

Enhancing Computational Thinking Skills through Participatory Learning among Indonesian Middle School Students in Jeddah

Andreyan Rizky Baskara¹, Yuslena Sari¹, Mutia Maulida^{1*},
Nurul Fathanah Mustamin¹, Erika Maulidiya¹, Sunardi²

¹Department of Information Technology, Universitas Lambung Mangkurat, Indonesia

²Department of Environmental Science, Universitas Lambung Mangkurat, Indonesia

Corresponding author: Mutia Maulida

ABSTRACT: Computational thinking (CT) is increasingly recognized as an essential competence for enabling students to approach problems systematically, logically, and efficiently. However, opportunities for explicit CT learning may vary across educational contexts, including Indonesian schools operating abroad. This study examined changes in middle school students' computational thinking performance following a participatory problem-solving intervention at Sekolah Indonesia Jeddah, Saudi Arabia. A one-group pretest–posttest design with 50 Grade VIII students. The three-day intervention introduced fundamental computational thinking concepts through contextual problem-solving activities, interactive games, group discussions, collaborative projects, presentations, and reflection. Students completed a 25-item assessment before and after the intervention. The mean score increased from 56.60 ($SD = 10.06$) on the pretest to 79.68 ($SD = 10.42$) on the posttest, an average improvement of 23.08 points. Because the difference scores were not normally distributed, a Wilcoxon signed-rank test was conducted and showed a statistically significant improvement in students' performance ($W = 0$, $p < .001$), with a large effect size ($r = .87$). The mean normalized gain was 0.555, indicating a moderate level of improvement, while the proportion of students reaching the mastery threshold of 70 increased from 4% to 84%. These findings suggest that participatory and contextual problem-solving activities can provide an accessible approach to strengthening computational thinking among middle school students. The study also contributes empirical evidence from an Indonesian overseas school context and highlights the potential of university–school collaboration in supporting computational thinking education.

Keywords - computational thinking, participatory learning, problem solving, middle school students, community service

I. INTRODUCTION

Computational thinking (CT) has become an increasingly important competency in contemporary education because it equips students with systematic approaches to understanding and solving problems. Wing [1] initially conceptualized computational thinking as a way to formulate problems and develop solutions based on fundamental computer science concepts. Since then, CT has evolved beyond its association with programming and has increasingly been incorporated into K–12 education as a broader problem-solving competency [2], [3]. Recent reviews also show that CT is now being integrated across subjects and educational settings, reflecting its growing relevance in twenty-first-century learning [4], [5]. Although computational thinking has been defined through different frameworks, educational contexts consistently emphasize several cognitive processes. These include decomposition, pattern recognition, abstraction, and algorithmic thinking. These processes make CT relevant not only to computer science but also to broader learning activities that require analysis, planning, and systematic problem solving [6], [7].

Evidence supports the educational value of CT by linking computational activities to higher-order cognitive development. Montuori et al. [8], through a systematic review and meta-analysis, found that computational interventions were associated with improvements in several cognitive processes, with the largest effects observed for problem-solving abilities. This finding reinforces the view that computational thinking should not be regarded merely as a technical competence related to coding but as a structured form of reasoning that can support students in approaching complex problems across learning contexts. Although programming provides a useful environment for practicing CT, students with limited prior experience in computer science

may initially struggle with programming syntax and technical procedures. Alternatively, unplugged computational thinking approaches introduce CT concepts through games, puzzles, physical activities, and contextual problem-solving tasks. Chen et al. [9] reviewed 49 K–12 studies and found that unplugged activities were commonly implemented through different forms of games and problem-solving activities.

Participatory problem-solving activities are particularly suitable for introducing fundamental CT concepts. Through collaborative tasks, students can break complex problems into smaller components, recognize patterns, identify relevant information, and construct systematic sequences of actions. A recent systematic review by Yin et al. [10] found that collaborative learning is widely used in K–12 CT education and identified multiple collaborative instructional designs that can support CT development. Empirical evidence also indicates that both plugged and unplugged CT approaches can support problem-solving development among middle school students [11]. Computational thinking development is also relevant for Indonesian students studying in overseas educational settings. However, empirical studies examining CT learning in Indonesian schools abroad remain limited. To address this context, Universitas Lambung Mangkurat collaborated with Sekolah Indonesia Jeddah (SIJ), Saudi Arabia, to conduct a computational thinking learning program for middle school students. The program aimed to introduce decomposition, pattern recognition, abstraction, and algorithms while encouraging systematic problem solving, logical reasoning, creativity, and teamwork.

The intervention took place over three days, using a participatory approach that combined conceptual instruction, interactive games, group discussions, and collaborative problem-solving activities. Students worked on logic problems such as the wolf–goat–cabbage and missionaries-and-cannibals problems, followed by collaborative projects and group presentations. To assess changes in computational thinking performance, students completed a pretest before the intervention and a comparable posttest after completing the learning activities. Both tests consisted of 20 multiple-choice questions and five short-answer questions covering decomposition, pattern recognition, abstraction, algorithms, logic, and analysis.

Although previous studies show that unplugged and participatory approaches can support computational thinking development, relatively little empirical attention has focused on Indonesian students studying in overseas school settings [12], [13], [14]. The present study addresses this contextual gap by examining the computational thinking performance of 50 Grade VIII students at Sekolah Indonesia Jeddah before and after participation in a participatory problem-solving intervention. In addition to testing whether students' performance changed significantly, the study examines the magnitude of improvement using normalized gain and mastery-level achievement. Accordingly, this study addresses the following research questions: (a) Is there a significant difference in students' computational thinking performance before and after participating in the participatory problem-solving intervention?; (b) How does students' performance change across the computational thinking domains of decomposition, pattern recognition, abstraction, and algorithmic thinking following the intervention?.

II. METHODOLOGY

This study employed a one-group pretest–posttest design to examine changes in students' computational thinking performance following a participatory problem-solving intervention. The same group of students completed an assessment before and after participating in the learning activities [15]. Figure 1 shows the research design of this participatory learning held by Universitas Lambung Mangkurat at Sekolah Indonesia Jeddah. The participants consisted of 50 Grade VIII students from Sekolah Indonesia Jeddah (SIJ), Saudi Arabia, comprising 17 students from Grade VIII-A, 17 students from Grade VIII-B, and 16 students from Grade VIII-C. The intervention was conducted as part of an international educational collaboration between Universitas Lambung Mangkurat, Indonesia, and Sekolah Indonesia Jeddah. The main computational thinking activities were implemented over three days.



Fig 1. Research Design

The intervention was designed around four fundamental computational thinking processes: decomposition, pattern recognition, abstraction, and algorithmic thinking. A participatory learning approach combined conceptual instruction with contextual problem-solving, interactive activities, collaborative discussion, and group-based tasks. The learning design emphasized applying CT concepts to structured problems rather than introducing computational thinking primarily through programming. Contextual logic problems engaged students in identifying problem constraints, examining alternative states, recognizing patterns, and constructing systematic sequences of actions. Collaborative activities encouraged students to articulate and compare problem-solving strategies, while group projects and presentations provided opportunities to integrate and communicate computational reasoning. These instructional characteristics were consistent with the program's emphasis on decomposition, pattern recognition, abstraction, algorithms, and systematic problem solving.



Fig 2. Computational Thinking Participatory Learning in Sekolah Indonesia Jeddah

Students' computational thinking performance was measured using a 25-item assessment consisting of 20 multiple-choice questions and five short-answer questions. The items covered decomposition, pattern recognition, abstraction, algorithms, logic, and analytical reasoning. The pretest and posttest used comparable formats, but the posttest included different question variations to reduce the chance that students would simply recall answers from the initial assessment. Scores were converted to a 0–100 scale, with higher scores representing stronger computational thinking performance. Table 1 shows the structure of the computational thinking intervention.

Table 1. Structure of Computational Thinking Intervention

Stage	Main Activity	Purpose
Pre-intervention	Pretest	Identify initial CT performance
Concept introduction	CT concepts and examples	Introduce fundamental CT processes
Interactive learning	Logic and contextual problems	Practice CT-based problem solving
Collaborative activity	Group discussion and project	Develop and integrate CT processes
Presentation and reflection	Presentation and discussion	Communicate and reinforce reasoning
Post-intervention	Posttest	Assess CT performance after intervention

Pretest and posttest scores were matched at the individual-student level, resulting in 50 complete paired observations. Descriptive statistics were first calculated to summarize students' performance, including the mean, standard deviation, median, minimum, and maximum scores. Shapiro-Wilk tests were used to examine the normality of paired difference scores. Because the distribution of the differences did not satisfy the normality assumption, changes between pretest and posttest performance were evaluated using the Wilcoxon signed-rank test. To complement statistical significance, we calculated an effect size to estimate the magnitude of the observed change. We interpreted values of approximately .10, .30, and .50 as small, moderate, and large effects, respectively. Students' relative learning improvement was additionally examined using the normalized gain (N-gain). In addition, the proportion of students reaching the mastery threshold of 70 was compared between the pretest and posttest to provide a complementary measure of learning achievement.

III. RESULTS AND DISCUSSION

3.1 Pre-test and Post-test Performance

Descriptive analysis showed a clear increase in students' computational thinking performance following the participatory problem-solving intervention. As shown in Table 2, the mean pretest score was 56.60 (SD = 10.06), increasing to 79.68 (SD = 10.42) on the posttest. The median score similarly increased from 60.00 to 80.50. The average individual improvement was 23.08 points, with score increases ranging from 15 to 30 points. All 50 students scored higher on the posttest than on the pretest.

Table 2. Descriptive Statistics of Pre-test and Post-test Scores

Measure	Pretest	Posttest
N	50	50
Mean	56.60	79.68
Standard deviation	10.06	10.42
Median	60.00	80.50
Minimum	35.00	54.00
Maximum	71.00	97.00

The normality of the difference scores was examined before conducting the inferential analysis. The Shapiro-Wilk test indicated that the paired differences significantly deviated from normality ($W = .933$, $p = .007$). Therefore, the non-parametric Wilcoxon signed-rank test was used to compare the pretest and posttest scores. The analysis revealed a statistically significant difference between students' pretest and posttest performance ($W = 0$, $p < .001$). The associated effect size was $r = .87$, indicating a large magnitude of change. Thus, the result supports the first research question, showing that students' computational thinking performance was significantly higher after participating in the intervention.

The substantial difference between the two assessments suggests that the instructional activities provided meaningful opportunities for students to practice computational problem solving. Rather than emphasizing programming syntax, the intervention required students to analyze constraints, evaluate alternative solutions, and construct systematic sequences of actions. This result is consistent with Chen et al. [9], whose meta-analysis reported a large overall positive effect of unplugged activities on computational thinking development among K–12 learners.

3.2 Improvement Across Computational Thinking Domains

To obtain a more detailed understanding of students' learning progress, the pretest and posttest scores were further analyzed according to the computational thinking domains measured by the assessment. Because the domains contained different numbers of items and therefore different maximum scores, the results were converted into percentages of the maximum possible domain score to facilitate comparison as shown in Table 3.

Table 3. Improvement Results of Computational Thinking Domains

CT Domain	Maximum Score	Pretest Mean (%)	Posttest Mean (%)	Improvement (percentage points)	Mean N-gain	Wilcoxon p	Effect Size (r)
CT Concept	20	60.50	76.40	+15.90	0.354	< .001	0.51
Decomposition	12	54.17	83.83	+29.67	0.612	< .001	0.81
Pattern Recognition	16	54.38	81.88	+27.50	0.510	< .001	0.78
Abstraction	16	54.50	82.63	+28.13	0.606	< .001	0.75
Algorithmic Thinking	32	57.06	77.56	+20.50	0.472	< .001	0.76

The domain-level results indicate that students' performance improved significantly across all computational thinking domains. The largest increase occurred in decomposition, which increased from 54.17% in the pretest to 83.83% in the posttest, representing an improvement of 29.67 percentage points. This domain also produced the highest mean N-gain ($g = 0.612$) and a large effect size ($r = .81$). A similarly substantial improvement was observed in abstraction, where performance increased from 54.50% to 82.63%, corresponding to a gain of 28.13 percentage points and a moderate N-gain of 0.606. Pattern recognition increased from 54.38% to 81.88%, representing a 27.50-percentage-point improvement and a mean N-gain of 0.510. Both domains also demonstrated large effect sizes.

Improvement was also evident in algorithmic thinking, although the relative increase was smaller than that observed for decomposition, abstraction, and pattern recognition. Students' performance increased from 57.06% to 77.56%, with an improvement of 20.50 percentage points and a mean N-gain of 0.472. Nevertheless, the Wilcoxon signed-rank test indicated a statistically significant difference ($p < .001$) with a large effect size ($r = .76$). The CT Concept domain showed the smallest relative improvement, increasing from 60.50% to 76.40%. Its mean N-gain was 0.354, which remained within the moderate category, and the pretest–posttest difference was statistically significant ($p < .001$). The comparatively smaller gain in this domain may partly reflect the higher baseline performance, as students already demonstrated greater initial familiarity with general CT concepts than with the application-oriented domains.

3.3 Learning Gain and Mastery Achievement

To complement the statistical comparison, we calculated normalized gain to examine the relative magnitude of students' improvement. The mean N-gain was 0.555, which falls within the moderate improvement category. Of the 50 students, 10 (20%) achieved a high N-gain, 39 (78%) demonstrated moderate gains, and only one (2%) was categorized as having a low gain. No student declined in performance as shown in Table 4.

Table 4. Learning Gain and Mastery Achievement

Indicator	Result
Mean score increase	23.08
Mean N-gain	0.555
N-gain category	Moderate
High gain	10 students (20%)
Moderate gain	39 students (78%)
Low gain	1 student (2%)
Pretest mastery (≥ 70)	2 students (4%)
Posttest mastery (≥ 70)	42 students (84%)
Effect size, r	0.87 (Large)

The mastery-level analysis provides an additional indication of the practical significance of the improvement. Before the intervention, only 2 of the 50 students (4%) achieved the mastery threshold of 70. Following the intervention, the number increased to 42 students (84%). Thus, the proportion of students reaching the expected performance level increased by 80 percentage points. The combination of a moderate average N-gain, a large effect size, and a substantial increase in mastery achievement indicates that the improvement was not limited to statistical significance. Instead, the pattern suggests a meaningful shift in students' overall performance. While the mean N-gain remained in the moderate category, this should be interpreted alongside the relatively broad baseline differences among participants. Students entered the intervention with pretest scores ranging from 35 to 71, yet the entire group improved.

These findings are consistent with recent empirical evidence supporting activity-based and unplugged computational thinking instruction. Aytekin and Topçu [11], for example, found that both plugged and unplugged CT approaches contributed to the development of problem-solving skills among middle school students, with the unplugged approach demonstrating particularly favorable outcomes compared with conventional instruction. These results extend this evidence by showing similar improvement in an Indonesian overseas school setting.

3.4 Discussion

The observed improvement can be interpreted in light of the intervention's instructional characteristics. The learning activities emphasized four interconnected computational thinking processes: decomposition, pattern recognition, abstraction, and algorithmic thinking. Contextual logic problems required students to identify a problem's components and constraints, recognize relationships among possible states, distinguish essential information, and develop logically ordered solution steps. These activities enabled students to engage with computational thinking as a problem-solving process rather than primarily as a programming exercise. This

interpretation is supported by Montuori et al. [8], whose systematic review and meta-analysis found that computational interventions were associated with improvements in higher-order cognitive processes and that the largest effects were observed for problem-solving abilities. The increase observed in the present study may therefore reflect the close alignment between CT processes and the structured problem-solving demands embedded in the intervention.

The domain-level analysis provides a more nuanced interpretation of the overall improvement. The strongest gains were observed in decomposition, abstraction, and pattern recognition, suggesting that the participatory problem-solving activities were particularly effective in supporting students' ability to structure problems, identify relevant information, and recognize recurring relationships. These processes were directly embedded in the contextual logic tasks used during the intervention. Algorithmic thinking also improved significantly, although its normalized gain was comparatively lower. This may be explained by the broader cognitive demands of the algorithmic domain, which included sequencing, conditional reasoning, repetition, and debugging[16]. Taken together, the findings indicate that the intervention supported multiple dimensions of computational thinking rather than improving only students' general familiarity with CT concepts[17].

The intervention's participatory nature may also have contributed to the learning experience[18]. Students did not solve all problems individually; instead, group discussion and collaborative activities required them to compare alternative strategies, explain reasoning, identify errors, and refine solutions. Collaborative interaction can make computational reasoning more explicit because students must articulate not only the final solution but also the logic underlying the solution process[19]. In their systematic review of 43 empirical studies, Yin et al. [10] identified collaborative learning as a widely adopted pedagogical approach in K–12 computational thinking education and highlighted its potential contribution to CT instructional design. Another relevant feature was using unplugged and contextual problems as the initial learning environment. For students with limited prior exposure to computer science, this approach keeps attention on reasoning and problem-solving rather than programming syntax.

Chen et al. [9] similarly found that unplugged activities—including games and problem-solving tasks—were widely used in K–12 CT education and produced a large overall effect on CT skills. The present findings therefore reinforce the potential value of introducing CT through accessible problem contexts before progressing toward more formal computing activities. The findings also have relevance beyond the immediate classroom context. Most existing CT studies have focused on conventional national school systems, computer science classrooms, or STEM learning environments[20]. Evidence from Indonesian schools operating abroad remains comparatively limited. The improvement observed among students at Sekolah Indonesia Jeddah therefore provides initial empirical evidence that participatory CT instruction can also be implemented meaningfully within an overseas Indonesian educational environment.

Nevertheless, the findings should be interpreted in accordance with the study design. Because the study used a one-group pretest–posttest design without a comparison group, the improvement cannot be attributed exclusively to the intervention with the same degree of confidence as in a randomized or controlled experimental design. The results show a large and statistically significant change following the intervention, but other influences cannot be completely ruled out. Future studies could therefore incorporate comparison groups, longer intervention periods, and delayed posttests to examine the persistence and transfer of computational thinking development.

IV. CONCLUSION

This study examined changes in computational thinking performance among 50 Grade VIII students at Sekolah Indonesia Jeddah following a participatory problem-solving intervention. The learning activities emphasized decomposition, pattern recognition, abstraction, and algorithmic thinking through contextual problems, interactive tasks, collaborative discussion, and group-based problem solving. The results showed a substantial increase in students' performance, with the mean score rising from 56.60 in the pretest to 79.68 in the posttest. The Wilcoxon signed-rank test confirmed that the difference was statistically significant ($p < .001$), while the large effect size ($r = .87$) indicated that the change was educationally meaningful.

The improvement was also reflected in the normalized gain and mastery results. The mean N-gain of 0.555 indicated moderate improvement, while the proportion of students achieving the mastery threshold of 70 increased from 4% to 84%. These findings suggest that participatory and contextual problem-solving activities can provide an accessible approach to strengthening computational thinking among middle school students, particularly when learners are introduced to computational reasoning through structured problems rather than programming syntax alone. The study also provides empirical evidence from an Indonesian overseas school setting, contributing to the relatively limited literature on computational thinking education among Indonesian students studying abroad.

Despite these positive findings, the results should be interpreted in light of the study's design limitations. The study employed a one-group pretest–posttest design without a comparison group, and the

intervention was conducted over a relatively short period. Consequently, the observed improvement cannot be attributed solely to the intervention, and the study did not examine long-term retention of students' computational thinking skills. Future research should incorporate comparison or control groups, longer intervention periods, and delayed posttests to evaluate the sustainability of learning outcomes. Further studies may also examine changes in specific computational thinking dimensions and explore how similar participatory approaches can be integrated into regular classroom instruction and replicated in other Indonesian schools abroad.

V. ACKNOWLEDGEMENTS

This research was financially supported by the Lambung Mangkurat University Research Grant for 2024 under grant number 2532.2/UN8.2/AM/2024.

REFERENCES

- [1]. J. Wing, "Computational Thinking," *Commun. ACM*, vol. 49, pp. 33–35, Mar. 2006, doi: 10.1145/1118178.1118215.
- [2]. S. J. Lee, G. M. Francom, and J. Nuatomue, "Computer science education and K-12 students' computational thinking: A systematic review," *Int. J. Educ. Res.*, vol. 114, p. 102008, 2022, doi: <https://doi.org/10.1016/j.ijer.2022.102008>.
- [3]. K.-D. Boom, M. Bower, J. Siemon, and A. Arguel, "Relationships between computational thinking and the quality of computer programs," *Educ. Inf. Technol.*, vol. 27, pp. 1–22, Jul. 2022, doi: 10.1007/s10639-022-10921-z.
- [4]. S. Yeni, N. Grgurina, M. Saeli, F. Hermans, J. Tolboom, and E. Barendsen, "Interdisciplinary Integration of Computational Thinking in K-12 Education: A Systematic Review," *Informatics Educ.*, vol. 23, Sep. 2023, doi: 10.15388/infedu.2024.08.
- [5]. Kathy A Mills, Jen Cope, Laura Scholes, and Luke Rowe, "Coding and Computational Thinking Across the Curriculum: A Review of Educational Outcomes," *Rev. Educ. Res.*, vol. 95, no. 3, pp. 581–618, Jun. 2025, doi: 10.3102/00346543241241327.
- [6]. Ting-Ting Wu, Andik Asmara, Yueh-Min Huang, and Intan Permata Hapsari, "Identification of Problem-Solving Techniques in Computational Thinking Studies: Systematic Literature Review," *Sage Open*, vol. 14, no. 2, p. 21582440241249896, Apr. 2024, doi: 10.1177/21582440241249897.
- [7]. H. Belmar, "Review on the teaching of programming and computational thinking in the world," *Front. Comput. Sci.*, vol. Volume 4-2022, 2022, doi: 10.3389/fcomp.2022.997222.
- [8]. C. Montuori, F. Gambarota, G. Altoé, and B. Arfé, "The cognitive effects of computational thinking: A systematic review and meta-analytic study," *Comput. Educ.*, vol. 210, p. 104961, 2024, doi: <https://doi.org/10.1016/j.compedu.2023.104961>.
- [9]. P. Chen, D. Yang, A. H. S. Metwally, J. Lavonen, and X. Wang, "Fostering computational thinking through unplugged activities: A systematic literature review and meta-analysis," *Int. J. STEM Educ.*, vol. 10, no. 1, p. 47, 2023, doi: 10.1186/s40594-023-00434-7.
- [10]. Stella Xin Yin, Dion Hoe-Lian Goh, and Choon Lang Quek, "Collaborative Learning in K-12 Computational Thinking Education: A Systematic Review," *J. Educ. Comput. Res.*, vol. 62, no. 6, pp. 1220–1254, Oct. 2024, doi: 10.1177/07356331241249956.
- [11]. A. AYTEKİN and M. S. TOPÇU, "Improving 6th Grade Students' Creative Problem Solving Skills Through Plugged and Unplugged Computational Thinking Approaches," *J. Sci. Educ. Technol.*, vol. 33, no. 6, pp. 867–891, 2024, doi: 10.1007/s10956-024-10130-y.
- [12]. S. Forsström and M. Bond, "Computational Thinking Assessment in Primary and Secondary Education: A Meta-synthesis of tools, Methods and Pedagogical Approaches," *Int. J. Comput. Sci. Educ. Sch.*, vol. 6, no. 4 SE-Full Papers, Nov. 2024, doi: 10.21585/ijcses.v6i4.208.
- [13]. M. Babazadeh and L. Negrini, "How is computational thinking assessed in European K-12 education? A systematic review," *Int. J. Comput. Sci. Educ. Sch.*, vol. 5, no. 4 SE-Full Papers, pp. 3–19, Oct. 2022, doi: 10.21585/ijcses.v5i4.138.
- [14]. S.-C. Kong and M. Lai, "Validating a computational thinking concepts test for primary education using item response theory: An analysis of students' responses," *Comput. Educ.*, vol. 187, p. 104562, 2022,

- doi: <https://doi.org/10.1016/j.compedu.2022.104562>.
- [15]. M. Zapata-Cáceres, P. Marcelino, L. El-Hamamsy, and E. Martín-Barroso, “A Bebras Computational Thinking (ABC-Thinking) program for primary school: Evaluation using the competent computational thinking test,” *Educ. Inf. Technol.*, vol. 29, no. 12, pp. 14969–14998, 2024, doi: [10.1007/s10639-023-12441-w](https://doi.org/10.1007/s10639-023-12441-w).
- [16]. C. Robledo-Castro, C. Hederich-Martínez, and L. F. Castillo-Ossa, “Cognitive stimulation of executive functions through computational thinking,” *J. Exp. Child Psychol.*, vol. 235, p. 105738, 2023, doi: <https://doi.org/10.1016/j.jecp.2023.105738>.
- [17]. Y. Dong, H. Ma, B. Jing, Q. Wang, N. Zhu, and P. Cao, “Effect of computational thinking professional development on in-service teachers’ teaching and their students’ learning,” *Educ. Technol. Soc.*, Feb. 2026.
- [18]. J.-F. Trani *et al.*, “The impact of a participatory intervention to improve learning outcomes and reduce school-based discrimination and community stigma in primary rural schools of Afghanistan: A cluster control randomized trial,” *Int. J. Educ. Dev.*, vol. 118, p. 103409, 2025, doi: <https://doi.org/10.1016/j.ijedudev.2025.103409>.
- [19]. H. Ceballos, T. van den Bogaart, S. van Ginkel, J. Spandaw, and P. Drijvers, “How collaborative problem solving promotes higher-order thinking skills: A systematic review of design features and processes,” *Think. Ski. Creat.*, vol. 59, p. 102001, 2026, doi: <https://doi.org/10.1016/j.tsc.2025.102001>.
- [20]. S. Yeni, J. Nijenhuis-Voogt, M. Saeli, E. Barensen, and F. Hermans, “Computational thinking integrated in school subjects – A cross-case analysis of students’ experiences,” *Int. J. Child-Computer Interact.*, vol. 42, p. 100696, 2024, doi: <https://doi.org/10.1016/j.ijcci.2024.100696>.

Corresponding author: Mutia Maulida

¹Department of Information Technology, Universitas Lambung Mangkurat, Indonesia