonesian students can compete with other countries, such as India, which have already implemented it. Technology education at the elementary level. Thus, the integration of coding in the education curriculum in elementary schools is very important as an effort to prepare the younger generation who are better prepared to face the demands of the times. Learning coding is expected to equip students with the skills needed to face future challenges while developing their potential more widely. Through learning coding at the elementary school level, it is hoped that students can develop a basic understanding of technology, as well as form a mindset that can support them in solving problems and creating new ideas. Therefore, schools need to integrate coding into the educational curriculum as part of efforts to prepare the younger generation who are better prepared to face the demands of contemporary developments.

The process of learning computer programs for children must be packaged easily so that they do not think computer programs are a complicated job. Material that is conveyed is so that children can understand how frequently used application processes work. Understanding the logic of computer programs can be done early in childhood (Sholeh et al., 2022). The process of cultivating children's interest in learning computer programs can be done using simple applications and by developing the logic of computer programs in gameplay.

In response to increasing demand, various online platforms have been introduced as effective tools for facilitating programming learning (Maura & Sutabri, 2024). These platforms offer a variety of resources, tutorials, and practical projects that allow students to learn coding independently or with the guidance of a teacher. There are many apps designed to teach coding to children at the elementary level. One of them is Scratch, a block-based application that allows children to create interactive animations, games, and stories without writing code. For younger children, ScratchJr offers a simpler interface, suitable for ages 5-7. Tynker is also popular, with game-based courses that teach coding through fun tutorials, while Blockly uses similar visual methods to Scratch to teach the basics of programming. Kodable teaches basic programming through an interesting adventure game for children aged 4-10 years, while Lightbot is a puzzle game that teaches programming concepts by solving puzzles. For younger children, Bee-Bot teaches basic coding through a physical robot or virtual app. Lastly, CodeCombat uses adventure games to teach programming languages such as Python or JavaScript. These apps allow children to learn coding in a fun and creative way, helping them develop programming skills from an early age.

According to Arslan & Çelik (2022), coding, recognized as a key 21st-century skill, has been incorporated into the curricula of many countries, ranging from preschool to high school, in recent years. Primary school teachers play a crucial role in teaching coding. This study aims to explore the coding education received by primary school students and its impact, from the perspectives of both classroom teachers and students. The renewed focus on teaching coding in

primary school classrooms has resulted in a shift towards using physical computing and coding to enhance students' abilities in algorithms, computational thinking, and problem-solving (Martin et al., 2024).

There are several previous studies related to learning coding at elementary or primary school. Ramadhan et al. (2020) explain that Al-Azhar Syifa Budi Elementary School Solo provides coding lessons to students as a response to advances in information technology. Looking at students who are familiar with gadgets, games, and even apps, schools feel the need to channel students' interest in technological advances by providing Coding for Kids learning. With this Coding learning program, children will have the ability to computational thinking, Complex Problem Solving, Limitless Imagination, and Creativity. With this computational ability, children can communicate thoughts in a logical and structured manner, such as instructions given to a computer, so that it can later be used to solve problems with computer-based technological assistance. Prasti et al. (2022) explain coding training at Lalebbata 1 Elementary School, Palopo City using Scratch. This Sctrach application was chosen because when used in learning coding, there is no need to code by typing commands. Coding is done using blocks, all commands used are packaged in blocks. The coding created is arranged based on blocks that have meaning and function and can be executed. The participants can understand coding logic, the use of decision logic, and loop logic. Students can create coding logic and create simple games using the Scratch application. Putro & Astuti (2022) explain that the coding extracurricular program is one of the popular approaches to increasing the interest and talent of UT Dharma Karya students in the field of informatics. The research shows that the use of Java applications in extracurricular coding significantly increases students' interest and talents. Java applications provide an interactive and educational platform for students to learn programming in a fun and engaging way. Using this application also increases student involvement in learning, strengthens understanding of programming concepts, and increases creativity in creating new programs. Arslan & Çelik (2022) describe that coding recognized as a key 21st-century skill, has been incorporated into the curricula of many countries, ranging from preschool to high school, in recent years. Primary school teachers play a crucial role in teaching coding. The study indicates that the Robotics and Coding course supports children's cognitive development and enhances metacognitive skills, such as creative and reflective thinking, while also improving their problem-solving abilities. Furthermore, the study concludes that the Robotics and Coding course should be introduced to students at an early age, suggesting its inclusion in primary school curricula. Anggraeni (2023) explains that based on the analysis of students' critical thinking skills in coding learning at the UPI Laboratory Primary School in coding extracurriculars. It is concluded that the critical thinking abilities of students who took coding extracurriculars were based on the Elementary Clarification indicator. The learning process of students' abilities is quite critical, this is supported by the results of interviews that on average students know how to focus questions, analyze arguments, and ask and answer questions. Dağ et al. (2023) reveal that the unplugged coding course led to a statistically significant improvement in the participants' CT skills, particularly in areas such as algorithmic design, abstraction, evaluation, decomposition, and generalization. Additionally, the findings showed that primary school students' CT skills were not influenced by their socio-demographic factors. Furthermore, a positive and statistically significant correlation was found between the students' CT skills and their collaboration and communication abilities. Then, (Mufidah & Majid, 2024) This study was conducted at SD Negeri Cimanggu Kecil in Central Bogor, focusing on the lack of student engagement and the failure to meet their learning needs in computer education. The N-gain test results indicated that the effectiveness in literacy and numeracy fell under the ineffective category, as the N-gain was less than 40%. The variables were tested using an independent sample t-test, showing homogeneous results. Anjani et al. (2023) explain that Coding for Kids training at Madrasah Diniyyah Sirojussibyan, Bogor, aims to overcome students' lack of access and understanding regarding programming. This program succeeded in introducing basic programming concepts, increasing digital literacy and student skills. Based on the evaluation, 100% of students felt they understood programming better and

felt more comfortable creating their own games or applications. Additionally, 83% of students are interested in learning more about creating games by coding. Despite the challenges, the program is effective in achieving its goal of increasing students' knowledge, skills, and interest in programming and technology.

Based on previous research, several gaps need to be explored further in studies on coding learning at the elementary school level. First, although much research has explored the short-term benefits of teaching coding, little has addressed its long-term impact on students' cognitive and academic development. Second, although various coding platforms such as Scratch and Java have been used, there has been no in-depth comparison of the effectiveness of each platform in different contexts, for example in schools with limited resources. Additionally, there is a lack of research examining teacher readiness and training in teaching coding, which is a key factor in the success of coding programs in elementary schools. Finally, although coding is known to improve critical thinking skills, there is little research that directly links learning to code with the development of deeper critical thinking skills, such as analytical and evaluation abilities. The novelty of this research lies in the comparison between various coding platforms (such as Scratch, Java, and unplugged methods) to assess their effectiveness in improving students' skills. This research also focuses on the long-term impact of coding on students' cognitive development and academic skills, as well as the important role of teacher training in implementing effective coding instruction. By analyzing various school contexts, including those with limited resources, it is hoped that this research will provide new insights into how coding can be taught more inclusively and with a wider impact.

Recent studies also have highlighted the importance of early coding education, but many of them focus solely on the implementation or short-term impacts of individual programs. With the variety of coding platforms now available, there is a critical need to explore not only how coding is taught but also how different platforms may influence students' learning experiences and outcomes. Moreover, while prior research has examined immediate gains in CT or problem-solving, limited attention has been given to the long-term impact of early coding education on students' development.

With the increasing interest and efforts to integrate coding into primary school curricula, various learning platforms such as Scratch, Tynker, and Code.org have been widely used. However, few studies have directly compared the effectiveness of these different coding platforms in enhancing computational thinking skills and their long-term impact on student development. Therefore, this study aims to offer a new contribution by using a comparative approach between several popular coding learning platforms and evaluating their impact not only in the short term but also their potential to sustainably strengthen 21st-century skills. The emphasis on this aspect represents the novelty of this research compared to previous studies that tend to focus on a single platform or short-term effects.

This research fills several gaps that exist in the study of coding education at the elementary school level. Some of these gaps include the long-term impact of learning coding on students' cognitive and academic development, a comparison of the effectiveness of various coding tools (such as Scratch, Blockly, and Java), and a lack of understanding regarding teacher readiness in teaching coding. In addition, this research will also examine the issue of student engagement, especially in schools with limited resources, as well as the relationship between coding and improving deeper critical thinking skills, not only in problem-solving. The uniqueness of this research includes a comparison of coding platforms used, an exploration of the long-term impact of coding on students' cognitive development, and the influence of teacher training on the success of coding programs. Therefore, this research aims to describe how coding is taught at the elementary school level and the benefits obtained from implementing coding based on existing research. Therefore, this study aims to fill these gaps by adopting a comparative approach to investigate multiple coding platforms used in elementary education and assess their respective impacts from both the teachers' and students' perspectives. Uniquely, this study also explores the perceived long-term benefits of coding education, beyond the classroom context. By combining a

comparative lens with a focus on lasting effects, the study contributes new insights to the growing discourse on computational thinking and digital education in primary schools.

Method

Library research is a type of research that relies on collecting data from available literary sources, such as books, journals, scientific articles, research reports, theses, dissertations, and other publications (Fitria, 2023). This research does not involve collecting data directly from the field or through experiments but rather analyzes and studies documents relevant to the topic under study. Library research is often used to explore existing theories, concepts, and findings, as well as to develop a framework or theoretical basis for research. In this research, researchers compile information from various literary sources to analyze and conclude matters related to the research topic, without collecting primary data through direct observation or interviews.

The data for this research were collected through document analysis from various scholarly sources, particularly literature. (Fitria, 2024) This discusses the application of coding in elementary school education, along with its benefits, such as improved computational thinking (CT), creativity, and problem-solving skills. Online academic databases such as Google Scholar, ERIC, and ScienceDirect were utilized to ensure access to reliable, peer-reviewed, and up-to-date resources. Only literature published within the last ten years (2014–2024) was selected to ensure the relevance and contemporaneity of the findings.

The criteria for selecting sources included: (1) publications that specifically address the use of coding or programming in elementary school learning; (2) studies that report on outcomes related to students' skill development (e.g., CT, creativity, collaboration, problem-solving); (3) peer-reviewed or reputable academic sources; and (4) accessibility in full-text format for thorough analysis. To ensure data validity, triangulation was applied by comparing findings from multiple sources addressing similar themes or outcomes. The reliability of each document was assessed based on the credibility of the publication source, the methodology used in the cited studies, and the consistency of findings across different references.

The data analysis technique used in this study is content analysis, which involves identifying, categorizing, and coding themes based on the main research focus. The data were classified as (1) types of coding teaching methods, (2) reported benefits of coding in elementary education, and (3) challenges or limitations in its implementation. Through this method, the research aims to provide a comprehensive understanding of how coding is integrated into elementary education and what outcomes are observed as a result.

Findings and Discussion

Findings

1. Coding

Coding is currently the most important part of developing technology and applications in today's digital era. Coding is an instruction written using a particular programming language (Suryanto et al., 2022). Coding means breaking down data, conceptualizing, and rearranging the data in a new way or arrange in a systematic order. In short, coding is a method for organizing and grouping data (Riasnugrahani & Analya, 2023). Coding is a way to give commands to a computer, application, telephone, or website to do something. Coding is a very important skill to master in the industrial era 4.0 and needs to be taught from an early age (Muklason et al., 2023).

As technology develops, coding has become the main language of the future which is very interesting to learn, especially the introduction of coding from an early age (Hondro, 2024; Sianturi & Butar-Butar, 2024). Learning to code interactively can be an alternative and a plus for children as a place to channel creativity and thinking that can improve skills such as problem-solving solving, being able to think logically and critically. One way to build coding logic in children is to introduce coding interactively to students. In the current digital era, the ability to think computationally is very important for students to have, computational thinking is now a form of literacy (Kristiyanto et al., 2023). Computer programming or coding is one way to improve

computational thinking skills. Coding trains students in problem-solving, and in programming many activities can improve students' logical, systematic, analytical, and creative thinking. However, introducing programming requires a fun way because previously students did not know the names of coding, programming, and algorithms.

Coding in general is writing a set of appropriate codes with certain writing rules (syntax) of the language programming used, such as; Java, Python, C/C++, etc. With that all said we can give you a list of instructions on computer devices according to our purposes. So coding is a technique for translating logic into language computer programming through a special application such as; Codeblocks, Visual Studio Code, Sublime Text, Scratch, etc (Ramadhan et al., 2020).

2. Example of Coding Application

The application of programming languages can be done using a learning application called Scratch (Sukri et al., 2024). Scratch is a visual programming tool designed to introduce programming concepts, especially to children, to train computational thinking in beginners. Scratch allows users to learn programming without having to master complex programming languages. In an era where technology has developed so sophisticatedly, Scratch has become a popular tool in teaching programming to children at the elementary school level. The learning system through programming language with scratch is a form of learning activity using the Science, Technology, Engineering, and Mathematics approach. The STEM approach has become crucial for educators because it must prepare students' abilities to face the modern era which is marked by technological innovation.

Scratch is a tool for helps to develop Apps without writing any code, just by stringing puzzles like this and it's not difficult for to children learn (Supriadi, 2020). This scratch is made especially for elementary to middle school children but can also be used for all age groups. Scratch is a tool that was developed by MIT (the University of Technology best in the world). To operate this Scratch. We only need a computer or laptop with a browser and also internet access for those who want to study online, for those who want to study online offline, just use a computer or laptop and an application scratch installed on the computer.

Scratch in programming learning has a simple programming language that can improve students' computational thinking skills (Putro & Astuti, 2022). Scratch is a programming language that is effective in improving the thinking skills of elementary school students. Learning with Scratch can help students develop creativity, algorithmic thinking, cooperativity, and mathematical thinking (Luthfiyyah et al., 2023). Scratch, a beginner-friendly programming language, helps enhance students' computational thinking. Given the ongoing digital advancements and evolving educational needs, Scratch plays a significant role in elementary schools, serving as a pioneer in promoting digital literacy (Putro & Astuti, 2023).

Scratch application is a coding programming application drag and drop, this application has a set of commands simple that can be understood by a computer, with applications Scratch students who take part in extracurricular learning can create stories, games, and even simple animations (Ramadhan et al., 2020). Looking at digital developments and educational needs now and in the future, Scratch has a quite good role in its application in elementary schools as a pioneer towards digital literacy.

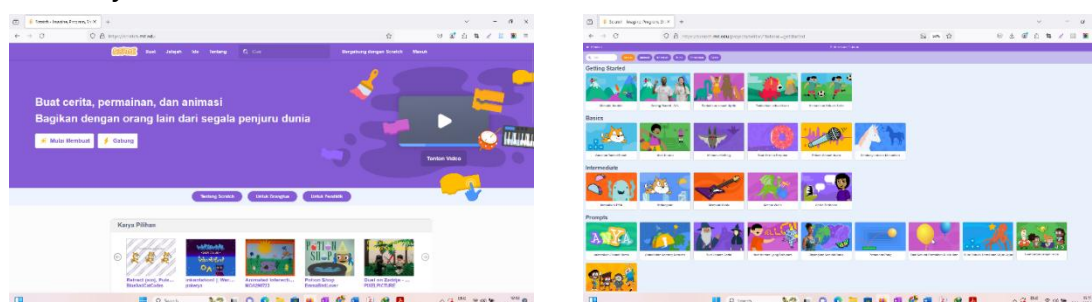


Figure 1a. Scratch for Coding

Source: The View of Scratch for Coding (<https://scratch.mit.edu/>)

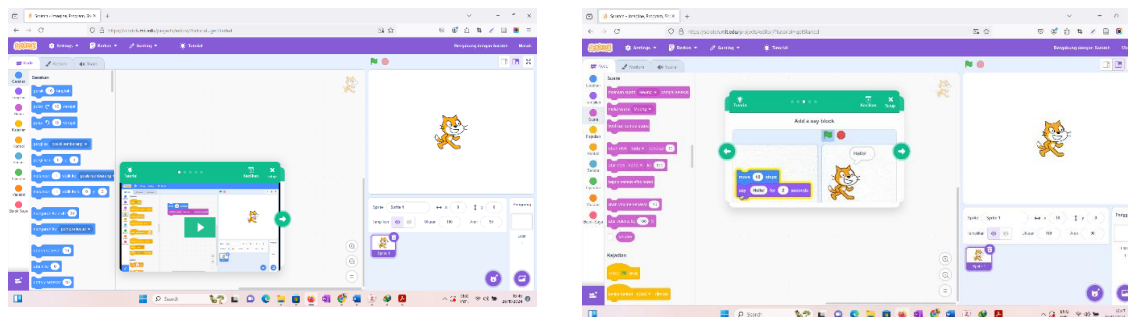


Figure 1b. Scratch for Coding

Source: The View of Scratch for Coding (<https://scratch.mit.edu/>)

To start using Scratch for simple coding, the first step is to visit the Scratch website. Once there, if you don't have an account yet, you can sign up or log in to save your projects. Although you can use Scratch without an account, having one allows you to save and share your projects. After logging in, create a new project by clicking the "Create" button at the top of the page. This will take you to the Scratch editor, where you can start writing code. Here, you can begin by choosing or creating a sprite, which is the character you will program to move or interact with within your project. You can either pick an existing sprite, such as the "Cat" sprite from the library, or create a new one as needed. Scratch uses a block-based coding system, resembling puzzle pieces, where each block represents an instruction or command. On the left side of the screen, you will see categories like Motion, Looks, Sound, Events, Control, and Sensing. To create a simple program, for example, to make a sprite move, select a block from the Motion category, such as move 10 steps. Drag this block into the middle area of the editor. To make the program run when the green flag is clicked, drag the when green flag clicked block from the Events category and connect it with the move 10 steps block. You can also add other interactions or effects to make the program more interesting. For example, to add sound when the sprite moves, select a block from the Sound category, then drag the play sound [sound] until the done block into your script area. You can choose a sound from the library or upload your own. This way, you can create a more dynamic program by combining sound and motion. To control the sequence of commands, Scratch provides control blocks that allow you to manage the flow of the program. Select a block from the Control category, then drag the repeat or wait block to delay certain commands. For instance, you can use the repeat 10 block and place the move 10 steps block inside it, making the sprite move 10 steps ten times. These control blocks are useful for creating more complex interactions. Once you've finished creating your code, click the green flag at the top left corner of the screen to test your program. If it works as expected, you can proceed to save and share your project. Click the File button in the top left to save your project and give it a name and description. Once saved, you can choose to share your project with the Scratch community so others can view it and provide feedback. As an example, a simple project you can create on Scratch is to make the sprite move and say "Hello." To do this, first, select the When green flag clicked block from the Events category, then add the Say "Hello!" for 2 seconds block from the Looks category, and the Move 10 steps block from the Motion category. By following these steps, you can create a simple coding project and develop basic programming skills.

3. Coding for Elementary Students

Currently, more and more elementary schools (SD) in Indonesia are offering coding extracurriculars, driven by the importance of digital skills in an era of rapid technological development. In Indonesia, more and more elementary schools (SD) are starting to teach coding, both as a lesson in the official curriculum and as an extracurricular activity. As technology advances and the importance of digital skills in the 21st century, many schools are realizing that teaching coding early can help students develop computational thinking, creativity, and problem-

solving skills that are important for their future. In extracurricular activities, coding allows students to solve problems in new ways because it is an internal game computers use a visual programming language that will train students' logic and thinking concepts so that students get used to it to solve problems systematically. Students get more smart in using technology. Not just only understand how to play applications on cellphones, but students can also create their own games or applications. Learning coding will encourage students' creativity because by coding they can create a game according to the ideas they have (Ramadhan et al., 2020).

Several factors encourage many elementary schools to offer coding extracurriculars, including government policy through the Ministry of Education and Culture (Kemdikbud) encouraging the application of technology in education, including the introduction of coding at the elementary school level. Apart from that, the use of technology in learning is also one of the reasons why many schools are starting to realize the importance of integrating technology in the teaching and learning process. No less important, the challenges of Industry 4.0 which increasingly dominates the world of work also make schools prepare their students to be ready to compete in a world that is increasingly connected to technology. However, although the number of elementary schools offering extracurricular coding is increasing, implementation still varies depending on location and the resources available at each school. Large schools in big cities tend to be quicker to adopt this program, while in more remote areas, there is still relatively little implementation.

Learning coding is very important for elementary school students because it can help them develop logical, creative, and problem-solving thinking skills. Coding teaches students to break down large problems into smaller, more manageable parts, thereby improving their analytical thinking skills. Additionally, coding also provides a space for students to create their digital projects, such as games or animations, which stimulates creativity. Given the increasing importance of digital skills in the future, learning to code at the elementary level can prepare students to face the challenges and opportunities that exist. Additionally, coding also teaches students to work in teams, improving their collaboration and communication skills. The technical skills gained through coding are very useful in various fields, be it science, mathematics, art, or design. The coding process that involves experimentation and error also helps students develop mental resilience and the ability to not give up easily when faced with challenges. By learning coding appropriately, elementary school students can gain skills that are useful in various aspects of their lives and careers in the future.

Elementary students need to learn coding because this skill has many benefits that can help them in various aspects of development. Coding teaches students to think systematically and break down large problems into smaller, more manageable parts, which improves logical and analytical thinking skills. Additionally, coding can also stimulate students' creativity, as they can create projects such as games, animations, or applications that allow them to express their ideas in new ways. Mastering coding from an early age also prepares students to face future challenges, where digital skills are increasingly important in the world of work. Apart from that, the process of learning to code teaches perseverance and problem-solving, because students will learn not to give up easily when facing challenges. Coding is also part of a broader set of digital skills, which are needed in various fields, whether in science, technology, arts, or mathematics. Thus, learning to code from an early age gives elementary school students an edge in facing rapid technological developments and prepares them for the future.

Implementing simple coding for elementary school students can be done using various tools and approaches that are easy to understand and fun. Here are some ways that can be implemented:

- a. Using a Visual Coding Platform (Like Scratch). Scratch is a visual coding platform designed especially for kids. Here, students can create animations, games, or interactive stories by dragging and arranging blocks of code. This approach teaches basic concepts such as sequence, repetition, and branching without requiring text programming.
- b. Making Simple Games. Teaching students to create simple games using Scratch or similar applications can be very interesting. For example, students can create "object click" or "character adventure" games, which teach them the logic of repetition and interaction.

- c. Introduction to Algorithms and Logic. Using games or activities that emphasize a sequence of steps, such as guessing games or "follow the pattern", can help students understand the basic algorithm and logic concepts. This is the basis of more complex programming later.
- d. Use of Coding Learning Applications for Children. Some apps like "Tynker", "Lightbot", and "Code.org" offer a fun and interactive introduction to coding. These applications use puzzles and challenges that encourage students to think logically while learning to create simple programs.
- e. Basic Robotics. The introduction of robotics with tools allows elementary students to learn coding in a more practical and fun way. They can program robots to move, complete certain tasks, or interact with sensors.
- f. Activities Without a Computer (Unplugged Coding). For an introduction to coding without digital devices, activities such as "human coding" can be performed. Students can role-play as computers following simple instructions from their friends, for example walking in a certain direction or arranging objects in a given order. It helps teach basic concepts of algorithms and logic without having to use electronic devices.

With the right methods and easy-to-use tools, simple coding can be implemented effectively at the elementary school level, providing a strong foundation for future programming skills.

4. Scratch

Scratch is an application that allows users to create stories, games, and interactive animations that can be shared over the Internet (Juliyana et al., 2024). Learning media created using scratch can be categorized as audio-visual media because it is media that can be heard and seen so that it can help students in learning activities which function to clarify or make it easier for students to understand the material through games in learning. Scratch also allows students to create a wide variety of games ranging from simple to complex, with an easy-to-use interface and easy-to-understand visual code blocks (Pramono, 2024). There are several advantages of using Scratch coding for elementary students.

Table 1. Summary of Research Findings on the Use of Scratch in Elementary School Coding Education

No	Author	Findings
1.	Wulandari et al. (2021)	With the Scratch application, students can learn important strategies for solving problems, designing work, and communicating ideas by programming. The majority of respondents already know about computational thinking skills and think that computational thinking skills can be trained through Scratch. Respondents also feel that Scratch is suitable, easy, and interesting to use for elementary school students. Then respondents thought that other skills that could be trained through scratch included: Accuracy, sensitivity to color, and critical thinking skills.
2.	Indrawan et al. (2021)	After participating in the training, students from SD Umeanyar showed improved understanding and gained experience in using computers. They also learned to create animations and simple games using the Scratch application. Initially, many students were unfamiliar with computers and software, finding it challenging to operate such tools. Continued guidance and support from the surrounding environment are essential for the students to fully benefit from the training. Therefore, collaboration between village leaders and schools is crucial to support students through various educational programs that enhance their knowledge and prepare them for higher levels of learning. Universities can also play a vital role in helping to implement these initiatives.

3.	Putro & Astuti (2022)	Application of Scratch in elementary school students' coding learning. The implementation of scratch at SD Plus Muhammadiyah Brawijaya, Mojokerto City is implemented in the form of learning coding for students. The material used in learning coding includes algorithms. Students use Scratch as an interactive and interesting coding learning medium because by using Scratch students can create animated videos, quizzes, games, and stories. By using the Technology Acceptance Model (TAM) it can be seen that Scratch is an application that can support coding learning activities. Even though there are still difficulties with indicators of implementation difficulty and feature complexity, when coding learning activities take place students seem to understand quite easily how to apply scratch. Apart from that, the application of scratch in coding learning also really supports the realization of an independent curriculum, namely training and improving national computational thinking abilities.
4.	Sholeh et al. (2022)	coding training at PAMS (Yogyakarta Syuhada Mosque Children's Education Institute). The application used is to use Scratch. This application was chosen, because using Scratch in learning computer programs does not require creating computer programs by typing commands. The computer program process is carried out using blocks, all the commands used are packaged in blocks. The computer programs created are arranged based on blocks that have meaning and function and can be executed. The use of blocks is similar to Lego games and this is certainly following the character of children who enjoy playing games. The participants can understand the logic of computer programs, the use of decision logic, and loop logic. Children can create logical computer programs to create fish-clicking games in the aquarium, catching snowballs, avoiding rolling eggs, and catching falling apples.
5.	Santoso & Churniawan (2022)	The results of the mentoring and training activities of the Scratch Jr program. For approximately 5 (five) months it has resulted in an increase in student competency in basic programming. The basic programming applied is programming in making games for elementary school (SD) children. The increase in competency after receiving training is quite significant, with an average of 8 (eight) points on a scale of 1 –10. Use of tools in ScratchJr programming. Which is user-friendly and makes it easier for students to absorb the material provided.
6.	(Luthfiyyah et al., 2023)	Codong learning activities can increase students' knowledge and interest in programming. Apart from that, meta-analysis shows that the use of Scratch in learning has a positive effect size on elementary school students' computational thinking skills, especially in the dimensions of concepts and practices. The research method used is experimentation using the pre-experimental design form with a one-shot case study design pattern. Thus, scratch has been proven to be an effective tool for improving elementary school students' computational thinking skills.

The table above summarizes important findings demonstrating the benefits of Scratch as a coding learning media for elementary students. The advantages of Scratch, from ease of use, and appeal to children, to improvements in computational thinking and logic skills, are consistently shown across various studies. Moreover, collaboration between schools and the community plays

an important role in supporting successful learning with Scratch. This makes Scratch an effective and enjoyable platform for introducing coding to children.

Using the Scratch application in learning coding provides various benefits for students, especially in improving computational thinking skills and programming understanding. Wulandari et al. (2021) found that Scratch helps students learn problem-solving strategies, design work, and communicate through programming, as well as practice other skills such as accuracy, color sensitivity, and critical thinking. Indrawan et al. (2021) reported that after participating in the training, Umeanyar Elementary School students showed an increased understanding of computer use and were able to create animations and simple games with Scratch. They also emphasized the importance of ongoing support from the local environment to maximize the benefits of training. Putro & Astuti (2022) revealed that Scratch supports interactive and interesting coding learning at SD Plus Muhammadiyah Brawijaya Mojokerto, where students can create animated videos, quizzes, games, and stories, although some students experience difficulties with the complexity of the features. Sholeh et al. (2022) also found that training using Scratch at PAMS Yogyakarta made students understand programming logic in a fun way, using programming blocks similar to Lego games, which suited the children's characters. Santoso & Churniawan (2022) reported that the use of Scratch Jr in training for five months succeeded in improving the basic programming competencies of elementary school students, especially in game creation, thanks to user-friendly tools. Lastly, Luthfiyyah et al. (2023) show that Scratch can increase elementary school students' knowledge, interest, and computational thinking skills, especially in the conceptual and practical dimensions. Overall, Scratch has proven effective in introducing programming, increasing creativity, and motivating students to learn more about technology.

Discussion

The findings reveal that coding has become an essential literacy in the digital era, aligning with global trends in 21st-century education. The integration of coding in early education reflects the shift from traditional literacies to digital and computational literacies, which are critical in developing problem-solving, creativity, logical reasoning, and analytical skills. The ability of students to interact with technology not just as users but as creators supports the notion of "student as producer," a key concept in digital pedagogy. Moreover, this supports the broader framework of constructivist learning, where students actively build knowledge through meaningful activities. Introducing coding at the elementary level enables young learners to engage in computational thinking—a skill considered as fundamental as reading or math in the current era (Kristiyanto et al., 2023). However, while the importance of coding is widely acknowledged, challenges remain in its effective implementation, particularly in schools lacking resources or trained educators.

The use of Scratch as a visual programming tool demonstrates how digital pedagogy can be made accessible and engaging for children. Its block-based interface aligns with the principles of scaffolded learning, allowing learners to understand complex programming concepts through visual and interactive tasks without being overwhelmed by syntax. This supports Papert's constructionism theory, where learners construct knowledge best when actively engaged in creating a meaningful product. Scratch not only enhances computational thinking but also fosters creativity, algorithmic reasoning, and teamwork, making it a powerful educational tool within STEM-based curricula. The fact that Scratch is widely accessible, both online and offline, further supports equitable access to digital learning tools. However, despite its simplicity, Scratch still requires adequate teacher guidance and structured activities to maximize its pedagogical potential. There is also a risk that students may remain at a superficial level of coding understanding unless guided to transfer block-based logic into text-based programming languages later on.

In terms of implementation in elementary schools, the increasing inclusion of coding as part of extracurricular activities signifies a progressive step toward digital readiness. However, this

implementation remains uneven. While urban schools may quickly adopt such programs due to better infrastructure and support, rural schools face significant barriers such as lack of internet access, limited devices, and insufficient teacher training. This creates a digital divide, challenging the equity principle in education. Nonetheless, government support through Kemdikbud's policy shows alignment with national strategies in educational digitalization. Integrating coding through both online tools (like Tynker or Code.org) and offline approaches (such as unplugged coding) reflects a flexible pedagogical model that can adapt to diverse school contexts. These methods reflect blended learning principles and Universal Design for Learning (UDL), which promote inclusivity. However, a limitation arises in sustaining student interest and progression in coding, as most current applications focus on isolated tasks rather than long-term learning pathways that develop from basic to advanced programming skills.

Currently, many elementary schools (SD) in Indonesia are starting to teach coding, both as a lesson in the official curriculum and as an extracurricular activity. As technology advances and the importance of digital skills in the 21st century, many schools may realize that teaching coding early can help students develop computational thinking, creativity, and problem-solving skills that are important for their future. Based on the findings, there are several advantages to learning coding in elementary students. Using Scratch in learning coding has been proven to provide many benefits for students. Scratch helps improve computational thinking skills, programming understanding, and creativity. Wulandari et al. (2021) found that Scratch trains problem-solving strategies and skills such as accuracy and critical thinking. Indrawan et al. (2021) reported that students were able to create animations and games after training, with the support of the surrounding environment being important. Putro & Astuti (2022) and Sholeh et al. (2022) stated that Scratch supports interactive and fun learning, and makes it easier to understand programming logic. Santoso & Churniawan (2022) reported an increase in basic programming competency with Scratch Jr, while Luthfiyyah et al. (2023) show that Scratch is effective in improving elementary school students' computational thinking skills. Overall, Scratch supports the development of students' technology skills in an engaging and easy-to-understand way.

In conclusion, the integration of coding into elementary education provides promising opportunities for enhancing digital literacy and aligning with contemporary pedagogical frameworks such as constructivism, STEM education, and digital pedagogy. However, to realize its full potential, several limitations must be addressed—particularly in teacher training, resource availability, and continuity of learning. Further research is needed to explore how coding can be sustained in long-term learning trajectories, and how its implementation can be scaled equitably across diverse educational settings.

Conclusion and Suggestion

Conclusion

In conclusion, integrating coding into elementary education offers significant potential for enhancing students' digital literacy and aligns with modern pedagogical frameworks such as constructivism, STEM education, and digital pedagogy. Scratch, as a visual and beginner-friendly coding platform, has been shown to improve students' computational thinking, programming understanding, creativity, and problem-solving skills while supporting fun, interactive learning environments. Students also benefit from creating animations and games, which boost their engagement and confidence. However, to fully harness these benefits, challenges such as teacher training, resource availability, and continuity of instruction must be addressed. This study contributes to the broader discourse by offering a comparative and longitudinal perspective on how students engage with coding in both formal and informal educational settings, highlighting the value of Scratch not only as a teaching tool but also as a platform for sustained learning and student agency. It underscores the importance of context in shaping learning outcomes and advocates for blended or hybrid models that combine structured teaching with flexible, creative exploration. These findings offer insights for educators and curriculum designers to scale and sustain coding education equitably across diverse school environments, while future research

should further investigate developmental trajectories and the long-term impacts of early coding education on student learning.

Suggestion

For teachers, it is important to start coding lessons in the classroom with simple basics, such as algorithms and programming logic, using a child-friendly platform like Scratch. Interactive approaches are highly recommended, such as inviting students to create fun small projects, such as simple games or animations. Teachers also need to teach the importance of collaboration, encourage students to work in groups, and emphasize the learning process rather than just focusing on the result. Integrating coding with other subjects, such as mathematics or science, can also make learning more relevant and interesting. In addition, providing sufficient support and repetition of exercises will help students gradually understand the material better.

Besides, for students, it is recommended to start with small, simple steps, such as making characters move or adding sound to their projects. It's also important for students to experiment with code and not be afraid to make mistakes because every mistake is an opportunity to learn. If confused, students must have the courage to ask a teacher or friend. Understand the logic behind the code to solve problems better and share projects with friends to increase self-confidence. Lastly, students must be patient in facing challenges and mistakes, because coding is a skill that takes time and patience to master.

References

- Anggraeni, -. (2023). Studi Kasus Terhadap Kemampuan Berpikir Kritis Siswa Pada Pembelajaran Coding di Sekolah Dasar Laboratorium UPI Kampus Cibiru: Penelitian Studi Kasus Pada Siswa yang Mengikuti Ekstrakurikuler Coding di Sekolah Dasar [Other, Universitas Pendidikan Indonesia]. <https://repository.upi.edu>
- Anjani, D., Bachtiar, Y., & Novianti, D. (2023). Pelatihan Coding For Kids Menggunakan Scratch Sebagai Upaya Meningkatkan Kecakapan Digital Bagi Siswa Madrasah Diniyyah Sirojussibyan, Bogor. *Jurnal Pengabdian Mandiri*, 2(7), 1439–1448.
- Arfé, B., Vardanega, T., & Ronconi, L. (2020). The effects of coding on children's planning and inhibition skills. *Computers & Education*, 148, 103807. <https://doi.org/10.1016/j.compedu.2020.103807>
- Arslan, S., & Çelik, Y. (2022). Primary School Teachers' and Students' Views about Robotic Coding Course. *African Educational Research Journal*, 10(2), 178–189.
- Budiman, G., Purnamasari, R., Rustam, Wahidin, & Basudewa, M. I. (2023). Pembelajaran Coding for Kids untuk Siswa dan Siswi di Sekolah Hamidah Sampurna Kabupaten Bandung. *Almufi Jurnal Pengabdian Kepada Masyarakat*, 3(2), 83–89.
- Dağ, F., Şumuer, E., & Durdu, L. (2023). The effect of an unplugged coding course on primary school students' improvement in their computational thinking skills. *Journal of Computer Assisted Learning*, 39(6), 1902–1918. <https://doi.org/10.1111/jcal.12850>
- Falloon, G. (2016). An analysis of young students' thinking when completing basic coding tasks using Scratch Jnr. On the iPad. *Journal of Computer Assisted Learning*, 32(6), 576–593. <https://doi.org/10.1111/jcal.12155>
- Fitria, T. N. (2023). A Library Research in English Education Research: A Guidance for Researchers in Writing Non-Research Articles. *Prosiding Seminar Nasional & Call for Paper STIE AAS*, 6(1), Article 1. <https://prosiding.stie-aas.ac.id/index.php/prosenas/article/view/266>
- Fitria, T. N. (2024). Qualitative Research Method in Education Field: A Guide for Researchers, Lecturers and Students (Metode Penelitian Kualitatif di Bidang Pendidikan : Panduan bagi Peneliti, Dosen dan Mahasiswa). *Eureka Media Aksara*. <https://repository.penerbiteurka.com/publications/578908/>
- Hondro, R. K. (2024). Peningkatan Pemahaman Siswa dalam Belajar Coding Interaktif Melalui Pengenalan Aplikasi Codecademy. *BERBAKTI: Jurnal Pengabdian Kepada Masyarakat*, 1(01), 18–22.

- Indrawan, G. B., Octavia, I. G. A., Saputra, G. A. A. P., Adi, I. G. K., Andrayuga, I. G. L. A., & Dewi, L. J. E. (2021). Pelatihan scratch programming untuk anak-anak SD Umeanyar. *Unri Conference Series: Community Engagement*, 3, 235–241. <https://doi.org/10.31258/unricsce.3.235-241>
- Juliyana, G., Boty, M., & Jadidah, I. T. (2024). Pengembangan Media Pembelajaran Berbasis Game Based Learning Menggunakan Scratch Pada Pembelajaran IPAS Di SD Negeri Mekar Sari Kabupaten Musi Banyuasin. *Jurnal Pendidikan, Sains Dan Teknologi*, 3(2), 282–289. <https://doi.org/10.47233/jpst.v3i2.1651>
- Kristiyanto, A., Romansyah, A., Septiani, S., Nazmudin, M., Fadilah, M. Y., Lestari, T., Malkan, F., & Safitri, H. (2023). Pengenalan Koding dan Program Scratch Bagi Siswa MA Mathla'ul Anwar Baros Upaya Meningkatkan Literasi Digital. *ABDIMASKU : JURNAL PENGABDIAN MASYARAKAT*, 6(2), 440–446. <https://doi.org/10.33633/ja.v6i2.1132>
- Lestari, I., Arifin, S. P., Widyasari, Y. D. L., Zain, M. M., & Rachmawati, H. (2023). The Coaching of Dunia Coding Program to Improve Computational Thinking Ability at As Shofa Junior High School Pekanbaru : *Dinamisia : Jurnal Pengabdian Kepada Masyarakat*, 7(1), 136–146. <https://doi.org/10.31849/dinamisia.v7i1.12455>
- Lloyd, M., & Chandra, V. (2020). Teaching coding and computational thinking in primary classrooms: Perceptions of Australian preservice teachers. *Curriculum Perspectives*, 40(2), 189–201. <https://doi.org/10.1007/s41297-020-00117-1>
- Luthfiyyah, R. Z., Nurhikmah, J., Najayanti, N., Alifah, A. N., Irsalina, S., Nabilah, S., & Alindra, A. L. (2023). Pengaruh Media Pembelajaran Berbasis Scratch Terhadap Motivasi Belajar Siswa Kelas IV di Salah Satu Sekolah Dasar Purwakarta. *Innovative: Journal Of Social Science Research*, 3(6), 5722–5731.
- Mannova, B. (2022). Teaching Coding in Schools. *EDULEARN22 Proceedings*, 5961–5967. 14th International Conference on Education and New Learning Technologies. <https://doi.org/10.21125/edulearn.2022.1399>
- Martin, D. A., Curtis, P., & Redmond, P. (2024). Primary school students' perceptions and developed artefacts and language from learning coding and computational thinking using the 3C model. *Journal of Computer Assisted Learning*, 40(4), 1616–1631. <https://doi.org/10.1111/jcal.12972>
- Maura, M. F., & Sutabri, T. (2024). Analisis Penggunaan Platform Replit dalam Pembelajaran Coding: Studi Kasus Terhadap Tingkat Keterlibatan Pengguna dan Efektivitas Pembelajaran. *IJM: Indonesian Journal of Multidisciplinary*, 2(3), 139–145.
- Mills, K. A., Cope, J., Scholes, L., & Rowe, L. (2024). Coding and Computational Thinking Across the Curriculum: A Review of Educational Outcomes. *Review of Educational Research*, 00346543241241327. <https://doi.org/10.3102/00346543241241327>
- Mufidah, T. H., & Majid, N. W. A. (2024). Pengaruh Peningkatan Computational Thinking Siswa Kelas 5 Melalui Pembelajaran Dasar Coding. *Buletin Literasi Budaya Sekolah*, 22–37. <https://doi.org/10.23917/blbs.v6i1.4231>
- Muklason, A., Riksakomara, E., Mahananto, F., Djunaidy, A., Vinarti, R. A., Anggraeni, W., Nurita, R. T., Utamima, A., Fauzia, R., Theresia, L. W., Fikri, M. A., Propitadewa, H., Habibah, J. H., Prasetyo, J. D., Permatasari, S. T. I., Risnina, N. N., Tsaniyah, N. D., & Maulana, M. D. (2023). Coding for Kids: Pengenalan Pemrograman untuk Anak Sekolah Dasar sebagai Literasi Digital Baru di Industri 4.0. *Sewagati*, 7(3), 393–404. <https://doi.org/10.12962/j26139960.v7i3.506>
- Popat, S., & Starkey, L. (2019). Learning to code or coding to learn? A systematic review. *Computers & Education*, 128, 365–376. <https://doi.org/10.1016/j.compedu.2018.10.005>
- Pramono, A. (2024). Learning Graphic Design and Basic Programming with the Scratch Application for Extracurricular Computer Students at SD Muhammadiyah 4 Surabaya. *KANGMAS: Karya Ilmiah Pengabdian Masyarakat*, 5(2), 54–60. <https://doi.org/10.37010/kangmas.v5i2.1596>
- Prasti, D., Rusdi, M. I., & Putri, I. K. (2022). Coding For Kids. *Abdimas Langkanae*, 2(2), 170–180. <https://doi.org/10.53769/jpm.v2i2.84>

- Price, C. B., & Price-Mohr, R. M. (2018). An Evaluation of Primary School Children Coding Using a Text-Based Language (Java). *Computers in the Schools*.
<https://www.tandfonline.com/doi/abs/10.1080/07380569.2018.1531613>
- Putro, Y. T. M., & Astuti, R. (2022). Penerapan Scratch dalam Pembelajaran Coding Siswa Sekolah Dasar. *Emergent: Journal of Educational Discoveries and Lifelong Learning (EJEDL)*, 1(4), 21–21. <https://doi.org/10.47134/emergent.v1i4.37>
- Putro, Y. T. M., & Astuti, R. (2023). Implementation of Scratch in Learning Coding for Elementary School Students: Penerapan Scratch dalam Pembelajaran Coding Siswa Sekolah Dasar. UMSIDA Preprints Server. <https://doi.org/10.21070/ups.1671>
- Ramadhan, D. R. P., Rosyada, A. Q., Marliza, W., Kasatri, D. E. P., & Yuliana, I. (2020). Pengaruh Ekstrakurikuler Coding Pada Siswa Sekolah Dasar Guna Meningkatkan Computational Thingking di Sekolah Al-Azhar Syifa Budi Solo. *Buletin Literasi Budaya Sekolah*, 2(1). <https://doi.org/10.23917/blbs.v2i1.11616>
- Riasnugrahani, M., & Analya, P. (2023). *Buku Ajar: Metode Penelitian Kualitatif*. Ideas Publishing.
- Rich, P. J., Browning, S. F., Perkins, M., Shoop, T., Yoshikawa, E., & Belikov, O. M. (2019). Coding in K-8: International Trends in Teaching Elementary/Primary Computing. *TechTrends*, 63(3), 311–329. <https://doi.org/10.1007/s11528-018-0295-4>
- Santoso, R., & Churniawan, A. D. (2022). Pendampingan Program Ekstrakurikuler Pembelajaran Coding di SD Anak Panah Kenjeran. *Jurnal Layanan Masyarakat (Journal of Public Services)*, 6(1), 109–118. <https://doi.org/10.20473/jlm.v6i1.2022.109-118>
- Sholeh, M., Pradnyana, I. W. J., & Ridhoni, I. W. (2022). Menumbuhkan Minat Anak-Anak dalam Belajar Koding dengan Menggunakan Aplikasi Scratch. *Abdiformatika: Jurnal Pengabdian Masyarakat Informatika*, 2(2), 72–79. <https://doi.org/10.25008/abdiformatika.v2i2.151>
- Sianturi, C. F., & Butar-Butar, L. H. O. (2024). Peningkatan Pemahaman Siswa Terhadap Coding Program Melalui Aplikasi Primaindisoft. *BERBAKTI: Jurnal Pengabdian Kepada Masyarakat*, 1(01), 1–5.
- Sukri, I. H., Amilia, W., Yeni, F., & Rahmayanti, E. (2024). Studi Komparatif Penggunaan Aplikasi Scratch dan Aplikasi Blockly pada Mata Pelajaran Informatika Siswa SD Islam Al-Azhar 32 Padang. *Indo-MathEdu Intellectuals Journal*, 5(4), 4399–4410. <https://doi.org/10.54373/imeij.v5i4.1569>
- Supriadi, D. (2020). Coding Scratch Basic—Scratch 3. Saung Coding.
- Suryanto, A. A., Arifia, A., Nurlifa, A., Muqtadir, A., Amaluddin, F., Haryoko, A., & Wijayanti, A. (2022). Pelatihan Pengenalan Coding Coding Bagi Guru Sd Menggunakan Scratch Jr. *Jurnal Pengabdian Pada Masyarakat METHABDI*, 2(2), 117–119. <https://doi.org/10.46880/methabdi.Vol2No2.pp117-119>
- Vlahu-Gjorgievska, E., Videnovik, M., & Trajkovik, V. (2018). Computational Thinking and Coding Subject in Primary Schools: Methodological Approach Based on Alternative Cooperative and Individual Learning Cycles. 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), 77–83. <https://doi.org/10.1109/TALE.2018.8615334>
- Wei, X., Lin, L., Meng, N., Tan, W., Kong, S.-C., & Kinshuk. (2021). The effectiveness of partial pair programming on elementary school students' Computational Thinking skills and self-efficacy. *Computers & Education*, 160, 104023. <https://doi.org/10.1016/j.compedu.2020.104023>
- Wulandari, W., Haftani, D. A., Ridwan, T., & Putri, D. I. H. (2021). Pemanfaatan Platform Scratch dalam Pembelajaran Koding di Sekolah Dasar untuk mengasah kemampuan Computational Thinking pada Siswa. *Renjana Pendidikan: Prosiding Seminar Nasional Pendidikan Dasar*, 2(1), 495–504.