INTERNATIONAL JOURNAL OF RESEARCH AND INNOVATION IN SOCIAL SCIENCE (IJRISS)
ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VIII August 2026

Gamification in Programming-Based STEM Learning: Trends, Challenges, and Future Directions

Lizawati Salahuddin[1*], Fathin Nabilla Md Leza[2], Nor Aiza Moketar[3], Noraswaliza Abdullah[4], Nurul Izrin Md Saleh[5], Noorrezam Yusop[6], Mohd Fazil Mohd Damin[7]

[1,2,3,4,5,6]Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia

[7]Pejabat Pendidikan Daerah Kuala Pilah, Kementerian Pendidikan Malaysia

[*]Corresponding Author

DOI: https://doi.org/10.47772/IJRISS.2026.100800391

Received: 17 August 2026; Accepted: 22 August 2026; Published: 06 September 2026

ABSTRACT

Gamification has gained increasing attention as an approach to improving engagement and learning in programming and STEM education. This narrative review examines research trends, gamification elements, implementation challenges, and future directions in programming-based STEM learning. The review draws on 16 selected literatures published between 2014 and 2026, covering programming education, computational thinking, STEM learning, educational robotics, Arduino, and micro:bit. The literature shows a clear shift from the use of basic game elements, such as points, badges, and leaderboards, towards more interactive and hands-on approaches involving challenges, project-based activities, collaboration, robotics, and physical-computing platforms. Key challenges include overreliance on rewards and competition, learner differences and personalisation, teacher and instructional design issues, and technological constraints. Future research would benefit from longitudinal studies, more adaptive and personalized approaches to gamification, a broader range of STEM contexts, and integrating emerging technologies. Overall, gamification shows promise for creating more engaging and meaningful programming-based STEM learning experiences when it is appropriately integrated with programming tasks and hands-on technologies.

Keywords: gamification; programming education; STEM learning; computational thinking; programming-based learning

INTRODUCTION

Science, Technology, Engineering, and Mathematics (STEM) education is increasingly importance to equip learners with the knowledge and skills the digital economy demands. Programming plays a central role in building computational thinking, logical reasoning, creativity, and problem-solving skills. Consequently, programming is recognised as a core part of STEM education rather than a subject in computer science program (Venter, 2020). Despites its importance, learning programming is difficult for many students. Abstract programming concepts, debugging process, and the demands of algorithmic thinking can reduce confidence and motivation, especially for novice students. These challenge leads to poor academic performance and higher dropout rates. As a result, educators look for teaching approaches that keep students engaged (Huang et al., 2020; Venter, 2020).

Gamification is one approach that has drawn substantial attention. It applies game elements such as points, badges, leaderboards, challenges, rewards, and feedback into learning activities. The aim is to make the lesson more interactive and motivating. Gamification layers game features onto existing activities to keep students participating (Hamari et al., 2014). Its use within programming-based STEM education has grown quickly. Previous research has already examined the use of gamification in programming education. Venter (2020), for example, provided a state-of-the-art review of gamification in programming education and highlighted the potential of game-based approaches for improving learner engagement and programming-related outcomes. However, the research landscape has continued to develop, particularly through the increasing use of physical-computing platforms, educational robotics, Arduino, micro:bit, and other technology-supported STEM environments. The present review therefore extends this earlier work by focusing specifically on empirical studies of programming-based STEM learning and by examining developments reported in the more recent literature. In particular, the review considers the programming platforms, gamification elements, implementation challenges, and emerging technology-supported approaches reported across the selected studies.

This paper aims to fill that gap with a narrative review of current research on gamification in programming-based STEM learning. It looks specifically at research trends, the gamification elements commonly used, implementation challenges, and directions for future research. By drawing together findings from recent empirical studies, the paper aims to offer practical guidance for educators, curriculum developers, and researchers seeking to build more engaging and effective programming-based STEM learning environments.

The rest of the paper is structured as follows. Section 2 outlines the review method. Section 3 discusses current research trends, followed by a look at gamification elements in programming-based STEM learning in Section 4. Section 5 turns to the challenges of implementing gamification in programming-based STEM learning, and Section 6 sets out directions for future research. Section 7 concludes.

LITERATURE REVIEW METHOD

This study adopts a narrative literature review approach to synthesize current research on gamification in programming-based STEM learning. A narrative review suits this study because it allows evidence from a wide range of research designs and educational settings to be critically examined emerging trends, research gaps, and where the field might head next (Baumeister and Leary, 1997; Ferrari, 2015). While systematic reviews follow strict search and screening protocols, a narrative review, by contrast, offers more flexibility for interpreting and synthesizing findings across studies.

The literature covered in this study includes peer-reviewed journal articles, and conference papers, published between 2014 and 2026. The literature search was conducted using the Google Scholar and Scopus database. The searches focusing on combinations of terms related to gamification, programming, STEM education, computational thinking, robotics, Arduino, micro:bit, and programming-based learning. Search terms were combined using Boolean operators to identify studies examining the application of gamification within programming-oriented STEM learning environments. The initial search produced a broader set of potentially relevant publications. Titles and abstracts were first screened to remove clearly irrelevant studies. The remaining publications were then examined against the inclusion and exclusion criteria. Studies were retained when they reported primary empirical findings involving learners and explicitly investigated gamification in programming-based or programming-related STEM learning. Studies that were review-based, systematic reviews, meta-analyses, or unrelated to programming-based STEM learning were excluded. Following the screening process, 16 primary studies were retained for the final narrative synthesis. The selected studies were then examined according to their educational context, programming platform, gamification elements, implementation characteristics, and reported challenges.

The literature was analysed through thematic synthesis, which made it possible to spot common patterns and key developments across the studies reviewed (Thomas and Harden, 2008). Rather than working through each study one by one, the findings were pulled together into four themes: (i) research trends in gamification, (ii) gamification elements in programming-based STEM learning, (iii) implementation challenges, and (iv) future research directions.

Research Trends In Gamification

The 16 primary studies reviewed in this study indicate several developments in the application of gamification to programming-based STEM learning. These developments concern the expansion of learning contexts, the use of different programming platforms, and the integration of emerging technologies. Rather than suggesting a general maturation of the entire field, the findings indicate a gradual broadening of how gamification is being applied within programming-related STEM education.

From Programming Courses to Broader STEM Learning

Early work in this area focused mainly on whether gamification could improve outcomes in introductory programming courses. Ibanez et al. (2014) used gamification in C programming, while Sprint and Cook (2015) applied a gamified activity to CS1 programming practice. These studies focused strongly on participation, enjoyment, and programming learning. Ibanez et al. (2014) reported significant improvement in C-programming comprehension after the intervention. Figueiredo and García-Peñalvo (2020) also reported improvements in attendance, participation, and proactivity when comparing a gamified programming course with a non-gamified year. These early studies established programming education as an important setting for gamification research.

Later studies expanded gamification into broader STEM contexts. Chhabra and Delaney (2022) examined meaningful gamification in STEM project-based learning, while Seliro Wangi and Wajdi (2022) investigated its effects on STEM abilities and soft skills among secondary students. The latter study reported significant improvements in STEM abilities and soft skills in the gamified group compared with the control group.

This broader focus is also evident in studies examining STEM motivation, engagement, and cognitive outcomes. Chen et al. (2023) combined gamification with educational robotics to support motivation and creativity. Mejias et al. (2024) further connected gamification with project-based STEM learning, and Zahid Sahito and Hussain Sahito (2024) examined gamification in high-school STEM classes with particular attention to critical thinking and problem solving. These studies indicate that programming and computational activities are increasingly being considered as part of wider STEM learning experiences rather than as isolated programming exercises.

Overall, the evidence suggests a shift from gamifying individual programming courses towards integrating gamification into broader STEM learning environments. However, the studies remain diverse in terms of educational level and learning context, so this trend should be understood as an expansion of application rather than evidence that gamification has become a universal instructional approach.

Programming Platforms

The literature shows that gamified STEM learning environments employ a variety of programming platforms. Early studies mostly remained within standard text-based programming courses. For example, Ibanez et al. (2014) focused on C programming, while Sprint and Cook (2015) examined gamification within an introductory programming course. Figueiredo and García-Peñalvo (2020) similarly investigated gamification in a conventional programming course.

More recent studies increasingly use visual programming environments, instructional robotics, physical computing platforms, and Internet of Things technologies. Programming principles are applied in practical, hands-on situations using tools. Scratch, Arduino, the micro:bit, and many instructional robotics kits are popular tools for teaching and learning programming. The micro:bit provides an accessible platform for introducing programming through visual and block-based programming activities in school settings. Rafti et al. (2025) used micro:bit programming with primary-school students through hands-on activities involving sequencing, LED control, debugging, and practical problem solving. The activities connected programming with creative and collaborative STEAM tasks, illustrating how physical computing can make programming concepts more concrete and accessible to younger learners. With Arduino, students may link programming to sensor-based applications and electronics. Araújo and Saúde (2026) found strong interest among middle-school students participating in Arduino programming activities, although students experienced difficulties with coding and electronic components. Nannim et al. (2025) similarly found strong support for hands-on programming, Arduino and microcontrollers, interactive project applications, instructional videos, and teamwork.

Educational robotics further enhances the connection between programming ideas and practical STEM applications by enabling students to create programming solutions that operate tangible objects (Araújo and Saúde, 2026; Sáez-López et al., 2026). Chen et al. (2023) showed that gamified educational robots could increase motivation and creativity. The study also used a smart-city STEM theme to connect programming and robotics with real-life applications. More recent robotics studies have moved towards interdisciplinary projects involving environmental problems, computational thinking, programming and machine learning. Sáez-López et al. (2026) integrated visual block programming, robotics, sensors and introductory machine learning in teacher education. The integration of gamification with these technologies has also supported the adoption of project-based learning, in which students complete design challenges rather than isolated programming exercises. This learning environment encourages teamwork, creativity, and higher-order thinking (Chen et al., 2023; Mejias et al., 2024).

Emerging Technologies

Gamification research has been influenced by emerging technology. Advances in artificial intelligence, learning analytics, educational robotics, IoT platforms, Arduino, and micro:bit enable more individualized, interactive, and authentic programming experiences. Teachers can create programming exercises that closely resemble real-world STEM practice using these tools, enabling adaptive feedback, collaborative learning, and problem-solving (Mejias et al., 2024; Sáez-López et al., 2026). Educational robotics is particularly important because it allows learners to see the direct consequences of their programming decisions. Chen et al. (2023) combined gamification with educational robots and a smart-city STEM theme, providing students with opportunities to apply programming through practical challenges. Similarly, Sáez-López et al. (2026) integrated robotics, sensors, visual programming, and introductory machine learning to support computational and STEM learning.

Taken together, the literature shows gamification research moving well beyond its early focus on isolated motivational techniques in introductory programming courses. It's now shifted toward more comprehensive, learner-centered approaches that bring together programming, STEM education, project-based learning, and emerging digital technologies. As Table 1 summarises, this development suggests a gradual shift towards learning environments in which programming, gamification, physical interaction, collaboration, and authentic STEM tasks are more closely connected.

Table 1 Major Trends in Gamification Research in Programming-Based STEM Learning

Research Trend	Early Focus	Current Focus	Educational Implications	Representative Studies
From Programming Courses to Broader STEM Learning	Gamification in introductory programming courses, mainly focusing on participation, enjoyment, and programming achievement	Broader programming-based STEM ecosystems incorporating creativity, collaboration, problem-solving, and STEM interest	Positions programming as an interdisciplinary STEM skill rather than an isolated technical subject	Chen et al. (2023); Chhabra and Delaney (2022); Figueiredo and García-Peñalvo (2020); Ibanez et al. (2014); Mejias et al. (2024); Seliro Wangi and Wajdi (2022); Sprint and Cook (2015); Zahid Sahito and Hussain Sahito (2024)
Programming Platforms	Conventional text-based programming, particularly C programming	Scratch, Arduino, micro:bit, educational robotics, visual/block-based programming, and physical computing	Provides more accessible, practical, and hands-on programming experiences	Araújo and Saúde (2026); Chen et al. (2023); Figueiredo and García-Peñalvo (2020); Ibanez et al. (2014); Mejias et al. (2024); Nannim et al. (2025); Rafti et al. (2025); Sáez-López et al. (2026); Sprint and Cook (2015)

Emerging Technologies	Limited use of emerging technologies in conventional programming courses	AI, learning analytics, educational robotics, IoT, Arduino, micro:bit, and other interactive technologies	Enables more interactive, personalised, authentic, and potentially adaptive learning experiences	Chen et al. (2023); Mejias et al. (2024); Sáez-López et al. (2026)

Gamification Elements In Programming-Based Stem Learning

The gamification components most frequently employed in programming-based STEM education are points, badges, leaderboards, levels, challenges, instant feedback, collaboration, projects and missions, and personalisation according to the literature. These are usually incorporated to increase learner motivation, retain students, and provide them with a real sense of accomplishment when they complete programming assignments.

Points and Badges

Points and badges are among the most common gamification elements used to recognise students' progress and achievements. In programming learning, points are often given for completing programming tasks, resolving code issues, or participating in class activities. Badges similarly indicate mastery of programming abilities or the accomplishment of project milestones. Ibanez et al. (2014), for example, incorporated badges and points into a gamified C programming environment. These elements provided students with visible indicators of their progress and achievement. The study reported improved programming comprehension following the intervention. Vang (2023) also incorporated points and badges into gamified STEM activities to encourage participation and task completion. However, these elements appear to be most useful when they are connected to meaningful learning activities. Chhabra and Delaney (2022) found that students preferred practical forms of gamification rather than rewards that were disconnected from the learning process. This suggests that points and badges should complement learning rather than become the main purpose of the activity.

Leaderboards

Leaderboards give students an advantage by displaying their accomplishments compared to those of their peers. They have been used to encourage participation and provide learners with a visible indication of their achievement. Ibanez et al. (2014), for example, incorporated recognition through leaderboards and badges alongside programming activities and peer interaction. More recent studies have produced more mixed findings. Ortiz-Rojas et al. (2025) examined leaderboard-based gamification with 175 university students and found improved learning performance, but the study did not find significant mediation through autonomous motivation or self-efficacy. Qualitative findings also suggested that students often experienced the leaderboard as a source of achievement and controlled motivation rather than increased autonomy. Zahid Sahito and Hussain Sahito (2024) used points, badges, and leaderboards alongside collaborative activities in high-school STEM learning. Their findings indicate that these elements can support participation and cognitive outcomes, but teachers also reported that some students became more interested in earning points and badges than in the learning content. Similarly, the findings suggest that leaderboards do not necessarily create competition only with other students. In some settings, students may interpret them as a way of monitoring their own progress. Therefore, the educational value of these elements depends strongly on how they are framed and connected to learning objectives.

Levels

Level allows students to move through increasingly difficult tasks or learning stages. In programming education, this can involve beginning with simple coding exercises before progressing towards more complex problems. Ibanez et al. (2014) used levels within its gamified programming environment. Levels are particularly useful for programming because students can gradually develop their skills instead of being exposed to complex programming problems immediately.

Challenges

Challenges and progression provide another common mechanism for structuring gamified programming activities. Programming itself involves repeated problem solving, debugging, and gradual mastery, making challenges and levels particularly suitable for this context. Challenges are particularly relevant to programming-based STEM learning because programming itself involves solving problems and overcoming errors. Gamified challenges can transform programming tasks into structured missions or problems that students progressively attempt to solve. Zahid Sahito and Hussain Sahito (2024) incorporated missions, weekly tasks, points, rewards, and competitions into STEM activities. Students completed tasks involving real-life problems, while both individual and group activities were used to support problem solving. These findings suggest that challenges are particularly appropriate for programming because they mirror the natural learning process: attempt, receive feedback, correct errors, and move to a more demanding task.

Feedback

Feedback is another important element in programming-based gamification. Programming learners frequently need immediate information about whether their code is functioning and logically correct. Gamified environments can therefore combine feedback with progress indicators, rewards, or challenge completion. Nannim et al. (2025) similarly highlighted the importance of interactive applications, instructional resources, and hands-on activities in supporting programming and robotics learning. The combination of programming tasks and timely feedback can encourage students to test, identify errors, and improve their solutions. The findings therefore suggest that feedback should not be treated simply as a reward. In programming-based STEM learning, its greater value lies in helping learners understand what they have achieved, where they made mistakes, and what they should do next.

Collaboration

Collaborative gamification allows students to work together to solve programming or STEM challenges, share ideas, and develop solutions. Sprint and Cook (2015) reported positive participation and interaction in a team-based gamified programming activity. Vang (2023) also incorporated collaborative activities into gamified STEM learning. Nannim et al. (2025) identified teamwork as an important component of programming and robotics learning. This suggests that gamification does not necessarily need to emphasise competition. Collaborative game-based activities can provide an alternative approach that combines engagement with communication and teamwork.

Projects and Missions

Projects and missions extend gamification beyond individual exercises by organising learning around a larger goal. This approach is particularly appropriate for programming-based STEM education because students can apply several concepts within a single project. Chhabra and Delaney (2022) demonstrated the value of integrating gamification with STEM project-based learning. Similarly, Mejias et al. (2024) highlighted the potential of combining gamification with project-based STEM activities. This approach shifts gamification towards meaningful learning, where students are motivated not simply to collect points but to complete a project or solve a problem.

Personalisation

Personalisation represents a more recent direction in gamification. Rather than providing the same challenges and rewards to every learner, personalised gamification attempts to adjust activities according to learners' knowledge, performance, motivation, or learning behaviour. Zhao et al. (2022) proposed a multi-layer gamification approach that considers learner characteristics and learning pathways. This direction is particularly

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VIII August 2026

relevant to programming education because students often enter programming courses with substantially different levels of prior experience. However, compared with more established elements such as points, badges and leaderboards, personalised gamification remains less developed in the reviewed primary studies. Further empirical research is therefore needed to determine how effectively adaptive game elements can support different types of programming learners. Table 2 summarizes the gamification in programming-based stem learning extracted from the review.

Table 2 Gamification Elements in Programming-Based STEM Learning

Gamification Element	Application in Programming/STEM Learning	Potential Role	Representative Studies
Points	Awarded for completing coding or STEM tasks	Provides immediate recognition and progress information	Chhabra and Delaney (2022); Ibanez et al. (2014); Vang (2023)
Badges	Awarded after achieving milestones	Recognises achievement and progress	Chhabra and Delaney (2022); Ibanez et al. (2014); Vang (2023)
Leaderboards	Ranking individual or group performance	Encourages competition and participation	Ibanez et al. (2014); Ortiz-Rojas et al. (2025); Zahid Sahito and Hussain Sahito (2024)
Levels	Increasing task difficulty	Provides a visible progression pathway	Ibanez et al. (2014)
Challenges	Coding problems, missions and STEM tasks	Encourages problem-solving and persistence	Zahid Sahito and Hussain Sahito (2024)
Feedback	Immediate responses to learning activities	Helps learners identify and correct errors	Nannim et al. (2025)
Collaboration	Team-based programming and projects	Supports communication and peer learning	Nannim et al. (2025); Sprint and Cook (2015); Vang (2023)
Projects and Missions	Authentic programming and STEM projects	Connects programming with real-world problems	Chhabra and Delaney (2022); Mejias et al. (2024)
Personalisation	Adapting learning pathways to learner characteristics	Adjust activities according to learners' knowledge, performance, motivation, or learning behaviour.	Zhao et al. (2022)

Challenges In Implementing Gamification In Prgramming-Based Stem Learning

Although the reviewed studies generally report positive experiences with gamification, they also identify several practical and pedagogical challenges. These challenges concern the balance between rewards and learning, differences between learners, teacher preparation, and access to suitable technology and resources.

Overreliance on Rewards

Points, badges, and leaderboards are among the most used gamification elements. However, their frequent use does not necessarily guarantee better learning. Zahid Sahito and Hussain Sahito (2024) reported that some students became more interested in earning points and badges than in the learning content. Teachers also observed that the motivational effect of game elements could decline over time. Ortiz-Rojas et al. (2025)

similarly found that leaderboard-based gamification improved learning performance, but students' experiences suggested stronger controlled rather than autonomous motivation. Chhabra and Delaney (2022) provide evidence for an alternative approach. Students in their STEM project-based learning study tended to favour practical gamification features over imaginary or intangible rewards. The implication is that rewards should support learning rather than become the purpose of the activity. In programming-based STEM education, meaningful challenges, useful feedback, progress, and authentic outcomes may be more sustainable than continuous reliance on points and rankings.

Learner Differences and Personalisation

Learners differ in their prior knowledge, abilities, interests, and preferred ways of learning. A gamified activity that is engaging for one student may be too easy or too difficult for another. Seliro Wangi and Wajdi (2022) reported positive effects of gamification on STEM abilities and soft skills but also acknowledged that the effectiveness of gamification can vary according to student characteristics, learning context, and implementation. Ortiz-Rojas et al. (2025) likewise showed that improved performance did not automatically translate into higher autonomous motivation or self-efficacy. Moreover, Chong et al. (2024) focused specifically on Generation Z students and found a preference for interactive and rewarding approaches, suggesting that learner expectations may influence how gamification is perceived. Personalisation remains particularly relevant to programming-based STEM learning, where students can have very different levels of programming experience. These findings highlight the need for more flexible gamification designs. Programming-based STEM environments could provide different levels of difficulty, optional challenges, personalised feedback, or alternative forms of recognition to accommodate learners with different needs.

Teacher and Instructional Design Issues

Successful gamification requires more than adding game mechanics to existing lessons. Teachers need to understand how game elements relate to learning objectives, assessment, student motivation, and classroom activities. Teachers are central to the successful use of gamification, but designing and managing a gamified learning environment can increase their workload. Zahid Sahito and Hussain Sahito (2024) reported that teachers needed preparation to implement gamified learning activities and identified concerns related to maintaining motivation and ensuring that game elements did not overshadow the learning objectives. The issue becomes more complex when gamification is combined with programming, robotics, Arduino, or micro:bit because teachers also need sufficient technical knowledge to support these technologies. Chhabra and Delaney (2022) emphasised the importance of carefully embedding gamification into project activities rather than simply adding rewards to an existing course. Their findings suggest that practical features that support communication, participation, and learning are more meaningful than isolated reward mechanisms. Studies involving robotics and programming education emphasise the importance of teacher preparation and training (Chen et al., 2023; Nannim et al., 2025; Sáez-López et al., 2026). Teacher readiness and instructional design should be considered central components of gamification rather than secondary implementation issues.

Technological and Resource Constraints

Technology can enhance gamified STEM learning, but it can also create practical barriers. Platforms involving Arduino, micro:bit, robotics, or other physical-computing devices require suitable hardware, software, connectivity, and technical support. These requirements may be difficult to maintain in settings with limited resources (Sáez-López et al., 2026). Nannim et al. (2025) identified limited access to robotics hardware, instructor training, and resource costs as challenges in robotics programming education. Zhao et al. (2022) further noted limitations in existing gamification systems, including limited support for advanced technology-enhanced learning and communication. Thus, successful implementation requires not only appropriate gamification design but also reliable infrastructure and institutional support. These issues can limit implementation, particularly in schools or institutions with restricted budgets. Table 3 summarizes the challenges in implementing gamification.

Table 3 Challenges in Implementing Gamification

Challenge	Description	Representative Evidence
Overreliance on rewards	Excessive emphasis on points, badges and competition may shift attention away from learning.	Chhabra and Delaney (2022); Ortiz-Rojas et al. (2025); Zahid Sahito and Hussain Sahito (2024)
Learner differences and personalisation	Fixed learning pathways may not accommodate differences in ability and motivation.	Chong et al. (2024); Ortiz-Rojas et al. (2025); Seliro Wangi and Wajdi (2022)
Teacher and instructional design issues	Effective gamification requires teachers to align game elements with curriculum objectives, learning activities, assessment, and student needs. Poorly designed gamification may emphasise game mechanics rather than learning.	Chen et al. (2023); Chhabra and Delaney (2022); Nannim et al. (2025); Sáez-López et al. (2026); Zahid Sahito and Hussain Sahito (2024)
Technological and resource constraints	Robotics and physical-computing activities require equipment and technical support.	Nannim et al. (2025); Sáez-López et al. (2026); Zhao et al. (2022)

Future Research Directions

Evaluating Long-Term Learning Outcomes

Many existing studies use relatively short interventions, making it difficult to determine whether improvements in motivation and engagement are maintained over time. Future studies should therefore examine learners over longer periods, such as an academic year or several learning units. This would provide better evidence of whether gamification produces lasting improvements in programming skills, STEM interest, and learning behaviour (Mejias et al., 2024; Ortiz-Rojas et al., 2025).

Personalized and Adaptive Gamification

Personalisation is another important area for future research. Learners differ in their prior knowledge, programming ability, motivation, and familiarity with games, yet many gamified activities provide similar experiences for all students. Zhao et al. (2022) proposed the use of learner profiles, including motivation, knowledge gaps, competency, and learning characteristics, to support more personalized gamification paths. Future systems could adapt task difficulty, feedback, and learning progression according to individual performance. However, empirical evidence on adaptive gamification in programming-based STEM learning is still limited. Future studies should therefore test whether personalized gamification provides better learning outcomes than conventional, non-adaptive approaches.

Broader STEM Contexts and Learner Groups

Future research should also expand gamification studies across different STEM disciplines and educational levels. Existing research has examined a range of STEM contexts, but differences between mathematics, science, engineering, technology, and programming may influence how students respond to gamification. Research should also consider learner characteristics such as prior gaming experience, academic self-concept, achievement orientation, and gender, as these factors may influence responses to competitive and reward-based activities.

For programming-based STEM education, this is particularly important because students often enter the classroom with very different levels of programming experience. Future studies could therefore investigate whether different gamification strategies are more appropriate for beginners, intermediate learners, or students with advanced programming skills.

Integrating Emerging Technologies

AI opens new opportunities for gamified programming and STEM learning, from adaptive challenges to personalized feedback to real-time learning analytics. Future systems could use learner performance and learning behaviour to adjust challenge difficulty, progression, feedback, and rewards. Zhao et al. (2022) highlight the potential of considering learner characteristics in gamified learning, while the findings of Vang (2023) and Ortiz-Rojas et al. (2025) suggest that learners may respond differently to gamification approaches. Nonetheless, the use of AI also raises concerns about data privacy, algorithmic fairness, transparency, and unequal access. Going forward, research shouldn't just examine whether AI makes gamification more effective, but also how it can be implemented in educational settings. Teachers still need to stay in the loop, especially in cases where automated systems are used to recommend learning activities or make decisions about learner progression. Future research should therefore investigate whether adaptive gamification can provide more appropriate learning experiences for learners with different levels of programming experience.

CONCLUSION

This review highlights that gamification is no longer confined to traditional programming courses. It's increasingly adopted in hands-on environments involving Arduino, micro:bit, and educational robotics. These technologies provide opportunities to combine programming with project-based activities, challenges, collaboration, and real-world problem-solving. This development suggests that gamification in programming education is shifting from simple points towards learning experiences that involve active participation and practical application. However, several challenges arise, including overreliance on rewards, learner differences and personalisation, teacher and instructional design issues, and technology constraints. Looking ahead, future research would benefit from longer-term studies, more personalized approaches to gamification, a wider range of STEM contexts, and integrating emerging tools. Emerging tools like AI and learning analytics also hold real promise for making learning more adaptive. However, this needs to go hand in hand with careful attention to privacy, fairness, and keeping teachers involved in the process. Overall, the reviewed primary studies suggest that gamification can support programming-based STEM learning through a combination of motivational, interactive, collaborative, and project-oriented approaches. However, the evidence remains heterogeneous, and the effectiveness of gamification approaches appears to depend on the learning context, learner characteristics, technology used, and intervention design.

ACKNOWLEGEMENT

This work was supported by the ON Semiconductor under the International Research Grant Scheme, No: ANTARABANGSA-ONSEMI/2025/FTMK/A00099.The authors would like to express gratitude to Fakulti Teknologi Maklumat dan Komunikasi (FTMK), Universiti Teknikal Malaysia Melaka (UTeM), for their invaluable support and resources provided throughout this research.

REFERENCES

1. Araújo, J. L., and Saúde, I. (2026). Exploring arduino programming in non-formal education context: enhancing middle school students' interest and motivation. Research in Science and Technological Education, 44(1), 296–313. https://doi.org/10.1080/02635143.2025.2472755
2. Baumeister, R. F., and Leary, M. R. (1997). Writing narrative literature reviews. Review of General Psychology, 1(3), 311–320. https://doi.org/10.1037/1089-2680.1.3.311
3. Chen, T. I., Lin, S. K., and Chung, H. C. (2023). Gamified Educational Robots Lead an Increase in Motivation and Creativity in STEM Education. Journal of Baltic Science Education, 22(3), 427–438. https://doi.org/10.33225/jbse/23.22.427
4. Chhabra, P., and Delaney, P. (2022). Using Gamification to Promote Student Engagement in STEM Project-Based Learning. IEEE Intelligent Informatics Bulletin, 22(1), 38–47.
5. Chong, M. M., Subramanian, P., Ting, M., and Ying, L. J. (2024). An Investigation into the Perception of Gen Z Students in Higher Education towards Gamification Techniques in Learning

Computing Subjects. 2024 International Visualization, Informatics and Technology Conference, IVIT 2024, 165–170. https://doi.org/10.1109/IVIT62102.2024.10692814

6. Ferrari, R. (2015). Writing narrative style literature reviews. Medical Writing, 24(4), 230–235. https://doi.org/10.1179/2047480615z.000000000329
7. Figueiredo, J., and García-Peñalvo, F. J. (2020). Increasing student motivation in computer programming with gamification. 2020 IEEE Global Engineering Education Conference (EDUCON), 997–1000. https://doi.org/doi:10.1109/EDUCON45650.2020.9125283
8. Hamari, J., Koivisto, J., and Sarsa, H. (2014). Does gamification work? - A literature review of empirical studies on gamification. Proceedings of the Annual Hawaii International Conference on System Sciences, 3025–3034. https://doi.org/10.1109/HICSS.2014.377
9. Huang, R., Ritzhaupt, A. D., Sommer, M., Zhu, J., Stephen, A., Valle, N., Hampton, J., and Li, J. (2020). The impact of gamification in educational settings on student learning outcomes: a meta-analysis. Educational Technology Research and Development, 68(4), 1875–1901. https://doi.org/10.1007/s11423-020-09807-z
10. Ibanez, M. B., Di-Serio, A., and Delgado-Kloos, C. (2014). Gamification for engaging computer science students in learning activities: A case study. IEEE Transactions on Learning Technologies, 7(3), 291–301. https://doi.org/10.1109/TLT.2014.2329293
11. Mejias, M., Arenas, F., and Duran, R. (2024). Exploring STEM Adolescent Education, using Project based Learning and Gamification. EDUNINE 2024 - 8th IEEE World Engineering Education Conference: Empowering Engineering Education: Breaking Barriers through Research and Innovation, Proceedings. https://doi.org/10.1109/EDUNINE60625.2024.10500645
12. Nannim, F. A., Ibezim, N. E., Agbo, G. C., Mgboji, C., Ngwoke, S. O. R., and Mosia, M. (2025). Development of a Project-Based Arduino Learning App: Fostering Robotics Programming Competence among Preservice Teachers of Computer and Robotics Education. ACM Transactions on Computing Education, 25(1), 1–24. https://doi.org/10.1145/3719016
13. Ortiz-Rojas, M., Chiluiza, K., Valcke, M., and Bolanos-Mendoza, C. (2025). How gamification boosts learning in STEM higher education: a mixed methods study. International Journal of STEM Education, 12(1). https://doi.org/10.1186/s40594-024-00521-3
14. Rafti, M., Milakis, E. D., Argyrakou, C. C., Vangeli, D. G., and Katsarou, M. C. (2025). Hands-On STEAM Programming: Microteaching with Micro:bit for Enhanced Learning. Journal of Hypermedia & Technology-Enhanced Learning, 3(2), 108–127.
15. Sáez-López, J. M., Redondo-Duarte, S., and De Lara-García Cervigón, S. (2026). Educational robotics with interactive and environmental projects using “micro:bit” and “Nezha” in nine primary schools. Eurasia Journal of Mathematics, Science and Technology Education, 22(3). https://doi.org/10.29333/ejmste/18107
16. Seliro Wangi, N. B., and Wajdi, M. B. N. (2022). Gamification: An Effective Strategy for Developing Soft Skills and STEM in Students. QALAMUNA: Jurnal Pendidikan, Sosial, Dan Agama, 14(1), 663–676. https://doi.org/10.37680/qalamuna.v14i1.4650
17. Sprint, G., and Cook, D. (2015). Enhancing the CS1 Student Experience with Gamification. 5th IEEE Integrated STEM Education Conference, 94–99. https://doi.org/doi:10.1109/ISECon.2015.7119953
18. Thomas, J., and Harden, A. (2008). Methods for the thematic synthesis of qualitative research in systematic reviews. BMC Medical Research Methodology, 8, 1–10. https://doi.org/10.1186/1471-2288-8-45
19. Vang, K. (2023). Assessing the Impact of Gamification in STEM Education: A Case Study of Middle School Classrooms. Journal of Asian Multicultural Research for Social Sciences Study, 4(4), 19–25. https://doi.org/10.47616/jamrsss.v4i4.465
20. Venter, M. (2020). Gamification in STEM programming courses: State of the art. 2020 IEEE Global Engineering Education Conference (EDUCON), 859–866.
21. Zahid Sahito, F., and Hussain Sahito, Z. (2024). Gamification as a Pedagogical Tool for Enhancing Critical Thinking and Problem-Solving Skills in STEM Education: A Case Study of High School Classrooms. Journal of Development and Social, 5(4), 316–331. https://doi.org/10.47205/jdss.2024(5-IV)29

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VIII August 2026

22. Zhao, D., Playfoot, J., De Nicola, C., Guarino, G., Bratu, M., Di Salvadore, F., and Muntean, G.-M. (2022). An Innovative Multi-Layer Gamification Framework for Improved STEM Learning Experience. IEEE Access, 10, 3879–3889. https://doi.org/10.1109/ACCESS.2021.3139729