frontiers in Computer Science

TYPE Original Research
PUBLISHED 23 March 2026
DOI 10.3389/fcomp.2026.1766830

Educational games as a tool for teaching programming in digital extracurricular computer science education

Bolat Tassuov[1], Ainur Nadyrbekova[1], Karakoz Ibragimova[1], Zhazira Taszhurekova[2]*, Nadira Niyetbayeva[1]*, Aigerim Abzhapparova[1] and Zarina Aidaraliyeva[3,4]

[1]Physics and Computer Science, Faculty of Natural Sciences, M.Kh. Dulaty Taraz University, Taraz, Kazakhstan, [2]Department of Applied Informatics and Programming, Faculty of Technology, M.Kh. Dulaty Taraz University, Taraz, Kazakhstan, [3]NCJSC "S.Seifullin Kazakh Agro Technical Research University", Astana, Kazakhstan, [4]EP Computer Sciences, M.Kh. Dulaty Taraz University, Taraz, Kazakhstan

OPEN ACCESS

EDITED BY
Jesus Martin-Vaquero,
University of Salamanca, Spain

REVIEWED BY
Bassey Isong,
North-West University, South Africa
Jorge Hochstetter,
University of La Frontera, Chile

*CORRESPONDENCE
Zhazira Taszhurekova
✉ zhk.taszhurekova@dulaty.kz
Nadira Niyetbayeva
✉ n_nadira79@mail.ru

RECEIVED 13 December 2025
REVISED 24 February 2026
ACCEPTED 25 February 2026
PUBLISHED 23 March 2026

CITATION
Tassuov B, Nadyrbekova A, Ibragimova K, Taszhurekova Z, Niyetbayeva N, Abzhapparova A and Aidaraliyeva Z (2026) Educational games as a tool for teaching programming in digital extracurricular computer science education.
Front. Comput. Sci. 8:1766830.
doi: 10.3389/fcomp.2026.1766830

This study explores how university students experience learning programming through educational games in extracurricular computer science education. A qualitative phenomenological design was employed to interpret the meanings students attribute to game-based learning rather than to evaluate instructional effectiveness. Data were collected through semi-structured interviews with 60 undergraduate students who participated in supervised extracurricular programming activities involving educational coding games. The analysis revealed that educational games transform programming from a formal academic subject into an exploratory problem-solving activity supported by immediate feedback. Students described a reduction of anxiety and perceived the environment as psychologically safe for experimentation. However, gameplay alone produced an intuitive yet fragmented understanding, requiring instructor explanation for conceptual structuring. The experience of learning was therefore characterized as a transition from experiential discovery to reflective comprehension. The findings indicate that educational games function as an experiential entry point through which students begin to interpret programming as a meaningful activity, while instruction supports conceptual articulation. This study contributes to understanding the pedagogical role of educational games not as substitutes for teaching but as mediating environments shaping students' perception of programming.

KEYWORDS
educational games, extracurricular learning, game-based learning, phenomenology, programming education, student experience

1 Introduction

In the context of rapid digital transformation, programming has become one of the key competencies required for successful participation in modern society. Higher education institutions face the challenge of not only providing students with technical knowledge but also fostering motivation, engagement, and problem-solving skills that are essential for working in technology-driven environments. However, traditional instructional approaches based primarily on lectures and textbooks are often perceived by students as abstract and insufficiently engaging, particularly in the early stages of learning programming. Extracurricular computer science education offers a flexible learning environment that allows the implementation of innovative pedagogical approaches beyond the constraints of formal curricula. Within this context, educational games have gained increasing attention as a potential tool for teaching programming. By integrating elements of interactivity, challenge, and feedback, educational games may support active learning and encourage students to engage more deeply with programming concepts.

Previous research suggests that game-based learning can enhance learners' interest and motivation, as well as support the development of logical thinking and problem-solving abilities. Nevertheless, the effectiveness of educational games largely depends on students' perceptions and experiences of using such tools in real learning contexts. Understanding how students perceive educational games is therefore essential for evaluating their pedagogical value and for improving instructional design.

Despite the growing body of research on game-based learning, there remains a lack of empirical studies focusing on students' subjective experiences of learning programming through educational games, particularly in extracurricular higher education settings. Many existing studies emphasize learning outcomes or instructional design, while less attention is paid to learners' lived experiences and personal interpretations of this approach.

The present study seeks to address this gap by exploring university students' perceptions of learning programming through educational games in extracurricular computer science education. The main objective of the research is to analyze students' views regarding motivation, engagement, and perceived learning benefits associated with the use of educational games. By focusing on students' experiences, this study aims to contribute to a deeper understanding of the role of educational games as a complementary tool in programming education.

2 Literature review

The integration of educational games into programming instruction has gained strong empirical support in recent years, particularly within digital and extracurricular learning contexts. A foundational contribution to this field is the meta-analysis conducted by Sailer and Homner (2020), which systematically examined the effects of gamification on cognitive, motivational, and behavioral learning outcomes. Their findings demonstrate that gamification positively influences achievement and engagement when game elements are meaningfully aligned with instructional objectives rather than superficially applied. Complementing this perspective, Zainuddin et al. (2020) conducted a comprehensive systematic review of empirical evidence and concluded that gamified environments significantly enhance learner motivation, self-regulation, and task persistence across digital learning settings. These findings are particularly relevant for programming education, where sustained engagement is essential due to the abstract and cognitively demanding nature of coding tasks. Tokac et al. (2019), in their meta-analysis of game-based learning in mathematics education, report moderate but statistically significant improvements in student achievement, suggesting that similar cognitive benefits may transfer to programming domains characterized by algorithmic reasoning and logical structure. Within computer science specifically, Papadakis and Kalogiannakis (2019) provide empirical evidence from an introductory programming course, demonstrating that a structured game-based learning approach positively influenced students' behavioral outcomes and competence development. Their study highlights reductions in anxiety and improvements in engagement, reinforcing the argument that educational games can create psychologically supportive environments for mastering foundational programming concepts. Collectively, these studies establish a strong theoretical and empirical foundation for incorporating game-based strategies into digital extracurricular programming education.

Beyond motivation and engagement, the pedagogical value of educational games is closely linked to the development of computational thinking, which constitutes a core competence in computer science. Denning and Tedre (2021) conceptualize computational thinking as a disciplinary framework encompassing abstraction, decomposition, algorithm design, and systematic problem-solving. Educational games offer experiential environments where these cognitive processes can be practiced through interactive challenges rather than passive instruction. Expanding on this perspective, Videnovik et al. (2023) conducted a scoping review of game-based learning in computer science education and identified consistent evidence that digital games support algorithmic reasoning, debugging skills, and conceptual understanding when integrated with curricular goals. Petri et al. (2022) similarly highlight that digital games in computing education enhance not only motivation but also conceptual retention and practical skill development, particularly when game mechanics simulate authentic problem-solving scenarios. Empirical illustrations further strengthen this argument. Kane et al. (2019) describe an escape-room–style educational game designed to teach Python programming, reporting increased student engagement and collaborative problem-solving among younger learners. Although earlier work by Malliarakis et al. (2014) provided foundational design principles for educational programming games, contemporary studies extend these frameworks into digital ecosystems characterized by interactivity and iterative feedback. Importantly, extracurricular settings provide flexible learning spaces where such game-based strategies can be implemented without the constraints of formal assessment, allowing learners to explore coding constructs in low-risk environments. This flexibility makes digital extracurricular education particularly conducive to game-based programming instruction.

Recent research further expands the scope of educational games through the integration of artificial intelligence and online collaborative structures. Wagan et al. (2023) propose an artificial intelligence–enabled game-based learning framework designed to optimize quality of experience through adaptive feedback and secure learning environments. Their model illustrates how AI-driven systems can personalize learning trajectories, dynamically adjust task difficulty, and provide real-time analytics, thereby supporting differentiated programming instruction. Parallel to these technological advancements, Zheng and Wang (2023) examine the effectiveness of online collaborative learning within gamified environments and report significant improvements in learner performance and peer interaction. These findings underscore the importance of social dimensions in digital extracurricular programming education, where collaborative problem-solving mirrors authentic software development practices. Taken together, the reviewed literature suggests that educational games function not merely as motivational tools but as structured pedagogical environments capable of fostering computational thinking, engagement, collaboration, and adaptive learning. However, the studies consistently emphasize the necessity of pedagogical alignment and guided reflection to ensure that gameplay translates into durable conceptual understanding. Therefore, contemporary scholarship positions educational games as complementary instructional instruments that are particularly effective in digitally mediated extracurricular computer science education when supported by thoughtful design and instructional scaffolding.

Overall, contemporary literature supports the effectiveness of digital games and gamification in programming education when pedagogical design is prioritized. Research increasingly emphasizes not whether game-based learning works, but how, under what conditions, and for which learners it produces meaningful learning outcomes.

2.1 Theoretical framework

This study is grounded in four complementary theoretical perspectives.

First, constructivist learning theory posits that learners actively construct knowledge through interaction with learning environments. Educational games support this process by enabling exploration, experimentation, and iterative learning.

Second, experiential learning theory emphasizes learning through experience and reflection. Programming games provide experiential learning opportunities by allowing students to test solutions, receive feedback, and refine their understanding through practice.

Third, gamification theory explains how motivational elements embedded in games—such as challenges, rewards, and feedback—can enhance engagement and persistence in learning tasks. In programming education, these elements may reduce anxiety associated with complex problem-solving.

Finally, this research adopts a phenomenological perspective, focusing on students' lived experiences and subjective interpretations of learning programming through educational games. This approach enables a deeper understanding of how students perceive the benefits and limitations of game-based learning beyond measurable performance indicators.

3 Materials and methods

In the modern world of information technology, programming is becoming an increasingly important and in-demand skill. Schools aim to provide their students with the opportunity to master the basics of programming to develop their creative abilities and prepare them for the future challenges of the digital era. In this context, the use of educational games as a means of teaching programming is becoming increasingly popular and recognized.

Educational games, or serious games, are educational programs or applications designed to teach specific skills or knowledge, in this case, programming. They incorporate elements of gameplay, making learning more engaging and interesting for students. Such games often provide opportunities to create their own programs, solve problems, and overcome obstacles, helping students develop skills in logical thinking and algorithmic reasoning.

3.1 Research method

To achieve the objective of the study, a phenomenological approach belonging to qualitative research methods was employed. Qualitative research is understood as an explanatory and interpretive process in which the researcher observes participants in a natural learning environment and analyzes the formation of events and experiences to identify the essential characteristics of individual and social phenomena (Mertens, 2019).

Phenomena in an educational context manifest in various forms, including experiences, perceptions, attitudes, meanings, and interpretations of learning activities. The phenomenological method focuses on how individuals perceive a phenomenon, how they describe and evaluate it, how they make sense of it, what emotions it evokes, and what understandings they construct about it. In the context of this study, the approach enables an exploration of how students experience learning programming through educational games, the meanings they assign to game-based learning, and how their understanding of programming develops through these experiences.

The use of the phenomenological approach is justified by the fact that learning programming through educational games represents a familiar yet not fully conceptualized educational phenomenon. The method is particularly suited to examining everyday learning practices that are commonly encountered but insufficiently understood in depth. By focusing on participants' lived experiences, the phenomenological design provides an appropriate framework for revealing how game-based learning influences engagement, perception, and interpretation of programming education (Tisdell et al., 2025).

3.2 Participants

The study involved 60 undergraduate students enrolled in computer science–related programs at universities in Kazakhstan. Participants had prior exposure to programming courses and voluntarily participated in extracurricular programming activities involving educational games.

The sample included students from all academic years (first to fourth year) and represented diverse levels of programming proficiency. Participation was voluntary and anonymous, and students were informed about the purpose of the research before the interviews.

The objective of participant selection was not statistical generalization but obtaining a range of experiences sufficient for

TABLE 1 Participants' demographic characteristics.

Class	Female	Male	Total
First-year	9	4	13
Second year	6	13	19
Third year	8	6	14
Fourth-year	4	10	14
Total	27	33	60

phenomenological saturation. The demographic characteristics of the participants are presented in Table 1. The sample included students from all 4 years of study, with 27 female and 33 male participants. This diversity allowed the study to capture perspectives from students with varying levels of academic experience in programming.

Table 1 depicts the breakdown of students by gender and class levels who participated in this study. Each number in the table represents the count of female and male students in each of the specified classes. The total column in the table represents the sum of female and male students in each class, and the overall total corresponds to the total number of research participants.

3.3 Learning environment and educational games

Educational games were used as part of extracurricular programming sessions organized by the instructor. Students interacted with beginner-level programming environments that required constructing algorithms, sequencing commands, and debugging solutions through visual or text-based interaction. Each session lasted approximately 60–90 min and was conducted once or twice per week over a period of 6–8 weeks. Tasks included solving levels, correcting program behavior, and collaboratively discussing possible solutions. The goal of gameplay was not assessment but exploration and familiarization with programming logic. Participants interacted with commonly used educational programming games and gamified environments, included in Table 2.

During sessions, students solved programming challenges, constructed algorithms, debugged code, and discussed solutions with peers and instructors. Games were used as learning triggers followed by reflection and explanation.

Data were collected using a semi-structured interview questionnaire developed specifically for this study. Semi-structured interviews were chosen to allow participants to express their experiences and opinions freely while ensuring that all interviews addressed the key research questions.

The interview form included demographic questions as well as open-ended and closed-ended questions related to students' experiences with educational games, perceived advantages and limitations, motivation, and skill development. To ensure clarity, relevance, and alignment with the research objectives, the interview questions were reviewed by two experts. Their feedback was incorporated into the final version of the interview instrument.

The semi-structured interview questionnaire, refined with input from experts, is presented in Table 3.

This data collection method allowed us to gain a deeper understanding of students' perspectives and experiences regarding learning programming through educational games in extracurricular computer education.

Closed-ended items were used only to provide contextual background information about participants and were not included in the phenomenological analysis.

3.4 Data collection procedure

Data collection was carried out through face-to-face interviews with students who voluntarily agreed to participate in the study. Before each interview, participants were informed about the purpose of the research and provided consent for audio recording.

All interviews followed the same semi-structured format and question order, without leading or directing participants' responses. Each interview lasted approximately 30–35 min. The extracurricular activities were organized as optional programming workshops conducted after regular classes. Students completed guided game-based tasks under instructor supervision rather than independent home use.

3.5 Data analysis

The collected data were analyzed using descriptive qualitative analysis. Audio recordings of the interviews were transcribed verbatim. Following transcription, the data were systematically coded and categorized according to recurring themes related to students' experiences, perceptions, motivation, and identified advantages and limitations of learning programming through educational games.

Frequency analysis was used to identify commonly occurring themes across participants' responses. To enhance transparency and credibility, direct quotations from participants were included in the Results section to illustrate key findings. The analysis process aimed to present participants' views clearly and consistently with the objectives of the study.

The purpose of the analysis was not to determine how many students expressed a particular opinion but to understand how participants interpreted their learning experience. Frequencies presented in tables serve only as descriptive context and do not represent statistical inference. The analysis focused on meaning structures rather than measurement, and numerical representations served only illustrative purposes.

3.6 Trustworthiness and ethical considerations

To enhance the trustworthiness of the study, the coding and categorization process was reviewed by the researchers to ensure consistency in theme identification. Discrepancies were discussed until agreement was reached.

TABLE 2 Educational games used in extracurricular programming sessions and their main programming concepts.

Educational game	Main programming concepts	Interaction type
Scratch	Sequence, loops, conditions	Visual block programming
Blockly games	Logic, conditions, functions	Puzzle-based
CodeCombat	Syntax, debugging, algorithms	Role-playing coding
Lightbot	Algorithmic thinking	Logic puzzles
Code.org challenges	Fundamental constructs	Guided tasks
Teacher-designed gamified tasks	Problem solving	Structured activities

TABLE 3 Semi-structured interview questionnaire form.

Questionnaire:	Interview:
Please indicate your age. Less than 18 years 18–25 years 26–35 years Over 35 years 1. In which university or educational institution are you currently enrolled? 2. How long have you been involved in programming? 3. Have you participated in courses or classes that used educational games for programming instruction? (Yes/No) 4. What educational games have you used in your programming education? Rate how much you believe educational games helped you learn programming on a scale from 1 to 5, where 1—not helpful at all, 5—very helpful. 5. What specific skills or aspects of programming do you think educational games helped you develop? 6. Do you believe that games stimulated your motivation to learn programming? (Yes/No) 7. Are there any limitations or drawbacks to using educational games for programming instruction that you would like to mention?	1. How do you evaluate your experience of learning programming using educational games? 2. What specific games have you used in your education, and what tasks or lessons did they offer? 3. What advantages and disadvantages do you see in using games in programming education? 4. Do you believe that the use of educational games has stimulated your motivation to learn programming? If yes, why? 5. Which specific skills or aspects of programming do you believe were best developed through educational games? 6. Are there any changes or improvements that, in your opinion, could make the learning process using games more effective? 7. Your opinion on how educational games can be better integrated into extracurricular computer education?

Ethical considerations were carefully observed throughout the research process. Participation was voluntary, and informed consent was obtained from all participants before data collection. Participants were assured of the confidentiality and anonymity of their responses, and the collected data were used solely for research purposes.

4 Results

The tables below provide contextual descriptive summaries and do not constitute the primary analytic procedure; interpretation is based on participants' narratives.

Quantitative summaries are presented only to contextualize participant characteristics and do not constitute the primary analytic procedure. The central analysis is thematic and interpretive.

Learning programming through educational games is an innovative and captivating approach that is gaining increasing popularity in the modern education system. This approach combines elements of entertainment and education, making the learning process more engaging and motivating.

The perspectives of university students on their experience of learning programming through educational games vary, and the study was conducted to uncover these perspectives. Students who participated in this research expressed their views on the teaching of programming through educational games.

Table 4 presents various perspectives of university students regarding their learning experience in programming using educational games, along with their ratings and comments.

Based on the data and comments from university students obtained during the research, the following conclusions and results can be drawn:

Positive Learning Experience: The majority of students expressed a positive attitude toward their learning experience in programming using educational games. They noted that this approach made the learning process more engaging and interesting.

Practicality and Interest: Students emphasized that they like the fact that they can immediately apply new skills in games, making learning more practical and interesting for them.

Motivation Boost: Educational games had a positive impact on students' motivation for the learning process. They expressed that games increase their motivation for studying programming.

TABLE 4 University students' perspectives on their experience of learning programming through educational games.

No.	Student's perspective	Rating of learning experience (from 1 to 5)	Comment
1	Positive	4.5	Games made the learning process engaging
2	Practical AND Interesting	4.0	I like that we apply skills in games immediately
3	Boosts motivation	4.2	It increases my motivation to learn
4	Develops problem- solving skills	3.8	I have improved my problem-analysis and problem-solving skills
5	Limitations in content	3.0	Some games do not cover all the necessary topics
6	Prefer traditional methods	2.5	I prefer traditional lectures and textbooks.
7	Depends on the quality of games	4.3	Good games have made learning more interesting
8	Enhances communication and collaboration	4.1	Games that require teamwork have helped me better understand collaboration

Development of Problem-Solving Skills: Research participants improved their problem-analysis and problem-solving skills as a result of learning through educational games.

Limitations in Content: Some students pointed out limitations in educational games, noting that not all necessary topics were well-covered in the games.

Preference for Traditional Methods: Despite the positive aspects of educational games, some students prefer traditional teaching methods, such as lectures and textbooks.

Dependence on Game Quality: The quality of educational games significantly influences the learning experience. Good games made learning more interesting and productive.

Enhancement of Communication and Collaboration: Games that require teamwork contribute to improving students' communication and collaboration skills.

Overall, the research results confirm that educational games can be an effective method for programming education and have a positive impact on students' motivation and interest in learning computer science. However, it is essential to consider the quality of games and ensure full coverage of necessary topics to ensure successful learning (Mathew et al., 2019).

Table 5 presents the opinions of university students regarding the advantages of learning programming through educational games. Let us rephrase their opinions:

1. Facilitating the learning process: More than half of the students (53.3%) noted that educational games significantly ease the learning process, making it easier to grasp information.
2. Creating a pleasant learning environment: Approximately 38.3% of students emphasized that educational games make the learning process engaging and allow them to have a pleasant learning experience.
3. Sustaining motivation in learning: 28.3% of students stated that educational games create a favorable learning environment and make learning more consistent.
4. Increasing motivation levels: 18.3% of students expressed that educational games have ignited their desire to learn and boosted their motivation to improve the learning process.
5. Assisting in problem-solving skill development: 8.3% of students claimed that educational games contribute to the development of rational decision-making skills and the ability to solve problems quickly.

From this table, it is evident that university students attach great importance to educational games as an effective means of learning programming. These games facilitate the learning process, create an enjoyable learning environment, sustain motivation, and contribute to the development of crucial skills.

Additionally, here are some opinions from students who participated in the study on the benefits of learning programming through educational games:

1. Student 1: "Games have made the learning process exciting. I feel more actively engaged in the learning process, making programming more interesting."
2. Student 2: "I like that we can immediately apply the skills we learn in games. It makes learning more practical, and I understand how to apply programming in practice better."
3. Student 3: "Games create a pleasant learning environment. I take pleasure in the process and attend classes with a lot of interest."
4. Student 4: "Learning programming through educational games sustains my motivation. I feel I can achieve success, making learning more consistent."
5. Student 5: "Educational games have helped me enhance my problem-solving and analytical skills. I've become more confident in my abilities."
6. Student 6: "Despite the advantages, some games are limited in content. Sometimes, they don't cover all the necessary topics, and I have to seek additional sources of information."
7. Student 7: "I prefer traditional learning methods. Educational games are good, but they can't replace classic lectures and textbooks."
8. Student 8: "The quality of games plays a significant role. Good games have made learning more interesting and effective."
9. Student 9: "Educational games have helped me better understand teamwork and communication. They foster the development of teamwork skills."

These students' opinions highlight the diversity of perceptions and experiences in learning programming through educational

TABLE 5 University students' opinions on the advantages of learning programming through educational games.

Rephrased themes:	Codes:	*F*	%
Facilitating the learning process	Facilitates learning	32	53.3
	Provides easy reinforcement of information		
Creating an enjoyable learning environment	Ensures engaging learning	23	38.3
	Allows for enjoyable learning experiences		
Sustaining motivation in learning	Creates a conducive learning environment	17	28.3
	Makes learning consistent		
Enhancing motivation levels	Ignites the desire for learning	11	18.3
	Awakens the desire to enhance the learning process		
Fostering problem-solving skills development	Provides the ability to generate rational solutions	5	8.3
	Facilitates the development of rapid problem-solving skills		

games. Some students see numerous advantages in this teaching method, while others prefer more traditional approaches.

Table 6 presents the opinions of university students who participated in the study regarding the drawbacks of learning programming through educational games.

Inadequacy in Educational Games:

- 35% of students noted drawbacks in educational games, with 21 of them highlighting the lack of suitable educational games. The absence of engaging educational games was also pointed out as a drawback.

Encountered Technical Issues:

- 25% of students identified shortcomings in the technical infrastructure. Online access problems were also mentioned as a drawback.

No drawbacks:

- 23.3% of participating students claim that there are no drawbacks to learning programming through educational games.

Information Infrastructure Drawbacks:

- 11.6% of students emphasized the lack of programming knowledge among novice students. Insufficient information in the information infrastructure was also noted.

These results demonstrate that students' opinions vary, and they identified various drawbacks in learning programming through educational games, such as inadequacies in available games and technical issues. However, some students do not see significant drawbacks in this learning method.

The following statements represent the opinions of some students who participated in the study regarding the drawbacks of learning programming through educational games:

1. Student 1: "I think one of the main problems is the lack of suitable educational games. Some of them simply do not cover all the necessary topics, making the learning incomplete."
2. Student 2: "We encountered issues with the technical infrastructure. Online access was not always stable, slowing down the learning process."
3. Student 3: "I would say we don't have significant drawbacks in learning programming through educational games. We can find a lot of useful information, and the learning process is enjoyable."
4. Student 4: "The lack of information in the information infrastructure could be a problem for newcomers. Some students might need more support and resources."

These comments illustrate the diversity of opinions among students regarding the drawbacks of learning programming through educational games. Some students identify clear issues, such as the absence of suitable games and technical problems, while others do not consider them to be significant drawbacks (Mathew et al., 2019).

The opinions of university students who participated in this study regarding motivation for learning programming through educational games represent a diverse spectrum of responses. The study revealed different levels of motivation and interest among students.

Table 7 presents the opinions of university students who participated in the study regarding their motivation to learn programming through educational games. The results show the following:

Approximately 16.7% of students expressed a strong desire to learn programming through educational games. This group of students is highly motivated to actively engage in the educational process and sees games as a stimulus for their learning.

The majority of participants, accounting for 41.7%, expressed a willingness to learn using educational games. They understand the value of this method, although their motivation may not be as strong as the first group. These students are ready to embrace the challenge of learning programming in a gaming format.

TABLE 6 University students' opinions on the drawbacks of learning programming through educational games.

Themes	Codes	F	%
Inadequacy in educational games	Lack of suitable educational games	21	35
	Absence of engaging educational games		
Technical issues encountered	Drawbacks of technical infrastructure	15	25
	Problems with online access		
No drawbacks	No drawbacks	14	23.3
	No disadvantage		
Drawbacks in information infrastructure	Lack of knowledge in programming among students, beginners	7	11.6
	Insufficient information infrastructure		

TABLE 7 University students' opinions on their motivation for learning programming through educational games.

Opinions of schoolchildren	F	%
I would love to	10	16.7
I never want to	25	41.7
I am unsure	12	20
I am not interested in	9	15
I would like to	4	6.6
Total	60	100

Approximately 20% of students were unsure about their motivation or found it difficult to provide a definite answer. This may indicate a lack of clarity in their motivation or the need for more detailed information about the teaching method; 15% of students stated that they are not interested in learning programming through educational games. This group of students likely prefers more traditional teaching methods.

Finally, 6.6% of students expressed a desire to learn using educational games, although their numbers are relatively small. This group may represent students who see a specific advantage in the gaming method.

In conclusion, the research results demonstrate that university students have varying levels of motivation and interest in learning programming through educational games, and taking into account this diverse motivation can be important when developing educational programs.

5 Discussion

The purpose of this study was to understand how university students experience learning programming through educational games in extracurricular computer science education. Rather than evaluating effectiveness, the findings reveal how students interpret this mode of learning and what role it plays in their cognitive and emotional engagement with programming.

The results indicate that educational games change the perceived nature of programming. Students no longer experience programming primarily as a rule-based academic subject but as an interactive problem-solving activity. The immediate feedback provided by games allows learners to construct understanding through experimentation. This supports constructivist perspectives in which knowledge emerges from interaction rather than transmission.

An important experiential shift identified in the study concerns emotional perception. Students described a transition from anxiety to curiosity. Errors inside game environments were interpreted as part of exploration rather than academic failure. This suggests that educational games function as psychologically safe learning spaces, reducing the fear commonly associated with introductory programming courses. Thus, the pedagogical value of games may lie less in content delivery and more in lowering the entry barrier to engagement.

At the same time, participants did not perceive games as complete instructional systems. They frequently described knowledge obtained through gameplay as intuitive but fragmented. Students could solve tasks but struggled to articulate underlying concepts until the instructor's explanation occurred. This indicates that game-based learning initiates understanding but does not finalize conceptual structuring. Traditional teaching, therefore, remains necessary for conceptual consolidation.

Another important finding concerns context dependence. The meaning students attributed to educational games varied depending on instructional integration. When gameplay was followed by discussion and reflection, students perceived meaningful learning. When used independently, games were experienced as puzzles rather than education. This suggests that the effectiveness of educational games is not inherent in the software itself but emerges from pedagogical mediation.

The reported "lack of content" should therefore be interpreted cautiously. Students' statements did not necessarily reflect deficiencies in the games but rather a mismatch between experiential discovery and formal curriculum structure. Games introduce concepts implicitly, whereas academic courses require explicit conceptual frameworks. The perceived gap arises between these two modes of learning rather than from the absence of learning itself.

The findings also show that students experience programming differently in social contexts. Collaborative gameplay transformed programming from an individual technical task into a shared reasoning activity. In this sense, educational games supported the development of communication and collective problem-solving practices associated with authentic software development processes.

Overall, the study suggests that educational games serve as an entry environment that initiates engagement, supports experimentation, and reshapes emotional attitudes toward programming. However, conceptual understanding emerges only when gameplay is integrated with explanation and reflection. Therefore, educational games function most effectively as a complementary pedagogical medium rather than an autonomous teaching method.

The detailed description of learning conditions enables analytical transferability to comparable educational environments.

6 Conclusion

This study explored university students' perceptions of learning programming through educational games in extracurricular computer science education. Using a qualitative phenomenological approach, the research focused on students' subjective experiences, motivations, and perceived benefits and limitations associated with game-based learning.

The findings show that students hold diverse views regarding the use of educational games in programming education. While many participants expressed interest in learning through games and associated this approach with higher engagement and motivation, others preferred traditional instructional methods. In addition, students identified several challenges related to the quality of available educational games and limitations in technical infrastructure.

These results suggest that educational games are perceived by students as a potentially useful complementary approach rather than a replacement for conventional teaching methods. The effectiveness of such approaches appears to depend on factors such as game quality, topic coverage, technical conditions, and individual learner preferences.

The study contributes to the understanding of students' perceptions of game-based programming education in extracurricular settings. Future research may further examine how different types of educational games and instructional designs influence students' learning experiences across diverse educational contexts.

The description of the learning context and tools enables analytical transferability of the findings to comparable educational settings.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding authors.

Ethics statement

Ethical review and approval was not required for the study on human participants in accordance with the local legislation and institutional requirements. The studies were conducted in accordance with the local legislation and institutional requirements. The participants provided their written informed consent to participate in this study.

Author contributions

BT: Writing – original draft, Project administration, Conceptualization. AN: Project administration, Software, Methodology, Writing – review & editing. KI: Writing – original draft, Formal analysis, Visualization. ZT: Methodology, Writing – original draft, Conceptualization. NN: Supervision, Writing – original draft, Software, Investigation. AA: Investigation, Formal analysis, Writing – review & editing. ZA: Methodology, Formal analysis, Software, Writing – review & editing.

Funding

The author(s) declared that financial support was not received for this work and/or its publication.

Conflict of interest

The author(s) declared that this work was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declared that generative AI was not used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

References

Denning, P. J., and Tedre, M. (2021). Computational thinking: a disciplinary perspective. *Inform. Educ.* 20, 361–390. doi: 10.15388/infedu.2021.21

Kane, I., Pham, C.-T., Lewis, A. W., and Miller, V. (2019). “Escape from the Python’s Den: an educational game for teaching programming to younger students,” in *ACM Southeast Regional Conference, April 18 - 20, 2019 GA, Kennesaw, USA*, 279–280. doi: 10.1145/3299815.3314477

Malliarakis, C., Satratzemi, M., and Xinogalos, S. (2014). *Educational Games for Teaching Computer Programming*. New York, NY: Springer, 87–98. doi: 10.1007/978-1-4614-6501-0_7

Mathew, R., Malik, S. I., and Tawafak, R. M. (2019). Teaching problem solving skills using an educational game in a computer programming course. *Inform. Educ.* 18, 359–373. Available online at: https://www.ceeol.com/search/article-detail?id=804181 (Accessed September 11, 2026).

Mertens, D. M. (2019). *Research and Evaluation in Education and Psychology: Integrating Diversity With Quantitative, Qualitative, and Mixed Methods, 5th Edn*. Sage publications. Available online at: https://www.scirp.org/reference/referencespapers?referenceid=3635745 (Accessed October 1, 2026).

Papadakis, S., and Kalogiannakis, M. (2019). Evaluating the effectiveness of a game-based learning approach in modifying students’ behavioural outcomes and competence, in an introductory programming course. A case study in Greece. *Int. J. Teach. Case Stud.* 10, 235–250. doi: 10.1504/IJTCS.2019.102760

Petri, G., von Wangenheim, C. G., Borgatto, A. F., Calderón, A., and Ruiz, M. (2022). “Digital games for computing education: what are the benefits?,” in *Research Anthology on Developments in Gamification and Game-Based Learning* (IGI Global Scientific Publishing), 1571–1598. Available online at: https://www.igi-global.com/chapter/digital-games-for-computing-education/293720 (Accessed September 30, 2026).

Sailer, M., and Homner, L. (2020). The gamification of learning: a meta-analysis. *Educ. Psychol. Rev.* 32, 77–112. doi: 10.1007/s10648-019-09498-w

Tisdell, E. J., Merriam, S. B., and Stuckey-Peyrot, H. L. (2025). *Qualitative Research: A Guide to Design and Implementation*. John Wiley and Sons.

Tokac, U., Novak, E., and Thompson, C. G. (2019). Effects of game-based learning on students’ mathematics achievement: a meta-analysis. *J. Comput. Assisted Learn.* 35, 407–420. doi: 10.1111/jcal.12347

Videnovik, M., Vold, T., Kiønig, L., Madevska Bogdanova, A., and Trajkovik, V. (2023). Game-based learning in computer science education: a scoping literature review. *Int. J. STEM Educ.* 10:54. doi: 10.1186/s40594-023-00447-2

Wagan, A. A., Khan, A. A., Chen, Y. L., Yee, P. L., Yang, J., and Laghari, A. A. (2023). Artificial intelligence-enabled game-based learning and quality of experience: a novel and secure framework (B-AIQoE). *Sustainability* 15:5362. doi: 10.3390/su15065362

Zainuddin, Z., Chu, S. K. W., Shujahat, M., and Perera, C. J. (2020). The impact of gamification on learning and instruction: a systematic review of empirical evidence. *Educ. Res. Rev.* 30:100326. doi: 10.1016/j.edurev.2020.100326

Zheng, E., and Wang, Q. (2023). Effectiveness of online collaborative learning in gamified environments. *Int. J. Emerg. Technol. Learn.* 18, 33–44. doi: 10.3991/ijet.v18i17.42851