Received: June 21, 2026
Revised: July 16, 2026
Accepted: July 31, 2026
Published: September 30, 2026

Elicit Journal of Education Studies
eISSN: 3121-8474
pISSN: 3093-0472

https://elicitpublisher.com

Empirical Paper

Development of a Scratch-Based Learning Application for Arithmetic Operations: An ADDIE Model Approach

Novia Setyoningtyasa[1], Sri Utaminingsih[2*], Ahmad Hariyadi[3]

Affiliations
[1]Faculty of Teacher Training and Education, Muria Kudus University, Indonesia

ORCID Identifiers

Sri Utaminingsih
0000-0002-2613-0052

Ahmad Hariyadi
0000-0001-8344-6718

DOI
10.65820/ejes-16vol2-issue2-2026

Abstract

Purpose: This study developed a Scratch-based learning application for teaching the four basic arithmetic operations using the Analysis, Design, and Development phases of the ADDIE model. It also examined stakeholders' perceptions of the application's interface, instructional content, and game mechanics before classroom implementation.

Methodology: A design and development research approach was adopted using the ADDIE instructional design framework. A Scratch-based application was developed, and qualitative data were collected through semi-structured interviews with four purposively selected stakeholders comprising a university lecturer, two Master of Education students, and an elementary school principal. Responses were analysed descriptively by identifying areas of agreement and disagreement.

Results: The Analysis stage identified declining student motivation as the primary instructional problem, guiding the application's visual design, game structure, and feedback mechanisms. Respondents agreed that the interface was intuitive and that colour and image-based visualisation improved content clarity. They also recommended teacher guidance, progressive learning tasks, peer collaboration, and improvements in device compatibility, reward mechanisms, and code maintainability before classroom implementation.

Novelty and Contribution: The study extends Scratch-based mathematics research by documenting the instructional development process prior to implementation and highlighting technical deployment considerations that have received limited attention in previous studies.

Practical and Social Implications: The findings provide practical guidance for educators and instructional designers developing Scratch-based mathematics applications and support the integration of interactive digital tools into elementary mathematics through systematic instructional design and stakeholder-informed refinement.

Keywords: Scratch; ADDIE model; mathematics education; arithmetic operations; educational game development

*Corresponding author
E-mail address: sri.utaminingsih@umk.ac.id

How to cite this article:

Setyoningtyas, N., Utaminingsih, S., & Hariyadi, A. (2026). Development of a Scratch-based learning application for arithmetic operations: An ADDIE model approach. *Elicit Journal of Education Studies, 2*(2), 200-209. https://doi.org/10.65820/ejes-16vol2-issue2-2026

1 Introduction

Learning mathematics has grown more demanding as classrooms are asked to prepare students for an increasingly technology-mediated world. Yet, one of its most basic components, operation count, encompassing addition, subtraction, multiplication, and division, remains a persistent source of difficulty for many students (Luneta, 2022; Mangarin & Caballes, 2024). Meeting the demands of Industry 4.0 calls for an education system capable of producing learners who are creative, innovative, and competitive, in part by making deliberate use of technology as an instructional tool (Hernandez-de-Menendez et al., 2020). Digital literacy has become part of what this requires, and among its components, the ability to understand programming or coding concepts has taken on particular importance for young learners (Maulidha & Utama, 2025).

Coding education for children is not simply a matter of teaching syntax. It introduces learners to a way of thinking, breaking problems into steps, recognising patterns and testing solutions that extends well beyond computer science itself (Bers, 2020). Scratch, a block-based visual programming platform, has become a common entry point for this kind of instruction: it lets users build interactive stories, animations, games, and simulations by dragging and connecting graphical blocks rather than writing conventional code (Batni et al., 2025). Because it removes the syntax barrier that often discourages young or novice learners, Scratch has been used not only to teach programming itself but also to support subject-area learning across several domains (Hu et al., 2021).

Mathematics is one such domain. Operation count, in particular, depends on students developing logical, systematic, and analytical habits of thought, as these underpin the cognitive skills needed for more advanced mathematics later on (Kodirun et al., 2025). Educational games and interactive applications can make this kind of learning more concrete: technology of this kind gives students an interactive and visual way to engage with material, helping abstract mathematical ideas become easier to grasp (Tarisna et al., 2025). Scratch has already been used to build mathematics learning games in earlier work (Bernard & Setiawan, 2020). The platform has also supported instructional applications in adjacent technical domains, such as teaching electronic components in design and technology courses (Mazlan et al., 2023). What remains less established in this line of work is a Scratch-based application built specifically around the four basic arithmetic operations, together with a documented account of how such an application is developed and reviewed before it reaches a classroom.

This study addresses that gap. It reports the development of a Scratch-based application for learning mathematics operation count, built and refined using the ADDIE model, and documents the feedback gathered from a university lecturer, graduate education students, and a school principal on the application's interface, content, and game design. The aim is not only to produce a working application but also to leave a clear record of the development process that other researchers and educators can examine, critique, or adapt when designing similar tools for elementary mathematics instruction. To address this gap, the study is guided by the following research questions.

RQ1. How is the ADDIE model applied in the analysis, design, and development of a Scratch-based application for learning mathematics operation count?

RQ2. How do relevant stakeholders evaluate the resulting application's interface design and content delivery?

RQ3. What improvements to the application's game mechanics and technical performance do these stakeholders identify as necessary before classroom use?

2 Literature Review

Learning Mathematics and Operation Count

Operation count, addition, subtraction, multiplication, and division sit at the foundation of primary mathematics, and mastering them thoroughly shapes how well students handle the mathematics that follows (Wu, 2026). Despite its basic status, this is one of the areas where young learners struggle most. Rejeki et al. (2024) found that second-grade students frequently misread multiplication as an isolated procedure rather than as repeated addition, a misunderstanding that is then carried forward into later errors with division. Their study also found that teachers themselves sometimes reinforce this confusion, for instance by treating "3×4" and "4×3" as procedurally different rather than equivalent. Difficulties of this kind are not limited to multiplication: errors in addition and subtraction have similarly been traced to an incomplete grasp of place value and number structure, rather than to carelessness (Mumpuni & Mulyawati, 2024). Two patterns emerge across this literature. First, operation-count errors are usually conceptual, not computational: students are applying a flawed mental model, not simply making mistakes under time pressure. Second, hands-on and visual instructional media are repeatedly proposed as a way to correct this, on the reasoning that concrete manipulation of quantities helps students build the underlying concept before they are asked to work with it in the abstract (Rejeki et al., 2024). This suggests that interactive, manipulable tools, including programming environments, offer a plausible means of addressing a problem that appears to stem from how mathematical operations are initially understood, not merely from how much they are practised.

Technology and Visual Programming in Mathematics Education

Technology has been proposed as one route to this kind of concrete, manipulable instruction, on the premise that interactive and visual tools make abstract mathematical ideas easier to grasp than static explanation alone (Mumpuni & Mulyawati, 2024). Scratch is a recurring example in this literature. Developed at the MIT Media Lab, it lets users build animations, games, and simulations by connecting graphical blocks rather than writing text-based syntax (Holstein & Cohen, 2025; Mazlan et al., 2023), which removes much of the barrier that discourages young or first-time learners from programming.

The theoretical basis for why these matters for learning goes back to Papert's (1980) constructionism, which holds that children build understanding most effectively by constructing something tangible and shareable, a program, in Scratch's case, rather than by receiving instruction passively. Scratch was designed with this principle in mind: it gives learners an "object to think with," and mathematical or logical ideas are worked out through the process of building and debugging a project rather than through direct explanation (Bernard & Setiawan, 2020).

Empirical work on Scratch in mathematics specifically supports this. Rodríguez-Martínez et al. (2020) conducted a quasi-experimental study with sixth-grade students, comparing pre- and post-instruction performance on word problems involving least common multiple and greatest common divisor, and found gains in both mathematical understanding and computational thinking after a Scratch-based intervention. Similar to the present study, Bernard and Setiawan (2020) developed a mathematics learning game using the Scratch language, demonstrating that the platform can be adapted specifically for content delivery rather than only for general programming instruction. Scratch's reach extends past mathematics as well; Mazlan et al. (2023) used it to teach electronic components in design and technology courses, which suggests the platform generalises reasonably well to structured technical content, not only creative or open-ended projects.

Taken together, this body of work establishes that Scratch can support subject-area learning, including mathematics, and that this rests on a workable theoretical basis (constructionism) rather than novelty alone (Papert, 1980). What it does not yet establish is how such an application should be built and validated before reaching a classroom; none of the studies above document a structured development process, and none focus specifically on the four basic operations at the elementary level rather than more advanced topics such as word problems involving multiples and divisors. That is the gap the present study addresses.

The ADDIE Model as a Development Framework

Where the studies above focus on outcomes after a tool already exists, the ADDIE model addresses how such a tool should be built in the first place. Originating in instructional design work for the U.S. military in the 1970s and refined through the 1980s, ADDIE structures development into five phases: Analysis, Design, Development, Implementation, and Evaluation, with evaluation understood as running throughout the process rather than only occurring at the end (Cahyadi, 2019). This framing matters methodologically: ADDIE is a development model, not itself a research design, and studies using it are more accurately described within a design and development research framework (Ofosu-Asare et al., 2019), defined as the systematic study of design, development, and evaluation processes aimed at producing an empirical basis for instructional products and the models that guide their creation (Richey & Klein, 2014). Positioning ADDIE within this framework rather than treating the two as interchangeable clarifies what kind of evidence a study built on ADDIE is expected to produce at each phase, including Implementation and Evaluation, which call for data from actual use rather than expert opinion alone.

3 Methodology

Research Design

This study employed a design and development research approach, following Richey and Klein's (2014) definition of the approach as the systematic study of design, development, and evaluation processes aimed at producing an empirical basis for instructional products. The ADDIE model- Analysis, Design, Development, Implementation, and Evaluation served as the operational framework for building the application (Cahyadi, 2019). Within that framework, this study reports the Analysis, Design, and Development phases in full. While the Implementation and Evaluation, which in a complete ADDIE, are addressed in this study at the level of expert projection rather than field testing: respondents were asked how they anticipated the application would perform and be evaluated once implemented, rather than being observed using it with students. Semi-structured interviews supplied the qualitative data needed to answer the three research questions guiding this study.

Participants

Four respondents participated in the study and were purposively selected based on their relevant expertise rather than to provide a representative sample of end-users. The respondents comprised a university lecturer at Muria Kudus University, two graduate students enrolled in the Master of Education programme at the same institution, and an elementary school principal. This composition gave the study perspectives from both teacher education and school leadership. Their feedback is used as expert and stakeholder validation of the application's design.

Instrument and Data Collection

Data were collected through a single interview protocol divided into three parts, administered to all four respondents. Part one gathered background information, gender, occupation, and institutional affiliation to contextualise each respondent's perspective. Part two asked about the application's content and instructional design: how the Scratch interface supported task completion, how student understanding of operation count could be evaluated through the application, whether the material was communicated clearly, and how the application's effectiveness could be further improved. Part three focused on visual design and game mechanics: whether the colour scheme supported engagement and understanding, whether the game's instructional objective came through clearly to students, and what specific improvements respondents would recommend. Interviews were conducted individually and recorded in written form for analysis.

Product Development Procedure

Alongside the interviews, the application itself was developed following the Analysis, Design, and Development stages of ADDIE. The Analysis stage identified the instructional problem, students' difficulty with operation count and a related decline in motivation for mathematics, along with the hardware and software requirements for building the application. The Design stage translated this into a five-slide Scratch project (https://scratch.mit.edu/projects/1101079800), sequenced to match the order in which the material would be presented and paired with background audio. The Development stage combined these design assets into working code, using Scratch's block-based drag-and-drop interface to build the game logic: movement controls and separate question-and-feedback sequences for each of the four operations (addition, subtraction, multiplication, division), each checking the student's answer and returning a correct or incorrect response.

Data Analysis

Interview responses were analysed descriptively, grouping respondents' answers by question and identifying points of agreement or disagreement across the four respondents for each of the three research questions. Where all four respondents converged on a judgment, for instance, on the appropriateness of the colour scheme, this is reported as a shared finding; where individual respondents raised distinct suggestions, such as the teacher recommendations on device performance, data structure, and code documentation, these are reported individually rather than folded into a consensus. This approach kept the analysis close to what respondents actually said, rather than imposing a coding scheme developed independently of the interview data.

4 Results

Application Development Outcomes (Analysis, Design, and Development stages)

The Analysis stage identified reduced student motivation toward mathematics, particularly around operation count, as the core problem the application needed to address, along with the basic hardware and software requirements for building a Scratch-based tool. This informed the Design stage, which produced a five-slide Scratch project sequenced to match the instructional order of the material and paired with background audio.

Figure 1 Application Prototype Slides

The Development stage combined these design assets into working game logic using Scratch's block-based interface. Movement was controlled through directional key inputs, and each of the four arithmetic operations, addition, subtraction, multiplication, and division, was implemented as a separate question-and-feedback sequence that checked the student's answer and returned a correct or incorrect response.

Stakeholder Backgrounds

Four respondents took part in the study and were purposively selected based on their relevant expertise in education and school leadership rather than their status as end-users of the application. Their backgrounds are summarised in Table 1.

Table 1 Respondent Backgrounds

Respondent	Gender	Background
R1	Male	Lecturer, Muria Kudus University
R2	Male	Master of Education student, Muria Kudus University
R3	Female	Master of Education student, Muria Kudus University
R4	Male	School Principal

Evaluation of Interface and Content Delivery (addressing RQ2)

Respondents were asked how the application's interface and instructional content supported operation-count learning. Their responses are synthesised in Table 2.

Table 2 Stakeholder Evaluation of Interface and Content

Question focus	Synthesis of respondent views
Interface design and task completion	Respondents agreed the Scratch interface was intuitive overall, and identified clear colour use and icon design as important for helping students understand what each code block does.
Evaluating student conceptual understanding	Respondents noted that evaluation should not rest on functionality alone; it should also draw on interviews, observation during development, and analysis of students' own code, supported by a comprehensive rubric covering functionality, design, and conceptual understanding together.
Clarity of instructional content	Respondents agreed the operation-count material was communicated clearly, attributing this to the consistent use of colour and image-based visualisation throughout the application.
Recommendations for instructional effectiveness	Respondents suggested starting with simple tasks before increasing complexity, building in opportunities for peer collaboration, incorporating reflection on errors, using real and contextual examples, and ensuring adequate teacher guidance alongside the application.

Across all four items, respondents converged on the view that the interface and content were age-appropriate and instructionally sound, while pointing toward pedagogical scaffolding (gradual difficulty, peer collaboration, teacher support) as necessary conditions for the application's effectiveness once used in a classroom.

Recommendations on Visual Design and Game Mechanics (addressing RQ3)

The third part of the interview asked respondents specifically about colour design, delivery of the game's instructional objective, and concrete suggestions for improving game mechanics. The responses are synthesised in Table 3.

Table 3 Stakeholder Recommendations on Design and Game Mechanics

Question focus	Synthesis of respondent views
Colour design and engagement	All four respondents agreed the colour scheme was appropriate and appealing, and supported understanding of the operation-count material.
Delivery of instructional objective	Respondents agreed the game's goals and content aligned with the intended operation-count material and could be understood by students.
Suggestions for improvement	Respondents recommended: (1) attention to performance on lower-specification devices, (2) selection of data structures appropriate to the application's needs, (3) addition of reward levels tied to task completion, and (4) use of code comments to improve readability and maintainability.

5 Discussion

The development process demonstrates that the ADDIE model provided more than a sequence of development activities; it ensured that the identified instructional problem informed each subsequent design decision. Rather than treating students' difficulties in arithmetic as purely conceptual, the Analysis stage identified declining motivation as the principal instructional concern. This diagnosis shaped the application's visual interface, game structure, background audio, and immediate feedback mechanisms. While previous studies have demonstrated the effectiveness of Scratch-based mathematics applications (Bernard & Setiawan, 2020; Rodríguez-Martínez et al., 2020), they provide limited explanation of how instructional problems are translated into specific design features. The present study therefore extends this literature by documenting the development pathway linking problem identification with instructional design.

The high level of agreement among respondents from different professional backgrounds also strengthens the evaluation findings. The lecturer, graduate students, and school principal independently concluded that the interface was intuitive and that colour and image-based visualisation supported understanding of arithmetic concepts. This finding is consistent with Bernard and Setiawan (2020), who found that Scratch effectively supports mathematics content delivery, and with Rodríguez-Martínez et al. (2020), who reported improvements in mathematical understanding following Scratch-based instruction. Unlike those classroom-based studies, however, the present evaluation was conducted before implementation, indicating that stakeholders recognised the instructional quality of the application before classroom use. This suggests that well-designed visual interfaces may represent an important prerequisite for successful implementation rather than merely contributing to learning outcomes after deployment.

The findings also provide a more nuanced interpretation of the constructionist perspective adopted in this study. Papert (1980) proposed that learning is strengthened when learners construct their own computational artefacts. By contrast, respondents evaluated an application that guides learners through structured arithmetic activities rather than encouraging them to create Scratch projects themselves. The application therefore functions primarily as an interactive learning environment rather than a fully constructionist programming experience. This does not contradict Papert's theory but indicates that Scratch can support mathematics instruction through guided interaction, extending its educational use beyond the learner-generated programming activities emphasised in constructionist research.

The recommendations for improving game mechanics reveal an important distinction between respondent groups. Whereas the lecturer and graduate students focused on instructional design, the school principal highlighted device performance, code maintainability, and compatibility with lower-specification hardware. This difference is understandable because school leaders are responsible for technology deployment under practical resource constraints. Earlier Scratch-based mathematics studies primarily examined instructional effectiveness (Bernard & Setiawan, 2020; Rodríguez-Martínez et al., 2020) and gave little attention to implementation issues such as hardware variability or software maintenance. Consequently, the present study contributes evidence on technical readiness for classroom adoption, an aspect that has received limited attention in previous research.

Although declining student motivation was identified during the Analysis stage as the central instructional problem, the evidence collected addresses usability, instructional clarity, and stakeholder perceptions rather than motivation itself. Respondents considered the application engaging and appropriate for elementary learners, but these judgments cannot be interpreted as evidence that student motivation improves through its use. Determining whether the application achieves its intended motivational objective requires classroom implementation involving elementary pupils during the Implementation and Evaluation phases of the ADDIE model. Recognising this boundary strengthens the interpretation by ensuring that the conclusions remain consistent with the evidence collected.

6 Conclusion

This study developed a Scratch-based learning application for teaching the four basic arithmetic operations using the Analysis, Design, and Development phases of the ADDIE model. The development process demonstrated how a clearly defined instructional problem can be systematically translated into design features that support mathematics learning. Identifying declining student motivation during the Analysis stage guided the selection of visual elements, game structure, background audio, and immediate feedback mechanisms incorporated into the application. Stakeholder evaluation showed broad agreement that the application possesses an intuitive interface, communicates instructional content clearly, and uses colours and visual representations effectively to support learners' understanding of arithmetic concepts. The consistency of these evaluations across respondents with different professional backgrounds strengthens confidence in the application's instructional design and usability. The findings also indicate that effective implementation should be accompanied by appropriate pedagogical support, including gradual progression of learning activities, teacher guidance, collaborative learning opportunities, and structured reflection on student errors. The study contributes to the growing literature on Scratch in mathematics education by documenting the instructional design process preceding classroom implementation rather than focusing exclusively on learning outcomes. It also identifies practical deployment considerations, including device compatibility and code maintainability, that have received limited attention in previous Scratch-based mathematics studies. At the same time, the findings suggest that the application represents a guided interactive learning environment rather than a fully constructionist programming experience. The evidence supports the application's instructional quality and readiness for classroom implementation. However, determining whether it improves student motivation and mathematical achievement requires empirical evaluation with elementary school learners during the remaining Implementation and Evaluation phases of the ADDIE model.

Practical and Social Implications

The study provides practical guidance for educators and instructional designers seeking to integrate Scratch into elementary mathematics instruction. By documenting the development process, it demonstrates how educational games can be designed systematically around identified learning needs rather than technological novelty. The findings suggest that intuitive interfaces, consistent visual design, immediate feedback, and structured progression of learning tasks can improve the clarity of mathematics instruction and make arithmetic concepts more accessible to young learners. Teachers can also strengthen the effectiveness of such applications by combining them with classroom discussion, collaborative activities, and guided reflection rather than relying on technology alone.

From a social perspective, the application has the potential to encourage greater engagement with mathematics by presenting arithmetic through an interactive learning environment that is familiar to digitally oriented learners. Strengthening foundational arithmetic skills is important because these concepts underpin later mathematical learning and broader problem-solving abilities. The study also highlights practical considerations that influence equitable implementation, particularly the need to ensure compatibility with lower-specification devices commonly found in schools with limited technological resources. Addressing these issues during development may improve accessibility across different educational settings and reduce barriers to adopting digital learning tools in primary education.

Implication for Policy

The findings support policies that encourage the integration of interactive digital learning tools into primary mathematics education through structured instructional design frameworks such as ADDIE. Education authorities should complement investments in educational technology with teacher professional development, curriculum guidance, and minimum technical standards that ensure applications function effectively across different school environments. The study also highlights the importance of evaluating educational applications before large-scale implementation, allowing usability, instructional quality, and technical compatibility to be assessed alongside curriculum alignment. Such measures can improve the effectiveness and sustainability of technology-supported mathematics instruction in primary schools.

Limitations of the Study and Suggestions for Future Research

Several limitations should be considered when interpreting the findings. First, the study reports only the Analysis, Design, and Development phases of the ADDIE model. Consequently, the application was not implemented or evaluated with elementary school students, preventing direct assessment of its effects on learning outcomes, motivation, or classroom engagement. Second, stakeholder feedback was obtained from four purposively selected respondents representing higher education and school leadership rather than the intended end-users. Although their expertise provided valuable validation of the application's design, their perspectives cannot substitute for evidence obtained through classroom implementation.

Another limitation is that the study relied primarily on qualitative interview data, which focused on perceptions of usability, instructional clarity, and design quality. Quantitative measures of learning effectiveness, usability, or motivation were beyond the scope of the present research.

Future studies should complete the remaining Implementation and Evaluation phases of the ADDIE model by conducting classroom trials with elementary school students. Experimental or quasi-experimental designs could be used to examine the application's effects on arithmetic achievement, motivation, engagement, and computational thinking. Future research should also investigate long-term classroom use, teacher adoption, and technical performance across schools with varying technological infrastructure. Such evidence would provide a more comprehensive assessment of the application's educational effectiveness and support wider implementation in primary mathematics education.

Acknowledgements

The author gratefully acknowledges the participants who contributed to the successful completion of this study. Their support and valuable contributions are sincerely appreciated.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Data Availability Statement

The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Conflict of Interest

The author declares no conflicts of interest.

Declaration of Generative AI

Declaration of Generative AI Usage: Generative AI tools were used solely for language editing, including grammar, clarity, and formatting. The author takes full responsibility for the content of this manuscript.

References

Batni, B., Junaini, S. N., Sidi, J., Mustafa, W. A., & Ismail, Z. I. A. (2025). Current research trends of scratch block-based programming for K-12: A systematic review. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *51*(2), 138-152. https://doi.org/10.37934/araset.51.2.138152

Bernard, M., & Setiawan, W. (2020). Developing math games media using Scratch language. *Journal of Physics: Conference Series, 1657*(1), 012064. https://doi.org/10.1088/1742-6596/1657/1/012064

Bers, M. U. (2020). *Coding as a playground: Programming and computational thinking in the early childhood classroom* (2nd ed.). Routledge. https://doi.org/10.4324/9781003022602

Cahyadi, R. A. H. (2019). *Pengembangan bahan ajar berbasis ADDIE model: Addie model-based teaching material development*. *Halaqa: Islamic Education Journal, 3*(1), 35–42. https://doi.org/10.21070/halaqa.v3i1.2124

Hernández-de-Menéndez, M., Escobar Díaz, C. A., & Morales-Menéndez, R. (2020). Engineering education for smart 4.0 technology: A review. *International Journal on Interactive Design and Manufacturing (IJIDeM), 14*(3), 789–803. https://doi.org/10.1007/s12008-020-00672-x

Holstein, S., & Cohen, A. (2025). Scratch teachers' perceptions of teaching computational thinking with school subjects in a constructionist approach. *Thinking Skills and Creativity, 56*, Article 101772. https://doi.org/10.1016/j.tsc.2025.101772

Hu, Y., Chen, C.-H., & Su, C.-Y. (2021). Exploring the effectiveness and moderators of block-based visual programming on student learning: A meta-analysis. *Journal of Educational Computing Research, 58*(8), 1467–1493. https://doi.org/10.1177/0735633120945935

Kodirun, Kadir, Busnawir, & Indrawati, W. O. (2025). Senior high school students' competence in logical operation and logical reasoning. *Frontiers in Education, 10*, Article 1493737. https://doi.org/10.3389/feduc.2025.1493737

Luneta, K. (2022). Special challenges in mathematics education in Sub-Saharan Africa. *Current Opinion in Behavioral Sciences, 48*, Article 101211. https://doi.org/10.1016/j.cobeha.2022.101211

Mangarin, R. A., & Caballes, D. O. (2024). Difficulties in learning mathematics: A systematic review. *International Journal of Research and Scientific Innovation, 11*(9), 401–405. https://doi.org/10.51244/IJRSI.2024.1109037

Maulidha, A. P., & Utama, C. (2025). A Literature Review on The Role of Coding Learning In Developing Elementary Students' Digital Literacy. *Proceedings Series of Educational Studies*, *2*(1), 270-277. https://conference.um.ac.id/index.php/pses/article/download/10595/4279

Mazlan, M. A., Azman, M. N. A., Evans, D. G., Pratama, H., & Lin, V. T. A. (2023). Development of Scratch applications for virtual smart homes in teaching electronic components in electronic design for design and tech subjects. *Journal of Technology and Humanities, 4*(2), 39–48. https://doi.org/10.53797/JTHKKSS.V4I2.5.2023

Mumpuni, F. O., & Mulyawati, I. (2024). Development of smart board media on the ability to count addition and subtraction materials in grade II elementary school students. *Prima: Jurnal Pendidikan Matematika, 8*(2), 424–435. https://doi.org/10.31000/prima.v8i2.11322

Ofosu-Asare, Y. A. W., Essel, H. B., & Bonsu, F. M. (2019). E-learning graphical user interface development using the ADDIE instruction design model and developmental research: the need to establish validity and reliability. *Journal of Global Research in Education and Social Science*, *13*(3), 1-6. https://www.researchgate.net/publication/332061094

Papert, S. (1980). Personal computing and its impact on education. In R. P. Taylor (Ed.), *The computer in the school: Tutor, tool, tutee* (pp. 197–202). Teachers College Press. https://www.scirp.org/reference/referencespapers?referenceid=3893569

Rejeki, S., Purnomo, Y. W., Maulyda, M. A., Nurul 'Aini, Y., & Asyifa, D. A. N. (2024). Improving primary school students' multiplication ability using "smart suitcase" media assisted by mathematical worksheets. *Pedagogical Research, 9*(3), Article em0211. https://doi.org/10.29333/pr/14635

Richey, R. C., & Klein, J. D. (2014). *Design and development research: Methods, strategies, and issues*. Routledge. https://doi.org/10.4324/978020382603

Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-López, J. M. (2020). Computational thinking and mathematics using Scratch: An experiment with sixth-grade students. *Interactive Learning Environments, 28*(3), 316–327. https://doi.org/10.1080/10494820.2019.1612448

Tarisna, K. S., Jayanta, I. N. L., & Ujianti, P. R. (2025). Educandy learning media in Indonesian language syllable learning media to improve reading and writing literacy. *Jurnal Media dan Teknologi Pendidikan, 5*(2), 363–371. https://doi.org/10.23887/jmt.v5i2.97953.

Wu, H. H. (2026). *Understanding numbers in elementary school mathematics*. American Mathematical Society. https://doi.org/10.4171/mag/56