

# **POOLAND: um jogo “figital” para auxiliar no ensino e aprendizagem de Programação Orientada a Objetos**

**Kaylaine A. S. Barbosa<sup>1</sup>, Jhonatan I. dos Santos<sup>1</sup>, William S. Menezes<sup>1</sup>,  
Stephanie K. A. de Sousa<sup>1</sup>, Fábio L. S. Prudente<sup>1</sup>,  
Phillipe C. Santos<sup>1</sup>, Gilson P. Santos Júnior<sup>1</sup>**

<sup>1</sup>Instituto Federal de Sergipe(IFC)  
Lagarto – SE – Brasil

{kaylaine.barbosa080, jhonatan.santos13, william.menezes099}@academico.ifs.edu.br

{stephanie.sousa, fabio.prudente, phillipe.santos, gilson.pereira}@ifs.edu.br

**Abstract.** This article presents *POOLAND*, a digital game with a physical board (“figital”) designed to gamify the teaching and learning of Object-Oriented Programming (OOP). The prototype combined elements from the “Snakes and Ladders” game and interactive quizzes. During the Make-a-thon 2024 at the Federal Institute of Sergipe, Campus Lagarto, students and teachers provided positive feedback, highlighting its attractiveness, clarity of instructions, and ease of use. Participants found the game engaging, and many expressed an intention to use it again. Thus, *POOLAND* proved promising in engaging students in OOP learning. Future work will include content customization by teachers.

**Resumo.** Este artigo apresenta o *POOLAND*, um jogo digital com tabuleiro físico (“figital”) para gamificar o ensino e aprendizagem de Programação Orientada a Objetos (POO). O protótipo combinou elementos de jogos do “Snakes and Ladders” e de quizzes interativos. Durante o Make-a-thon 2024, no Instituto Federal de Sergipe, Campus Lagarto, estudantes e professores forneceram feedbacks positivos, destacando sua atratividade, clareza das instruções e facilidade de uso. Os participantes consideraram o jogo envolvente, e muitos expressaram a intenção de utilizá-lo. Assim, o *POOLAND* mostrou ser promissor no engajamento de estudantes no aprendizado de POO. Trabalhos futuros incluirão a personalização do conteúdo pelos professores.

## **1. Introdução**

Historicamente, o ensino de programação apresenta elevadas taxas de evasão e retenção em cursos da área das ciências exatas e da terra, especialmente em computação e engenharia [JOSÉ RICHTER et al. 2019]. Isso ocorre porque aprender a programar é uma tarefa complexa que envolve tanto a assimilação da linguagem de programação quanto o desenvolvimento da habilidade de resolver problemas computacionalmente. A segunda está diretamente relacionada ao aprimoramento do pensamento computacional, enquanto a primeira abrange o estudo de paradigmas de programação, além da sintaxe e semântica da linguagem. Apesar das dificuldades, aprender a programar favorece o desenvolvimento do raciocínio lógico, a habilidade de resolução de problemas, a capacidade de abstração e a noção de causa e efeito [GOMES 2015].

Contudo, os métodos tradicionais de ensino, pautados na reprodução de “o que aprendii”, “como aprendi” e “com os mesmos exemplos que aprendi” [SANTOS JÚNIOR 2023], na “educação bancária” criticada por [Freire 2001] e na “pedagogia da transmissão” [SILVA and CLARO 2007], têm se mostrado insuficientes para motivar e engajar os estudantes no aprendizado da programação. Além disso, dificuldades na compreensão de conceitos, no raciocínio lógico, na abstração, na decomposição de problemas, no reconhecimento de padrões e na criação de soluções algorítmicas, somadas a fragilidades nos fundamentos matemáticos e na interpretação de texto, intensificam os desafios no aprendizado de paradigmas de programação, como o Paradigma Orientado a Objetos.

A Programação Orientada a Objetos (POO) é um paradigma que representa entidades do mundo real ou conceitos abstratos por meio de objetos. Sua estrutura complexa e a necessidade de abstração tornam seu aprendizado desafiador para os estudantes. Tornar o processo de aprendizagem interativo, atrativo e concreto pode contribuir para superar essas barreiras. Para tanto, optamos pela aprendizagem baseada em jogos (ABJ) [Prensky 2010, Prensky 2012], do inglês *Game-Based Learning*.

Os jogos têm o potencial de transformar o aprendizado em uma experiência envolvente, estimulando a participação ativa dos estudantes, a reflexão sobre suas ações e o aprendizado a partir dos erros [Kapp 2012]. Essa característica é essencial no aprendizado de programação, no qual o processo de tentativa e erro auxilia na construção e evolução dos algoritmos. Além disso, os jogos promovem o desenvolvimento do raciocínio lógico, a criação de estratégias, o fortalecimento da memória e a ampliação cultural [SILVA and KIRNER 2010].

Em uma recente revisão da literatura, [de Freitas 2023, p. 40-41] constatou que a gamificação do ensino de POO no ensino superior tem explorado abordagens como a implementação de jogos pelos estudantes, o uso de elementos gamificadores no processo de ensino e o uso de jogos como ferramenta de auxílio na aprendizagem, sendo esta última a metodologia adotada por 61,5% dos 23 trabalhos analisados no estudo.

Diante desse contexto, este artigo apresenta o POOLAND, um jogo digital com tabuleiro físico (“figital”) para a gamificação do processo de ensino e aprendizagem de Programação Orientada a Objetos. O POOLAND promove uma experiência lúdica e imersiva por meio de uma metodologia híbrida, na qual o jogo digital exibe desafios de POO em formato de quiz e fornece *feedback* imediato sobre as respostas, enquanto o tabuleiro físico ilustra a jornada e o progresso dos jogadores, incentivando a interação social e o trabalho em equipe. Dessa forma, essa abordagem busca promover o engajamento dos estudantes e facilitar a compreensão dos conceitos fundamentais do paradigma, preparando-os melhor para os desafios do desenvolvimento de sistemas orientados a objetos.

## 2. Referencial Teórico

### 2.1. Programação Orientada a Objetos (POO)

A Programação Orientada a Objetos (POO) é um dos paradigmas mais utilizados no desenvolvimento de software, sendo amplamente adotado em linguagens como Java, C# e Python. Börstler et al.[2003] destacam que a POO é, e deve ser, um tópico principal na ciência da computação, pois permite um desenvolvimento modular e reutilizável.

Seus principais conceitos incluem classes e objetos, encapsulamento, herança e polimorfismo. As classes definem atributos e comportamentos, enquanto objetos são suas instâncias [Sommerville 2011]. O encapsulamento protege os dados internos do objeto [Larman 2002], a herança possibilita a reutilização de características [Gamma et al. 1995] e o polimorfismo permite flexibilidade na implementação de métodos [Stroustrup 1997].

Apesar de sua importância, aprender a programar orientado a objetos é um desafio. A abstração dos conceitos e a abordagem tradicional de ensino dificultam a assimilação, elevando os índices de reprovação [COSTA et al. 2017]. Ademais, é comum que estudantes apresentem dificuldades em relacionar teoria e prática, o que impacta sua motivação e aprendizado [BÖRSTLER et al. 2003].

Ao longo dos anos, algumas abordagens metodológicas vêm sendo exploradas para tornar o ensino de POO mais acessível e engajador. Dentre essas abordagens, destacam-se a gamificação e a aprendizagem baseada em jogos, que têm-se mostrado promissoras.

## **2.2. Gamificação e ABJ no Processo de Ensino e Aprendizagem de POO**

Gamificação e aprendizagem baseada em jogos são conceitos relacionados, porém distintos. Enquanto a gamificação utiliza elementos de jogos digitais em contextos não lúdicos para engajar e motivar os participantes, a ABJ refere-se à aplicação de jogos com objetivos educativos. Ambas têm mostrado resultados promissores em disciplinas de programação.

A gamificação não consiste em criar um jogo, mas em aplicar mecânicas, dinâmicas e estéticas de jogos para engajar as pessoas a resolver problemas [Kapp 2012]. Tal abordagem tem ganhado espaço no ensino de programação, pois o aprendizado ocorre com a prática, durante o processo de tentativa e erro, essencial na criação de algoritmos. Já a ABJ é uma abordagem na qual o uso de jogos digitais é o meio de ensino [Tang et al. 2009].

De acordo com [de Freitas 2023], o processo de gamificação no ensino e aprendizagem de POO ainda é incipiente e explora a tríade PBL (Pontos (Points), emblemas (Badges) e ranking (Leaderboards)), além de desafios, recompensas, narrativas e competições. Por outro lado, o uso de jogos como recurso didático teve um notável avanço, envolvendo jogos de interpretação de papéis (RPG), jogos de tabuleiro e jogos digitais. Essa diversificação de abordagens realça a busca por meios para aperfeiçoar o processo de ensino e aprendizagem em POO.

Dentre os jogos apresentados na Tabela 1, “The Legend of Zelba” e “Monotony” foram os únicos jogos classificados como jogos de tabuleiro por [de Freitas 2023] no estudo. Esses jogos foram utilizados por [Jr and Bennett 2006] para ensinar polimorfismo, um conceito essencial na programação orientada a objetos.

## **3. POOLAND**

O POOLAND é um jogo digital com tabuleiro físico (“figital”) desenvolvido para gamificar o processo de ensino e aprendizagem de POO. O protótipo foi criado durante o Make-a-thon 2024 no Instituto Federal de Sergipe (IFS), Campus Lagarto, de 01 a 08 de novembro, um evento organizado pelo Laboratório de Inovação e Criatividade (LaBIC) em parceria com a Pró-Reitoria de Extensão e Pesquisa (Propex) do IFS, que reuniu alunos em um ambiente de inovação e colaboração.

**Tabela 1. Ferramentas e jogos no processo de ensino e aprendizagem de POO.**  
**Adaptado de [de Freitas 2023].**

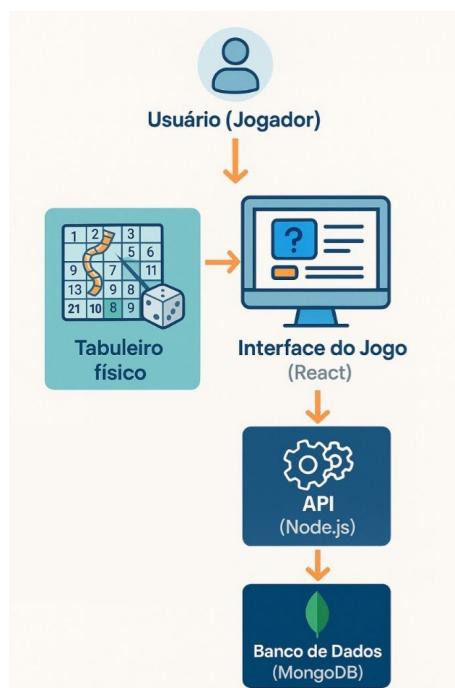
Classificação	Nome
Ferramentas	Alice 2D, Greenfoot, Darwin, CodeCombat, RoboCode, Jeroo, Scratch, MinimUML, GameMaker, PlayLogo3D, Code Spells, Dream Coders, GameMaker e IDE BlueJ
Jogos	jAVANT-GARDE, JavabotWars, The Game of Code: Lost in JavaLand, Extension of the Microworld Object Karel, “Rise of the Java Emperor”, Amiga, Karel the Robot, E-Adventure, Nice-Game, E-TrainingDS, Approach to Support Programming Skills (GAPS 1.0), Minecraft, “The Legend of Zelba”, “Monotony” e “The Odyssey of Phoenix”

Durante o evento, foram oferecidos diversos mini-cursos para capacitar os participantes, seguido por uma competição de projetos em grupos de quatro integrantes. Os participantes tiveram duas semanas para desenvolver protótipos físicos alinhados aos Objetivos de Desenvolvimento Sustentável (ODS), tema central do evento. Essa iniciativa permitiu que os estudantes colcassem em prática seus conhecimentos, trabalhando em equipe para criar soluções criativas e sustentáveis, promovendo a aplicação da teoria na resolução de problemas reais. O protótipo criado combinou mecânicas, dinâmicas e estéticas inspiradas no “*Snakes and Ladders*” e em jogos de quiz, como QuizUp e Kahoot, visando a Educação de Qualidade (ODS 4) e a Indústria, Inovação e Infraestrutura (ODS 9).

O “*Snakes and Ladders*” é um jogo de tabuleiro originado na Índia antiga, no século II d.C. Esse jogo popular consiste em uma corrida baseada na sorte, na qual os jogadores percorrem o tabuleiro do início ao fim, utilizando escadas para avanços rápidos e enfrentando penalidades representadas por cobras. O número de casas percorridas por cada jogador em uma rodada é determinado pelo valor sorteado no lançamento do dado.

Jogos de quiz têm o objetivo de testar o conhecimento dos jogadores de maneira interativa e competitiva, apresentando uma sequência de perguntas e atribuindo pontuações com base na precisão e rapidez das respostas. Esses jogos desafiam a agilidade e o raciocínio dos participantes, que devem responder às perguntas dentro de um tempo determinado ou em turnos.

Ao combinar elementos do “*Snakes and Ladders*” e de jogos de quiz, foi desenvolvido um jogo híbrido, que possibilita até quatro jogadores simultâneos, cujo vencedor é o jogador que alcança a última casa primeiro (casa 50). O jogo é realizado em turnos, nos quais cada participante visualiza uma pergunta de múltipla escolha sobre os pilares da POO – abstração, herança, polimorfismo e encapsulamento – no jogo digital. Ao verbalizar a resposta, em caso de acertos, o jogador avança no tabuleiro; em caso de erro, o jogador tem seu movimento bloqueado ou retrocede no tabuleiro. A mecânica de movimento é mediada por um dado digital no jogo, adicionando, portanto, um elemento de aleatoriedade às jogadas. A Figura 1 apresenta a arquitetura do sistema, evidenciando os principais componentes envolvidos no funcionamento do jogo e sua interação, enquanto na Figura 2 está ilustrado o tabuleiro do POOLAND.



**Figura 1. Diagrama de Arquitetura do POOLAND**

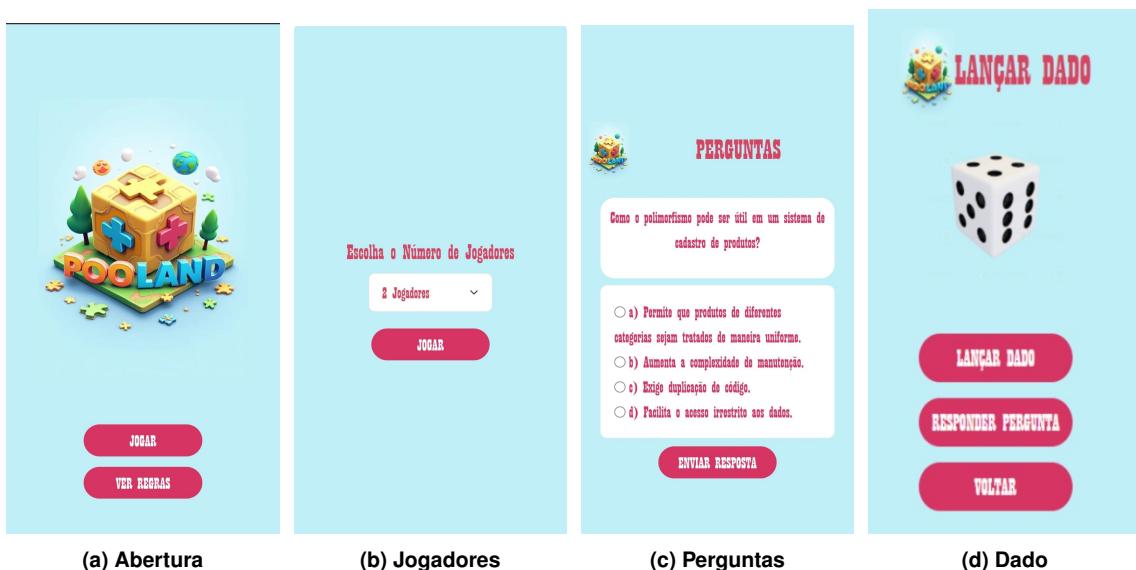
Para além da abordagem teórica, as perguntas foram elaboradas com base em situações comuns do cotidiano da programação, como a necessidade de reutilização de código por meio de herança, a definição de classes genéricas utilizando abstração, o controle de acesso com encapsulamento e a sobreescrita de métodos em diferentes classes para simular comportamentos distintos por meio do polimorfismo. Dessa forma, busca-se promover nos jogadores a reflexão sobre como os conceitos da orientação a objetos se aplicam em decisões práticas de modelagem e desenvolvimento, estimulando o raciocínio além da memorização. Assim, o mecanismo de perguntas contribui diretamente para que os estudantes consigam mapear os princípios da POO para estruturas concretas de código, superando uma das principais dificuldades encontradas no processo de ensino-aprendizagem da programação orientada a objetos.



**Figura 2. Tabuleiro físico do POOLAND**

Conforme ilustrado na Figura 2, o tabuleiro é uma matriz 5x10, totalizando 50 casas numeradas de 1 a 50, da esquerda para a direita e de baixo para cima. Os jogadores se movimentam no tabuleiro de acordo com o número obtido no dado digital e com as respostas ao quiz. Os elementos especiais de movimentação no tabuleiro incluem: (i) escadas, que permitem que o jogador avançar casas, incentivando acertos consecutivos; (ii) cobras, que fazem o jogador retroceder casas, promovendo maior desafio; e estrelas, que protegem o jogador de regredir ao ser ultrapassado por outro.

É importante destacar a convergência entre as tecnologias digital (jogo digital) e física (tabuleiro) no POOLAND. Enquanto o tabuleiro físico promove a experiência lúdica, oferecendo os elementos visuais e tátteis que tornam o jogo mais atrativo, o jogo digital gerencia as perguntas e respostas, o dado digital e as interações entre os jogadores. Para tanto, o jogo digital apresenta as telas de: abertura (Figura 3a), escolha de jogadores (Figura 3b), perguntas (Figura 3c) e dado (Figura 3d).



**Figura 3. Telas do POOLAND**

Conforme ilustrado na Figura 3, a tela de abertura (Figura 3a) é carregada para iniciar o jogo e visualizar suas regras. Ao clicar em "jogar", o jogo digital solicita a escolha do número de jogadores e a informação de seus nomes na tela de jogadores (Figura 3b). Em seguida, a tela do dado é exibida para rodá-lo e movimentar o jogador (Figura 3d). Por fim, a tela de perguntas (Figura 3c) é exibida com a pergunta e as opções de resposta para os estudantes, fornecendo feedback imediato sobre o desempenho ao final.

A infraestrutura tecnológica do jogo digital foi projetada para garantir flexibilidade, escalabilidade e eficiência. O banco de dados utilizado foi o MongoDB, um banco NoSQL escolhido por sua capacidade de armazenar dados de forma flexível e facilitar a prototipagem rápida. As questões do jogo são armazenadas como documentos JSON dentro da coleção *questions*, permitindo futuras expansões e modificações com facilidade.

No backend, foi utilizado Node.js para o desenvolvimento da API, aproveitando sua escalabilidade, eficiência e suporte a aplicações em tempo real. Já o frontend foi desenvolvido com React.js, garantindo um desenvolvimento modular e dinâmico. Além

disso, o Bootstrap foi incorporado para assegurar responsividade e padronização do design, proporcionando uma melhor experiência para os usuários.

Por fim, é relevante destacar que a aleatoriedade incluída com o dado digital e as escadas e cobras espalhadas no tabuleiro modificam o fluxo linear do jogo, tornando-o atrativo e desafiador para os estudantes. Ademais, dado seu propósito, o POOLAND pode ser classificado como um “jogo sério” (*Serious Game*), pois os participantes confirmam os conceitos aprendidos ao acertarem as respostas, além de aprenderem novos ou corrigirem definições quando erram suas respostas. Assim, é um jogo que promove o ensino e aprendizado de POO, em vez de apenas entreter os jogadores.

#### **4. Conclusão**

O presente trabalho apresentou o POOLAND, um jogo digital e um tabuleiro físico para o processo de ensino e aprendizagem de Programação Orientada a Objetos, denominado (*POOLAND*). Este protótipo foi apresentado à comunidade do IFS, Campus Lagarto, durante o Make-a-thon, momento em que foram recebidos *feedbacks* de estudantes e professores, inclusive de outras áreas de conhecimento. Os participantes destacaram a atratividade da abordagem “figital”, a partir da combinação entre jogo digital e um tabuleiro físico, tornando o aprendizado de Programação Orientada a Objetos mais dinâmico e acessível. A recepção do jogo foi amplamente positiva, com os participantes destacando a clareza das instruções e a facilidade de uso. Muitos consideraram a experiência envolvente e indicaram que utilizariam o jogo novamente ou o recomendariam para outros estudantes.

Os comentários dos participantes demonstraram o potencial do *POOLAND* para promover o engajamento e facilitar a assimilação dos conceitos de POO em um ambiente descontraído e interativo. Além disso, o jogo mostrou-se eficaz na motivação dos estudantes, proporcionando um ambiente de aprendizagem interativo e seguro, o que é fundamental para o desenvolvimento de habilidades cognitivas, como a resolução de problemas e o pensamento crítico.

Como trabalhos futuros da pesquisa, será construída a tela de cadastro de questões para permitir aos professores a inserção e o gerenciamento de questões diretamente pela interface. Ademais, serão realizados estudos de avaliação qualitativa e quantitativa do protótipo quanto à aceitação dos estudantes e professores, e à contribuição no aprendizado dos conceitos de POO.

O POOLAND foi apresentado a comunidade do IFS, Campus Lagarto, durante o Make-a-thon, momento em que foram recebidos *feedbacks* de estudantes e professores, inclusive de outras áreas de conhecimento. Os participantes destacaram a atratividade da abordagem “figital”, a partir da combinação entre jogo digital e um tabuleiro físico, tornando o aprendizado de Programação Orientada a Objetos mais dinâmico e acessível. A recepção do jogo foi amplamente positiva, com os participantes destacando a clareza das instruções e a facilidade de uso. Muitos consideraram a experiência envolvente e indicaram que utilizariam o jogo novamente ou o recomendariam para outros estudantes. Esses retornos demonstram o potencial do *POOLAND* para promover o engajamento e facilitar a assimilação dos conceitos de POO em um ambiente descontraído e interativo. Além disso, o jogo mostrou-se eficaz na motivação dos estudantes, proporcionando um ambiente de aprendizagem interativo e seguro, o que é fundamental para o desenvolvimento

de habilidades cognitivas como a resolução de problemas e o pensamento crítico.

Como trabalhos futuros da pesquisa, será construída a tela de cadastro de questões para permitir aos professores a inserção e gerenciamento de questões diretamente pela interface. Ademais, será realizado estudos de avaliação qualitativa e quantitativa do protótipo quanto a aceitação dos estudantes e professores e da contribuição no aprendizado dos conceitos de POO.

## Referências

- BÖRSTLER, J., BRUCE, K., and MICHIELS, I. (2003). Sixth workshop on pedagogies and tools for learning object oriented concepts.
- COSTA, A. C., SILVA, J. R., and SOUZA, M. A. (2017). Uma abordagem contextualizada para o ensino de programação orientada a objetos no ensino médio profissionalizante. In *Anais do Workshop de Computação e Educação (WCBIE)*.
- de Freitas, T. T. (2023). O uso da gamificação como estratégia para o ensino de programação orientada a objetos: uma revisão sistemática da literatura.
- Freire, P. (2001). *Pedagogia da autonomia: saberes necessários à prática educativa*. Paz e Terra, Rio de Janeiro, 18 edition.
- Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, Boston.
- GOMES, M. C. P. (2015). Os benefícios do ensino de linguagem de programação no currículo regular. *Administradores.com*.
- JOSÉ RICHTER, C., BERNARDI, G., and ZANKI CORDENONSI, A. (2019). O ensino de programação mediado por tecnologias educacionais: uma revisão sistemática de literatura. *Revista Novas Tecnologias na Educação*, 17(1):517–526.
- Jr, T. S. P. and Bennett, C. (2006). A framework for teaching polymorphism using game programming. *Journal of Computing Sciences in Colleges*, 22(2):154–161.
- Kapp, K. (2012). *The gamification of learning and instruction: Game-based methods and strategies for training and education*. Pfeiffer, San Francisco, CA.
- Larman, C. (2002). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Prentice Hall, Upper Saddle River, NJ, 3rd edition.
- Prensky, M. (2010). *Não me atrapalhe, mãe - Eu estou aprendendo! Como os videogames estão preparando nossos filhos para o sucesso no século XXI - e como você pode ajudar!* Phorte, São Paulo.
- Prensky, M. (2012). *Aprendizagem baseada em jogos digitais*. Editora Senac São Paulo, São Paulo.
- SANTOS JÚNIOR, G. P. d. (2023). *Pensamento computacional com a educação on-line no ensino médio integrado*. Tese (doutorado em educação), Universidade Federal de Sergipe, São Cristóvão.
- SILVA, K. A. C. and KIRNER, C. (2010). Vantagens educacionais no uso de jogos em realidade aumentada. *Revista Novas Tecnologias na Educação*, 8(3). Acesso em: 19 nov. 2024.

- SILVA, M. and CLARO, T. (2007). A docência online e a pedagogia da transmissão. *Boletim Técnico do Senac*, 33(2):81–89. Acesso em: 30 out. 2021.
- Sommerville, I. (2011). *Software Engineering*. Addison-Wesley, Boston, 9th edition.
- Stroustrup, B. (1997). *The C++ Programming Language*. Addison-Wesley, Reading, MA, 3rd edition.
- Tang, S., Hanneghan, M., and El Rhalibi, A. (2009). Introduction to game-based learning. In *Games Based Learning Advancements for Multi-Sensory Human Computer Interfaces*, pages 1–17. IGI Global, New York.