Adaptive Gamification Model Using Scratch to Enhance Computational Thinking and Real-World Case-Handling Skills in Higher Education

Resmi Darni [1], Yulyanti Harisman [2], Muhammad Anwar [1], Hary Suswanto [3], Evania Yafie [4], Achmad Hamdan [3], Andhika Herayono [1]

[1] *Faculty of Engineering, Universitas Negeri Padang, Padang, West Sumatra, Indonesia*
[2] *Faculty of Mathematics and Natural Sciences, Universitas Negeri Padang, Padang, West Sumatra, Indonesia*
[3] *Faculty of Engineering, Universitas Negeri Malang, Malang, East Java, Indonesia*
[4] *Faculty of Education, Universitas Negeri Malang, Malang, East Java, Indonesia*

***Abstract* – This study aims to develop and evaluate an adaptive gamification model using Scratch to enhance computational thinking (CT) and real-world case-handling skills among university students. Grounded in the ADDIE development framework, the model was implemented in a quasi-experimental setting involving control and experimental groups. Data were collected through pre- and post-tests, reflective journals, and observation rubrics focusing on CT dimensions (decomposition, abstraction, algorithm design, etc.) and problem-handling performance. Quantitative results showed a significant increase in the experimental group's scores compared to the control group. Chi-Square analysis also revealed a significant association between the instructional model and students' success rates, confirming the model's effectiveness. Qualitative findings supported these results, highlighting increased motivation, cognitive engagement, and the ability to translate coding concepts into real-life scenarios.**

This study concludes that adaptive gamification fosters deeper learning and practical skill development in programming education.

***Keywords* – Gamification, Scratch, computational thinking, real-world problem-solving, adaptive learning.**

DOI: 10.18421/TEM153-97
https://doi.org/10.18421/TEM153-97

Corresponding author: Resmi Darni,
Faculty of Engineering, Universitas Negeri Padang, Padang, West Sumatra, Indonesia
Email: resmidarni@ft.unp.ac.id

Received: 12 July 2025.
Revised: 03 February 2026.
Accepted: 01 April 2026.
Published: 27 August 2026.

1. Introduction

Era of rapid digital transformation is where the ability to think computationally and solve real-world problems is considered a core competency in the field of informatics and software engineering [1]. Despite its significance, many graduates from higher education institutions still struggle to meet the demands of the digital industry, especially in applying programming skills to authentic case scenarios. A tracer study conducted by Universitas Negeri Padang (2024) revealed that nearly 30% of informatics graduates lack programming skills aligned with industry requirements, highlighting a significant skills gap that must be addressed through innovative pedagogical interventions [2]. Traditional methods of teaching programming often emphasize syntax and theoretical constructs rather than practical application and contextual problem-solving. This approach limits students' engagement, creativity, and ability to independently formulate solutions using computational principles [3]. Furthermore, many students fail to transfer theoretical knowledge into practical solutions that meet the expectations of today's fast-evolving technology sector [4]. As a result, graduates may face difficulties in adapting to real work environments where critical thinking, case-handling, and adaptive decision-making are essential.

Gamification has emerged as a promising pedagogical strategy that leverages game elements such as points, levels [5], rewards, and challenges to enhance student motivation and engagement. Studies have shown that integrating gamification into coding instruction can significantly increase students active participation, motivation, and comprehension of complex programming concepts [6]. However, many gamification approaches remain generic and fail to connect students with real-world problem-solving contexts or industry-relevant skill development [7]. To bridge this gap, this study proposes an adaptive gamification model that incorporates Scratch-based coding activities within a combined project-based learning (PjBL) and challenge-based learning (CBL) framework. This model emphasizes authentic learning experiences that simulate real-life problem-solving scenarios, allowing students to design, develop, and debug functional games that reflect real-world applications [8]. Moreover, the adaptive nature of the model ensures that learning challenges align with individual student competencies, fostering deeper engagement and mastery of computational skills [9].

The integration of Scratch in higher education provides an intuitive and visual approach to programming that helps students develop algorithmic thinking without the cognitive overload of complex syntax [10]. When combined with gamified challenges and meaningful feedback, Scratch becomes a powerful tool for cultivating computational thinking, creativity, and real skill case-handling abilities [11]. This study is grounded in several intersecting theoretical perspectives that support the integration of gamification in educational settings to improve computational thinking and real-world case-handling abilities. One of the core theoretical foundations is the Self-Determination Theory (SDT) developed, which emphasizes the importance of intrinsic motivation in learning [12]. SDT proposes that learning environments which support autonomy, competence, and relatedness foster greater engagement and sustained motivation [13]. Gamification, when designed appropriately, has the potential to fulfill these psychological needs by offering students choices in how they engage with tasks (autonomy), clear indicators of progress (competence), and collaborative or competitive elements that enhance social connection (relatedness). These motivational dynamics are critical in promoting persistence in complex learning domains such as programming [14].

The cognitive basis of the study draws on computational thinking as conceptualized Wing and Papert, who highlight it as a structured way of solving problems that can be applied across disciplines [15].

Computational thinking encompasses key processes such as decomposition, pattern recognition, abstraction, and algorithm design. These components align naturally with the learning outcomes of programming courses and can be effectively fostered through structured, challenge-based activities [16]. As students engage with programming challenges through gamified tasks, they practice breaking down problems into smaller parts, identifying recurring patterns, filtering out irrelevant information, and constructing step-by-step solutions all of which mirror real-world coding practices. Furthermore, this research is informed by the principles of Project-Based Learning (PjBL) and Challenge-Based Learning (CBL) [17], [18]. These pedagogical approaches emphasize student-centered learning through the completion of meaningful, context-rich tasks that reflect authentic problems. PBL allows students to construct knowledge by developing digital products such as games or applications, while CBL emphasizes iterative experimentation and the application of conceptual understanding in dynamic problem contexts. Integrating gamification into these frameworks enables learners to receive continuous feedback, maintain sustained engagement, and develop practical competencies that mirror the expectations of the technology industry [19].

The use of Scratch as the coding platform supports the theoretical objective of lowering the entry barrier for learners by offering a visual, block-based programming environment [20]. Scratch simplifies the abstraction of complex programming logic, making it accessible to novice learners while still providing room for sophisticated problem-solving. When embedded in a gamified, adaptive instructional design, Scratch becomes not just a learning tool but a simulated workplace environment where students iteratively build and refine their coding solutions [10]. Overall, this study is situated at the intersection of motivational theory, cognitive development, and applied pedagogy, where gamification serves as both a motivational enhancer and a cognitive scaffold. By aligning gamified instructional design with computational thinking, case-handling, and adaptive challenge mechanisms, the proposed model aims to produce deeper learning outcomes that are both transferable and industry-relevant [21].

This study aims to design, implement, and evaluate the effectiveness of the proposed adaptive gamification model in enhancing students' computational thinking and real-world problem-solving skills [22]. It contributes not only to the development of innovative instructional methods in computer science education but also to efforts in aligning academic outcomes with the digital industry's evolving competency demands.

2. Methodology

This research employed a mixed-method approach, combining Research and Development (R&D) using the ADDIE instructional design model with a quasi-experimental method to evaluate the effectiveness of the developed gamification model. The ADDIE framework comprising Analysis, Design, Development, Implementation, and Evaluation was adopted to guide the systematic design and integration of gamification into a coding course using the Scratch platform [23]. Each phase of ADDIE was tailored to support the construction of an adaptive learning model that aligns with the competencies demanded in the digital industry. During the Analysis phase, a needs assessment was conducted to identify gaps in students' computational thinking and real-world case-handling abilities. This involved analyzing tracer study data, curriculum relevance, and feedback from instructors and industry stakeholders. The Design phase involved creating instructional blueprints, defining learning outcomes, and mapping out gamification elements such as badges, levels, and real-case coding tasks [24], [25]. The Development phase focused on creating the instructional content and gamified modules within Scratch, including scenarios that simulate real industry challenges. During the Implementation stage, the gamification model was deployed in a classroom setting involving undergraduate informatics students. Finally, the Evaluation phase was conducted through pre- and post-tests, rubric-based project assessments, and student feedback to assess the model's effectiveness and usability [26].

To assess the impact of the model, a quasi-experimental research design was applied involving two student groups: An experimental group of 30 students who engaged with the gamified Scratch-based coding model, and a control group of 30 students who received conventional instruction. Both groups completed a pre-test to measure baseline skills in computational thinking and real-case handling. After the intervention, a post-test was administered, and the results were analyzed to identify any significant improvements attributable to the gamification-based learning model. Quantitative data were collected from the test scores and analyzed using descriptive and inferential statistics, including gain score comparisons and hypothesis testing through t-tests or Z-tests [10]. In addition, qualitative data were gathered through observation, reflective student journals, and semi-structured interviews to capture students' perceptions, engagement levels, and challenges encountered during the learning process.

This dual approach enabled the researchers to not only measure the effectiveness of the model but also explore how and why it influenced student learning behaviour [27].

By incorporating both the systematic design process of ADDIE and empirical validation through quasi-experimentation, the methodology ensured that the gamification model was not only pedagogically sound but also demonstrably effective in enhancing 21st-century skills in a real educational context [28]. Figure 1 illustrates the sequential representation of the ADDIE development model used in this study.

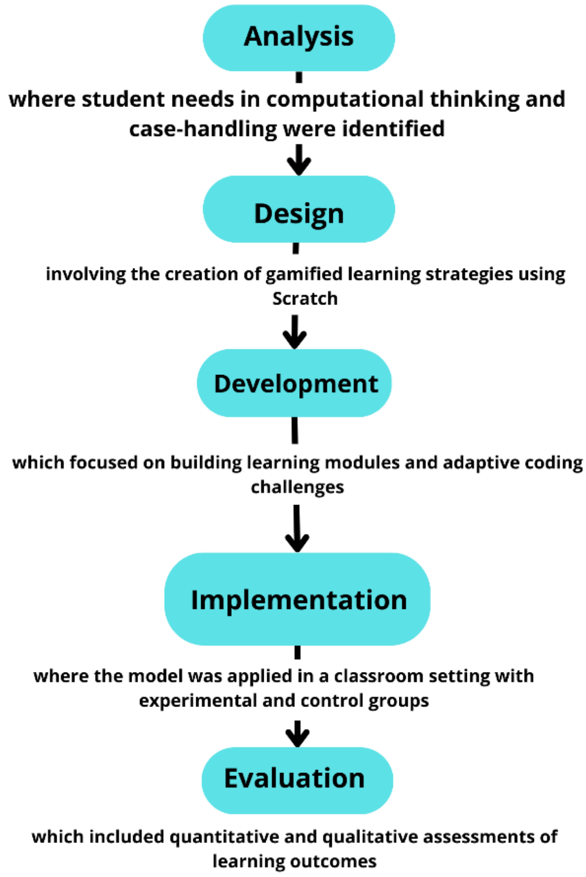

Figure 1. Sequential representation of the ADDIE development model

To assess students' computational thinking (CT) abilities in the context of adaptive gamified learning using Scratch, a rubric was developed based on the theoretical constructs proposed by [29]. This rubric serves to evaluate key dimensions of CT, including decomposition, pattern recognition, abstraction, algorithm design, and optional debugging capabilities. Each dimension is assessed using specific indicators and scored on a scale from 1 to 4. The rubric was used to assess student-created projects developed during the implementation phase of the study.

With score interpretation shown in Table 2 from the lowest with needs improvement level to the highest with excellent level. To assess students' computational thinking abilities in the context of adaptive gamified learning using Scratch, a rubric was developed based on established theoretical constructs [22].

This rubric evaluates key dimensions of CT, including decomposition, pattern recognition, abstraction, algorithm design, and debugging capability. Each dimension is assessed using specific indicators and scored on a scale from 1 to 4, representing increasing levels of mastery. The detailed rubric used to evaluate these dimensions is presented in Table 1.

Table 1. Rubric for assessing Computational Thinking (CT)

Dimension	Indicator	Scoring Criteria	Score Range
Decomposition	Ability to break down complex problems into manageable parts	4 - Identifies all relevant sub-problems clearly 3 - Identifies most but with minor issues 2 - Limited breakdown with unclear components 1 - No attempt or irrelevant	1–4
Pattern Recognition	Ability to identify similarities or patterns across problems	4 - Accurately identifies patterns across multiple scenarios 3 - Identifies some patterns 2 - Vague or partially incorrect patterns 1 - Fails to recognize patterns	1–4
Abstraction	Ability to remove unnecessary details and focus on essential information	4 - Clearly abstracts essential elements 3 - Some abstraction present 2 - Redundant/irrelevant info remains 1 - No abstraction	1–4
Algorithm Design	Ability to design step-by-step solution (logic and flow)	4 - Logical, efficient algorithm with clear steps 3 - Mostly correct, some steps unclear 2 - Incomplete or inefficient steps 1 - Illogical or missing steps	1–4
Debugging (optional)	Ability to identify and correct errors in code or logic	4 - Detects and resolves all issues 3 - Detects most errors, minor issues remain 2 - Partial identification, major issues remain 1 - No attempt to debug or ineffective	1–4

To interpret students' overall computational thinking performance, the total scores obtained from the rubric were categorized into four performance levels. The interpretation of these score ranges is presented in Table 2, ranging from Needs Improvement as the lowest level to Excellent as the highest level.

Table 2. Score interpretation

Score Range	CT Level
17–20	Excellent
13–16	Good
9–12	Fair
≤ 8	Needs Improvement

To evaluate students' competencies in addressing real-world challenges through coding activities, a specialized rubric was developed to assess their real-world case-handling skills. In the context of informatics education, real-world problem-solving is not limited to writing functional code; it also involves the ability to interpret authentic problems, design feasible solutions, implement them effectively using appropriate tools such as Scratch, and critically reflect on the outcomes.

The rubric designed in this study focuses on five core dimensions: problem definition, solution planning, implementation, evaluation, and communication. These dimensions were selected based on their relevance to 21st-century competencies, including analytical thinking, design thinking, technical implementation, reflective evaluation, and communication skills. For instance, defining a problem within a realistic context demonstrates a student's analytical thinking and situational awareness, while planning and implementation reflect their technical and logical reasoning abilities. Evaluation encourages reflective practice and improvement, whereas communication measures students' ability to articulate the rationale behind their solutions, which is essential in collaborative and professional environments. The complete rubric specification used to assess these competencies is presented in Table 3.

Table 3. Rubric and specification for real-world case-handling

Dimension	Indicator	Scoring Criteria	Score Range
Problem Definition	Ability to define and contextualize real-world problems	4 - Clearly defines the problem with relevant real-world context 3 - Defines the problem with minor gaps in context 2 - Vaguely defines the problem with unclear context 1 - Fails to define the problem or lacks context	1–4
Solution Planning	Ability to plan a logical and practical solution	4 - Presents a logical and feasible plan aligned with problem needs 3 - Plan is mostly logical with minor practical gaps 2 - Plan lacks feasibility or clarity 1 - No plan or illogical steps presented	1–4
Implementation	Effectiveness in applying the solution through coding	4 - Implements solution effectively with functional code 3 - Implementation is mostly functional with minor errors 2 - Partially functional or buggy implementation 1 - Ineffective or no implementation	1–4
Evaluation	Ability to assess and reflect on solution effectiveness	4 - Accurately evaluates solution impact and limitations 3 - Basic evaluation with minor analysis 2 - Limited or surface-level evaluation 1 - No evaluation or inaccurate reflection	1–4
Communication	Clarity in presenting the solution and rationale	4 - Communicates solution and rationale clearly and logically 3 - Mostly clear explanation with minor gaps 2 - Unclear or incomplete communication 1 - Poor or no explanation provided	1–4

The rubric employs a four-point scale (1–4) for each indicator, enabling both formative and summative evaluation of students' performance. This instrument was applied during the implementation phase of the adaptive gamification model to measure how effectively students were able to transfer computational thinking skills into practical solutions within gamified coding scenarios. Furthermore, the rubric provides valuable insights for instructors to identify areas where instructional strategies can be refined in order to better simulate the complex demands of real-world problem-solving.

3. Results

The experimental results revealed a significant difference in learning outcomes between the experimental and control groups.

In the experimental group, which implemented the adaptive gamification model based on Scratch, the mean score increased from 65.3 (SD = 7.1) in the pre-test to 84.7 (SD = 6.8) in the post-test. In contrast, the control group demonstrated a more modest improvement, with scores increasing from 64.9 (SD = 6.9) to 72.5 (SD = 7.5).

This substantial improvement in the experimental group indicates the effectiveness of the gamification-based learning approach in enhancing both computational thinking and real-world case-handling skills. The gamified learning model not only increased student engagement but also encouraged deeper cognitive processing, as reflected in students' improved ability to abstract problems, design algorithms, and implement coding solutions.

The comparison of pre-test and post-test results between the two groups is illustrated in Figure 2.

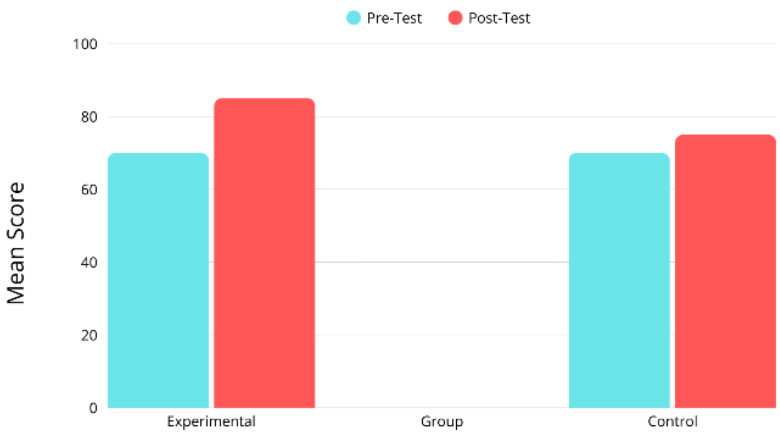

Figure 2. Pre and post test result

To further evaluate the effectiveness of the adaptive gamification model beyond mean score improvements, a Chi-Square Test of Independence was conducted to examine the association between the instructional approach (gamified vs. conventional) and students' achievement status. In this study, a passing score was defined as 70 or above. The contingency distribution of student achievement is presented in Table 4.

Table 4. Contingency table of student achievement

Group	Pass (≥70)	Not Pass (<70)	Total
Experimental	27	3	30
Control	18	12	30
Total	45	15	60

To verify the statistical assumptions of the Chi-Square test, the expected frequencies under the null hypothesis were calculated and are shown in Table 5.

Table 5. Expected frequencies under the null hypothesis

Group	Pass	Not Pass
Experimental	22.5	7.5
Control	22.5	7.5

The results of the Chi-Square test are summarized as follows:

- Chi-Square Value (χ^2): 5.69
- Degrees of Freedom (df): 1
- p-value: 0.017
- Significance Level (α): 0.05

Since the calculated p-value (0.017) is lower than the significance level of 0.05, the null hypothesis is rejected. This indicates a statistically significant association between the instructional model and students' performance categories. In other words, students who participated in the adaptive gamification learning model had significantly higher probabilities of achieving the passing score compared to those who learned through conventional instructional methods.

This categorical finding strengthens the earlier analysis based on mean score improvements, confirming that gamification not only improves average academic performance but also increases the proportion of students who achieve satisfactory learning outcomes.

These findings are consistent with previous research reported in [30], which demonstrated that gamified learning environments can enhance student motivation and cognitive engagement, particularly when integrated with visual programming platforms such as Scratch.

However, the present study extends these findings by applying gamification in a higher education context involving real-world problem scenarios, thereby improving the external validity of the results [31].

Furthermore, the higher number of failing students in the control group may indicate reduced engagement when conventional instructional methods are used to teach complex concepts such as algorithmic logic and problem decomposition. In contrast, the adaptive gamification model enabled students to connect theoretical learning objectives with authentic problem-solving tasks, leading to improved engagement, deeper cognitive processing, and more consistent learning outcomes.

4. Conclusion

This study demonstrates that the adaptive gamification model implemented using Scratch is effective in enhancing students' computational thinking and real-world problem-handling skills in higher education. The findings show substantial improvements in students' ability to decompose problems, design algorithms, apply abstraction, and translate computational concepts into practical and meaningful solutions. These improvements were evident not only in the increase of post-test scores but also in the higher proportion of students achieving the expected performance threshold when compared to conventional instructional approaches.

The adaptive gamification approach provides several important benefits. First, it supports deeper cognitive engagement by aligning learning challenges with students' competency levels, allowing learners to progress at an appropriate pace while maintaining motivation. Second, the integration of real-world scenarios into gamified coding activities enables students to connect abstract programming concepts with authentic problem contexts, which is essential for developing transferable skills relevant to professional practice. Third, the use of a visual programming environment facilitates learning by reducing syntactic complexity while still fostering logical reasoning and structured problem-solving.

Despite these strengths, this study has several limitations. The implementation was conducted within a single course context and involved a relatively limited sample size, which may affect the generalizability of the findings. In addition, the duration of the intervention focused on short-term learning outcomes, and long-term retention or transfer of skills was not examined. These limitations suggest opportunities for further refinement and broader validation of the proposed model.

Future research is recommended to explore the scalability of adaptive gamification across different disciplines, educational levels, and programming platforms. Longitudinal studies could investigate the sustainability of learning gains and the impact of gamified instruction on long-term problem-solving performance and career readiness. Further studies may also examine how adaptive mechanisms can be enhanced through data-driven personalization to better support diverse learner profiles. Overall, this study contributes to the advancement of innovative instructional strategies that support the development of essential 21st-century skills in higher education.

Acknowledgement

The authors gratefully acknowledge the support provided by the Institute for Research and Community Service (LP2M), Universitas Negeri Padang, and Universitas Negeri Malang for facilitating and supporting the implementation of this research collaboration. This research was funded under Contract No. 1533/UN35.15/LT/2025.

References:

[1]. Al-Jubari, I., Hassan, A., & Liñán, F. (2019). Entrepreneurial intention among University students in Malaysia: integrating self-determination theory and the theory of planned behavior. *International entrepreneurship and management journal*, *15*(4), 1323-1342.

[2]. Martín-Lara, M. Á., et al. (2025). Boosting engagement and learning in the economic analysis of chemical processes through gamification. *Education for Chemical Engineers*, *51*, 79-86. Doi: 10.1016/j.ece.2025.02.004

[3]. Anwar, K., et al. (2025). Peningkatan Ketrampilan Berfikir Computational Thinking Dalam Pembelajaran Berbasis Programming-Matematika Menggunakan Menggunakan Scratch:(Enhancing Computational Thinking Skills in Mathematics-Based Programming Learning Using Scratch). *Jurnal JUPEMA*, *4*(1), 50-60.

[4]. Chau, M., & Arruzza, E. (2026). Learning through play: Educators' autoethnography on gamification in medical radiation science. *Journal of Medical Imaging and Radiation Sciences*, *57*(3), 102203. Doi: 10.1016/j.jmir.2026.102203

[5]. Christopoulos, A., & Sprangers, P. (2021). Integration of educational technology during the Covid-19 pandemic: An analysis of teacher and student receptions. *Cogent Education*, *8*(1), 1964690. Doi: 10.1080/2331186X.2021.1964690

[6]. Darni, R., et al. (2025). *Work Readiness in Vocational Education : Perception and Correlation Between Measurement Level and System Integration*. *14*(2), 1388–1397. Doi: 10.18421/TEM142

[7]. Darni, R., & Mursyida, L. (2021). Career Exploration System (C-EXSYS) in era society 5.0 based on expert system. *Jurnal Teknologi Informasi Dan Pendidikan*, *14*(2), 131-143. Doi: 10.24036/tip.v14i2.491

[8]. Dharma, A. S., Tampubolon, L. A., & Purba, D. S. (2020). Drug Stock Prediction at Balige HKBP Hospital Using Adaptive Neuro-Fuzzy Inference System. *Sinkron: jurnal dan penelitian teknik informatika*, *5*(1), 26-34. Doi: 10.33395/sinkron.v5i1.10529

[9]. Febrianti, R., et al. (2023). Implementation of project-based learning for improve students' critical thinking skills in creative product and entrepreneurship subjects. *Jurnal Pendidikan Teknologi Kejuruan*, *6*(4), 240-247.

[10]. Al Husaeni, D. F., et al. (2022). The effect of using web-based interactive learning media for vocational high school students to understanding of looping: Qualitative approach. *Journal of Science Learning*, *5*(1), 115–126. Doi: 10.17509/jsl.v5i1.35534

[11]. Hidayat, H., et al. (2021). The Empirical Analysis of Industrial Work Challenges in the Industrial Revolution 5.0 Towards a Grade Point Average (GPA) for Electronic Engineering Education Students. *International Journal of Online & Biomedical Engineering*, *17*(9). Doi: 10.3991/ijoe.v17i09.25679

[12]. Jaramillo-Mediavilla, L., et al. (2024). Impact of gamification on motivation and academic performance: A systematic review. *Education Sciences*, *14*(6), 639.

[13]. Kim, K. T. (2019). The structural relationship among digital literacy, learning strategies, and core competencies among south korean college students. *Educational sciences: theory and practice*, *19*(2), 3-21. Doi: 10.12738/estp.2019.2.001

[14]. Kiong, T. T., & Zulkifli, M. A. N. B. (2024). Employability challenges of vocational college graduates in the state of Johor. *Jurnal Pendidikan Teknologi Kejuruan*, *7*(2), 76-90.

[15]. Ma, Z. M., & Xu, Z. S. (2020). Computation of generalized linguistic term sets based on fuzzy logical algebras for multi-attribute decision making. *Granular Computing*, *5*(1), 17-28. Doi: 10.1007/s41066-019-00199-x

[16]. Maryati, I. (2018). Penerapan model pembelajaran berbasis masalah pada materi pola bilangan di kelas vii sekolah menengah pertama. *Mosharafa*, *7*(1), 63-74.

[17]. Muhamediyeva, D. T., & Egamberdiyev, N. A. (2019). Algorithm and the program of construction of the fuzzy logical model. *2019 International Conference on Information Science and Communications Technologies (ICISCT)*, 1-4. Doi: 10.1109/ICISCT47635.2019.9011960

[18]. Namaziandost, E., Çakmak, F., & Ashkani, P. (2025). How do you feel? Unmasking ambiguity tolerance, learning adaptability, self-management, and learning outcomes in gamification vs. ludicization in foreign language learning. *Computers in Human Behavior Reports*, *18*, 100651. Doi: 10.1016/j.chbr.2025.100651

[19]. Nelson, S., Darni, R., & Haris, F. (2022). Development Augmented Reality (AR) learning media for pencak silat course at faculty of sports and science universitas negeri padang. *Educational Administration: Theory and Practice*, *28*(1), 37-46.

[20]. Nofrianto, H., et al. (2020). Validity of Cooperative-Discovery Learning Model to Improve Competencies of Engineering Students. *Systematic Reviews in Pharmacy*, *11*(12).

[21]. Piquer-Martinez, C., et al. (2025). Mobile gamification in pharmacy education: A comparative study of learning outcomes and perceptions across gender. *Currents in Pharmacy Teaching and Learning*, *17*(12), 102480. Doi: 10.1016/j.cptl.2025.102480

[22]. Pritasari, O. K., et al. (2021). The Use of Android Smartphone as An Effort to Increase Students' Creativity in Learning Modern Hair Bun Course at Vocational High School in the Pandemic Era. *Journal of Vocational and Career Education*, *6*(2).

[23]. Ranuharja, F., et al. (2021). Development of interactive learning media edugame using ADDIE model. *Jurnal Teknologi Informasi Dan Pendidikan*, *14*(1), 53-59. Doi: 10.24036/tip.v14i1.412

[24]. Reigeluth, C. M., & An, Y. (2020). *Merging the instructional design process with learner-centered theory: The holistic 4D model*. Routledge. Doi: 10.4324/9781351117548

[25]. Shang, X. (2025). Mapping the integration of gamification into foreign language learning in E-learning settings: A bibliometric analysis. *Results in Engineering*, 108452. Doi: 10.1016/j.rineng.2025.108452

[26]. Sheikhkhoshkar, M., et al. (2019). Automated planning of concrete joint layouts with 4D-BIM. *Automation in construction*, *107*, 102943. Doi: 10.1016/j.autcon.2019.102943

[27]. Darni, R., Harisman, Y., & Setiawan, I. N. A. F. (2025). The implementation and empirical analysis of adaptive virtual mentor: Mobile technology empowers introverts' business communication skills. *International Journal of Interactive Mobile Technologies*, *19(*8), 159–173.

[28]. Todd, P. E., & Zhang, W. (2020). A dynamic model of personality, schooling, and occupational choice. *Quantitative Economics*, *11*(1), 231-275. Doi: 10.3982/qe890

[29]. Wang, C., Fang, T., & Gu, Y. (2020). Learning performance and behavioral patterns of online collaborative learning: Impact of cognitive load and affordances of different multimedia. *Computers & Education*, *143*, 103683. Doi: 10.1016/j.compedu.2019.103683

[30]. Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, *49*(3), 33–35.

[31]. Zhou, C., & Zhang, W. (2024). Computational thinking (CT) towards creative action: developing a project-based instructional taxonomy (PBIT) in AI education. *Education Sciences*, *14*(2), 134.