Abdimas Umtas: Jurnal Pengabdian Kepada Masyarakat
LPPM-Universitas Muhammadiyah Tasikmalaya
Volume: 8, Issue: 3, July, 2025; pp: 1145-1153
E-ISSN: 2614 - 8544

Improving Digital Literacy Using a C++-Based LMS: A Case from SMP Al Azhar Syifa Budi Parahyangan

Selly Meliana[a)], I Gede Manggala Putra[b)], Hani Nurrahmi[c)], Putu Padmanaba[d)], Rafi Arkan Fachreza Arfianta[e)]

Telkom University

[a)] Corresponding author: sellym@telkomuniversity.ac.id
[b)] igedemanggalaputra@telkomuniversity.ac.id
[c)] haninurrahmii@telkomuniversity.ac.id
[d)] putupadmanaba@student.telkomuniversity.ac.id
[e)] rezaarfianta@student.telkomuniversity.ac.id

ABSTRACT

This community service program aimed to introduce digital literacy and basic programming to junior high school students at SMP Al Azhar Syifa Budi Parahyangan, an A-accredited school lacking a formal ICT curriculum. The program utilized a Learning Management System (LMS) built with C++ programming exercises and auto-grading features to support digital learning. The LMS was enhanced based on an Iterative and Incremental Development (IID) approach. It was implemented through four face-to-face mentoring sessions conducted between February and May 2025. Program outputs included 231 coding problems, seven instructional videos, structured modules, interactive quizzes, and a Junior Coding Contest. Based on a survey of 11 respondents, 45.45% expressed satisfaction with the materials and mentoring. However, feedback highlighted challenges related to schedule and material complexity. Recommendations include integrating the program into formal class hours, using block-based programming tools such as Scratch, and involving teachers for sustainability. The project supports the goals of the Merdeka Curriculum and demonstrates that tailored LMS systems can be effective tools for digital education in schools with limited infrastructure.

ARTICLE INFO

Article History :
Submitted/Received 2 July 2025
First Revised 10 July 2025
Accepted 29 July 2025
First Available online 31 July 2025
Publication Date 31 July 2025

Keyword :
Digital literacy
LMS
C++ programming
Junior high school
Educational technology
Programming education

https://doi.org/10.35568/abdimas.v8i3.6869

INTRODUCTION

In the era of rapid digital transformation, computational thinking and programming have become essential 21st-century skills. However, in many parts of Indonesia, access to formal informatics education remains limited. While many national curriculum reforms now emphasize flexible, technology-integrated learning, several junior high schools continue to operate without formal informatics education or adequate digital infrastructure. This gap limits students' exposure to fundamental programming concepts at an early age.

One example of a school facing this challenge is SMP Al Azhar Syifa Budi Parahyangan, a private school established in 2009 in West Bandung, fully accredited (Grade A), and recognized for its commitment to holistic student development. However, the school has no formal ICT curriculum, posing a challenge for developing students' 21st-century digital competencies. Learning Management Systems (LMS) have emerged as effective tools to support structured and flexible digital learning. When designed with interactive features and automated assessment, LMS platforms can help learners build digital literacy and independent learning habits. However, most existing applications are developed for older students or rely on simplified programming environments. Despite its complexity, C++ was chosen in this project to expose students to structured and rigorous programming logic early on, while still supported by simplified scaffolding through videos and auto-grading.

Community service initiatives in digital education have demonstrated promising outcomes when learning technologies are combined with teacher facilitation. One such program in Indonesia showed that flipped classroom approaches in teacher training significantly improved pedagogical strategies and student engagement in literacy-focused settings, offering a useful reference for structuring informal digital instruction.

This raises a critical question: can a C++-based LMS, if properly structured and combined with direct mentoring, improve students' digital competence in schools without formal ICT instruction? The present study addresses this challenge by designing and deploying a specialized LMS that integrates coding exercises, video tutorials, auto-grading features, and gamified assessment. The system was implemented as part of an extracurricular coding club for junior high school students and evaluated through structured mentoring sessions and a programming competition. This approach differentiates itself by combining formal learning resources, interactive assessment, and real-time mentoring within a community service context.

Therefore, the purpose of this article is to present the development process, learning outcomes, and practical insights from deploying a tailored LMS-based learning model for junior high school students, and to evaluate its potential as a scalable solution for similar underserved school contexts. This initiative was also a response to the Merdeka Curriculum's push for digital literacy in secondary schools, particularly for institutions with limited teaching resources and digital infrastructure.

METHOD

This study, involving 11 junior high school participants, applied an Iterative and Incremental Development (IID) approach to build and refine a Learning Management System (LMS) tailored for junior high school programming education. IID is a widely used method in educational technology development that allows continuous improvement based on user feedback while ensuring each development cycle is functional and testable. The program was built on a previous community service initiative and was conducted at the request of the partner school, which viewed LMS as a potential solution for integrating digital learning aligned with the Merdeka Curriculum.

https://doi.org/10.35568/abdimas.v8i3.6869

Needs Analysis

The team conducted structured interviews with teachers and students at SMP Al Azhar Syifa Budi Parahyangan to identify learning challenges, technological readiness, and curricular gaps related to digital literacy and programming, including topics such as students' prior exposure to coding, access to devices, and teacher readiness.

Content and System Design

Based on the analysis, the LMS content was structured into seven learning modules covering basic C++ concepts, including data types, loops, conditionals, arrays, functions, and recursion. Each module included short video tutorials, interactive quizzes, and automated coding exercises using a grader engine. This mirrors integrative instructional designs that incorporate mathematical reasoning to enhance programming comprehension (Pattichis et al., 2024).

Each of the seven videos covered specific topics: data types, loops, conditionals, combination of loops and conditionals, arrays, functions and procedures, and recursion. These modules were supported by instructional videos covering core programming concepts, as illustrated in Figure 1.

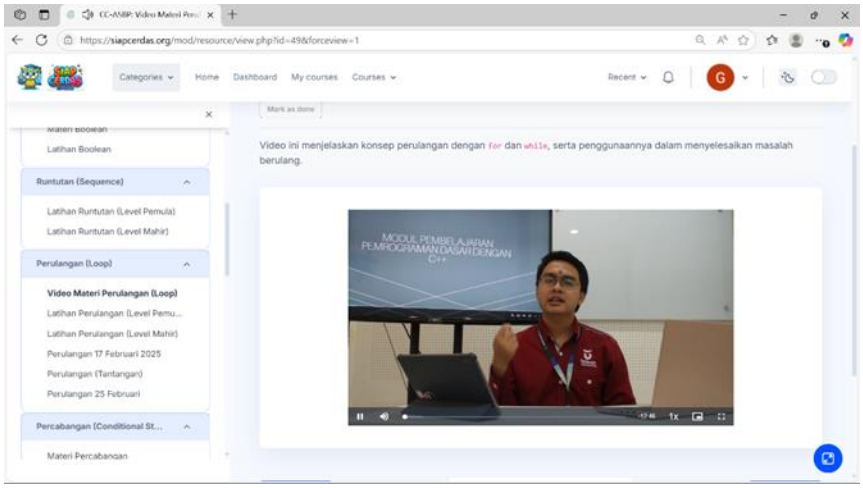


FIGURE 1. Integration of Instructional Video Content in LMS

These visual materials were designed to reinforce key programming concepts and serve as accessible learning aids for students with no prior coding experience.

Development and Deployment

The LMS was developed on the Moodle platform, with a user-friendly interface adapted for middle school learners. Features were deployed incrementally to ensure technical stability and student usability, and the final system supported login access for students, progress tracking, and real-time code evaluation.

Mentoring and Implementation

Four mentoring sessions were conducted over three months. Each session included LMS orientation, collaborative problem solving, and individual coding practice. Students were guided by facilitators to explore materials at their own pace, supported both in-class and during asynchronous practice. This

https://doi.org/10.35568/abdimas.v8i3.6869

combination of structured mentoring and technology use reflects prior teacher training programs using flipped classrooms that successfully improved instructional delivery (Irianti, Rachmawati, & Friatin, 2020).

To better illustrate the structured progression of this Iterative and Incremental Development (IID)-based program, Figure 2 provides a comprehensive timeline that maps out the six-month implementation process, beginning with early needs assessment and culminating in the final evaluation and knowledge transfer. This visual timeline is designed to help readers understand the logical sequencing and overlap of activities, including module development, system testing, mentoring, and evaluation. By presenting each phase in chronological order with corresponding durations and key tasks, the figure clarifies how each iterative cycle builds upon the previous stage to refine both the pedagogical content and technical infrastructure of the LMS. This systematic timeline also emphasizes the continuous nature of feedback loops, demonstrating how students and teachers input informed ongoing improvements throughout the program's lifecycle.

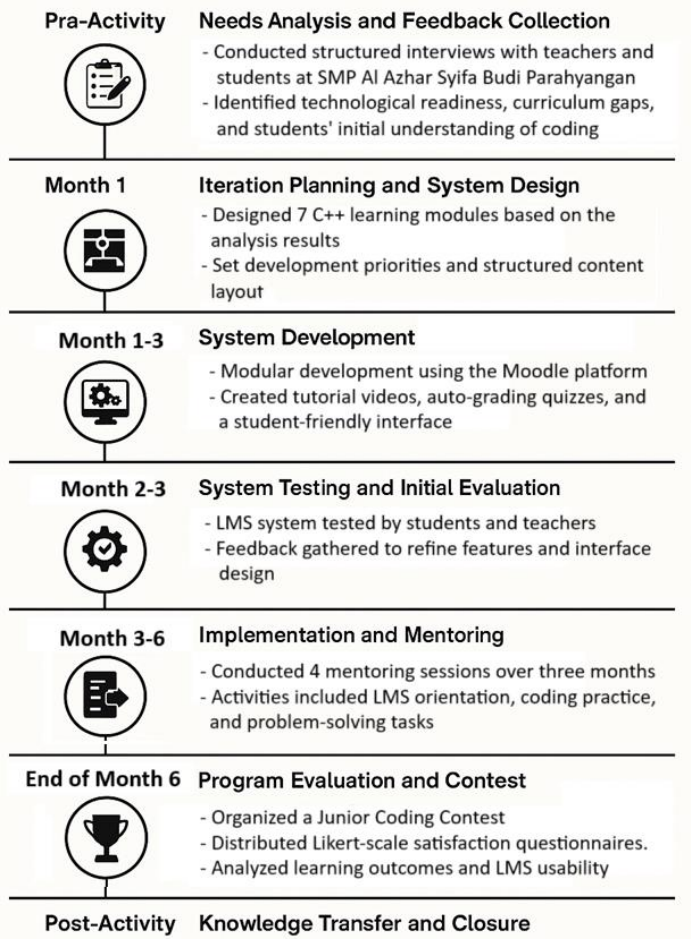


FIGURE 2. Timeline of Activities

Evaluation

Program impact was assessed through a Junior Coding Contest and a Likert-scale satisfaction questionnaire. The questionnaire included indicators such as material clarity, ease of use of LMS, facilitator support, and perceived learning improvement. Data were analyzed to identify student learning outcomes and gather feedback for future system improvement.

https://doi.org/10.35568/abdimas.v8i3.6869

This methodological framework ensured that both technological and pedagogical aspects were addressed, allowing the system to evolve responsively while directly supporting the learning needs of the students.

RESULTS AND DISCUSSION

The implementation of the LMS-based coding program was carried out over four mentoring sessions held at SMP Al Azhar Syifa Budi Parahyangan between February and May 2025. The LMS platform was developed using Moodle and designed specifically for junior high school students with no prior exposure to formal ICT subjects.

LMS Content and Features

The LMS was structured into seven progressive modules covering topics such as data types, conditionals, loops, arrays, functions, and recursion. Each module included:

- Short video tutorials with voice narration and live coding demonstrations
- Interactive quizzes and coding exercises
- Automatic grading using a custom CodeRunner plugin

See Figure 3 for the LMS interface designed for junior high school students, featuring structured modules and navigation adapted for beginners.

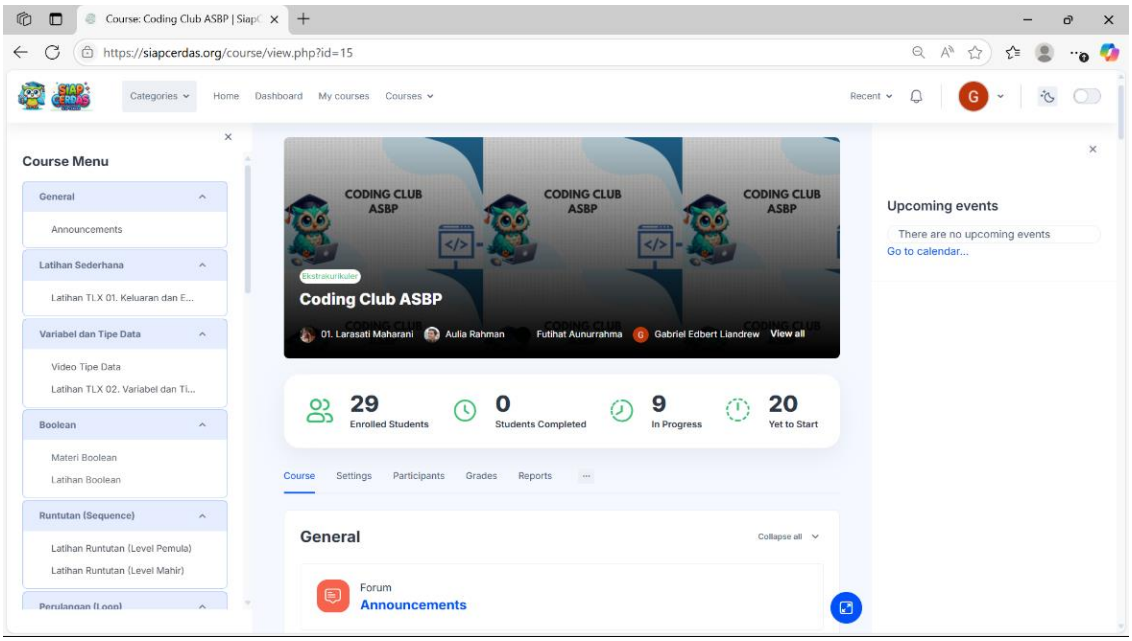
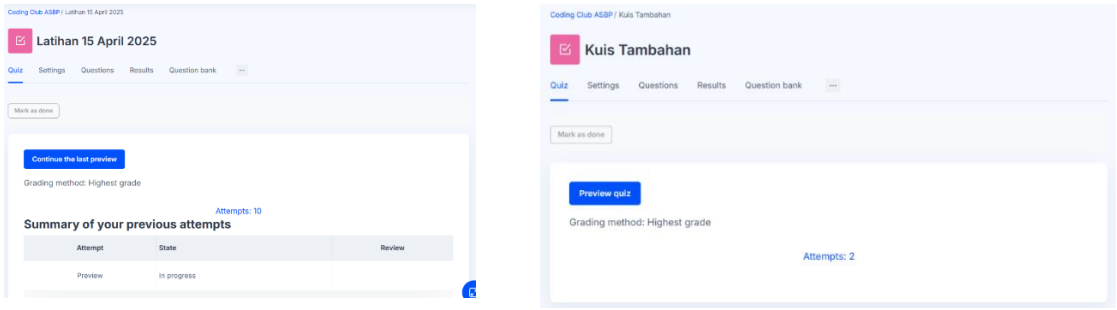


FIGURE 3. User Interface of the LMS Homepage

In total, 231 coding exercises were integrated into the platform, allowing students to receive instant feedback and retry incorrect solutions. The platform supported both synchronous and asynchronous learning modes. In-class sessions involved real-time guided exercises, while students also completed self-paced tasks outside scheduled meetings. See Figure 4 for illustrations of both guided and independent practice scenarios.

https://doi.org/10.35568/abdimas.v8i3.6869



(a) (b)

FIGURE 4. Student practice activities using LMS: (a) in-class synchronous session; (b) independent asynchronous session

These scenarios demonstrate the flexibility of LMS use in both guided and independent settings. In addition to video tutorials and quizzes, students also accessed downloadable teaching modules. These documents provided structured summaries, annotated code samples, and guided exercises to reinforce the lessons. Figure 4 illustrates one of the modules designed for independent use. This LMS-based approach aligns with recent studies on the effectiveness of such systems in secondary education (Zhafira, Simangunsong, & Harahap, 2024). See Figure 5 for an example of a module designed for independent use.

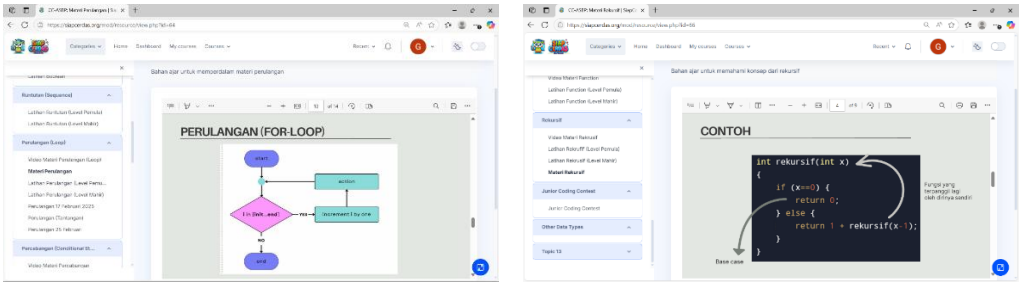


FIGURE 5. Interactive Teaching Module Sample

Prior studies have shown that automatic feedback and scaffolded content delivery are effective in increasing engagement and programming mastery among younger learners (Grover & Pea, 2013). These components reflect key instructional strategies proven effective for teaching programming in secondary education, such as chunking, gamification, and self-paced learning via LMS environments (Salleh, Dewi, Liyana, & Nasir, 2021).

Learning Activities and Mentoring Sessions

Each face-to-face session was designed to guide students through LMS features, assist them in solving problems, and encourage collaboration. Students accessed the platform using laptops provided by the school or their own devices. The facilitators provided support during both synchronous in-class sessions and asynchronous learning outside the classroom. Similar approaches using online platforms for programming education have shown positive engagement among middle school students (Alebaikan et al., 2022).

The program concluded with a Junior Coding Contest, where students independently solved ten randomly generated coding problems using the LMS system. Performance was measured based on accuracy and the number of attempts. In the final contest, performance was ranked based on the number of correct answers and number of attempts: students with fewer attempts received higher rankings in tie

https://doi.org/10.35568/abdimas.v8i3.6869

situations. This competition-based learning model has been shown to foster computational thinking and perseverance in middle school students (Ober et al., 2022).

Evaluation and Student Feedback

To assess satisfaction and perceived effectiveness, a five-point Likert-scale questionnaire was distributed to all 11 participants. Although the total number of participants was relatively small (11 students), the feedback provides valuable insights. Key findings include:

- 45.45% of students agreed or strongly agreed that the program was useful and engaging
- The highest satisfaction score was given to the mentoring support and clarity of materials
- However, 32.73% of responses were neutral, and 21.81% indicated dissatisfaction due to late class schedules and high material complexity

These results suggest that while the LMS and mentoring were generally effective, certain aspects require improvement for broader impact and scalability. Figure 6 illustrates the distribution of student feedback across all five questionnaire indicators, highlighting the variation in satisfaction and areas needing further attention.

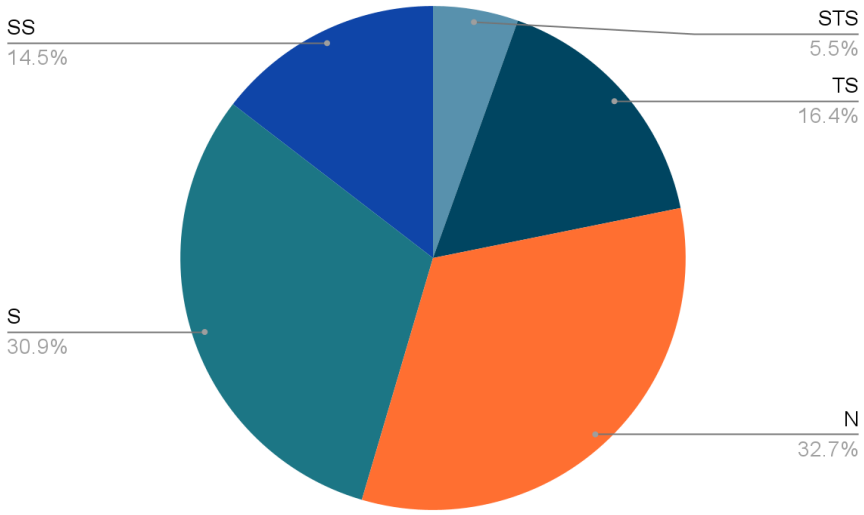


FIGURE 6. Distribution of Participant Feedback

These results confirm that while most participants perceived the program positively, a significant portion faced obstacles that must be addressed in future iterations. Despite the encouraging results, a combined 54.55% of students responded either neutrally or negatively, indicating a need for significant improvement in scheduling, content delivery, and teacher engagement. This evaluation highlights the importance of refining both pedagogical and logistical elements to enhance student outcomes. Future versions may also consider reward systems or gamified progress tracking to boost motivation, especially during self-paced learning.

Challenges and Recommendations

The primary challenges identified were:

- Learning Time: sessions held after regular school hours led to fatigue and lower concentration.

https://doi.org/10.35568/abdimas.v8i3.6869

- Material Complexity: using C++ for students without prior coding experience proved difficult for some participants.
- Limited Teacher Involvement: few teachers were directly engaged in facilitating or continuing the learning process.

To address these issues, future iterations of the program should:

- Integrate sessions into regular class hours
- Begin with block-based programming platforms like Scratch before introducing C++. This aligns with findings that game-based and block-oriented programming activities such as Scratch help lower cognitive barriers for beginners and encourage deeper engagement (Seralidou & Douligeris, 2020).
- Train teachers and provide editable teaching modules to promote sustainability. Similar teacher/community training initiatives focused on digital tools like Google Workspace have proven effective in building digital literacy and sustaining technology-based programs (Kamil et al., 2025).

This recommendation aligns with successful hybrid learning models implemented to enhance digital literacy in Indonesian schools (Zubaidah et al., 2023). These recommendations are expected to support the refinement and sustainability of similar digital learning programs in schools with limited ICT capacity.

CONCLUSION

This study demonstrates that integrating a Learning Management System (LMS) with structured C++ programming content and direct mentoring can enhance digital literacy and programming competence among junior high school students in schools without formal ICT curricula. The development of 231 graded coding problems, instructional videos, and modular content allowed students to engage in self-paced, feedback-driven learning, skills that align with 21st-century learning competencies (Trilling & Fadel, 2009).

Although 45.45% of participants expressed satisfaction with the program, challenges such as scheduling constraints, content difficulty, and limited teacher involvement were noted. These factors highlight the need for more accessible and adaptive instructional approaches.

To improve future implementation, the program should consider integrating into regular class hours, adopting simpler block-based programming environments at the initial stage, and training teachers to lead and sustain the learning process. These enhancements will not only increase participation and comprehension but also establish a sustainable model for informatics education at the middle school level.

Importantly, this study confirms the hypothesis that a tailored LMS, when combined with direct mentoring, scaffolded content, and gamified assessments, can serve as an effective, replicable strategy for advancing digital education in rural and underserved school contexts. Beyond its success in one institution, the LMS model has strong potential to be scaled across similar schools nationwide, especially those constrained by limited infrastructure or lacking formal digital literacy programs. This initiative also aligns with the goals of the Merdeka Belajar policy, which promotes the integration of digital tools into formal education systems in Indonesia (Prayitno & Mahmudi, 2025).

ACKNOWLEDGMENTS

https://doi.org/10.35568/abdimas.v8i3.6869

The authors would like to express their sincere gratitude to SMP Al Azhar Syifa Budi Parahyangan for their cooperation, enthusiasm, and active participation throughout this community service program. Special appreciation is extended to the students at the school's Coding Club, whose involvement was crucial in the implementation and evaluation of the LMS.

We also thank the Faculty of Informatics, Telkom University, for funding and supporting this initiative as part of the university's internal community service grant scheme. This activity was carried out under the mandate of Tri Dharma Perguruan Tinggi and would not have been possible without the support of student facilitators, teachers, and school administrators who contributed to the success and continuity of the program.

REFERENCES

Alebaikan, R., Alsaif, A., & Alnahdi, G. (2022). Experiences of middle school programming in an online learning environment. Journal of Educational Technology & Society, 25(2), 45–60.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. Educational Researcher, 42(1), 38–43.

Irianti, L., Rachmawati, E., & Friatin, L. Y. (n.d.). Pelatihan guru-guru Bahasa Inggris dalam penggunaan flipped classroom model di era literasi digital. Jurnal Abdimas Umtas: Jurnal Pengabdian kepada Masyarakat, 4(2), 754–759.

Kamil, I., Yolifiandri, Wahyuningsih, E., Irawan, I. A., & Azkiah, N. (2025). Utilization Of Google Workspace Technology To Improve Understanding Of Accounting And Financial Reporting In Ukm Kemuning Mandiri. ABDIMAS: Jurnal Pengabdian Masyarakat, 8(2), 902–911. https://doi.org/10.35568/abdimas.v8i2.6414

Ober, T. M., Cheng, Y., Coggins, M. R., & others. (2022). Charting a path for growth in middle school students' attitudes toward computer programming. Computer Science Education, 34, 4–36.

Pattichis, M. S., Kameas, A., & Kameas, X. (2024). Teaching computer programming using mathematics: An integrative approach. International Journal of STEM Education, 11(1), 112–130.

Prayitno, M., & Mahmudi, M. R. (n.d.). Effectiveness of the Merdeka Belajar policy: A case study. Journal of Curriculum Studies, 18(1), 22–38.

Salleh, F., Dewi, D. A., Liyana, N., & Nasir, N. R. M. (n.d.). A model for teaching and learning programming subjects in public secondary schools of Malaysia. Proceedings of the 2021 International Conference on Computing and Informatics, 362–373.

Seralidou, E., & Douligeris, C. (2020). Learning programming by creating games through the use of structured activities in secondary education in Greece. Education and Information Technologies, 26, 859–898.

Trilling, B., & Fadel, C. (2009). 21st Century Skills: Learning for Life in Our Times. Jossey-Bass.

Zhafira, D., Simangunsong, I. P., & Harahap, S. S. (n.d.). Analyzing the utilization of learning management systems in secondary schools. Indonesian Journal of Educational Technology, 10(3), 77–95.

Zubaidah, S., Nugroho, H., & Susanto, M. R. (2023). Penguatan literasi digital siswa melalui pendekatan hybrid learning. Jurnal Pendidikan Informatika, 9(2), 122–130.

https://doi.org/10.35568/abdimas.v8i3.6869