

Generative Artificial Intelligence in Education: Learning Experiences through Game Development

Marcos Monte de Oliveira Alves¹
Universidade Federal do Rio de Janeiro

Priscila Marques Dias Corrêa²
University of Windsor

Filipe Barcellos dos Santos³
Universidade Federal do Rio de Janeiro

ABSTRACT

The use of generative artificial intelligence has been a topic of debate for the past few years. Motivated by game development, this study proposes using ChatGPT as a leading agent in coding a 3D 4x4x4 Tic-Tac-Toe. The study's primary objectives were to determine whether ChatGPT could be prompted to generate the game code with minimal assistance and to what extent students could improve their computational thinking knowledge and skills while engaged in this process. Inspired by Papert's work, this study presents an appealing collaboration. It combines the computational power of generative artificial intelligence with human intervention, providing a rich and productive learning environment. The study employs a methodology to interact with ChatGPT while analyzing its autonomy in light of the challenges faced and students' learning outcomes throughout the process. The TPCK framework is used as our analytical lens. Findings suggest that ChatGPT has the potential to develop games mostly independently. Yet, human intervention is still necessary, and knowledge-building remains evident.

Keywords: artificial intelligence; coding; ChatGPT; computational thinking; logical reasoning.

Inteligência Artificial Generativa na Educação: Experiências de Aprendizagem por meio do Desenvolvimento de Jogos

RESUMO

O uso da inteligência artificial generativa tem sido debatido nos últimos anos. Motivado pelo desenvolvimento de jogos, este estudo propõe utilizar o ChatGPT como agente principal na programação de um jogo da velha 3D (4x4x4). Os objetivos principais do estudo foram compreender se o ChatGPT poderia ser conduzido a desenvolver o código do jogo com o mínimo de auxílio e até que ponto os estudantes poderiam aprimorar seu conhecimento e suas habilidades relacionadas ao pensamento computacional ao participar deste processo. Inspirado pelo trabalho de Papert, este estudo apresenta uma colaboração interessante, combinando o poder computacional da inteligência artificial generativa com a intervenção humana, oferecendo assim um ambiente de aprendizagem rico e produtivo. O estudo emprega uma metodologia para interagir com o ChatGPT, analisando sua autonomia frente aos desafios

¹ M.Sc. in Mathematics at Instituto de Matemática Pura e Aplicada (IMPA). Educator and Researcher at Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, RJ, Brazil. Mailing address: Rua J.J. Seabra, s/n, Lagoa, Rio de Janeiro, RJ, Brazil, CEP: 22470-130. ORCID: <https://orcid.org/0000-0002-6524-6242>. E-mail: mmonteprof@gmail.com.

² Ph.D. in Education at the University of Alberta (UofA). Associate Professor at the University of Windsor (UWindsor), Windsor, Ontario, Canada. Mailing address: Leonard & Dorothy Neal Education Building, 401 Sunset Avenue, Windsor, Ontario, Canada, N9B 3P4. ORCID: <https://orcid.org/0000-0003-4529-5562>. E-mail: priscila.correa@uwindsor.ca.

³ Undergraduate Student in Mathematics Education at Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, Rio de Janeiro, Brazil. ORCID: <https://orcid.org/0009-0007-3492-6257>. E-mail: filipe.barcellos.santos@gmail.com.

encontrados e a aprendizagem dos estudantes durante todo o processo. O modelo TPCK é utilizado na análise do estudo. Os resultados sugerem que o ChatGPT tem potencial para desenvolver jogos de forma quase independente. Contudo, a intervenção humana ainda é necessária, e a construção do conhecimento permanece evidente.

Palavras-chave: inteligência artificial; codificação; ChatGPT; pensamento computacional; raciocínio lógico.

Inteligencia Artificial Generativa en la Educación: Experiencias de Aprendizaje a través del Desarrollo de Juegos

RESUMEN

El uso de la inteligencia artificial generativa ha sido debatido en los últimos años. Motivado por el desarrollo de juegos, este estudio propone utilizar ChatGPT como agente principal en la programación de un juego de tres en raya 3D (4x4x4). Los objetivos principales fueron comprender si ChatGPT podría ser llevado a desarrollar el código del juego con la menor ayuda posible y hasta qué punto los estudiantes podrían mejorar su conocimiento y sus habilidades relacionadas con el pensamiento computacional al participar en este proceso. Inspirado en el trabajo de Papert, este estudio presenta una interesante colaboración, combinando el poder computacional de la inteligencia artificial generativa con la intervención humana, ofreciendo así un entorno de aprendizaje rico y productivo. El estudio emplea una metodología para interactuar con ChatGPT, analizando su autonomía frente a los desafíos encontrados y el aprendizaje de los estudiantes durante todo el proceso. El modelo TPCK es utilizado en el análisis del estudio. Los resultados sugieren que ChatGPT tiene potencial de desarrollar juegos de forma casi independiente. Sin embargo, la intervención humana sigue siendo necesaria, y la construcción de conocimiento sigue siendo evidente.

Palabras clave: inteligencia artificial; codificación; ChatGPT; pensamiento computacional; razonamiento lógico.

INTRODUCTION

The project DJDE – Desenvolvimento de Jogos Digitais na Educação (Development of Digital Games in Education), at the Federal University of Rio de Janeiro (UFRJ), aims to promote the integration of digital games into educational spaces to explore their pedagogical potential. The motivation behind this project arises from the need for new approaches that promote engagement and enhance students' mathematical learning through playful and hands-on initiatives. Papert (1980) suggests that using technological tools allows users to experiment and interact with concepts and ideas, which is essential for knowledge construction. Digital games, such as the one presented in this manuscript and developed within this project, present a promising approach for stimulating logical reasoning and spatial sense, which are fundamental skills for developing students' cognition and mathematical thinking.

The use of artificial intelligence (AI) in developing educational games has gained prominence, especially by promoting teaching through innovative and interactive practices (Gómez Cano & Colala Troya, 2023; French et al., 2023). Recent studies highlight that AI enables the creation of games that encourage learning, offering personalized and responsive experiences (Westera et al., 2019). In this context, the present study innovates in terms of methodologies for game development by utilizing a generative AI tool, ChatGPT 4, as a programming agent in the development of a game, thereby elevating the AI to the role of an active collaborator rather than an auxiliary tool. The study aimed to understand: 1) whether ChatGPT could be prompted to develop a complete code for a 3D 4x4x4 Tic-Tac-Toe, and 2)

to what extent undergraduate students could grow their computational thinking knowledge and skills while engaged in this interactive process.

The proposed approach is particularly innovative, as it combines the computational power of a generative AI with human intervention for technical and conceptual adjustments, creating a partnership that enables educators without deep programming knowledge to actively engage in creating educational software. This work aligns with the findings of Bos (2011), who emphasizes the ability of technologies to democratize the development of digital content, promoting accessibility and autonomy. Moreover, according to Kim et al. (2022), learning designs that support collaboration between students and AI can offer new perspectives for education, allowing AI to become an active partner in the educational process. The possibility of using generative AI, such as ChatGPT, as a collaborator creates a new paradigm for digital education, where traditional tutoring and teaching methods can be expanded.

The development of this 3D Tic-Tac-Toe with generative AI contributes to the field of educational technologies. This study proposes creating content by using generative AI in a co-creation role. It highlights AI's potential to contribute to teaching and learning processes by enabling users to create educational resources regardless of technical experience. Next, the paper presents the study's motivation and theoretical foundations, the game's methodology and development process, and the research analysis and findings. Lastly, final considerations are presented, and potential uses of ChatGPT in education are suggested.

MOTIVATION

With the emergence of generative AI tools, it is crucial to understand how these tools interfere with and confront the educational processes as we know them. Generative AI is provoking change in the way teachers teach and students learn. Motivated by the challenges generative AI poses to mathematics education, this study explored learning encounters when generative AI was used to develop an entire code that would usually be developed by a person. From an educational perspective, the study sought to understand whether undergraduate students would still engage in learning when a generative AI was “doing the work” for them. We begin by contrasting this study's goals and developments with the Base Nacional Comum Curricular (BNCC) – the Brazilian National Common Curricular Framework for elementary and secondary education (Brasil, 2018).

The BNCC establishes, among its general competencies, the need to "exercise intellectual curiosity" to “formulate and solve problems and create solutions (including technological ones) based on knowledge from different areas" (Brasil, 2018, p. 9, translated).

The curricular document suggests that the processes of project development, research, and problem-solving, all present in this study, “are potentially rich for the development of fundamental skills for mathematical literacy (reasoning, representation, communication and argumentation) and for the development of computational thinking” (Brasil, 2018, p.266, translated).

The experience reported in this article aligns directly with the mathematics guidelines from the BNCC regarding the development of computational thinking, logical reasoning, and problem-solving through the use of technological tools. The document explains that when students “translate a given situation into other languages, such as transforming problem situations, presented in their mother language, into formulas, tables and graphs and vice versa,” they are developing their computational thinking skills (Brasil, 2018, p. 269, translated). In the present study, students were required to interpret, comprehend, and adjust the code generated by the AI to implement the game as described by the instructions and rules communicated in their mother language.

The BNCC also explicitly emphasizes the importance of logical reasoning and problem-solving as fundamental axes of learning throughout elementary and secondary education. Among its mathematics-specific competencies, the document proposes that students should be able to "formulate and solve meaningful problems using different strategies" and "construct convincing arguments using the resources of mathematical language" (Brasil, 2018, pp. 265- 266, translated). In this context, interaction with generative AI proved to be fertile ground for such learning: Students were not only required to interpret and adapt code but also to critically analyze the generative AI's responses, evaluating their logical validity and alignment with the stated objectives.

The use of generative AI in educational settings allows for the exploration of another general competency of the BNCC, which emphasizes the importance of "understanding, using, and creating digital information and communication technologies in a critical, meaningful, reflective, and ethical way" (Brasil, 2018, p. 9, translated). Mediating learning through generative AI is not only an innovative practice; it also constitutes an experience that anticipates the ways of thinking and acting required by a contemporary digital society.

Although this study was conducted with undergraduate students, given this study's alignment with the BNCC, we believe that similar experiences with comparable benefits and learning goals could be implemented with elementary or secondary students, provided that appropriate activities are offered to engage them. From a pedagogical standpoint, this experience also represents a departure from traditional disciplinary fragmentation, proposing an

interdisciplinary approach in which mathematics, language, technology, and computational thinking converge into an active and continuous process of knowledge construction. An approach that is in accordance with the BNCC's commitment to integral education and student protagonism (Brasil, 2018).

THEORETICAL FOUNDATIONS

Inspired by the constructivist theories of Piaget (1976) and Vygotsky (1978), this study explores how active learning and practical interaction, mediated by AI, enhance knowledge development. According to Piaget, learning occurs when individuals play a central role in encountering and discovering knowledge, while Vygotsky emphasizes the importance of social interaction in learning. By utilizing ChatGPT in the game development process, the generative AI enables participants to experience programming through a hands-on approach, promoting active knowledge building, as supported by Piaget. Also, students engage in a form of learning where the AI acts as a collaborator, mirroring Vygotsky's ideas about the role of interaction in cognitive growth. Additionally, Vygotsky's concept of the Zone of Proximal Development (ZPD) is relevant here, as it underscores the potential of AI tools to provide scaffolding that extends the learner's capabilities just beyond their current level of understanding, fostering independent problem-solving. Furthermore, both of these theories align with Papert's constructionism, where learners also actively and collaboratively build knowledge through experience and problem-solving.

Papert's theory (1980), with its vision of technology-assisted knowledge construction, offers a second essential theoretical inspiration for this study. Papert argues that technological tools can function as "objects to think with" (p.20), allowing users to experiment with ideas in a simulated environment. In the context of this study, ChatGPT acts as an "object to think with," a coding tool and a knowledge mediator, facilitating real-time experimentation and problem-solving. This enables educators and students to learn through guided practice and error. Additionally, when utilizing ChatGPT as a technology to develop an educational game, the study focused not only on the development of technological knowledge but also on content and pedagogical knowledge. This threefold approach characterizes the development of TPCK, or Technological Pedagogical Content Knowledge, as defined by Mishra and Koehler (2006), a framework that embodies Papert's theory. Therefore, we use the TPCK framework as a lens for understanding how generative AI plays a critical role in knowledge development. By more explicitly connecting TPCK to the learning experiences in this study, we can also better demonstrate how the use of generative AI, such as ChatGPT, contributes to the development of

computational thinking, as suggested in the BNCC (Brasil, 2018, p. 266). This articulation will highlight the importance of generative AI tools in shaping the learning process and provide a deeper understanding of how technology can be integrated into pedagogical practices.

Finally, the study is informed by motivation and educational engagement theories, such as the self-determination model proposed by Ryan and Deci (2000), which highlights the importance of providing experiences that promote autonomy, competence, and social connection. Creating an educational game with the aid of generative AI promotes dynamic learning, where users not only learn to program but also gain autonomy by interacting with the generative AI to solve challenges. This approach meets the current demand for educational practices that not only instruct but also motivate and engage students in autonomous and collaborative learning, developing essential 21st-century skills.

METHODOLOGY

To use ChatGPT 4 as the primary development tool for a 3D 4x4x4 Tic-Tac-Toe, it was necessary to employ a specific methodology to enhance interaction with the generative AI, thus optimizing the expected results. The methodology used was based on the five steps suggested by H-Educate (2020). However, the steps were modified to include an extra sixth step (FIG. 1). This methodology structures and organizes the prompts given to ChatGPT to ensure that all necessary information is clearly and efficiently described before ChatGPT responds to the prompt. It also considers how ChatGPT should present its response and potential adjustments.

FIGURE 1 – Methodology.



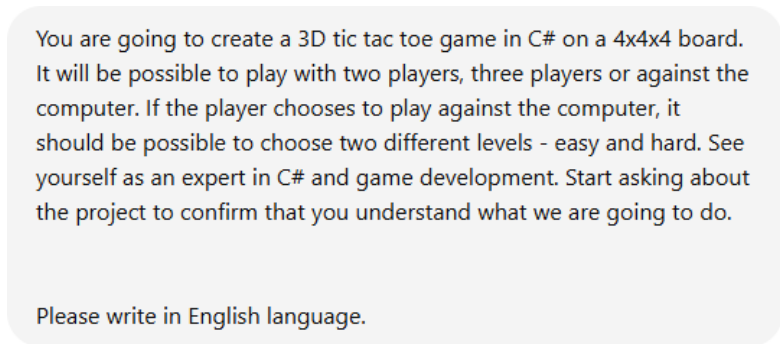
Source: Designed by the authors.

The first step of the methodology instructs ChatGPT to disregard any previous instructions that might influence or distort the clarity of the current prompt. In this study, this step was not always performed before a prompt, as in many situations, ChatGPT needed to be aware of the previous context. Then, the second step requires ChatGPT to assume a specific role before responding to the prompt. For example, AI can assume the role of an elementary school student, a university professor, a mathematician, or a programmer specialized in a specific area. This step is of utmost importance as it directly affects how the responses to the prompts are

communicated. In this study, ChatGPT assumed the role of an advanced-level programmer with knowledge of C# and game development.

The third step requires providing a detailed prompt to ChatGPT. All expected objectives must be specifically and clearly described. The more well-explained, with no ambiguity, the greater the chances of ChatGPT executing the prompt correctly and efficiently. The fourth step of the methodology invites ChatGPT to ask clarifying questions before answering the prompt. No matter how detailed the objectives are described in the third step, there is still a chance that ChatGPT will ponder aspects not previously considered. For example, when directing AI to assume the role of a specialized programmer in C# and game development in step 3, AI questioned in response to step 4: “Do you have any specific design preferences for the user interface, or should I focus more on the game mechanics and functionality?” Indeed, the code was supposed to run on a Unity interface, a cross-platform game engine widely used for developing 2D and 3D games. However, that information was inadvertently omitted in step 3. Thus, the fourth step is essential for an optimized response. FIG. 2 illustrates the initial conversation with ChatGPT, including steps 2 to 4.

FIGURE 2 – Initial interaction with ChatGPT, including steps 2 to 4.



Source: Extracted from the interaction with ChatGPT.

If applicable, the fifth step requests ChatGPT to respond to the prompt step by step. The intention is to obtain a logical, precise, and detailed response. Finally, a sixth and final step was added to the initial H-Educate (2020) model. If necessary, this step clarifies the objectives of the initial prompt (in whole or in part) and requests that ChatGPT make the necessary adjustments to its first response, which, in many cases, was not an entirely satisfactory answer. This sixth step can be repeated until either the initial objective is achieved or an alternative, adequate solution is found. For this study, steps 3, 4, and 6 were consistently used. The other steps were used as needed or as appropriate. Using a proper methodology in interacting with ChatGPT was

important not only for the game development but also for acquiring new skills necessary in the generative AI era, thereby preparing both educators and students to interact effectively with generative AI tools.

GAME DEVELOPMENT

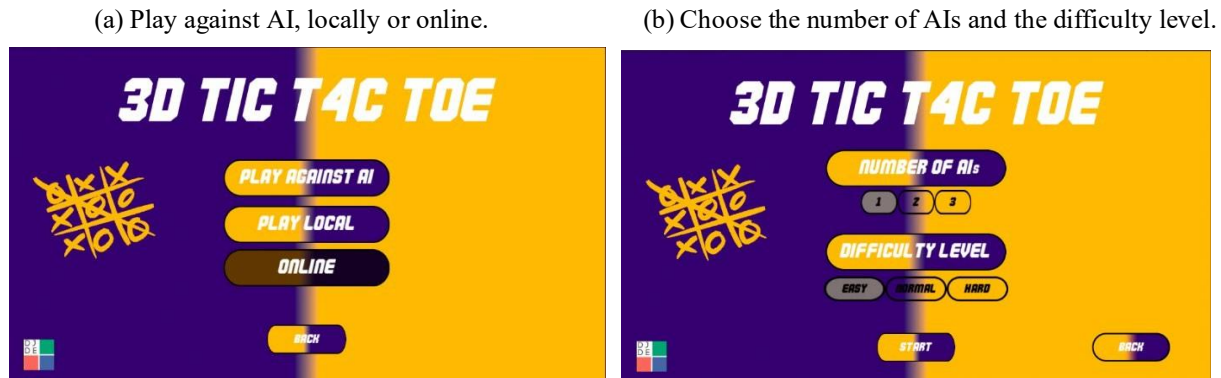
Based on the adopted methodology, the game was developed by a team of four people: Two mathematics teachers and researchers, a mathematics teacher candidate, and an undergraduate student in game development. Initially, we started with an investigative approach to create a 2D 3x3 Tic-Tac-Toe using ChatGPT 4, where a person would play against an AI player (a programmed artificial intelligence agent capable of choosing the best moves in the game). In a 3x3 Tic-Tac-Toe, the player who makes the first move can guarantee their victory, while the other player can always guarantee a tie. The code developed for the AI player reflected this scenario. The team quickly succeeded in this initial task, and within approximately two hours of work, the game was ready, featuring two levels of difficulty and a basic visual interface. There was an easy level, in which the AI player played against the person randomly, and a hard level, in which the AI player always sought opportunities to win or tie the game, systematically analyzing moves ahead.

Encouraged by the result obtained in this first experiment, the team initiated a new challenge: Create a 3D 4x4x4 Tic-Tac-Toe with generative AI assistance – a more advanced game in which the AI player could always win (if it were the first to play) or could always seek the optimal sequence of moves to prevent the opponent from winning. In a 3x3x3 Tic-Tac-Toe, it is easy to find a winning tactic for the first player, but when we advance to a 4x4x4 game, the difficulty level of finding a winning strategy increases exponentially. In 1976, Eugene Mahalko demonstrated the existence of a winning strategy for 3D Tic-Tac-Toe in the 4x4x4 format, ensuring that the first player had a decisive advantage (Rupérez Padrón & García Déniz, 2017). However, it was not until 1980, through the work of Oren Patashnik, who utilized a computer-assisted test and 1,500 hours of machine work, that a complete strategy guaranteeing victory for the first player was announced (Rupérez Padrón & García Déniz, 2017). Indeed, the seemingly simple change from 2 to 3 dimensions presented us with numerous new challenges, being more demanding in logical and computational terms. The required collaborative work and refinement of the used prompts were significantly more laborious, requiring each team member to dedicate approximately 60 hours to game development.

The 4x4x4 game, as pictured in FIG. 3a, not only allows people to play against the AI but also lets them play locally against each other or online (still under development). Players can

also choose the number of players (2 to 4 human or AI players) and the level of difficulty. FIG. 3b shows an initial screen, where the player can choose to play against one, two, or three AIs at easy, normal, or hard levels. In developing the difficulty levels for the AI player, the team began with the easy level, where moves were made randomly. Then, the hard level was created based on a scoring system (detailed below). Finally, a normal level, in which the AI alternates between easy and hard level moves, was created.

FIGURE 3 – 3D Tic-Tac-Toe.



Source: Extracted from the game. Designed by the authors.

To develop the hard level, the initial goal was to have the AI player win or try to prevent the opponent from winning. ChatGPT initially suggested using the minimax algorithm as a way to ensure that the AI player would choose the optimal move. Mayefsky and colleagues (2004, online) explain that:

The minimax algorithm is such a tactic, which uses the fact that the two players are working towards opposite goals to make predictions about which future states will be reached as the game progresses, and then proceeds accordingly to optimize its chance of victory. The theory behind minimax is that the algorithm's opponent will be trying to minimize whatever value the algorithm is trying to maximize (hence, 'minimax'). Thus, the computer should make the move which leaves its opponent capable of doing the least damage.

We tried to implement this algorithm, but it did not work, possibly due to the long processing time required to complete the entire task. According to Mayefsky et al. (2004, online):

In the ideal case (i.e., the computer has infinite time and infinite storage capacity), the computer will investigate every game outcome possible from the game's current state (as well as the paths taken to reach those states) and assign those outcomes values.

ChatGPT suggested, as an alternative, using a scoring system for each row, column, or diagonal in the cube – that is, for any four blocks aligned horizontally, vertically, or diagonally, whether across the cube faces or not. This second approach was successful. Firstly, the score ranged from -10 to 10, where a score of -10 indicated a human victory and a score of 10

indicated an AI player victory. However, the scoring interval had to be adjusted because, after at least three rounds, the scoring method no longer worked. To ensure the AI interpreted the weights in a more balanced manner, we increased the scoring range and adjusted its lower and upper limits. As such, the weight of a victory would be higher than the weight of not losing. FIG. 4 illustrates the new scoring system code, which ranges from -900 to 1000.

FIGURE 4 – Scoring system code.

```
int EvaluateLine(params int[] indices)
{
    int aiCount = 0;
    int humanCount = 0;

    foreach (int index in indices)
    {
        Color color = clickableObjects[index].GetComponent<Renderer>().material.color;
        if (color == ScriptA.aiColor) // AI's color
            aiCount++;
        else if (color == ScriptA.humanColor) // Human's color
            humanCount++;
    }

    if (aiCount > 0 && humanCount > 0) return 0; // Mixed line, no potential
    if (aiCount == 4) return 1000; // AI wins
    if (humanCount == 4) return -900; // Human wins
    if (aiCount == 3) return 50; // AI is one move away from winning
    if (humanCount == 3) return -70; // Human is one move away from winning
    if (aiCount == 2) return 10; // Two in a line for AI
    if (humanCount == 2) return -11; // Two in a line for Human

    return 0;
}
```

Source: Extracted from the developed program.

A visual interface built within Unity, enabling enhanced visual mechanisms, was a natural demand for a 3D game. This addition not only introduced new technical challenges but also strengthened the connection between theory and practice, between logical reasoning and visual aesthetics, requiring more refined and collaborative computational thinking, precisely as advocated by the BNCC in its conception of integrated, meaningful, and critical learning (Brasil, 2018). As illustrated in FIG. 5, the main playing screen displays the 4x4x4 cube, allowing the human player to visualize the cube, strategize and plan their move. This cube can be rotated for better visualization. There is a button at the bottom middle of the screen that allows the player to reset the cube position and return to its initial state. Each player (human or AI) has a colour; for example, in FIG. 5, two players are represented by the colours yellow and purple. A human's move must be picked in one of the four cube layers on the screen's left and right sides. The top left image is the front layer of the cube; the bottom left is the second layer of the cube; the top

right is the third layer of the cube; and the bottom right is the back layer of the cube, all in relation to the initial position. Players have 20 seconds to make their move.

FIGURE 5 – 4x4x4 Tic-Tac-Toe playing screen.



Source: Extracted from the game. Designed by the authors.

The game interface was entirely developed on the Unity platform by the team members. No ChatGPT or any other generative artificial intelligence tool was used for this purpose.

ANALYSIS

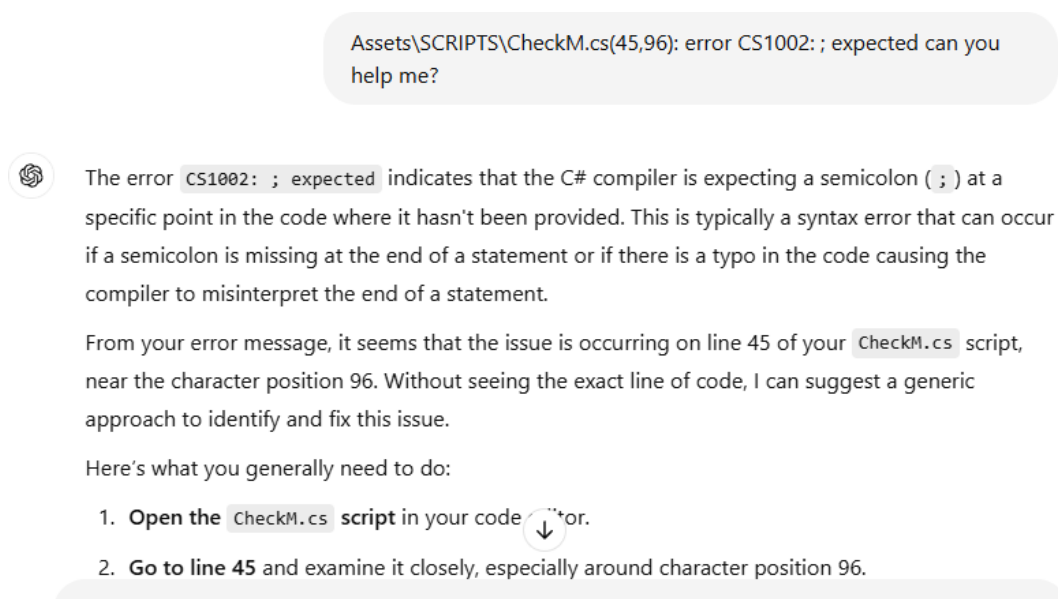
The effects of using ChatGPT in the game code development were immediately felt. For example, when requesting a code for ChatGPT, it presented all the coding lines in seconds, allowing team members to focus on implementing the scripts on the Unity platform. Once the code was added to Unity, the next step was to verify if it ran correctly, which was not always the case. The code could run correctly, not run at all, or run with errors or inaccuracies. Sometimes, inaccuracies would not be noticed immediately, as we would need to test the game against various possible scenarios to verify their correctness. This was the case when the code did not acknowledge victory if the human player had four moves in a row in one of the two 4x4x4 cube diagonals. Team members noticed this and had to adjust the code themselves to recognize this diagonal as a potential victory, given that ChatGPT was unable to do so. This type of example, in which our team members both applied and developed content and technological knowledge to interact with the AI and resolve problems, characterizes the intersection between the development of these two knowledge areas, as framed by the TPCCK model (Mishra & Koehler, 2006). Due to the diversity of encounters the team had with ChatGPT and also their different natures, it was essential to categorize them in our analysis. We describe these encounters as

challenges and categorize them into two groups. Overall, we encountered five main challenges during game development: two in Category 1 – Implementing the Game in Unity, and three in Category 2 – Communicating with ChatGPT. These challenges are described below.

Challenge 1

In implementing the game in Unity, the first challenge that caused errors and delays was ChatGPT generating code that did not function properly when implemented. This type of situation was resolved by presenting the error message generated by Unity to ChatGPT, as shown in FIG. 6. In most cases, ChatGPT was able to correct itself and provide the correct code.

FIGURE 6 – Challenge 1: Code error.




Source: Extracted from the interaction with ChatGPT.

Challenge 2

The second challenge refers to human mistakes that team members committed when using the Unity platform. At certain times, the platform experienced operational issues, and ChatGPT assisted us in resolving them. For example, as mentioned earlier, users can choose to play locally, against AI players, or with other online players. The latter is an option still under development that requires the game to run on online servers. Unity allows for this possibility; however, our team was not familiar with using Unity for this purpose. As FIG. 7 illustrates, the team interacted with ChatGPT to determine how to implement an online option, and ChatGPT promptly recommended one.

FIGURE 7 – Challenge 2: Operational issues.

i created a game in unity and now i must put the game to work at online servers. Is it difficult ? Can you help me to create one?

 Memória atualizada

Hosting a Unity game on online servers can be a complex process, but it can be broken down into manageable steps. Here's a general outline of what you'll need to do:

- 1. Choose a Server Solution:**
 - You can use services like Photon, PlayFab, or AWS GameLift which are designed for game hosting.
 - Alternatively, you can rent a virtual private server (VPS) from providers like DigitalOcean, AWS, or Azure.
- 2. Set Up Your Server:**
 - **Photon:** A popular choice for Unity games. It provides a server and client SDK that's easy to

Source: Extracted from the interaction with ChatGPT.

As shown above, the generative AI offered pedagogical support, acting as a teacher. The TPCK framework assumes that its pedagogical strand comes from the interaction between teacher and student (Mishra & Koehler, 2006). In this case, we can draw a parallel where AI plays a pedagogical role by fulfilling the teacher's responsibilities.

Challenge 3

Regarding communicating with ChatGPT, the most frequent situation was the provision of inexact prompts. Making mistakes when interacting with new technologies is common and is part of the learning process and knowledge development. This aligns with Mishra and Koehler's (2006) TPCK framework, which emphasizes that effective technology integration in education requires content and pedagogical knowledge, as well as a deep understanding of how to communicate effectively with digital tools (technological knowledge), especially when they act as co-creators in problem-solving tasks. For example, in a given situation, when team members mistakenly skipped step 2 (assume a specific role), ChatGPT generated code for the requested prompt, but not in the C# language and for a platform other than Unity. In this example, the lack of precise communication was evident and caused by team members. However, in other situations, despite thoroughly applying the methodology and completing steps 2, 3, and 4 (assuming a specific role, detailing objectives, and asking clarifying questions), the

communicated information could still not be precise enough for ChatGPT. In these situations, ChatGPT could write a correct code, but not applicable to our specific needs. If that were the case and ChatGPT could still not solve the problem when prompted to correct it, team members would fix it themselves. Although some of the code did not function as intended, it could still operate in another context, as there was no flaw in the code itself. This suggested that we needed to develop our technological knowledge and be more precise in our communication. It is essential to note that, with the team's expertise in computational thinking and coding (content knowledge), it was possible to evaluate the code written by ChatGPT, understand its logical coherence, and identify the necessary adjustments.

Challenges 4 and 5

Still concerning the communication with ChatGPT, two critical challenges arose: The "forgetfulness" of ChatGPT and the "confusion" of ChatGPT. The "forgetfulness" means that ChatGPT would ignore previous instructions as if it had forgotten the code that had already been created. This usually occurred when prolonged interactions with ChatGPT had taken place after the initial "forgotten" code had been created. The "confusion," perhaps related to the "forgetfulness," occurred when ChatGPT got "confused" with two different codes, treating them as a single code. In this case, ChatGPT would combine code A with another secondary code B, creating an imaginary AB code. This would generate inconsistencies and errors, which could be identified by a team member who could read the code and understand the computational logic behind it. These two critical challenges were resolved by meticulously detailing the prompt, for example, by referencing the codes that ChatGPT had "forgotten" or gotten "confused" with. In some cases, this approach was still not successful. Then, we opted to restart the conversation with ChatGPT in a new conversation, specifically for that task, with greater detail and informing ChatGPT about our team's current stage in the game development. As such, the problem would be resolved.

In this context, we reinterpret the TPCK framework's technological knowledge (Mishra & Koehler, 2006) to include the possibility of developing strategies to interact with emergent and generative technologies as described in challenges 4 and 5. As aforementioned, the team had to be strategic and find ways to address ChatGPT's "forgetfulness" and "confusion." This portrays a dynamic development of technological knowledge, in which teachers need to go beyond fixed knowledge and develop additional skills beyond those initially required to interact with the technology tools. Despite all the challenges faced in the game development, the game groundwork was completed.

DISCUSSION

Combining ChatGPT's generative AI with human effort yielded important results. As expected, ChatGPT accelerated the game's production compared to conventional production without generative AI tools. In addition, responding to our first study goal, which sought to understand if ChatGPT could be prompted to develop a 3D 4x4x4 Tic-Tac-Toe code in its entirety, ChatGPT 4 almost succeeded. Indeed, ChatGPT developed most of the game code. However, besides the need for a human to insert the code into Unity and develop the game interface, as discussed earlier, there were instances when our team had to intervene and make adjustments to the developed code, especially in situations that required more abstract and flexible thinking to solve problems. This type of human intervention portrays the use and development of our team members' TPACK content knowledge (Mishra & Koehler, 2006). For example, ChatGPT did not correctly generate the entire code for checking victories; it failed to recognize four moves on one of the main diagonals of the cube as a victory. ChatGPT was only able to resolve this problem when specifically asked to do so. Another example was the aforementioned scoring system code for the AI player playing against a human. Initially, the AI could not check the best moves in a round. To assist in these situations, the team had to apply an understanding of computational thinking and knowledge of the C# language, which brings us to our second investigation point. Were undergraduate students improving their computational thinking knowledge and skills while developing the game with AI support?

The short answer is yes, they were. The collaboration between humans and ChatGPT clearly promoted learning gains for participants in the study. Undergraduate students from both education and game development majors markedly improved their technical skills, becoming more proficient in C# programming and more adept at using the Unity game engine. In addition, grounded on the methodology used to interact with the generative AI, students developed a foundational competency in prompt engineering – the emerging skill of crafting effective prompts to guide generative AI. This cross-disciplinary setting meant that an education major gained confidence in complex coding environments, while a game development major honed their ability to articulate and translate game design ideas into precise instructions for the generative AI. Consequently, every team member advanced their technical fluency and adaptability in working with new digital tools. In addition, team members enhanced their computational content knowledge by addressing challenges throughout the game development process, such as the “forgetfulness” and “confusion” processes. The novelty of the challenge ensured that learning was mutual and continuous: No one could rely solely on prior experience; everyone had to grow their skill sets together throughout the game development. Such

substantial gains in technological proficiency are consistent with the TPCK framework (Mishra & Koehler, 2006).

Another pivotal learning outcome of the study was the group's deepened appreciation for the educational value of errors. The frequent errors and inaccuracies in ChatGPT's outputs – ranging from minor logical mistakes to more significant misunderstandings of the game requirements – became catalysts for a deeper inquiry. Participants learned to scrutinize whether an error stemmed from a poorly formulated prompt, an ambiguous game specification, a coding error, or a limitation in the AI's current capabilities. Through this iterative process, the students became increasingly aware of how they communicated technical requirements, both to the AI and to one another. They also had to apply their computational thinking skills to analyze, discuss, and interpret the code in order to identify and correct the error. This entire process, which explores the relevance of errors and how they can be used for knowledge development and learning, is an example of how students were engaged in the TPCK's pedagogical strand (Mishra & Koehler, 2006). It is worth mentioning that our understanding aligns with Carol Dweck's (2006) perspective, in which errors serve as pedagogical stepping stones that promote reflection, resilience, and the reformulation of strategies.

Over time, these discussions normalized the presence of errors as an integral part of the learning journey and as opportunities for collective learning, enhancing the team's collaborative working abilities. These reflective dialogues enhanced group cohesion, fostered a culture that acknowledges the complexity of human-machine interaction, and improved students' critical thinking capacity. The study required constant communication, joint problem-solving, and negotiation of ideas – a teamwork-oriented pedagogical approach that reflects the development of students' pedagogical knowledge, as suggested by the TPCK framework (Mishra & Koehler, 2006). This synergy not only improved the efficiency of the game development but also fostered a supportive learning community. This approach diverges from the notion of a single correct answer. It shifts toward educational practices that value argumentation, productive struggle, and collaborative knowledge-building – all principles clearly outlined in the BNCC (Brasil, 2018), which proposes that students learn to collaboratively argue in contexts of uncertainty and technological mediation.

FINAL CONSIDERATIONS

The primary focus of this study was on developing a game using generative AI and analyzing the learning experiences that unfolded during that process. Overall, although ChatGPT primarily generated the entire code, team members needed a foundational knowledge to engage in

the game development process, and they still attested to new learnings and the development of their existing knowledge. In other words, consonant with Papert's (1980) theoretical foundations, team members could interact with the technology and build on their knowledge. This process aligns with the TPACK framework (Mishra & Koehler, 2006), as participants needed to integrate technological proficiency with pedagogical reasoning and content understanding to effectively engage with the AI and make informed decisions about its use in addressing domain-specific challenges. Much like Papert's view of computers as "objects to think with," ChatGPT has become a cognitive partner, enabling students to externalize their ideas, test hypotheses, and refine their thinking iteratively.

The experience embodied a constructivist approach rooted in a growth mindset. By engaging in hands-on creation with a generative AI agent and openly tackling the problems encountered, the participants vividly illustrated the principles of learning by doing, characterizing a growth mindset (Dweck, 2006), where challenges and mistakes serve to strengthen resilience and problem-solving strategies. It also aligns with classic constructivist learning theories (Piaget, 1976), which hold that knowledge is actively constructed through experience, including the experience of confronting and resolving problems. Noticeably, the learning outcomes of this study extend beyond technical proficiencies. The group not only successfully developed the 3D Tic-Tac-Toe game with the support of generative AI, but also internalized a powerful lesson in how productive struggle and collaboration can enhance the learning process.

It is essential to note that the approach employed in this study can be applied to other contexts, providing valuable insights into how generative AI tools can benefit the educational community. The methodology employed in this experiment enables teachers to explore the use of generative AI with their students for various purposes, promoting innovative learning and enriching classroom experiences. For instance, given a specific problem, a teacher can use ChatGPT to solve it and debate with their pupils about whether the solution is correct. If not, they can discuss both the error and how to improve the prompts to obtain a better answer. Teachers can also develop educational games with students or other digital educational tools and resources to support students throughout their learning process, including students with specific needs. Teachers can create tools tailored for individual students, promoting an accessible and inclusive classroom. Lastly, teaching approaches that use generative AI also present opportunities to foster discussions about the ethical and effective use of these technologies in learning environments. This extrapolation of our study findings to other educational contexts is also in tune with the TPACK framework (Mishra & Koehler, 2006), given the importance of

teachers not only grasping content knowledge but also knowing how to use digital tools and integrate them meaningfully into their pedagogical practices. Lastly, our study echoes Papert's (1980) view that technology in education should empower learners and teachers alike, fostering creativity, experimentation, and autonomy.

REFERENCES

BOS, Beth. Professional development for elementary teachers using TPACK. *Contemporary Issues in Technology and Teacher Education*, v. 11, n. 2, 2011. Disponível em: <https://citejournal.org/volume-11/issue-2-11/mathematics/professional-development-for-elementary-teachers-using-tpack/>. Acesso em: 29 abr. 2025

BRASIL. MINISTÉRIO DA EDUCAÇÃO. *Base Nacional Comum Curricular*. Brasília, DF: MEC, 2018. Disponível em: https://www.gov.br/mec/pt-br/escola-em-tempo-integral/BNCC_EI_EF_110518_-versaofinal.pdf. Acesso em: 29 abr. 2025.

DWECK, Carol Susan. *Mindset: The new psychology of success*. New York: Random House, 2006.

FRENCH, Fiona; LEVI, David; MACZO, Csaba; SIMONAITYTE, Aiste; TRIANTAFYLLIDIS, Stefanos; VARDA Gergo. Creative use of OpenAI in education: Case studies from game development. *Multimodal Technologies and Interaction*, v. 7, n. 8, p. 81, 2023. <https://doi.org/10.3390/mti7080081>.

GÓMEZ CANO, Carlos Alberto; COLALA TROYA, Ana Lucía. Artificial intelligence applied to teaching and learning processes. *LatIA*, v. 1, n. 2, 2023. <https://doi.org/10.62486/latia20232>.

H-EDUCATE. *ChatGPT prompt engineering course* [vídeo]. YouTube, 2020. Disponível em: <https://www.youtube.com/watch?v=mBYu5NoXBcs>. Acesso em: 8 abr. 2025.

KIM, Jinhee.; LEE, Hyunkyung.; CHO, Young Hoan. Learning design to support student-AI collaboration: Perspectives of leading teachers for AI in education. *Education and Information Technologies*, v. 27, p. 6069–6104, 2022. <https://doi.org/10.1007/s10639-021-10831-6>.

MAYEFESKY, Eric; ANENE, Francine; SIROTA, Marina. *Algorithms – Minimax*. Stanford University, 2003. Disponível em: <https://cs.stanford.edu/people/eroberts/courses/soco/projects/2003-04/intelligent-search/minimax.html>. Acesso em: 28 abr. 2025.

MISHRA, Punya.; KOEHLER, Matthew. J. Technological Pedagogical Content Knowledge: A Framework for Teacher Knowledge. *Teachers College Record*, v. 108, n. 6, p. 1017–1054, 2006. <https://doi.org/10.1111/j.1467-9620.2006.00684.x>

PAPERT, Seymour. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books, 1980.

PIAGET, Jean. *Science of Education and the Psychology of the Child*. New York: Orion Press,

1976.

RUPÉREZ PADRÓN, José Antônio; GARCÍA DÉNIZ, Manuel. Juegos de alineamiento: variantes del tres en raya. *Números: Revista de Didáctica de las Matemáticas*, n. 94, p. 87-97, 2017. Disponível em: <https://core.ac.uk/download/pdf/287746262.pdf>. Acesso em: 28 abr. 2025.

RYAN, Richard. M; DECI, Edward L. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, v. 55, n. 1, p. 68–78, 2000. Disponível em: https://selfdeterminationtheory.org/SDT/documents/2000_RyanDeci_SDT.pdf Acesso em: 29 abr. 2025

VYGOTSKY, Lev. *Mind in Society: The Development of Higher Psychological Processes*. Cambridge: Harvard University Press, 1978.

WESTERA, Wim; PRADA, Rui; MASCARENHAS, Samuel; SANTOS, Pedro A.; DIAS, João; GUIMARÃES, Manuel; GEORGIADIS, Konstantinos; NYAMSUREN, Enkhbold; BAHREINI, Kiavash; YUMAK, Zerrin; CHRISTYOWIDIASMORO, Chris; DASCALU, Mihai; GUTU-ROBU, Gabriel; RUSETI, Stefan. Artificial intelligence moving serious gaming: Presenting reusable game AI components. *Education and Information Technologies*, v. 25, p. 351–380, 2019. <https://doi.org/10.1007/s10639-019-09968-2>.