



RESEARCH ARTICLE

Mendorong Kreativitas Digital: Peran Literasi Koding dalam Pengembangan Teknologi Pembelajaran

Amilia Khasanah Marzuki

Universitas Adi Buana Surabaya

Author email: 250020069@student.unipasby.ac.id

Riwayat Artikel

Pengiriman: 10 April 2026

Revisi: 6 Oktober 2026

Diterima: 9 Oktober 2026

Dipublikasikan: 12 Oktober 2026

Informasi Publikasi

DOI: 10.36456/7k7jg477

Volume/Issue: Vol 2 No 2 (2025)

Pages: 7 - 13

Kata Kunci

literasi koding; berpikir komputasional; kreativitas digital; teknologi pembelajaran; instructional design

ABSTRAK

Artikel ini bertujuan menganalisis peran literasi koding sebagai kompetensi pedagogis yang mendorong kreativitas digital dalam pengembangan teknologi pembelajaran. Penelitian menggunakan studi literatur naratif dengan sintesis tematik terhadap artikel jurnal terindeks, proceeding akademik, dan dokumen kebijakan yang relevan pada rentang 2020–2025, serta melibatkan karya-karya seminal yang menjadi landasan konseptual. Penelusuran dilakukan menggunakan kata kunci coding literacy, computational thinking, digital creativity, educational technology, dan low-code/no-code, kemudian sumber diseleksi berdasarkan relevansi topik, kredibilitas publikasi, dan keterkaitan dengan konteks pendidikan. Hasil kajian menunjukkan bahwa literasi koding berperan pada tiga level. Pertama, memperkuat computational thinking melalui aktivitas dekomposisi, abstraksi, algoritma, debugging, dan evaluasi. Kedua, menjembatani teori konstruktivisme–konstruksionisme dengan praktik desain pembelajaran berbasis proyek, prototyping, dan pengembangan media interaktif. Ketiga, mentransformasi peran pendidik dan teknolog pendidikan dari pengguna teknologi menjadi perancang pengalaman belajar. Kajian juga menunjukkan bahwa pendekatan low-code/no-code berpotensi menurunkan hambatan teknis, tetapi implementasinya tetap memerlukan scaffolding pedagogis, dukungan infrastruktur, dan integrasi kurikulum. Implikasinya, program studi Teknologi Pendidikan dan pengembangan profesional guru perlu menempatkan literasi koding sebagai kompetensi desain, bukan sekadar keterampilan teknis. Dengan demikian, literasi koding layak diposisikan sebagai fondasi pengembangan teknologi pembelajaran yang kreatif, kontekstual, dan berkelanjutan.

PENDAHULUAN

Transformasi digital, kecerdasan artifisial, dan ekonomi berbasis pengetahuan telah mengubah cara pendidikan memaknai kompetensi abad ke-21. Pendidikan tidak lagi cukup menempatkan peserta didik sebagai pengguna teknologi, tetapi juga perlu menyiapkan mereka menjadi perancang solusi, kreator media, dan pemecah masalah yang adaptif. Dalam kerangka Education 4.0, kreativitas, inovasi, pemecahan masalah kompleks, dan kecakapan teknologi dipandang sebagai kompetensi utama yang harus dibangun melalui pengalaman belajar yang lebih aktif dan produktif (World Economic Forum, 2020).

Di antara kompetensi tersebut, literasi koding menempati posisi strategis karena menyediakan bahasa kerja untuk menerjemahkan gagasan menjadi artefak digital yang dapat diuji, diperbaiki, dan dikembangkan. Literasi koding tidak hanya berkaitan dengan penguasaan sintaks, tetapi juga melibatkan dekomposisi masalah, abstraksi, algoritma, debugging, dan refleksi. Dalam kerangka Brennan dan Resnick (2012), praktik-praktik tersebut membentuk computational thinking melalui konsep, praktik, dan perspektif komputasional. Sejalan dengan itu, Nouri et al. (2020) menunjukkan bahwa aktivitas pemrograman mendorong perkembangan computational thinking, kompetensi digital, kolaborasi, dan creative problem solving, sedangkan Jiang dan Li (2021)

menemukan bahwa pembelajaran Scratch berkontribusi positif terhadap kreativitas, cooperativity, dan critical thinking peserta didik.

Meskipun demikian, praktik dan literatur yang ada masih cenderung memposisikan koding sebagai keterampilan teknis atau sekadar luaran belajar peserta didik. Kajian mengenai Scratch dan pembelajaran pemrograman lebih banyak berfokus pada efektivitas alat, penggunaan di kelas, atau peningkatan skor computational thinking (Montiel & Gomez-Zermeño, 2021). Di sisi lain, studi tentang low-code/no-code menyoroti manfaat, tantangan adopsi, dan persepsi guru terhadap citizen development, tetapi belum banyak menjelaskan bagaimana kompetensi tersebut diintegrasikan ke dalam kerja desain pembelajaran dan pengembangan teknologi pembelajaran (Ajimati et al., 2025; McHugh et al., 2024). Dengan kata lain, masih terbatas kajian yang secara eksplisit menempatkan literasi koding sebagai kompetensi pedagogis inti bagi pendidik dan teknolog pendidikan.

Berdasarkan celah tersebut, artikel ini bertujuan menganalisis peran literasi koding sebagai kompetensi pedagogis yang mendorong kreativitas digital dalam pengembangan teknologi pembelajaran. Artikel ini secara khusus menjelaskan bagaimana literasi koding: (1) memperkuat computational thinking, (2) menjembatani teori belajar dengan praktik instructional design, dan (3) mentransformasi peran pendidik dan teknolog pendidikan melalui pemanfaatan pendekatan low-code/no-code. Kontribusi utama artikel ini adalah menawarkan kerangka konseptual yang mengintegrasikan computational thinking, konstruksionisme, dan model ADDIE ke dalam satu alur argumentatif yang lebih operasional bagi pengembangan kurikulum, pendidikan guru, dan profesi Teknologi Pendidikan.

METODE

Penelitian ini menggunakan studi literatur kualitatif dengan orientasi narrative review dan sintesis tematik. Pendekatan ini dipilih karena fokus penelitian bersifat konseptual dan bertujuan menyusun argumentasi integratif mengenai posisi literasi koding dalam bidang teknologi pembelajaran, bukan menguji hubungan kausal atau menghitung effect size seperti pada meta-analisis.

Penelusuran sumber dilakukan pada basis data akademik yang lazim digunakan dalam kajian pendidikan, terutama Google Scholar, ERIC, dan Scopus, dengan kombinasi kata kunci coding literacy, computational thinking, digital creativity, educational technology, instructional design, low-code/no-code, dan programming education. Rentang waktu yang diprioritaskan adalah publikasi 2020–2025 agar pembahasan merepresentasikan perkembangan mutakhir, namun karya-karya seminal seperti Papert (1980), Wing (2006), dan Brennan dan Resnick (2012) tetap dimasukkan sebagai landasan teoretis.

Sumber yang diikutkan meliputi artikel jurnal peer-reviewed, proceeding akademik, dan dokumen kebijakan dari lembaga kredibel yang secara langsung membahas salah satu atau beberapa tema berikut: literasi koding, computational thinking, kreativitas digital, desain pembelajaran, serta penggunaan low-code/no-code dalam konteks pendidikan. Sumber yang bersifat duplikatif, terlalu umum, atau tidak memiliki implikasi pendidikan yang jelas dieliminasi dari sintesis.

Analisis dilakukan melalui tiga tahap. Pertama, screening dan kategorisasi sumber berdasarkan fokus tema. Kedua, open coding untuk mengidentifikasi tema-tema utama seperti peran koding dalam membangun computational thinking, hubungan koding dengan kreativitas digital, kontribusi terhadap instructional design, serta tantangan implementasi. Ketiga, perbandingan antarsumber dilakukan untuk menemukan titik temu, perbedaan penekanan, dan implikasi praktis. Dengan prosedur ini, pembahasan tidak berhenti pada deskripsi pustaka, tetapi

diarahkan pada dialog antarsumber dan perumusan kerangka konseptual yang relevan bagi bidang Teknologi Pendidikan.

HASIL DAN PEMBAHASAN

Literasi Koding sebagai Fondasi Kreativitas Digital

Sintesis literatur menunjukkan bahwa literasi koding merupakan medium yang memungkinkan peserta didik dan pendidik mengekspresikan gagasan dalam bentuk artefak digital yang dapat dimodifikasi secara iteratif. Dalam perspektif computational thinking, proses tersebut mengaktifkan dekomposisi, abstraksi, algoritma, debugging, dan evaluasi (Wing, 2006). Hasil penelitian menambahkan bahwa koding juga membentuk perspektif komputasional, yaitu kemampuan memandang diri sebagai kreator dan memandang teknologi sebagai medium ekspresi, bukan sekadar alat konsumsi(Brennan dan Resnick (2012)).

Dialog dengan penelitian terdahulu memperlihatkan bahwa hubungan antara koding, computational thinking, dan kreativitas digital tidak bersifat otomatis, tetapi muncul ketika aktivitas pemrograman dirancang sebagai proses membuat, menguji, memperbaiki, dan merefleksikan karya. Hasil penelitian menunjukkan bahwa pembelajaran pemrograman di sekolah mendorong perkembangan digital competence, collaborative attitudes, dan creative problem solving(Nouri et al. (2020)). Temuan penelitian menguatkan hal tersebut, Scratch meningkatkan kreativitas, cooperativity, dan critical thinking, meskipun peningkatan algorithmic thinking dan problem solving memerlukan durasi belajar dan scaffolding yang lebih kuat(Jiang dan Li (2021)). Artinya, kreativitas digital lahir ketika literasi koding diposisikan sebagai praktik design-making-reflection, bukan latihan sintaks yang terputus dari konteks pembelajaran.

Menjembatani Teori Belajar dan Praktik Desain Pembelajaran

Secara teoretis, literasi koding beririsan kuat dengan konstruktivisme dan konstruksionisme. Menurut ahli menegaskan bahwa peserta didik membangun pengetahuan secara lebih bermakna ketika mereka membuat objek yang dapat diamati, diuji, dan didiskusikan(Papert (1980)). Dalam konteks ini, aktivitas membuat simulasi, animasi, game edukasi, atau media interaktif melalui koding menjadi bentuk nyata belajar melalui penciptaan. Koding menyediakan ruang bagi peserta didik untuk memadukan logika, imajinasi, dan refleksi kritis di dalam satu proses pengembangan.

Jika dibandingkan dengan kajian penelitian yang menekankan Scratch sebagai alat inovatif untuk menumbuhkan computational thinking (Montiel dan Gomez-Zermeño (2021)), artikel ini menempatkan literasi koding lebih jauh sebagai penghubung antara teori belajar dan alur instructional design. Dalam model ADDIE, literasi koding dapat berperan pada tahap analisis untuk memetakan kebutuhan belajar dan kelayakan teknis, pada tahap desain untuk merumuskan alur interaksi dan umpan balik, pada tahap pengembangan untuk membangun prototipe media, pada tahap implementasi untuk menguji penggunaan di kelas atau LMS, dan pada tahap evaluasi untuk membaca data interaksi serta melakukan debugging. Dengan posisi tersebut, koding bukan aktivitas tambahan, melainkan bagian dari kompetensi desain pembelajaran itu sendiri.

Table 1. Integrasi Literasi Koding dalam Pengembangan Teknologi Pembelajaran

Tahap	Landasan Konseptual	Praktik Literasi Koding	Implikasi Bagi Teknologi Pendidikan
Input	Computational thinking, konstruktivisme	Unplugged coding, block-based coding, pengenalan logika dasar	Membangun kesiapan kognitif, literasi digital, dan orientasi problem solving

Tahap	Landasan Konseptual	Praktik Literasi Koding	Implikasi Bagi Teknologi Pendidikan
Desain	Instructional design ADDIE	Perancangan alur, algoritma, interaksi, dan umpan balik pembelajaran	Mendukung lahirnya media dan pengalaman belajar yang kontekstual
Pengembangan	Learning-by-design, project-based learning	Pembuatan prototipe simulasi, animasi, game edukasi, atau aplikasi sederhana	Menguatkan peran teknolog pendidikan sebagai learning architect
Implementasi	Constructivist learning environment	Integrasi produk hasil koding ke kelas, LMS, atau aktivitas proyek	Meningkatkan keterlibatan, kebermaknaan, dan kolaborasi belajar
Evaluasi	Reflective practice, learning analytics	Debugging, revisi kode, pembacaan data interaksi, refleksi desain	Mendorong perbaikan berkelanjutan berbasis bukti
Outcome	Digital creativity, 21st century skills	Artefak digital edukatif yang dapat diuji dan dibagikan	Peserta didik dan pendidik bergerak dari pengguna menuju kreator teknologi pembelajaran

Tabel 1 menunjukkan bahwa literasi koding dapat dibaca sebagai kompetensi lintas-tahap dalam pengembangan teknologi pembelajaran. Kerangka ini menegaskan bahwa praktik koding memiliki relevansi pedagogis sejak analisis kebutuhan hingga evaluasi, sehingga nilai utamanya tidak berhenti pada penguasaan bahasa pemrograman, tetapi pada kapasitas merancang pengalaman belajar yang bermakna.

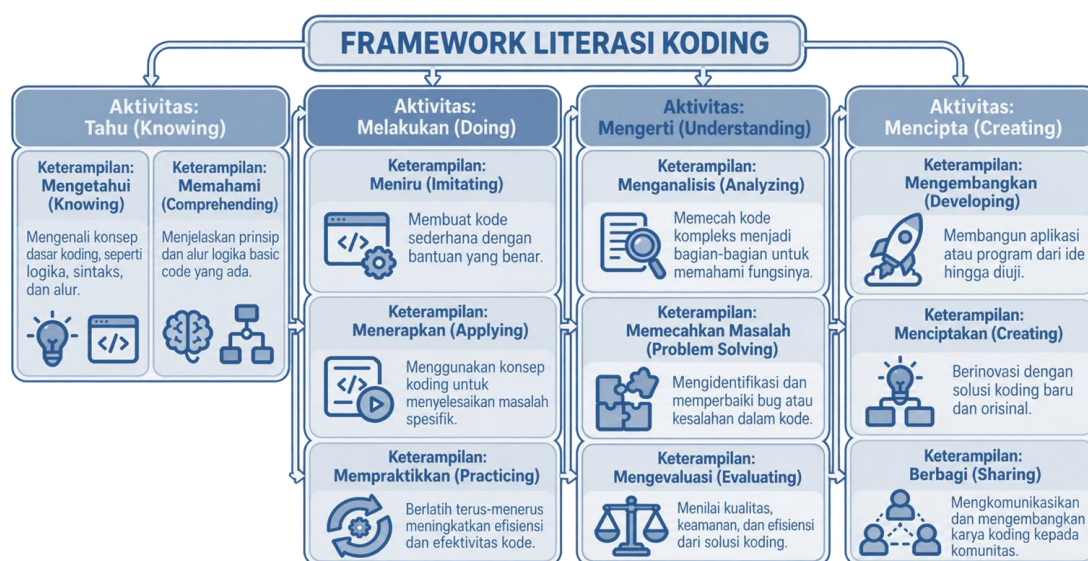
Transformasi Peran Teknolog Pendidikan

Literasi koding memperluas horizon kerja teknolog pendidikan dari pengguna perangkat menjadi perancang solusi belajar. Pada tahap analisis, teknolog pendidikan yang memahami koding dapat lebih kritis menilai kelayakan pedagogis dan teknis suatu media. Pada tahap desain dan pengembangan, mereka lebih siap membuat prototipe cepat, mengadaptasi alur interaksi, atau menghubungkan berbagai layanan digital. Pada tahap implementasi dan evaluasi, literasi koding mendukung pembacaan log data, perbaikan bug, serta penyempurnaan pengalaman belajar secara berkelanjutan.

Temuan ini sejalan dengan Ajimati et al. (2025) yang menunjukkan bahwa low-code/no-code memberi manfaat berupa empowerment, innovation development, process improvement, dan usability, tetapi sekaligus menghadirkan tantangan seperti kesenjangan keterampilan, keamanan, governance, dan fragmentasi platform. Dalam konteks pendidikan menengah, McHugh et al. (2024) juga menegaskan bahwa low-code/no-code dapat menjadi jalan masuk yang lebih inklusif bagi guru untuk mengintegrasikan aspek-aspek ilmu komputer tanpa menuntut mereka menjadi programmer profesional. Dibandingkan kedua studi tersebut, artikel ini menambahkan penekanan bahwa low-code/no-code di bidang Teknologi Pendidikan harus dipahami sebagai enabler pedagogis: alat untuk mempercepat desain, prototyping, dan validasi media pembelajaran, bukan sekadar instrumen efisiensi teknis.

Gambar 1 memperlihatkan progresi literasi koding dari tahap knowing, doing, understanding, hingga creating. Progresi ini penting karena menegaskan bahwa tujuan pendidikan koding tidak berhenti pada mengenal sintaks, tetapi bergerak menuju kemampuan menganalisis, mengevaluasi, mengembangkan, dan membagikan solusi digital. Dengan demikian, kreativitas

digital dipahami sebagai hasil dari akumulasi praktik berpikir dan praktik desain, bukan hasil instan dari penguasaan alat tertentu.



Gambar 1. Framework literasi koding sebagai progresi dari knowing hingga creating

Dialog dengan Penelitian Terdahulu dan Implikasi

Kekuatan utama artikel ini terletak pada upaya mempertemukan beberapa rumpun kajian yang selama ini berjalan relatif terpisah. Nouri et al. (2020) dan Jiang dan Li (2021) terutama membahas perkembangan keterampilan peserta didik ketika belajar pemrograman. Montiel dan Gomez-Zermeño (2021) menyoroti Scratch sebagai perangkat inovatif untuk pendidikan K-12. Sementara itu, Ajimati et al. (2025) dan McHugh et al. (2024) berbicara tentang adopsi low-code/no-code serta persepsi guru terhadap citizen development. Artikel ini menghubungkan seluruh rumpun tersebut dengan memosisikan literasi koding sebagai kompetensi pedagogis inti yang menopang instructional design, produksi media, dan pengembangan kreativitas digital dalam bidang Teknologi Pendidikan. Dengan kata lain, kebaruan artikel ini bukan pada klaim bahwa koding itu penting, melainkan pada cara artikel ini merumuskan posisi strategis koding di dalam ekosistem pengembangan pembelajaran.

Implikasi praktis dari sintesis ini setidaknya ada tiga. Pertama, kurikulum program studi Teknologi Pendidikan perlu memasukkan literasi koding sebagai kompetensi desain dan prototyping, bukan sekadar mata kuliah teknis. Kedua, pengembangan profesional guru sebaiknya dimulai dari identifikasi masalah belajar, perancangan pengalaman belajar, dan pemilihan alat yang relevan; low-code/no-code dapat menjadi jembatan yang realistis bagi guru non-IT. Ketiga, asesmen literasi koding perlu menilai proses berpikir dan proses desain melalui artefak proyek, dokumentasi algoritma, jejak debugging, serta refleksi penggunaan teknologi secara etis dan pedagogis.

Tantangan Implementasi dan Arah Lanjut

Meskipun menjanjikan, integrasi literasi koding dalam teknologi pembelajaran menghadapi sejumlah tantangan yang perlu dipertimbangkan secara kritis.

- Reduksi pedagogis ke sintaks.** Jika pembelajaran terlalu berfokus pada aturan penulisan kode, tujuan pedagogis dapat bergeser dari pemecahan masalah dan desain pembelajaran ke latihan teknis yang sempit. Oleh sebab itu, konteks masalah dan tujuan belajar harus selalu menjadi titik tolak sebelum alat dan bahasa dipilih.

- (b) Beban kognitif bagi pendidik non-IT. Guru atau teknolog pendidikan yang belum memiliki pengalaman pemrograman membutuhkan jalur masuk yang bertahap. Temuan McHugh et al. (2024) memperlihatkan bahwa keterbukaan guru terhadap low-code/no-code tinggi, tetapi hambatan psikologis dan dukungan institusional tetap menentukan keberhasilan adopsi.
- (c) Kesenjangan akses dan keberlanjutan. Adopsi platform digital masih sangat dipengaruhi ketersediaan perangkat, konektivitas, dan dukungan teknis. Ajimati et al. (2025) juga menunjukkan bahwa fragmentasi platform, isu keamanan, dan masalah governance dapat melemahkan keberlanjutan implementasi bila tidak diantisipasi sejak awal.

Berdasarkan tantangan tersebut, arah tindak lanjut yang disarankan adalah integrasi lintas disiplin, scaffolding bertahap dari unplugged coding ke block-based coding lalu low-code/no-code atau text-based coding sesuai kebutuhan, serta penguatan budaya kolaboratif antara guru, teknolog pendidikan, dan pengembang. Penelitian berikutnya perlu menguji secara empiris kerangka konseptual yang diusulkan dalam konteks desain media, pengembangan LMS, atau proyek pembelajaran berbasis masalah di jenjang sekolah menengah dan pendidikan tinggi.

KESIMPULAN

Literasi koding tidak lagi memadai dipahami hanya sebagai keterampilan teknis untuk menulis program. Dalam bidang Teknologi Pendidikan, literasi koding layak diposisikan sebagai kompetensi pedagogis yang memungkinkan pendidik dan teknolog pendidikan membangun computational thinking, merancang pengalaman belajar, membuat prototipe media, dan menghasilkan artefak digital yang kreatif serta kontekstual. Sintesis literatur menunjukkan bahwa nilai penting literasi koding terletak pada kemampuannya menghubungkan proses berpikir, proses desain, dan proses produksi pembelajaran di dalam satu ekosistem kerja yang iteratif.

Pendekatan low-code/no-code membuka peluang implementasi yang lebih inklusif, tetapi tidak dapat berdiri sendiri tanpa scaffolding pedagogis, dukungan institusional, dan asesmen yang menilai proses berpikir secara autentik. Karena itu, pengembangan kurikulum, pendidikan guru, dan profesi Teknologi Pendidikan perlu memberi ruang yang lebih eksplisit bagi literasi koding sebagai kompetensi desain. Dengan langkah tersebut, literasi koding dapat menjadi fondasi yang kuat bagi pengembangan teknologi pembelajaran yang kreatif, relevan, dan berkelanjutan.

DEKLARASI

Pendanaan

Penelitian ini tidak menerima pendanaan eksternal.

Konflik Kepentingan

Penulis menyatakan tidak terdapat konflik kepentingan.

Persetujuan Etik dan Persetujuan Setelah Penjelasan

Tidak berlaku.

Pernyataan Penggunaan AI

Grammarly digunakan untuk membantu penyuntingan bahasa. Penulis tetap bertanggung jawab untuk memverifikasi keakuratan dan integritas isi akhir manuskrip.

PUSTAKA

- Ajimati, M. O., Carroll, N., & Maher, M. (2025). Adoption of low-code and no-code development: A systematic literature review and future research agenda. *Journal of Systems and Software*, 222, Article 112300. <https://doi.org/10.1016/j.jss.2024.112300>
- Angeli, C., & Valanides, N. (2020). Developing and validating a scale for the assessment of preservice teachers' computational thinking. *Educational Technology Research and Development*, 68(3), 1323–1343.
- Bahrour, Z., Anane, C., Ahmed, V., & Zacca, A. (2023). Transforming education: A comprehensive review of generative artificial intelligence in educational settings through bibliometric and content analysis. *Sustainability*, 15(17), Article 12983. <https://doi.org/10.3390/su151712983>
- Bandonio, A., & Pramujiono, A. (2025). Landasan teknologi pendidikan. Madza Media.
- Brennan, K. A., & Resnick, M. (2012, April 16). Using artifact-based interviews to study the development of computational thinking in interactive media design. American Educational Research Association Annual Meeting, Vancouver, BC, Canada. <https://scratched.gse.harvard.edu/stories/scratch-and-scratched-aera-2012>
- Chao, P. Y. (2022). The impact of a no-code web development course on students' attitudes toward computing. *Journal of Educational Computing Research*, 60(2), 488–513
- Dúo-Terrón, P. (2023). Analysis of Scratch software in scientific production for 20 years: Programming in education to develop computational thinking and STEAM disciplines. *Education Sciences*, 13(4), Article 404. <https://doi.org/10.3390/educsci13040404>
- Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42(1), 38–43. <https://doi.org/10.3102/0013189X12463051>
- Li, Q (2023). The effect of Scratch programming on primary students' digital creativity.
- Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51–61. <https://doi.org/10.1016/j.chb.2014.09.012>
- Marifah, S. N., Mu'iz, D. A., Wahid, R., M. (2022). Systematic Literatur Review : Integrasi Computational Thinking dalam Kurikulum Sekolah Dasar di Indonesia. *Creative of Learning Students Elementary*, 5(5), 928-938. <https://doi.org/10.22460/collase.v5i5.12148>
- McHugh, S., Carroll, N., & Connolly, C. (2024). Low-code and no-code in secondary education—Empowering teachers to embed citizen development in schools. *Computers in the Schools*, 41(4), 399–424. <https://doi.org/10.1080/07380569.2023.2256729>
- Moreno-León, J., & Robles, G. (2020). The effect of integrated Scratch lessons on computational thinking skills. *IEEE Transactions on Education*, 63(2), 113–119
- Sari, R. P (2023). Tantangan guru dalam implementasi literasi digital dan koding di Indonesia.
- Su, J., & Yang, W. (2023). A systematic review of integrating computational thinking in early childhood education. *Computers and Education Open*, 4, Article 100122. <https://doi.org/10.1016/j.caeo.2023.100122>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>
- World Economic Forum. (2020). Schools of the future: Defining new models of education for the fourth industrial revolution. <https://www.weforum.org/publications/schools-of-the-future-defining-new-models-of-education-for-the-fourth-industrial-revolution/>
- Zhang, L., & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education*, 141, Article 103607. <https://doi.org/10.1016/j.compedu.2019.103607>