Revista Poíesis Pedagógica

Ano 2025 – Jan./Dez.
Volume 23, e2025011
Páginas 1–22
ISSN 2178–4442
doi 10.69532/2178-4442.v23.75010

Artigo Original

Pensamento Computacional na Educação Básica: o *Game Design* Document (GDD) como proposta pedagógica

Computational Thinking in Basic Education: the Game Design Document (GDD) as a pedagogical proposal

Pensamiento Computacional en la Educación Básica: el Documento de Diseño de Juegos (GDJ) como propuesta pedagógica

Douglas Aquino Moreno
Centro Universitário Luterano de Palmas

Gustavo Cunha de Araujo
Universidade Federal do Norte do Tocantins

Resumo

Esta pesquisa é um recorte de um estudo mais amplo desenvolvido no Mestrado Profissional em Educação, que teve como principal objetivo desenvolver o *Game Design Document* (GDD) de um jogo para dispositivos móveis, visando auxiliar o ensino por parte dos professores e a aprendizagem do Pensamento Computacional por estudantes da Educação Básica. Baseia-se em uma abordagem qualitativa do tipo interpretativa. Como estratégia metodológica, utilizou-se a revisão bibliográfica. Dentre alguns resultados, identificou-se que o processo de elaboração do GDD pode se tornar para a Educação Básica um recurso pedagógico inovador que pode ser utilizado pelos professores para auxiliar na introdução do Pensamento Computacional em sala de aula. Além disso, o GDD oferece diretrizes claras para o desenvolvimento de jogos que não apenas entretêm, mas também educam, contribuindo para a formação dos alunos e para a incorporação das habilidades exigidas pela Base Nacional Comum Curricular (BNCC).
Palavras-chave: Pensamento Computacional. Tecnologias Digitais. Educação Básica.

Abstract

This research is an excerpt from a broader study carried out as part of the Professional Master's Degree in Education, the main objective of which was to develop the Game Design Document GDD of a game for mobile devices, with the aim of helping teachers to teach and basic education students to learn Computational Thinking. It is based on a qualitative, interpretive approach. A literature review was used as a methodological strategy. Among some of the results, it was identified that the process of developing the GDD can become an innovative pedagogical resource for Basic Education that can be used by teachers to help introduce Computational Thinking in the classroom. In addition, the GDD offers clear guidelines for the development of games that not only entertain, but also educate, contributing to students' education and the incorporation of the skills required by the National Common Curricular Base (BNCC).
Keywords: Computational Thinking. Digital Technologies. Basic Education.

Resumen

Esta investigación es un extracto de un estudio más amplio realizado en el marco del Máster Profesional en Educación, cuyo objetivo principal era desarrollar el *Game Design Document* (GDD) de un juego para dispositivos móviles, con el fin de ayudar a los profesores a enseñar y a los alumnos de primaria a aprender Pensamiento Computacional. Se basa en un enfoque cualitativo interpretativo. Como estrategia metodológica se utilizó la revisión bibliográfica. Entre algunos de los resultados, se identificó que el proceso de creación de un DDJ puede convertirse en un recurso pedagógico innovador para la educación primaria que puede ser utilizado por los profesores para ayudar a introducir el Pensamiento Computacional en el aula. Además, la DDJ ofrece pautas claras para el desarrollo de juegos que no sólo entretengan, sino que también eduquen, contribuyendo a la formación de los estudiantes y a la incorporación de las competencias requeridas por el Base Nacional Común Curricular (BNCC).
Palabras clave: Pensamiento Computacional. Tecnologías Digitales. Educación Básica.

CC BY

OPEN ACCESS

1. INTRODUÇÃO

O Pensamento Computacional é mencionado na Base Nacional Comum Curricular (BNCC) como uma habilidade transversal a ser trabalhada com estudantes da Educação Básica (Brasil, 2018), com o objetivo de proporcionar novas competências sobre o uso consciente e ético da tecnologia (Barcelos & Silveira, 2013). Nesse contexto, as atividades lúdicas, especialmente com abordagem em jogos, podem se constituir em importantes recursos metodológicos no processo de ensino e aprendizagem, pois não se limitam ao entretenimento, mas funcionam como ferramentas educacionais que facilitam o desenvolvimento de novas habilidades (Kiya, 2014; Resende, 2013). Além disso, essas atividades podem auxiliar pedagogicamente em diferentes disciplinas, tanto na Educação Básica quanto no Ensino Superior (Barbosa, 2014).

No entanto, para que esse processo seja dinâmico e produtivo, é fundamental planejar, descrever e registrar as ações do mesmo. Para isso, uma das formas utilizadas para orientar na construção de um jogo é o *Game Design Document* (GDD). Conforme Rogers (2013), o GDD é um documento que descreve de forma precisa o que o jogo deve conter, como informações técnicas, detalhes sobre os personagens e outras informações necessárias para o jogo, sendo relevante ferramenta para facilitar a comunicação entre os membros da equipe de desenvolvimento.

Esta pesquisa é um recorte de um estudo mais amplo desenvolvido no Mestrado Profissional em Educação, que teve como principal objetivo desenvolver o GDD de um jogo para dispositivos móveis, visando auxiliar o ensino por parte dos professores e a aprendizagem do Pensamento Computacional por estudantes da Educação Básica. Neste artigo, devido a sua extensão, socializamos de forma descritiva e analítica, partes deste jogo criado. É importante salientar que o objetivo principal deste GDD foi subsidiar o desenvolvimento de um jogo educativo voltado ao fomento do Pensamento Computacional entre estudantes da Educação Básica.

É importante destacar que a proposta central desta pesquisa não se limita à utilização de jogos digitais como ferramenta educativa, mas enfatiza a construção do GDD como uma estratégia pedagógica ativa. A elaboração do GDD, quando realizada com a mediação dos professores e a participação dos estudantes, constitui um processo capaz de fomentar o Pensamento Computacional ao envolver habilidades como decomposição, abstração, reconhecimento de padrões e criação de algoritmos.

2. METODOLOGIA

Neste artigo, a presente pesquisa baseia-se em uma abordagem qualitativa do tipo interpretativa (Erickson, 1989). Como estratégia metodológica, utilizou a revisão bibliográfica (Pilão, 2009), com o objetivo de analisar o Pensamento Computacional e o uso de jogos educacionais para seu desenvolvimento, principalmente no que concerne à área educacional. Conforme Strauss e Corbin (2008, p. 23), uma pesquisa de abordagem qualitativa é “qualquer tipo de pesquisa que produza resultados não alcançados através de procedimentos estatísticos ou de outros meios de quantificação”.

Dito com outras palavras, a abordagem qualitativa enfatiza os aspectos e ações da realidade estudada, destacando a importância das interações entre pesquisadores e participantes para desenvolver as pesquisas. Com efeito, ao explorar as dinâmicas sociais e as percepções subjetivas,

esse quadro metodológico, assumido neste artigo, pode proporcionar uma compreensão mais profunda dos fenômenos estudados (Denzin & Lincoln, 2018).

O jogo *Logic Think* foi desenvolvido até o estágio de prototipagem como parte de um projeto de dissertação de mestrado, com o objetivo de explorar como jogos educacionais podem ser utilizados para promover o Pensamento Computacional no ensino básico. No entanto, neste artigo, é apresentado exclusivamente o GDD, que descreve a estrutura e os elementos do jogo, sem abordar dados empíricos de sua aplicação com estudantes. O desenvolvimento do GDD foi realizado utilizando a plataforma *Unity*, que permite a criação de *scripts* com as linguagens *JavaScript* e *C#*.

No que diz respeito ao *design* do jogo e seus componentes gráficos que fazem parte do GDD, foram utilizados o *Photoshop* e o *Aseprite*. O processo de elaboração do GDD incluiu a criação de um roteiro detalhado com informações sobre cenários, diálogos e outros elementos essenciais para o desenvolvimento do jogo. O GDD, portanto, consolidou a ideia central do *Logic Think* e seu *design*, servindo como o principal produto do trabalho até o momento.

3. O PENSAMENTO COMPUTACIONAL NA EDUCAÇÃO BÁSICA

A BNCC desempenha um papel fundamental como instrumento de gestão pedagógica, estabelecendo diretrizes para o desenvolvimento de competências essenciais que todos os estudantes devem adquirir ao longo de sua trajetória educacional (Brasil, 2018). Com o objetivo de promover uma educação de qualidade e garantir a equidade, a BNCC orienta a formulação de políticas pedagógicas nas escolas brasileiras, fornecendo um referencial comum para a construção dos currículos (Brasil, 2018).

De acordo com Vicari *et al*. (2018), a BNCC propõe as "Dez Competências Gerais", que visam uma formação integral dos estudantes, abordando aspectos cognitivos e socioemocionais. Essas competências buscam desenvolver habilidades como curiosidade intelectual, uso crítico das tecnologias digitais e valorização da diversidade. Vicari *et al*. (2018) destacam competências alinhadas ao Pensamento Computacional, como o uso de conhecimentos sobre o mundo físico, social e digital para compreender a realidade e colaborar para uma sociedade justa e democrática; a utilização de diferentes linguagens para expressar ideias; e o uso ético de tecnologias digitais, promovendo o protagonismo na vida pessoal e coletiva.

A inserção do Pensamento Computacional na Educação Básica pode auxiliar os alunos a enfrentar desafios complexos. Essa abordagem permite que os estudantes analisem problemas de forma estruturada, decompondo tarefas em partes menores e identificando padrões e abstrações relevantes. Além disso, eles têm a oportunidade de experimentar soluções e aprimorá-las por meio da utilização de algoritmos e iterações.

De acordo com a Sociedade Brasileira de Computação (SBC, 2017), a incorporação da computação na educação básica proporciona uma oportunidade para o desenvolvimento de habilidades e competências computacionais, fortalecendo as ciências e suas diversas áreas de conhecimento. Essas habilidades e competências são essenciais para o Pensamento Computacional, que busca a resolução de problemas e a criação de processos mais eficientes e objetivos.

Com base nos referenciais da Sociedade Brasileira de Computação (SBC, 2017), é possível identificar um conjunto de habilidades específicas relacionadas ao Pensamento Computacional que podem ser desenvolvidas progressivamente ao longo das etapas da Educação Básica. Essas habilidades estão alinhadas aos pilares do PC — decomposição, reconhecimento de padrões, abstração e algoritmos — e são adaptadas às capacidades cognitivas e pedagógicas de cada fase escolar. Para tornar essas informações compreensíveis e acessíveis, a seguir apresenta-se um quadro com a síntese dessas competências, organizadas conforme o nível de ensino:

Quadro 1 – Habilidades do Pensamento Computacional na Educação Básica

Etapa de Ensino	Habilidades Relacionadas ao Pensamento Computacional
Educação Infantil	• Compreender uma determinada situação e identificar sequências de padrões e passos para a resolução de problemas; • Representar passos simples de uma determinada tarefa através da representação gráfica de desenhos organizados; • Desenvolver estratégias para solucionar problemas de movimento corporal e trajetórias espaciais.
Ensino Fundamental: Anos Iniciais	• Utilizar representações concretas para abstrair problemas e experiências; • Identificar abstrações-chave para criar processos envolvendo escolhas, composições, repetições e simulações, definindo passos simples que representam situações cotidianas dos alunos; • Utilizar recursos lúdicos visuais para representar passos estruturados; • Entender técnicas para decompor problemas e formas de como solucioná-los.
Ensino Fundamental: Anos Finais	• Utilizar recursos visuais e linguagem nativa para representar processos; • Formalizar conceitos de algoritmos estruturados de forma clara e precisa; • Empregar conceitos de recursão para aprofundar o conhecimento de técnicas de decomposição de problemas; • Elaborar soluções de problemas reconhecendo padrões e aplicando técnicas de generalização para permitir o reuso dessas soluções em diferentes contextos. • Estabelecer uma relação entre algoritmos descritos em linguagens visuais e sua representação em linguagens de programação.
Ensino Médio	• Desenvolver projetos interdisciplinares em equipe, solucionando problemas por meio do uso de recursos tecnológicos; • Compreender técnicas de solução de problemas através do reconhecimento de padrões; • Analisar algoritmos para identificar custos e eficiência, a fim de justificar soluções adequadas aos requisitos e tomar decisões entre diferentes abordagens; • Argumentar sobre correções de algoritmos, para permitir que uma solução possa realmente resolver um problema proposto; • Reconhecer o conceito de programação como uma forma de generalizar um problema; • Entender os limites que a Computação possui para diferenciar o que pode ou não ser automatizado, buscando uma compreensão mais ampla dos processos envolvidos.

Fonte: adaptado de SBC (2017).

As habilidades destacadas pela SBC (2017) indicam que o Pensamento Computacional abrange uma variedade de conhecimentos que podem contribuir para a formação dos estudantes, especialmente ao oferecer ferramentas conceituais úteis para enfrentar desafios de forma estruturada.

Embora a presença dessas habilidades no currículo não seja, por si só, uma garantia de resultados educacionais, elas oferecem um ponto de partida relevante para práticas pedagógicas voltadas à resolução de problemas. Assim, a introdução do Pensamento Computacional pode representar uma oportunidade pedagógica significativa, desde que articulada a estratégias didáticas bem planejadas e contextos educacionais adequados.

De acordo com Vicari *et al*. (2018), o Brasil é um país caracterizado por diferenças regionais. Nesse contexto, argumenta-se que o ensino do Pensamento Computacional pode desempenhar um papel crucial no aprendizado dos alunos, considerando suas necessidades e diferenças individuais. Além disso, é necessário levar em conta fatores como a disponibilidade de energia elétrica, a presença ou ausência de conexão com a internet, a existência de equipamentos eletrônicos e, principalmente, a disponibilidade de professores para trabalhar com conceitos e tecnologias relacionadas.

É importante destacar que o ensino do Pensamento Computacional não se limita ao uso de dispositivos eletrônicos. Ele envolve o desenvolvimento de habilidades cognitivas, como raciocínio lógico, resolução de problemas e criatividade, que podem ser exploradas de diversas maneiras, mesmo em ambientes com recursos limitados. Portanto, é essencial buscar alternativas e adaptar o ensino do Pensamento Computacional às realidades regionais, promovendo uma educação inclusiva e de qualidade para todos os estudantes (Vicari *et al*., 2018; SBC, 2017).

3.1. Jogos no Auxílio do Pensamento Computacional

No âmbito do ensino básico, são amplas as estratégias disponíveis para facilitar a assimilação do Pensamento Computacional. Entre elas, destaca-se a incorporação de jogos com mecânicas que fomentam o engajamento dos alunos e sustentam a concentração deles nas atividades pedagógicas propostas. O Pensamento Computacional, caracterizado por habilidades analíticas e sistêmicas, encontra terreno fértil nos ambientes dos jogos, onde os alunos são instigados a decompor problemas complexos em etapas gerenciáveis e a desenvolver estratégias para alcançar objetivos.

Conforme observado por Oliveira (2020), a aplicação de jogos tem a capacidade de induzir uma sensação de imersão no próprio ambiente do jogo, resultando em uma permanência prolongada por parte do jogador. Através dessa abordagem, é possível explorar e fortalecer os pilares fundamentais do Pensamento Computacional, contribuindo significativamente para a sua compreensão e desenvolvimento. A natureza dos jogos também contribui para o engajamento dos alunos nas atividades didáticas. A interatividade e a imersão proporcionadas pelos jogos despertam a curiosidade dos alunos e estimulam a exploração ativa.

De acordo com a pesquisa de Magno *et al*. (2018), a integração de jogos como ferramenta de apoio ao Pensamento Computacional demonstra a capacidade de promover uma aprendizagem motivadora, amplificando assim o desenvolvimento do raciocínio lógico. Os jogos têm a capacidade de integrar-se em diversas esferas do conhecimento. Por conseguinte, é possível empregá-los como ferramentas para fomentar a alfabetização, estimular o engajamento e enriquecer uma ampla gama de atividades educacionais (Magno, *et al*., 2018).

Segundo Vicari *et al*. (2018), os principais desafios na utilização de jogos digitais como recurso educacional são: a resistência dos professores a novas metodologias de ensino, a falta de

formação específica em como utilizar jogos digitais como ferramentas pedagógicas, a dificuldade em encontrar jogos capazes de engajar verdadeiramente os alunos e que explorem todo o potencial pedagógico dessas ferramentas, o *design* limitado de muitos jogos disponíveis, a infraestrutura escolar inadequada e, principalmente, a dificuldade de alinhar os jogos com os objetivos de aprendizagem e os conteúdos curriculares.

Na Figura 3, são apresentadas duas opções tecnológicas que desempenham um papel de auxiliar na compreensão de um dos pilares fundamentais do Pensamento Computacional: os algoritmos. Estes jogos supracitados, como o *Hour of Code* e o *The Foos*, adotam tal enfoque visando conferir ao processo de aprendizado da programação um caráter mais imersivo e atrativo.

Figura 3 - Jogos para ensino de Pensamento Computacional: Hora do Código (à esquerda) e *The Foos* (à direita)



Fonte: www.hourofcode.com e www.thefoos.com.

A Figura 3 apresenta dois jogos educativos, o "*Hour of Code*" (à esquerda) e o "*The Foos*" (à direita), que representam uma abordagem inovadora no campo do ensino da programação. Ao adotarem uma linguagem de programação visual, na qual os comandos são simbolicamente representados por blocos que podem ser manipulados por meio de arrastar e soltar, essas plataformas visam tornar o aprendizado da programação mais acessível e intuitivo, especialmente para públicos novatos, incluindo crianças.

As ferramentas *Hour of Code* e *The Foos*, usam uma abordagem visual para tornar o aprendizado da programação mais interativo e divertido. Aqui está um pouco mais de informação sobre cada um deles:

- ***Hour of Code***: iniciativa que visa introduzir pessoas de todas as idades ao mundo da programação por meio de atividades interativas. Eles oferecem tutoriais e jogos que ajudam os participantes a entender conceitos básicos de programação. A abordagem visual torna mais fácil para os iniciantes entenderem os fundamentos da codificação (Code.Org, 2023).
- ***The Foos***: jogo educativo que ensina programação para crianças. Ele permite que os jogadores resolvam quebra-cabeças usando uma interface de arrastar e soltar, onde os blocos de comando são organizados para criar programas que controlam os personagens no jogo. Isso ajuda as crianças a compreenderem sequências lógicas e estruturas de programação (Code.Spark, 2021).

Através de abordagens lúdicas e interativas, é possível desmistificar a complexidade inicialmente intimidadora do Pensamento Computacional, oferecendo aos indivíduos, mesmo sem experiência prévia, a oportunidade de compreender e aplicar os conceitos básicos de maneira par-

ticipativa. O Pensamento Computacional pode ser trabalhado por meio de diversas metodologias, e os jogos digitais surgem como uma opção promissora, conforme demonstrado. Na próxima seção, será apresentada a construção do *Game Design Document*.

4. CONCEITO DE *GAME DESIGNER DOCUMENT* (GDD)

4.1. Descrição e análise do *Game Design Document*

Game Design é o processo de construção das ideias centrais e da essência do jogo. Para Schell (2010), ao criar um jogo, os *designers* precisam refletir sobre a tomada de decisões, o tipo de experiência desejada, os elementos-chave que manterão os jogadores engajados durante o jogo e os componentes que poderão fazer parte dele.

Autores como Schuytema (2008) descrevem o *Game Design* como a etapa de elaboração de uma planta baixa, onde o *designer* é responsável por criar a combinação de técnicas, dinâmicas, componentes e esforços a partir dessa planta. O *Game Design* é um plano de elaboração, uma forma de rascunhar ideias que os desenvolvedores e *designers* podem explorar, esboçando mecânicas, dinâmicas e elementos que constituem um jogo.

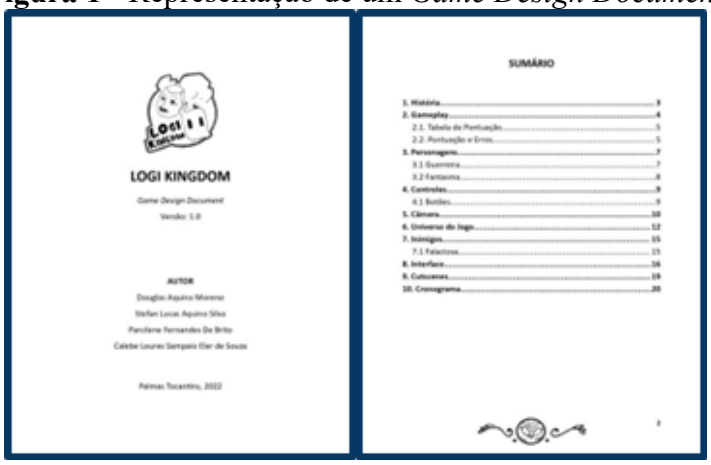
Durante o processo de elaboração do *Game Design*, é criado um documento de orientação conhecido como *Game Design Document* (GDD). O GDD é um documento guia que contém todas as informações necessárias para a elaboração e desenvolvimento do jogo, ou seja, é um documento que descreve com detalhes os aspectos técnicos e artísticos do jogo a ser desenvolvido (Leite; Mendonça, 2013; Perucia *et al*., 2005).

De acordo com Moreira e Castro (2016), o GDD é um registro que engloba as características que compõem um projeto de jogo, tanto do ponto de vista conceitual quanto tecnológico. O documento descreve as etapas necessárias para a criação de um jogo e pode ser apresentado de diversas formas, sem seguir um padrão pré-estabelecido. Não existe um modelo de GDD padronizado para a criação de jogos, e cada equipe pode adotar sua própria metodologia de desenvolvimento, sem depender de modelos preestabelecidos.

Schell (2010) apresenta duas finalidades principais do GDD. Primeiro, como uma ferramenta de memória, o GDD registra todas as fases e decisões importantes do desenvolvimento do jogo, garantindo que nada seja esquecido e que problemas possam ser resolvidos rapidamente. Segundo, o GDD serve como um meio de comunicação, permitindo que as decisões sejam compartilhadas entre todos os membros da equipe. Isso facilita a colaboração, permitindo que todos contribuam, identifiquem problemas e encontrem soluções de forma conjunta.

Nesse sentido, os trabalhos de Moreno (2022) e Silva (2022) consistiram no desenvolvimento de um jogo chamado "*Logi Kingdom*", que tem como objetivo ensinar aos usuários sobre Lógica Proposicional e Tabela-Verdade. Os jogadores são imersos em um universo fictício semelhante ao planeta Terra, porém em menor escala e localizado em outra galáxia. Além disso, os autores apresentam o GDD como parte de seu trabalho, detalhando os elementos do jogo desenvolvido por eles. A Figura 1 ilustra a estrutura inicial do GDD adotada por esses autores.

Figura 1 - Representação de um *Game Design Document*



Fonte: Moreno (2022) e Silva (2022).

A Figura 1 apresenta a estrutura do GDD desenvolvido por Moreno (2022) e Silva (2022). O GDD produzido por eles foi dividido em dez áreas principais: História, *Gameplay*, Personagens, Controles, Câmera, Universo do Jogo, Inimigos, Interface, *Cutscenes* e Cronograma. Cada uma dessas áreas foi fundamental para a construção do jogo. Além disso, o GDD de Moreno e Sousa foi estruturado para abranger os elementos-chave que foram essenciais ao longo do desenvolvimento do projeto.

De acordo com Silva *et al*. (2021), durante a fase de planejamento e *brainstorming*, diferentes ideias e conceitos podem surgir entre os membros da equipe, sendo necessário documentar essas informações de maneira acessível, pois podem fazer parte do jogo. Dessa forma, o GDD tem o objetivo principal de servir como um material de referência para o desenvolvimento de um jogo específico.

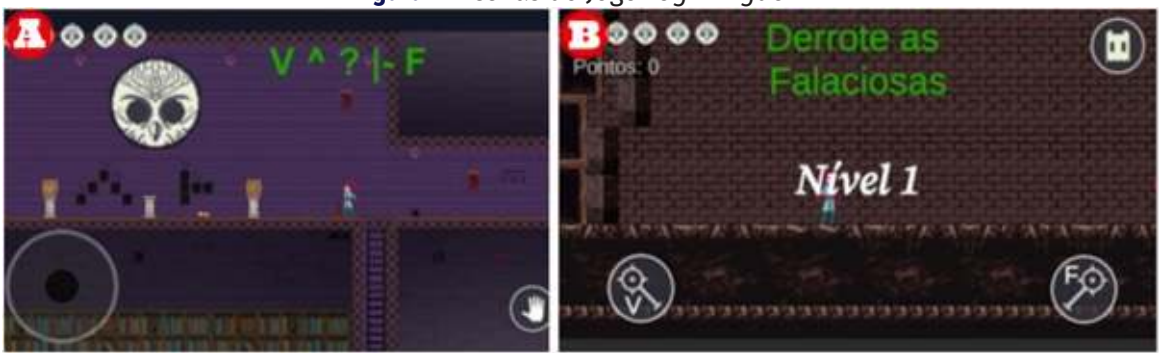
Para Silva *et al*. (2021), o processo de documentação não é uma tarefa simples para a equipe de desenvolvimento, especialmente devido às constantes mudanças no *design* do jogo, que resultam em atualizações frequentes. No entanto, a documentação do *design* é um passo importante para garantir que a equipe esteja alinhada e que nenhuma informação seja perdida ou ignorada durante a fase de desenvolvimento.

4.1.1. Jogos mobile: o exemplo do Logi Kingdom

Os jogos *mobiles* se tornaram parte da cultura contemporânea, especialmente devido à sua acessibilidade e disponibilidade em dispositivos móveis, como *tablets* e *smartphones*. De acordo com Bulgam (2015), o mercado de entretenimento está extremamente aquecido devido ao crescimento exponencial do público e, consequentemente, à maior demanda por conteúdo. Nesse contexto, os jogos *mobile* se destacam como um dos nichos que mais têm crescido nos últimos anos.

Os jogos para dispositivos móveis já fazem parte da realidade de diversas escolas e faculdades em todo o mundo, sendo utilizados desde os primeiros anos, quando a criança aprende a ler e escrever, até no ensino superior, em disciplinas como lógica e matemática. Um exemplo de jogo *mobile* educacional é o "*Logi Kingdom*" de Moreno *et al*. (2022), uma proposta de jogo educativo para o ensino lúdico e intuitivo da Lógica Proposicional e Tabela-Verdade. A Figura 2 a seguir apresenta algumas telas do jogo.

Figura 2 - Cenas do Jogo *Logi Kingdom*



Fonte: Moreno (2022) e Silva (2022).

Conforme mencionado por Moreno *et al*. (2022), o jogo *Logi Kingdom* pode ser visto como uma ferramenta que pode incentivar o ensino do pensamento lógico e capaz de motivar os alunos no aprendizado da Lógica Proposicional e Tabela-Verdade. Além disso, é evidente que os jogos para dispositivos móveis têm sido empregados em diversas situações para aprimorar o ensino de disciplinas variadas, bem como auxiliar na compreensão do pensamento lógico.

Os jogos *mobiles*, por serem altamente acessíveis, têm uma variedade de usos além do entretenimento, incluindo o ensino em diversas disciplinas e áreas. Nas escolas, os jogos *mobiles* são frequentemente utilizados para auxiliar no ensino de disciplinas como lógica matemática, digitação e outras, incorporando a jogabilidade com os temas da disciplina e aproveitando o poder dos elementos dos jogos para ensinar princípios e conceitos (Adams; Dormans, 2012).

De acordo com Ferreira (2024), quando os jogos digitais são aplicados adequadamente no currículo escolar e alinhados aos objetivos pedagógicos, podem estimular de forma prazerosa habilidades cognitivas, sociais e emocionais dos estudantes. A autora também ressalta a importância de formar professores para integrar a tecnologia em sala de aula, potencializando o desenvolvimento dos alunos.

Segundo Pereira e Rocha (2023), jogar é uma atividade prazerosa que gera emoções e insere o elemento lúdico no *gameplay*. Por isso, deve ser trabalhado pelos professores como uma ferramenta complementar de ensino. A multidisciplinaridade dos temas abordados nos jogos permite que eles despertem experiências únicas nos alunos, combinando elementos como dinâmica, mecânica e estéticas.

5. RESULTADOS E DISCUSSÃO

5.1. Breve história do Jogo criado a partir do GDD

Em um futuro não tão distante, a humanidade deu um passo além da inteligência artificial generativa que conhecemos hoje, criando um mundo virtual paralelo ao mundo real onde computadores alcançaram uma forma de consciência própria. Essas inteligências artificiais evoluíram para uma sociedade própria dentro do mundo virtual, conhecido como “Código Vivo”, repleto de cidades digitais, paisagens e seres inteligentes formados por linhas de códigos.

Em *Logic Think*, o jogador assume o papel de Allan, um jovem estudioso e apaixonado por tecnologias emergentes. Com anseio em desbravar os segredos das tecnologias digitais o impulsio-

na a desenvolver um algoritmo para acessar o "Código Vivo" através de seus óculos de Realidade Virtual. Ao colocar os óculos, ele consegue acessar e interagir com esse mundo. Fascinado pelo potencial da inteligência artificial no mundo real, Allan sonha em desvendar os segredos do Pensamento Computacional para compreender os seres virtuais que residem em "Código Vivo".

No entanto, o mundo virtual enfrenta grandes desafios, e a relação entre humanos e seres virtuais não é das melhores. Fora da rede, temerosos com a rápida evolução tecnológica e com a crescente onda da inteligência artificial que ameaça seus empregos, os seres humanos começaram a atacar o mundo virtual. Eles conectam na rede aparelhos infectados com vírus, espalhando o caos pelo ambiente virtual. Em meio a esse caos, os seres virtuais começaram a desaparecer. Os que sobraram se protegem dos perigos vindos do mundo exterior. Isso resultou em uma desaceleração do avanço da tecnologia e causou instabilidade em ambos os mundos.

Determinado a mostrar que os humanos e a Inteligência Artificial podem coexistir sem problemas, Allan embarca em uma emocionante jornada pelo mundo virtual, utilizando seus óculos de Realidade Virtual e guiado pelos quatro pilares do Pensamento Computacional: Decomposição; Reconhecimento de padrões; Abstração e; Algoritmos.

A jornada de Allan pelo "Código Vivo" é repleta de desafios, emoções e aprendizados. Ele busca não apenas aprender sobre o Pensamento Computacional, mas também sobre amizade, empatia e a verdadeira essência da vida, mesmo que seja composta por zeros e uns. Nessa história, Allan se torna não apenas um aprendiz do mundo virtual, mas um elo entre a humanidade e a tecnologia.

5.2. Personagens

Logic Think é um jogo que apresenta um garoto chamado Allan como personagem principal, um habilidoso programador que os jogadores controlam para proteger o "Código Vivo" de ameaças de vírus. Allan interage com dois Non-Player Character NPCs: os *Pixeisilicons* e os *Bitnomads*. Os *Pixeisilicons* são especialistas em processamento e organização de dados, além de serem especialistas em estratégias digitais e em resolução de problemas. Já os *Bitnomads* são uma comunidade nômade de seres digitais focados em liberdade e criatividade. Cada grupo contribui de maneira única para a história e para a jogabilidade.

Os personagens foram desenvolvidos com base nos conteúdos abordados nas disciplinas de computação, tanto em suas características quanto em seus nomes. No entanto, o estilo artístico precisou ser adaptado ao contexto do jogo, já que o presente GDD se refere a um jogo *mobile* de plataforma 2D que utiliza a técnica de *pixel art*. Moreno *et al*. (2021) e Klein *et al*. (2020) destacam que os jogos podem ser utilizados como ferramentas eficazes para o ensino de diversas disciplinas. Nesse contexto, a criação de personagens com características específicas é uma estratégia para contextualizar e aplicar conteúdos acadêmicos no ambiente virtual. Adams e Dormans (2012) também ressaltam a importância do *design* de personagens na construção de uma conexão emocional com o público, elemento essencial para o engajamento e o aprendizado.

5.2.1. Personagem Allan

Allan é um jovem estudioso e apaixonado por tecnologias. Movido pelo desejo de desvendar os segredos das tecnologias digitais, ele desenvolveu um algoritmo para acessar o "Código

Vivo" através de seus óculos de Realidade Virtual. Ao colocar os óculos, Allan é "levado" para o mundo virtual, podendo interagir com seus habitantes. Fascinado pelo potencial da inteligência artificial no mundo real, Allan sonha em desvendar os segredos do Pensamento Computacional. Seu objetivo é compreender os seres virtuais que residem no "Código Vivo" e ajudá-los a combater os vírus que estão destruindo o mundo virtual. A Figura 4, a seguir, apresenta o personagem.

Figura 4 - Personagem Allan



Fonte: Elaborado pelos autores.

Allan é um jovem dinâmico, inteligente e apaixonado por tecnologias. Ele demonstra habilidade notável em aprender e aplicar os pilares do Pensamento Computacional em problemas complexos. Além disso, utiliza óculos de realidade virtual que emitem raios desintegradores para eliminar inimigos no jogo, como vírus. Ele também possui uma chave especial que emprega para reparar danos causados pelos inimigos.

Durante sua infância, o jovem Allan esteve imerso na solidão do mundo real, mas tudo mudou quando ele desenvolveu algoritmos que o permitiram acessar o "Código Vivo". Nesse mundo digital, Allan encontrou um refúgio, experimentando euforia e felicidade ao fazer amizades com seres digitais.

No entanto, a alegria inicial foi abalada quando Allan descobriu que a humanidade estava ameaçando o "Código Vivo" com vírus disseminados pela rede. Determinado a salvar esse mundo que havia se tornado sua nova casa, Allan embarca em uma jornada para executar missões e realizar ações específicas para proteger o "Código Vivo" e garantir a segurança de seus habitantes digitais. O Quadro 3 apresenta as ações que o personagem poderá executar.

5.3. Sistemas do Jogo

Ao explorar o ambiente e coletar equipamentos e ferramentas, Allan aprimora suas características. Esses itens também servem como armas contra os vírus, como óculos de realidade virtual que disparam raios (Figura 5 - A) e uma chave que funciona como arma de ataque e reparo contra os vírus (Figura 5 - B). Além disso, Allan pode adquirir um escudo de algoritmos para se proteger de vírus mais agressivos. O escudo é alimentado por pontos de força, aumentando sua durabilidade (Figura 5 - C).

Figura 5 - Sistemas de aprimoramento



Fonte: Elaborado pelos autores.

A maioria dos jogos digitais possuem um sistema de vidas para os personagens que são controlados pelo jogador. O GDD desenvolvido neste projeto adota um sistema de vidas inspirado no gênero *Role-Playing Game* (RPG). O personagem principal do *Logic Think* possui um total de cinco vidas ao longo de todo o jogo. Portanto, o jogador precisa se preocupar em conservar o maior número possível de vidas durante a jogabilidade, pois a perda delas pode resultar no reinício do jogo. As vidas de Allan serão representadas por cinco corações, conforme a Figura 6.

Figura 6 - Sistemas de vidas



Fonte: Elaborado pelos autores.

À medida que o jogador elimina os vírus presentes no «Código Vivo», ele é recompensado com uma quantidade específica de pontos. Com o avanço nos níveis, essa pontuação aumenta gradualmente, acompanhando a crescente complexidade dos desafios enfrentados. Esses pontos podem ser utilizados para fortalecer o sistema de defesa do personagem. O escudo de Allan é alimentado por esses pontos, que podem ser investidos em aumentar a sua força, proporcionando assim uma maior durabilidade à proteção.

No jogo, cada nível apresenta desafios específicos que contribuem para a jornada do jogador no mundo digital do "Código Vivo". No Nível 1, o jogador é introduzido ao jogo e aos controles básicos, enquanto no Nível 2, ele precisa compreender conceitos de decomposição para enfrentar desafios no *Datacenter*. No Nível 3, o foco é no reconhecimento de padrões, seguido pelo desenvolvimento da habilidade de abstração no Nível 4. Finalmente, no Nível 5, o jogador enfrenta desafios que exigem a aplicação de algoritmos para eliminar vírus mais desafiadores.

Dentro do universo do "Código Vivo", o jogador enfrentará quatro desafios distintos. Na primeira missão, é essencial que o jogador aprenda e aplique os conceitos de decomposição para eliminar vírus e corrigir problemas no *Datacenter*. Concluída essa etapa, o desafio seguinte envolve a correção de problemas relacionados aos padrões de imagens gerados no jogo. Ao término da segunda missão, o jogador precisa compreender os conceitos de abstração e organizar um conjunto de dados. Na quarta e última missão, o jogador é desafiado a criar algoritmos para proteger o "Código Vivo" das ameaças virais e salvar o mundo digital dessas ameaças. A conclusão de cada missão recompensa o jogador com uma insígnia, reconhecendo suas habilidades e conquistas dentro do jogo.

5.4. Cenários e *Design* do Jogo

O mundo de "Código Vivo" é vasto, diversificado e abrange uma variedade de cenários digitais, desde cidades de algoritmos até a floresta de dados. Cada cenário é único, com suas próprias características peculiares e desafios específicos. Por exemplo, as florestas de dados são repletas de árvores e criaturas geradas por dados, enquanto as cidades são vastas extensões de informações organizadas em códigos. A jogabilidade é influenciada pelo ambiente, com quebra-cabeças e desafios únicos surgindo conforme o cenário em que o jogador se encontra.

A Figura 7 apresenta o primeiro cenário do jogo, mostrando o quarto de Allan. No ambiente, Allan trabalha no desenvolvimento do código para seus óculos de realidade virtual, cercado por monitores e equipamentos eletrônicos. O cenário destaca a dedicação e o foco de Allan em seu projeto de tecnologia avançada.

Figura 7 - Primeiro cenário do jogo



Fonte: Elaborado pelos autores.

O segundo cenário do jogo apresenta uma cidade repleta de luzes cintilantes, que Allan acessa após finalizar o código para seus óculos e colocá-los (Figura 8). O ambiente é composto por arranha-céus iluminados, criando uma atmosfera distópica e futurística. Nesse cenário, Allan encontra *Pixeisilicon*, que o introduz ao jogo e às missões que ele deve realizar para proteger o "Código Vivo" dos vírus.

Figura 8 - Segundo cenário do jogo



Fonte: Elaborado pelos autores.

O terceiro cenário do jogo que abrange a "Decomposição" é uma sala de *Datacenter*, onde começam as missões de Allan (Figura 9). Este ambiente é composto por diversos servidores e computadores, e o personagem principal deve consertá-los para eliminar os vírus que estão causando problemas aos *Pixeisilicons*. A sala tem um *design* tecnológico e imersivo, com luzes de LED piscando, fios e cabos conectando os servidores, e uma aparência que transmite um senso de complexidade.

Figura 9 - Terceiro cenário do jogo



Fonte: Elaborado pelos autores.

A Figura 9 ilustra a primeira missão do jogo, onde o jogador precisa desmontar e remontar o *Datacenter* utilizando a lógica da numeração binária. Essa tarefa desafia o jogador a desenvolver uma compreensão aprofundada dos elementos e mecânicas do jogo, exigindo atenção aos detalhes e aplicação prática dos conceitos de numeração binária. Além de promover o domínio dos componentes do *Datacenter*, a missão incentiva o jogador a desenvolver habilidades de resolução de problemas e a exercitar o pensamento lógico, essencial para avançar nas etapas subsequentes do jogo.

O quarto cenário que abrange o "Reconhecimento de Padrões" é composto por um ambiente externo cheio de neon, luzes vibrantes, e paisagens tecnológicas (Figura 10). Dentro do ambiente, existem imagens projetadas em telas flutuantes, que mudam de forma e cor à medida que o jogador se aproxima. O jogador deve interagir com as telas para reorganizar as imagens ou escolher a sequência correta de padrões.

Figura 10 - Quarto cenário do jogo



Fonte: Elaborado pelos autores.

A Figura 10 apresenta a segunda missão do jogo que consiste na habilidade de reconhecimento de padrões. Nesse desafio, o jogador precisa ajustar um dispositivo, ligando e desligando pontos conforme o padrão exibido no *card* de referência ao lado. O objetivo é alinhar o dispositivo ao padrão correto, representado por pontos pretos (ligados) e brancos (desligados). Esse exercício ajuda o jogador a desenvolver a capacidade de identificar e reproduzir padrões, uma habilidade importante para resolver problemas complexos dentro do universo do jogo.

O quinto cenário que abrange a "Abstração" é uma sala digital representada como um espaço fluido, com painéis de dados e fileiras de informações em formato de tabelas e gráficos (Figura 11). O ambiente é visualmente dinâmico, com uma mistura de cores brilhantes e tons metálicos, re-

fletindo a complexidade dos dados. No cenário, Allan tem acesso a ferramentas virtuais para filtrar, classificar e organizar os dados.

Figura 11 - Quinto cenário do jogo



Fonte: Elaborado pelos autores.

A Figura 11 apresenta a terceira missão do jogo, que explora o conceito de abstração. Nessa missão o jogador deve separar as informações apresentadas, classificando-as em diferentes categorias de acordo com os ícones representados nas caixas na parte inferior da tela. O objetivo é que o jogador foque nos aspectos essenciais de cada elemento e ignore detalhes irrelevantes para completar a tarefa com sucesso. Esse exercício reforça a habilidade de abstração, treinando o jogador a identificar e organizar apenas as informações mais importantes, essencial para resolver problemas complexos no ambiente virtual do jogo.

O sexto cenário que abrange os "Algoritmos" é um laboratório de programação futurística, com painéis holográficos, telas e circuitos digitais visíveis (Figura 12). O ambiente é altamente tecnológico, com uma estética que lembra um laboratório de alta tecnologia ou um centro de controle digital. O cenário inclui fragmentos de código representados como blocos de programação que o jogador deve analisar e organizar. Esses blocos podem incluir comandos, *loops*, condições e funções.

Figura 12 - Sexto cenário do jogo



Fonte: Elaborado pelos autores.

A Figura 12 apresenta a quarta missão do jogo, que explora o conceito dos algoritmos. Nessa missão, o jogador deve conduzir o *Bitnomad* até o processador, localizado no canto inferior direito da grade, utilizando uma sequência de comandos diretos (setas para cima, baixo, esquerda e direita). O jogador precisa planejar o caminho, evitando os vírus, e organizando a ordem das instruções na parte inferior da tela. Esse exercício reforça a habilidade de construção de algoritmos, treinando o jogador a criar uma sequência lógica de passos, corrigir erros (caso *Bitnomad* colida com os vírus)

e identificar o caminho mais eficiente para atingir o objetivo final, essencial para resolver problemas complexos no ambiente virtual do jogo.

Os cenários foram projetados para garantir a fluidez do jogo, incorporando elementos tecnológicos que enriquecem o ambiente e facilitam a compreensão do Pensamento Computacional. Com esses elementos, o jogo proporciona entretenimento e pode servir como uma ferramenta educativa, promovendo o desenvolvimento das habilidades de Pensamento Computacional de maneira lúdica.

5.5. Desafios e Pontuações

Os desafios do *Logic Think* estão relacionados com os pilares do Pensamento Computacional e aos vírus presentes no jogo. O jogador ganhará certa quantidade de pontos a cada vírus que exterminar. Com o passar dos níveis essa pontuação será maior, e consequentemente quanto maior o nível, maior será a pontuação e a complexidade dos problemas enfrentados em "Código Vivo". A descrição da pontuação de acordo com o nível é apresentada no quadro a seguir.

Quadro 2 - Distribuição de Nível por Conteúdo

Nível	Conteúdo	Tempo de Vírus	Recompensa
1	Introdução	10 segundos	1 ponto por vírus
2	Decomposição	8 segundos	2 pontos por vírus
	Concluir a Missão	-	Insígnia - Decomposição
3	Reconhecimento de Padrões	6 segundos	3 pontos por vírus
	Concluir a Missão	-	Insígnia - Reconhecimento de Padrões
4	Abstração	4 segundos	4 pontos por vírus
	Concluir a Missão	-	Insígnia - Abstração
5	Algoritmos	2 segundos	5 pontos por vírus
	Concluir a Missão	-	Insígnia - Algoritmos

Fonte: Elaborado pelos autores.

O Quadro 2 apresenta a distribuição dos conteúdos por nível, delineando os desafios encontrados pelo jogador em cada etapa. Cada nível é identificado por um número que representa sua sequência de progressão. Os conteúdos descrevem os tipos de conhecimento que o jogador poderá adquirir. Por exemplo, no nível 1, o jogador é introduzido ao jogo através de um tutorial que ensina as mecânicas básicas.

O "Tempo de Vírus" indica o período que o jogador tem para derrotar um vírus antes que novos surjam e o ameacem. Esse tempo diminui à medida que a dificuldade aumenta, demandando maior agilidade e raciocínio por parte do jogador. Já a "Recompensa" representa a quantidade de pontos que o jogador recebe por cada vírus derrotado. Essa pontuação por vírus também se eleva conforme a dificuldade aumenta, incentivando o progresso do jogador.

A tabela apresenta “Missões” que o jogador enfrentará ao atingir o término de cada nível e ter estudado sobre o pilar do Pensamento Computacional interligado a esse nível. Essas missões funcionam como uma etapa final do nível para validar o conhecimento adquirido pelo jogador. Ao concluir o nível, o jogador é recompensado com uma “Insígnia” relacionada ao tema abordado. Para avançar nos níveis do jogo, o jogador deve garantir a sobrevivência de Allan, derrotar os vírus que cruzarem seu caminho e completar a missão de cada nível. Os corações representam as vidas disponíveis para o personagem.

No Nível 1 – “Ambientação e Apresentação do Jogo”, o jogador é imerso no universo do jogo e é apresentado a *Pixeisilicons*, que delineia a missão de Allan. Durante esta etapa inicial, os primeiros vírus começam a surgir, e Allan é recompensado com 1 ponto por cada vírus eliminado. O principal objetivo deste nível é familiarizar o jogador com os controles e a dinâmica do jogo, preparando-o para os desafios subsequentes.

No Nível 2 – “Missão: Decomposição”, os vírus continuam a representar uma ameaça, porém a pontuação por vírus eliminado é dobrada. A característica distintiva deste nível é a necessidade de o jogador compreender os conceitos de decomposição transmitidos por *Pixeisilicons* para resolver o desafio do *Datacenter*. A missão envolve desmontar e remontar o *Datacenter* de acordo com a numeração binária, exigindo do jogador uma compreensão mais profunda dos elementos do jogo. Como recompensa por completar esta missão, o jogador é agraciado com uma insígnia de Aprendizagem.

No Nível 3 – “Missão: Reconhecimento de Padrões”, os vírus ainda representam uma ameaça, porém a pontuação por vírus eliminado é triplicada. Os vírus começam a surgir em intervalos mais espaçados, com um novo vírus aparecendo a cada 6 segundos. Percebendo o avanço de Allan, *Pixeisilicons* apresenta uma nova missão centrada no reconhecimento e aplicação de padrões dentro do mundo digital. A missão desafia o jogador a identificar padrões de imagens, exigindo uma compreensão mais profunda do pilar de Reconhecimento de Padrões. Ao concluir esta missão, o jogador é agraciado com uma insígnia de Observador.

No Nível 4 – “Missão: Abstração”, o jogador progride para uma fase mais avançada de sua jornada, concentrada na habilidade de Abstração. Um novo NPC, o *Bitnomads*, é introduzido ao jogo. Reconhecendo o crescimento de Allan, o *Bitnomads* apresenta uma nova missão que demanda a capacidade de abstrair e filtrar informações irrelevantes para resolver desafios complexos. Os vírus continuam representando uma ameaça, mas agora o jogador é desafiado a ganhar 4 pontos por cada vírus eliminado, refletindo a necessidade de focar nos aspectos essenciais do problema.

No Nível 4, Allan precisa organizar uma grande quantidade de dados virtuais coletados durante uma investigação no mundo digital. No entanto, os dados estão desorganizados e misturados, contendo informações relevantes e muitos elementos irrelevantes. Allan precisa aplicar sua habilidade de abstração para classificar e filtrar os dados, separando o que é importante do que é supérfluo. Ao concluir esta missão, o jogador é recompensado com uma insígnia de Abstração.

No Nível 5 – “Missão: Algoritmos”, Allan enfrenta desafios que requerem a aplicação de algoritmos para resolver problemas complexos dentro do mundo digital. Os vírus tornam-se ainda mais insidiosos, exigindo que Allan desenvolva algoritmos eficientes para neutralizá-los. Agora, para

cada vírus eliminado, Allan ganha 5 pontos, refletindo a crescente complexidade dos desafios. Os vírus surgem em intervalos de apenas 2 segundos, aumentando a pressão sobre Allan para reagir rapidamente e aplicar os algoritmos de forma eficaz.

No Nível 5 - "Missão Algoritmos", para proteger o sistema e evitar danos, Allan precisa criar um algoritmo que leve o *Bitnomad* até o processador central. O jogador deve conduzir o *Bitnomad*, utilizando uma sequência de comandos, planejando o caminho e evitando os vírus. O exercício reforça a habilidade de construção de algoritmos, treinando o jogador a criar uma sequência lógica de passos e encontrar o caminho mais eficiente até o objetivo final do jogo.

Martins e Munhoz (2016) e McGonigal (2012) argumentam que as recompensas nos jogos podem moldar o comportamento dos jogadores ao incentivá-los a explorar novas áreas ou a completar missões específicas. Isso ajuda a orientar a experiência do jogador em direção aos objetivos estabelecidos pelo jogo. Além disso, a utilização de sistemas de recompensas com elementos como pontos, níveis ou troféus aumenta a sensação de progresso e realização, contribuindo para uma experiência mais satisfatória.

Portanto, a partir dos dados apresentados neste artigo a partir do jogo *Logic Think*, inspirado nos pilares do Pensamento Computacional, foram utilizados quatro conceitos fundamentais: Decomposição, Reconhecimento de Padrões, Abstração e Algoritmos. Cada vez que um jogador conclui um nível, é recompensado com uma insígnia relacionada a um desses pilares, reforçando o aprendizado de habilidades computacionais, conforme abordado por autores como Code.Org (2016), Brackmann (2017), BBC Learning (2015) e Liukas (2015).

6. CONSIDERAÇÕES FINAIS

A pesquisa desenvolvida permitiu compreender, dentre alguns resultados, que os jogos digitais educacionais podem proporcionar novas formas de ensino para estudantes da Educação Básica, uma vez que podem ser importantes facilitadores na compreensão de conteúdos trabalhados em sala de aula, de diferentes disciplinas curriculares, a partir do desenvolvimento de um projeto de jogo digital lúdico.

Mas, para que esse processo se efetive, o GDD deve propor novas formas ativas de ensino, ao servir de recurso metodológico para estudantes e professores, desde que o GDD seja elaborado com o objetivo de tornar mais acessível o conteúdo trabalhado em sala de aula, importante para que os alunos aprendam de maneira mais interativa, potencializando o seu aprendizado.

O GDD produzido nesta pesquisa possibilita também aos professores e estudantes terem contato inicial com o Pensamento Computacional em sala de aula, que podem auxiliá-los a entender como os jogos digitais podem oferecer uma gama de recursos metodológicos para o planejamento, avaliação e conhecimento de diferentes conteúdos do currículo escolar, o que torna o processo de ensino e aprendizagem mais dinâmico.

É essencial assinalar que o GDD produzido incorpora as habilidades exigidas pela BNCC, no que concerne ao Pensamento Computacional na Educação Básica contribui para preencher as lacunas quanto ao desenvolvimento de novas habilidades e competências que promovam o uso consciente, ético e crítico da tecnologia no dia a dia.

Entretanto, implementar a tecnologia no ensino, especificamente no que tange ao uso do GDD em ambiente escolar, traz importantes desafios. O primeiro se refere à falta de infraestrutura de algumas escolas para receber esse tipo de tecnologia, e a falta de conhecimento dos professores em seu uso. Esses desafios ressaltam a necessidade de mais cursos de formação continuada que tratem de temas relacionados às tecnologias educacionais e maior investimento público em escolas para o uso e manutenção contínua adequadas dessas tecnologias.

7. REFERÊNCIAS

ADAMS, Ernest; DORMANS, Joris. ***Game Mechanics: Advanced Game Design***. Berkeley: New Riders Publishing, 2012.

BARBOSA, Ruy Madsen. **Aprendo com jogos**. São Paulo: Grupo Autêntica, 2014.

BARCELOS, Thiago Schumacher; SILVEIRA, Ismar Frango. Pensamento computacional e educação matemática: Relações para o ensino de computação na educação básica. *In*: **Anais.** XX Workshop sobre Educação em Computação, Curitiba, 2012, p. 23.

BBC *LEARNING*. ***What is computational thinking***?, 2015. Disponível em: http://www.bbc.co.uk/education/guides/zp92mp3/revision. Acesso em: 6 jul. 2024.

BRACKMANN, Christian Puhlmann. **Desenvolvimento do Pensamento Computacional através de atividades desplugadas na educação básica**. 226 f. Tese (Doutorado) - Curso de Informática na Educação, Centro Interdisciplinar de Novas Tecnologias na Educação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**. Ministério da Educação, Brasília, DF. MEC, 2018. Disponível em: http://www.mec.gov.br. Acesso em: 3 mar. 2023.

BULGAM, Rodolfo Helfenstein. **Kouri**: desenvolvimento de jogos Mobile com Unity, 2015.

CODE.ORG. **Aprendendo a Ciência da Computação**. Disponível em: < https://code.org/>. Acesso em: mar. 2023.

CODE.SPARK. **Jogo eletrônico de plataforma**. 2021. Disponível em: https://codespark.com/play/. Acesso em: 23 ago. 2024.

DENZIN, Norman K.; LINCOLN, Yvonna S. (Eds.). ***The Sage handbook of qualitative research***. 5 ed. Thousand Oaks: Sage, 2018.

ERICKSON, Frederick. *Métodos cualitativos de investigación*. *In*: WITTROCK, M. C. (Ed.). ***La investigación de la enseñanza***. Barcelona- Buenos Aires-México: Paidos, 1989, p. 195-299.

FERREIRA, Natália Uchôa Almeida. **Jogos Digitais na Educação Infantil**: Uma sequência didática para alunos da escola municipal Guadalupe (Porto Velho - RO). 31 f. Trabalho de Conclusão de Curso (Licenciatura em Pedagogia) - Fundação Universidade Federal de Rondônia, Porto Velho, 2024.

ISTE. Standards for Students. (2016). ***International. Society for Technology in Education***. Disponível em: https://www.iste.org/standards/for-students#startstandards. Acesso em: 25 mar. 2024.

KIYA, Márcia Cristina da Silveira. **O uso de Jogos e de atividades lúdicas como recurso pedagógico facilitador da aprendizagem**. Universidade Estadual de Ponta Grossa-UEPG, 2014.

KLEIN, Danieli Regina; CANEVESI, Fernanda Cristina Sanches; FEIX, Angela Regina; GRESELE, Jizéli Fonseca Parreira; WILHELM, Elizane Maria de Siqueira. Tecnologia na educação: evolução histórica e aplicação nos diferentes níveis de ensino. **Educere-Revista da Educação da UNIPAR**, v. 20, n. 2, 2020. https://doi.org/10.25110/educere.v20i2.2020.7439

LEITE, Patrícia da Silva; DE MENDONÇA, Vinícius Godoy. Diretrizes para *game design* de jogos educacionais. **Proc. SBGames, *Art Design Track***. 2013, p. 132-141.

LIUKAS, Linda. ***Hello Ruby***: *adventures in coding*. Feiwel & Friends, 2015.

MAGNO, Carlos *et al*. MD Investigações: um jogo educacional aberto para auxiliar a aprendizagem de lógica. **Anais** do XXIX Simpósio Brasileiro de Informática na Educação (Sbie 2018), Brasília, p. 675-684, 2018. https://doi.org/10.5753/cbie.sbie.2018.675.

MARTINS, Darlan Rodrigues Martins; MUNHOZ, Antonio Siemsen. **Gamificação: Perspectivas de Utilização no Ensino Superior**, 2016. UNINTER. Curitiba/PR.

McGONIGAL, J. **Realidade em jogo: por que os games nos tornam melhores e como eles podem mudar o mundo**. Rio de Janeiro: Best Seller, 2012.

MORENO, Douglas Aquino; SILVA, Stefan Lucas Aquino; SOUZA, Calebe Loures Sampaio Eler de; SANTOS, Natanna Rocha; BRITO, Parcilene Fernandes de. Desenvolvimento de uma plataforma gamificada baseada em uma HQ de Lógica. *In*: **Anais.** ENCOINFO - Congresso de Computação e Tecnologias da Informação, 23., 2021, Palmas - TO. Palmas - TO: CEULP/ULBRA, 2021, p. 69 - 81. Disponível em: https://ulbra-to.br/encoinfo/edicoes/2021/artigos/desenvolvimento-de-uma-plataforma-gamificada-baseada-em-uma-hq-de-logica/. Acesso em: 06 jul. 2024.

MOREIRA, Emerson Elder; CASTRO, Adriane Belluci Belório de. **Proposta de jogo digital para aprendizagem de língua portuguesa utilizando o *game design document***. SIED: EnPED Simpósio Internacional de Educação a Distância e Encontro de Pesquisadores em Educação a Distância, 2016.

MORENO, Douglas Aquino. ***Logi Kingdom*: *Game Design*** e Protótipo de um Jogo para Dispositivos Móveis Baseado em uma HQ de Lógica. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação). Centro Universitário Luterano de Palmas, Palmas, Tocantins, 2022. Disponível em: <http://ulbra-to.br/bibliotecadigital/publico/home/documento/2026>. Acesso em: 10 jun. 2024.

OLIVEIRA, Adriano Zarlam Peixoto de. **O ensino-aprendizagem de lógica matemática na educação básica**. 2020. 58 f. TCC (Graduação) - Curso de Matemática, Instituto de Ciências Exatas, Universidade Federal Rural do Rio de Janeiro, Seropédica, 2020.

PEREIRA, Diego Rodrigues; ROCHA, Luciano Beiestorf. O Uso De Jogos Digitais Como Ferramenta De Aprendizagem. *In*: **Anais.** Trilha De Educação – Artigos Completos - Simpósio Brasileiro De Jogos E Entretenimento Digital (Sbgames), 22, 2023, Rio Grande/RS. Porto Alegre: Sociedade Brasileira de Computação, 2023, p. 858-868. https://doi.org/10.5753/sbgames_estendido.2023.233891 .

PERUCIA, Alexandre; BERTHÊM, Antônio de; BERTSCHINGER, Guilherme; CASTRO, Roberto R. **Desenvolvimento de Jogos Eletrônicos: Teoria e Prática**. 1a ed. São Paulo: Novatec, 2005.

PILLÃO, Delma. **A pesquisa no âmbito das relações didáticas entre matemática e música: Estado da Arte**. 109f. Dissertação (Mestrado em Educação) – Faculdade de Educação, Universidade de São Paulo, São Paulo, 2009.

RESENDE, Gustavo Barreto. **Jogos educacionais para dispositivos móveis, como e quando utilizá-los**. 39 f. TCC (Graduação) - Curso de Licenciatura em Computação, Universidade de Pernambuco, Pernambuco, 2013. Disponível em: http://www.upe.br/garanhuns/wp-content/uploads/2019/09/Monogra%CC%81fia-GustavoBarretoResende.pdf. Acesso em: 10 out. 2024.

ROGERS, Scott. ***Level up***: um guia para o *design* de grandes jogos. São Paulo: Editora Blucher, 2013.

SBC. **Referenciais de Formação em Computação**: Educação Básica. Porto Alegre: Editora da Sociedade Brasileira de Computação, 2017. Disponível em: http://www.sbc.org.br/noticias/10-slideshow-noticias/1996-referenciais-de-formacao-em-computacao-educacao-basica. Acesso em: 29 mai. 2023.

SCHELL, Jesse. **A arte de *Game Design***: o livro original. Rio de Janeiro: Elsevier, 2010.

SCHUYTEMA, Paul. ***Design* de games: uma abordagem prática**. São Paulo: Cengage Learning, 2008. 447 p.

SILVA, Gabriel Fonseca; SILVA, Tatyane Souza Calixto da; FILHO, Michele Perrone; ARRIVABENE, Rafael Mariano; SPERANZA, Ygor. ***Game Design***. Porto Alegre: Sagah, 2021.

SILVA, Stefan Lucas Aquino. ***Logi Kingdom*:** Implementação das mecânicas e dinâmicas de um jogo para dispositivos móveis baseado em uma HQ de lógica. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação). Centro Universitário Luterano de Palmas, Palmas, Tocantins, 2022. Disponível em: <http://ulbra-to.br/bibliotecadigital/publico/home/documento/2036>. Acesso em: 06 jul. 2024.

STRAUSS, Anselm; CORBIN, Juliet. **Pesquisa qualitativa:** técnicas e procedimentos para o desenvolvimento de teoria fundamentada. Porto Alegre: Artmed, 2008.

VICARI, Rosa Maria; MOREIRA, Alvaro Freitas; MENEZES, Paulo Fernando Blauth. **Pensamento Computacional**: revisão bibliográfica. 2018.

Como citar – ABNT

MORENO, Douglas Aquino; ARAUJO, Gustavo Cunha de. Pensamento Computacional na Educação Básica: o Game Design Document (GDD) como proposta pedagógica. **Revista Poíesis Pedagógica**, Catalão/GO, Brasil, v. 23, e2025011, outubro, 2024. https://doi.org/10.69532/2178-4442.v23.75010

Como citar – APA

Moreno, D. A., & Araujo, G. C. de (2025). Pensamento Computacional na Educação Básica: o Game Design Document (GDD) como proposta pedagógica. *Revista Poíesis Pedagógica*, *23*, e2025011. https://doi.org/10.69532/2178-4442.v23.75010

Apêndice – Informações sobre o artigo

Histórico editorial
Submetido: 12 de março de 2025.
Aprovado: 29 de setembro de 2025.
Publicado: 01 de outubro de 2025.

Conflito de interesse
Nada a declarar.

Declaração de disponibilidade de dados
Todos os dados foram apresentados/gerados no presente artigo.

Contribuição dos autores
Resumo/Abstract/Resumen: Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Introdução ou Considerações iniciais:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Referencial teórico:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Metodologia:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Análise de dados:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Discussão dos resultados:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Conclusão ou Considerações finais:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Referências:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Revisão do manuscrito:** Douglas Aquino Moreno, Gustavo Cunha de Araujo; **Aprovação da versão final publicada:** Douglas Aquino Moreno, Gustavo Cunha de Araujo.

Direitos Autorais
Os direitos autorais são mantidos pelos autores, os quais concedem à Revista Poíesis Pedagógica os direitos exclusivos de primeira publicação. Os autores não serão remunerados pela publicação de trabalhos neste periódico. Os autores têm autorização para assumir contratos adicionais separadamente, para distribuição não exclusiva da versão do trabalho publicado nesta revista (ex.: publicar em repositório institucional, em site pessoal, publicar uma tradução, ou como capítulo de livro), com reconhecimento de autoria e publicação inicial nesta revista. Os editores da Revista Poíesis Pedagógica têm o direito de realizar ajustes textuais e de adequação às normas da publicação.

Open Access
Este artigo é de acesso aberto (***Open Access***) e sem cobrança de taxas de submissão ou processamento de artigos dos autores (***Article Processing Charges – APCs***). O acesso aberto é um amplo movimento internacional que busca conceder acesso online gratuito e aberto a informações acadêmicas, como publicações e dados. Uma publicação é definida como 'acesso aberto' quando não existem barreiras financeiras, legais ou técnicas para acessá-la–ou seja, quando qualquer pessoa pode ler, baixar, copiar, distribuir, imprimir, pesquisar ou usá-la na educação ou de qualquer outra forma dentro dos acordos legais.

Licença de uso
Este artigo é licenciado sob a Licença **Creative Commons Atribuição 4.0 Internacional (CC BY 4.0)**. Esta licença permite compartilhar, copiar, redistribuir o artigo em qualquer meio ou formato. Além disso, permite adaptar, remixar, transformar e construir sobre o material, desde que seja atribuído o devido crédito de autoria e publicação inicial nesta revista.

Verificação de Similaridade
Este artigo foi submetido a uma verificação de similaridade utilizando o software de detecção de texto **iThenticate** da Turnitin, através do serviço ***Similarity Check*** da Crossref.

Processo de avaliação
Revisão por pares duplo-cega (***Double blind peer review***).

Editora
Cláudia Tavares do Amaral

Fomento
O artigo foi editado, diagramado e publicado com o apoio do auxílio financeiro concedido pela **FAPEG Edital nº 10/2023** – Programa de Apoio a Periódicos Científicos de Instituições de Ensino Superior do Estado de Goiás.

Publisher
Este artigo foi Publicado na **Revista Poíesis Pedagógica** vinculada ao Programa de Pós-Graduação em Educação da Faculdade de Educação da **Universidade Federal de Catalão - UFCAT**. A Revista Poíesis Pedagógica publica artigos de natureza técnico-científica, provenientes de estudos e pesquisas que ofereçam subsídios para o desenvolvimento do conhecimento educacional, propiciando um diálogo entre os diferentes campos da educação no Portal de Periódicos da UFCAT. As ideias expressadas neste artigo são de responsabilidade de seus autores, não representando, necessariamente, a opinião do corpo editorial ou da referida universidade. Na **Avaliação CAPES (2017-2020)** a Revista Poíesis Pedagógica obteve **Qualis B1.**