



UNIVERSIDADE FEDERAL DO TOCANTINS
PROGRAMA DE PÓS-GRADUAÇÃO PROFISSIONAL EM EDUCAÇÃO
MESTRADO PROFISSIONAL EM EDUCAÇÃO

DOUGLAS AQUINO MORENO

LOGIC THINK: GAME DESIGN DOCUMENT DE UM JOGO PARA O
DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL

Palmas-TO
2024

DOUGLAS AQUINO MORENO

**LOGIC THINK: GAME DESIGN DOCUMENT DE UM JOGO PARA O
DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL**

Trabalho apresentado ao Programa de Pós-Graduação Profissional em Educação da Universidade Federal do Tocantins (UFT), na Linha de Pesquisa: Métodos e técnicas de ensinar e aprender na educação básica, como requisito parcial para obtenção do título de Mestre em Educação.

Orientador: Prof. Dr. Gustavo Cunha de Araújo.

**Palmas-TO
2024**

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da Universidade Federal do Tocantins

- M8431 Moreno, Douglas Aquino.
 LOGIC THINK: GAME DESIGN DOCUMENT DE UM JOGO PARA O
 DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL. / Douglas
 Aquino Moreno. – Palmas, TO, 2024.
 100 f.
- Dissertação (Mestrado Profissional) - Universidade Federal do Tocantins
 – Câmpus Universitário de Palmas - Curso de Pós-Graduação (Mestrado)
 Profissional em Educação, 2024.
 Orientador: Gustavo Cunha de Araújo
1. Game Design Document. 2. Pensamento Computacional. 3. Jogos
 Educativos. 4. Base Nacional Comum Curricular. I. Título

CDD 370

TODOS OS DIREITOS RESERVADOS – A reprodução total ou parcial, de qualquer
forma ou por qualquer meio deste documento é autorizado desde que citada a fonte.
A violação dos direitos do autor (Lei nº 9.610/98) é crime estabelecido pelo artigo 184
do Código Penal.

**Elaborado pelo sistema de geração automática de ficha catalográfica da
UFT com os dados fornecidos pelo(a) autor(a).**

DOUGLAS AQUINO MORENO

**LOGIC THINK: GAME DESIGN DOCUMENT DE UM JOGO PARA O
DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL**

Trabalho apresentado ao Programa de Pós-Graduação Profissional em Educação da Universidade Federal do Tocantins (UFT), na Linha de Pesquisa: Métodos e técnicas de ensinar e aprender na educação básica, como requisito parcial para obtenção do título de Mestre em Educação.

Orientador: Prof. Dr. Gustavo Cunha de Araújo.

Data de aprovação: 03/12/2024

Banca Examinadora

Prof. Dr. Gustavo Cunha de Araújo, orientador e presidente da banca, Universidade
Federal do Norte do Tocantins (UFNT/UFT/PPPGE)

Profa. Dra. Marluce Evangelista Carvalho Zacariotti, examinadora interna, Universidade
Federal do Tocantins (UFT/PPPGE)

Prof. Dr. José Carlos Miguel, examinador externo, Universidade Estadual Paulista
(PPGE/UNESP/Marília)

Palmas-TO
2024

Dedico este trabalho ao jovem que teve a coragem de deixar sua cidade natal e enfrentar os desafios de uma nova cidade, mesmo diante das incertezas e dúvidas sobre suas escolhas. Dedico também ao recém-formado que, sem saber exatamente qual caminho seguir, confiou no fluxo da vida e permitiu que as oportunidades se desdobrassem de maneira inesperada.

AGRADECIMENTOS

Agradeço primeiramente a Deus por todas as oportunidades que me foram concedidas, pela força e serenidade nos momentos de fraqueza e desafios.

Minha gratidão imensa à minha família, por sempre apoiar minhas decisões, e ao meu namorado, por sua paciência, apoio incondicional, e por ter estado ao meu lado nos momentos de ansiedade e estresse, oferecendo sempre palavras de conforto e incentivo.

Aos meus amigos, agradeço por todo o auxílio nas dúvidas e pela contribuição essencial para a realização desta pesquisa. Incluo aqui meus antigos professores, que hoje considero amigos e colegas de trabalho, cuja orientação foi fundamental ao longo do meu caminho.

Minha profunda gratidão ao meu orientador, Prof. Dr. Gustavo Cunha de Araújo, por sua orientação e apoio dedicado ao longo de todo o processo. Sua expertise e comprometimento foram fundamentais na construção desta pesquisa e na elaboração dos artigos que desenvolvemos juntos. Agradeço também a banca examinadora, Prof. Dr. José Carlos Miguel e Profa. Dra. Marluce Zacariotti, pelas valiosas contribuições.

Aos meus colegas de mestrado, agradeço pelas discussões enriquecedoras sobre a área pedagógica e pelos debates que tanto contribuíram para o desenvolvimento do meu pensamento crítico construtivo.

RESUMO

A Base Nacional Comum Curricular (BNCC) se apresenta como um documento normativo que padroniza o ensino em todo o território brasileiro ao elencar competências e habilidades a serem desenvolvidas pelos alunos em diversas áreas do conhecimento, inclusive o Pensamento Computacional. Com o interesse de inovar nas práticas pedagógicas, esta pesquisa teve como principal objetivo desenvolver um *Game Design Document* (GDD) para um jogo mobile que estimule o desenvolvimento do pensamento computacional em alunos do Ensino Básico. De abordagem qualitativa, do tipo Estado da Arte, os procedimentos metodológicos incluíram ainda uma pesquisa bibliográfica para identificar conceitos discutidos por outros pesquisadores, a concepção da ideia do jogo, a elaboração do roteiro e a construção do GDD. Assim, esta pesquisa resultou na criação do Produto Final, um GDD que contém todos os elementos necessários para o desenvolvimento do jogo Logic Think, garantindo que os pilares do Pensamento Computacional sejam incorporados de maneira eficaz.

Palavras-chaves: *Game Design Document*. Pensamento Computacional. Jogos Educacionais. Base Nacional Comum Curricular.

ABSTRACT

The National Common Curricular Base (BNCC) presents itself as a normative document that standardizes education across Brazilian territory by listing competencies and skills to be developed by students in various areas of knowledge, including Computational Thinking. With the goal of innovating pedagogical practices, this research aims to develop a Game Design Document (GDD) for a mobile game that encourages the development of computational thinking in elementary school students. With a qualitative approach, of the State of the Art type, the methodological procedures also included a bibliographical research to identify concepts discussed by other researchers, the conception of the game idea, and the construction of the GDD. Thus, this research resulted in the creation of the Final Product, a GDD that contains all the necessary elements for the development of the game Logic Think, ensuring that the pillars of Computational Thinking are effectively incorporated.

Key-words: Game Design Document. Computational Thinking. Educational Games. Common National Curriculum Base.

LISTA DE ILUSTRAÇÃO

Figura 1- Decomposição no Pensamento Computacional	23
Figura 2- Sequência de padrões e subpadrões	24
Figura 3- Exemplo de Abstração	26
Figura 4- Exemplo de Algoritmos	28
Figura 5- Reconhecimento de Padrões de Imagens	33
Figura 6- Aprendendo Algoritmos com o Scratch	34
Figura 7- Telas do jogo AlgoBot	35
Figura 8- Tétrade Elemental	45
Figura 9- Evolução das plataformas de jogos digitais	46
Figura 10- Representação de um Game Design Document	50
Figura 11- Cenas do Jogo Logi Kingdom	51
Figura 12- Jogos para ensino de Pensamento Computacional: Hora do Código (à esquerda) e The Foos (à direita)	53
Figura 13- ENgAGED - Educational Games Development	59
Figura 14- Metodologia AMAEG	61
Figura 15- Processo Metodológico	62
Figura 16- Dica sobre o pilar da Decomposição	67
Figura 17- Dica sobre o pilar do Reconhecimento de Padrões	68
Figura 18- Dica sobre o pilar da Abstração	69
Figura 19- Dica sobre o pilar dos Algoritmos	70
Figura 20- Personagem Allan	73
Figura 21- Personagem Pixeisilicon	76
Figura 22- Aplicação da Dica sobre o pilar da Decomposição	76
Figura 23- Personagem Bitnomad	77
Figura 24- Aplicação das Dicas sobre o pilar Reconhecimento de Padrões	77
Figura 25- Personagem Vírus	78
Figura 26- Sistemas de aprimoramento	79
Figura 27- Sistemas de vidas	79
Figura 28- Primeiro cenário do jogo	81
Figura 29- Segundo cenário do jogo	81
Figura 30- Terceiro cenário do jogo.	82
Figura 31- Quarto cenário do jogo.	83

Figura 32- Quinto cenário do jogo.

84

Figura 33- Sexto cenário do jogo.

85

LISTA DE QUADROS

Quadro 1- Textos referenciados para a Metodologia Estado da Arte	57
Quadro 2- Requisitos funcionais e não-funcionais	64
Quadro 3- Mecânicas e ações do personagem	74
Quadro 4- Distribuição de Nível por Conteúdo	86

LISTA DE ABREVIATURAS E SIGLAS

AMAEG	Agille Methods Applied To Educacional Games
AVAMEC	Ambiente Virtual de Aprendizagem do Ministério da Educação
BBC	British Broadcasting Corporation
BIE	Buck Institute for Education
BNCC	Base Nacional Comum Curricular
ENgAGED	Educational Games Development
GBL	Game Based Learning
GDD	Game Design Document
GEDD	Game Educacional Design Document
ISO	International Standards Organization
ISTE	International Society for Technology in Education
NPCs	Non-Playable Character
PBL	Problem Based Learning
PPPGE	Programa de Pós-Graduação Profissional em Educação
RPG	Role-Playing Game
SBC	Sociedade Brasileira de Computação
TDIC	Tecnologias Digitais de Informação e Comunicação
TICs	Tecnologias de Informação e Comunicação
UFT	Universidade Federal do Tocantins
RV	Realidade Virtual

SUMÁRIO

SUMÁRIO.....	13
1 MEMORIAL DE FORMAÇÃO.....	14
2 INTRODUÇÃO.....	17
3 FUNDAMENTAÇÃO TEÓRICA.....	20
3.1 Pensamento Computacional.....	20
3.1.1 Pilares do Pensamento Computacional.....	22
3.1.2 O Pensamento Computacional na Educação Básica.....	29
3.1.3 Abordagens para o Ensino do Pensamento Computacional.....	32
3.2 Práticas Pedagógicas: Tecnologia e Inovação.....	36
3.2.1 Nativos Digitais.....	36
3.2.2 Metodologias Ativas de Aprendizagem.....	37
3.2.3 Inserção de Tecnologias nas Práticas Pedagógicas.....	39
3.2.4 Avaliação em Ambientes Tecnológicos.....	41
3.3 Jogos Digitais.....	42
3.3.1 Definição e Características dos Jogos.....	43
3.3.1.1 Estrutura dos Jogos Digitais.....	44
3.3.1.2 Gênero de Jogos Digitais.....	46
3.3.2 Game Design Document.....	48
3.3.3 Jogos Mobiles.....	51
3.3.4 Jogos no Auxílio do Pensamento Computacional.....	52
4 PROCESSO METODOLÓGICO.....	55
4.1 Abordagem e Tipo de Pesquisa.....	55
4.2 Metodologia Aplicada ao Produto Tecnológico.....	58
4.2.1 ENGAGED.....	58
4.2.2 AMAEG.....	60
4.2.3 Materiais e Métodos.....	62
5 RESULTADOS E DISCUSSÕES.....	64
5.1 Projeto do Jogo.....	64
5.2 Conteúdo de Aprendizagem.....	66
5.3 Concepção do Jogo.....	70
5.3.1 História do Jogo.....	71
5.3.2 Personagens.....	72
5.3.2.1 Personagem Allan.....	73
5.3.2.2 Personagem Pixeisilicon.....	75
5.3.2.3 Personagem Bitnomads.....	77
5.3.2.4 Personagem Inimigos.....	78
5.3.3 Sistemas do Jogo.....	79
5.3.4 Cenários e Design do Jogo.....	80
5.3.5 Desafios e Pontuações.....	85
5.4 Possibilidades de Aplicação.....	88
6 CONSIDERAÇÕES FINAIS.....	89
REFERÊNCIAS.....	92
APÊNDICE A - LOGIC THINK: Game Design Document.....	100

1 MEMORIAL DE FORMAÇÃO

Ingressar no Mestrado Profissional foi uma decisão tomada após concluir a graduação em Ciência da Computação, representando uma conquista significativa tanto para o meu aperfeiçoamento profissional quanto pessoal. O Programa de Pós-Graduação Profissional em Educação (PPPGE) da Universidade Federal do Tocantins (UFT) proporcionou a oportunidade de estudar diferentes perspectivas sobre a educação e suas aplicações práticas.

As disciplinas foram cursadas ao longo de dois semestres. No primeiro semestre, foram estudadas as disciplinas "Educação dos Povos Tradicionais" e "Prática Educativa e Relação com o Saber". No segundo semestre, as disciplinas "Gestão da Educação no Território Municipal", "Projeto de Qualificação do Trabalho Final" e "Saberes Docentes e Processos de Ensino" foram abordadas.

Na disciplina "Educação dos Povos Tradicionais", foi possível explorar a Política Nacional de Desenvolvimento Sustentável dos Povos e Comunidades Tradicionais com os professores Dr. Paulo Fernando Martins e Dr. Valtuir Soares. As discussões foram enriquecedoras, e, mesmo no contexto de aulas síncronas, os professores conduziram os diálogos com paciência e empatia ao longo da disciplina. Além disso, a disciplina proporcionou um entendimento aprofundado dos conceitos de Comunidade Tradicional e seu impacto nas políticas públicas contemporâneas. Conforme destaca Thum (2017, p. 165), “a produção dessa condição de Povo e Comunidade Tradicional tem contributos, tanto do mundo acadêmico como dos Movimentos Sociais e seus protagonistas”.

Na disciplina "Prática Educativa e Relação com o Saber", foram discutidas políticas públicas relacionadas ao trabalho didático-pedagógico nas escolas e à gestão do ambiente educacional, com ênfase nas etapas de elaboração de propostas pedagógicas. Os professores Dr. José Carlos da Silveira Freire e Dra. Dilsilene Maria Ayres de Santana apresentaram conceitos e definições sobre a educação e a relação com o saber, que envolvem a interação de um sujeito com o mundo ao seu redor, consigo mesmo e com os outros, ou seja, a relação com o tempo e o aprendizado. Segundo Charlot (2000) e Brandão (2007), tudo que é importante para o sujeito ou para o mundo ao seu redor existe como algum tipo de saber e se manifesta em algum modo de ensinar.

Na disciplina "Gestão da Educação no Território Municipal", ministrada pelo professor Dr. Roberto Francisco de Carvalho, foram abordados temas como o estudo da escola como local de trabalho e as tendências atuais da gestão municipal no Brasil. Além disso, foram discutidos aspectos da Gestão Democrática e a elaboração de projetos pedagógicos como eixo

de integração da gestão escolar. Dessa forma, foi possível estudar o Sistema Nacional de Educação alinhado com o Plano Nacional de Educação, que, conforme Saviani (2010), representa a unidade dos diversos aspectos e serviços educacionais mobilizados por determinados países, reunidos em um conjunto coerente que opera de forma eficaz no processo de educação da população do determinado país.

A disciplina "Projeto de Qualificação do Trabalho Final", ministrada pelos professores Dr. Valtuir Soares, Dr. Paulo Fernando e Dr. Damião Rocha, incluiu discussões sobre o projeto de trabalho final e a exposição das propostas de pré-projeto submetidas como parte da avaliação para ingresso no Programa de Pós-Graduação Profissional em Educação (PPPGE). Além disso, foi discutida a Portaria nº 60, de 20 de março de 2019, que dispõe sobre os mestrados e doutorados profissionais no âmbito da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) (Brasil, 2019). As discussões foram enriquecedoras e esclareceram dúvidas dos mestrandos sobre o funcionamento das avaliações e a importância do programa para o aprofundamento do conhecimento.

A disciplina "Saberes Docentes e Processos de Ensino", ministrada pelo professor Dr. José Carlos da Silveira Freire, incluiu diálogos sobre o cotidiano escolar e de sala de aula, além de como desenvolver estratégias de saberes docentes para atualizar e aperfeiçoar processos de ensino para diferentes públicos. Além disso, foram discutidas as tecnologias utilizadas no processo metodológico e como aplicá-las de maneira a enriquecer o conhecimento dos alunos. Outro ponto importante foi a discussão sobre a identidade docente, que, segundo Garcia, Hypolito e Vieira (2005), compreende as posições de sujeito atribuídas por diferentes discursos e agentes sociais no exercício de suas funções em contextos laborais concretos. Refere-se a um conjunto de representações colocadas pelos discursos relativos às funções desempenhadas em instituições educacionais.

Paralelamente aos estudos das disciplinas, o referencial do projeto de pesquisa era estudado em conversas e reuniões com o orientador, professor Dr. Gustavo Cunha de Araújo, que orientava nas melhorias e adaptações da redação e construção do produto final do programa. Além disso, em todas as disciplinas, foram produzidos diversos trabalhos acadêmicos, como artigos e memoriais, o que resultou em crescimento, evolução e consolidação das práticas de pesquisa.

Alguns dos artigos resultantes das disciplinas foram publicados em revistas e simpósios. Entre eles, destaca-se "Uma Revisão Sistemática dos Benefícios da Gamificação no Ensino Básico", publicado na Revista Singular. Sociais e Humanidades, disponível em: <http://ulbra-to.br/singular/index.php/SingularSH/article/view/227>. O artigo apresenta uma

Revisão Sistemática da Literatura que explora os benefícios e as técnicas de gamificação aplicáveis ao ensino básico. Além disso, foi publicado o artigo "Terraverde: um jogo para conscientização ambiental", apresentado no XXIII Simpósio Brasileiro de Jogos e Entretenimento Digital realizado na cidade de Manaus, disponível em: <https://sbgames.org/sbgames2024/>.

As diversas disciplinas cursadas, os artigos publicados, e as discussões com orientadores e colegas proporcionaram um aprofundamento teórico e prático valioso no que tange a educação. O conhecimento adquirido ao longo do programa ampliou a compreensão das complexidades da educação e suas diversas facetas, além de fornecer ferramentas para enfrentar os desafios do campo educacional de maneira inovadora.

O impacto deste aprendizado reflete-se na capacidade de analisar criticamente questões educacionais, e também na habilidade de aplicar métodos e estratégias que promovam o desenvolvimento e a conscientização em diferentes contextos educacionais. Dessa forma, o Mestrado Profissional em Educação revelou-se uma etapa fundamental e transformadora, que continuará a influenciar positivamente a carreira e a atuação no campo da educação.

2 INTRODUÇÃO

A educação está em constante transformação, adaptando-se aos novos cenários sociais, tecnológicos, políticos e econômicos. Essas mudanças refletem a urgência de preparar os estudantes para um mundo cada vez mais globalizado e interconectado, onde as tecnologias permeiam diversas áreas do conhecimento. Nesse contexto, metodologias ativas ganham destaque por sua flexibilidade e capacidade de atender às necessidades individuais dos alunos, conforme defendido por autores como Jean Piaget (2010) e Lev Vygotsky (2009), que enfatizam a importância da interação e do aprendizado ativo.

Integrar tecnologias digitais em ambientes educacionais, como salas de aula, pode contribuir na adaptação do ensino, criando novas formas de ensinar e de aprender. Para isso, é necessário desenvolver estratégias que utilizem tecnologias aplicadas a diferentes níveis de conhecimento para diferentes grupos de alunos. Uma das estratégias de integrar a tecnologia na educação é inseri-la através de disciplinas que envolvam raciocínio lógico e o Pensamento Computacional.

A Base Nacional Comum Curricular (BNCC) engloba o Pensamento Computacional como uma habilidade transversal a ser desenvolvida pelos alunos do ensino básico em todas as áreas do conhecimento (Brasil, 2018). O objetivo de inserir o Pensamento Computacional no âmbito escolar é fazer com que os alunos possam adquirir competências e habilidades que estão relacionadas ao uso crítico, ético e consciente da tecnologia (Barcelos e Silveira, 2013). No entanto, existem vários conceitos, abordagens e práticas de aplicação do Pensamento Computacional que são difíceis de serem aprendidas em sala de aula.

Algumas estratégias para o ensino do Pensamento Computacional estão relacionadas à adoção de metodologias lúdicas, que, segundo Moreno *et al.* (2021), podem ser utilizadas para motivar o processo de ensino e aprendizagem. Recursos tecnológicos podem ser usados como ferramentas para engajar os alunos em disciplinas que exigem tempo de dedicação e possuem conceitos difíceis de serem abstraídos.

Conforme destacado por Klein *et al.* (2020), a integração de tecnologia no âmbito educacional visa instigar os alunos a explorarem mudanças e evoluções na prática de aprendizagem. Quando o professor incorpora recursos tecnológicos na entrega de conteúdo, ele não apenas facilita a transmissão de conhecimento, mas também introduz novas ferramentas de aprendizagem. Entre essas ferramentas, os jogos educacionais se destacam por seu potencial de engajamento. Eles tornam o processo de aprendizagem mais atraente e

interativo, além de ajudar a desenvolver diversas habilidades, como o Pensamento Computacional, a resolução de problemas e a colaboração entre alunos.

Para Kiya (2014) e Resende (2013), os jogos são atividades lúdicas que podem ser utilizadas como recursos metodológicos para aprimorar o aprendizado, tornando o trabalho educacional mais dinâmico. Nesse contexto, os jogos não servem apenas para fins de entretenimento, mas também como ferramentas educacionais em escolas e instituições de ensino, facilitando a transmissão de conhecimentos e estimulando as habilidades dos alunos.

Os jogos podem ser utilizados como ferramentas motivacionais para o ensino e a aprendizagem de diversas disciplinas, tanto no ensino básico quanto na graduação (Moreno, 2022). Corroborando isso, Barbosa (2014) argumenta que os jogos são ferramentas promissoras que, em conjunto com metodologias ativas, podem ser utilizados para o ensino de disciplinas que exigem pensamento lógico. O jogo pode motivar pessoas de diferentes faixas etárias, envolvendo-as significativamente em diversas atividades.

Para despertar o efeito motivador que os jogos proporcionam, é essencial planejar e documentar o processo criativo. Uma das técnicas utilizadas para auxiliar na construção de um jogo é o *Game Design Document* (GDD). Segundo Rogers (2013), o GDD é um documento que descreve de forma precisa o que o jogo deve conter, incluindo informações técnicas, detalhes sobre os personagens e outras informações necessárias para a equipe, servindo como uma ferramenta para facilitar a comunicação entre os membros da equipe de desenvolvimento.

A Base Nacional Comum Curricular (BNCC) enfatiza a importância da utilização de tecnologias como parte essencial para o desenvolvimento do aprendizado dos alunos. Nesse sentido, de que forma pode ser criada uma estratégia de ensino dos pilares do Pensamento Computacional através de um GDD para a educação básica? A busca por responder a essa indagação é a motivação para o desenvolvimento desta pesquisa, que visa elaborar uma estratégia de ensino inovadora.

Consequentemente, como objetivo geral, busca-se 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, de tal forma que os jogadores (estudantes) aprendam enquanto se divertem. Para ajudar a alcançar esse objetivo, foram estabelecidos os seguintes objetivos específicos: identificar as características essenciais a serem incluídas em um GDD para apoiar o ensino do Pensamento Computacional na Educação Básica; determinar as características pedagógicas necessárias para incorporar os pilares do Pensamento Computacional em um jogo educacional; analisar

metodologias de desenvolvimento de jogos educacionais visando a construção de um jogo *mobile* com foco nos pilares do Pensamento Computacional; e examinar como diferentes abordagens de ensino impactam a compreensão e aplicação dos pilares do Pensamento Computacional por alunos da Educação Básica.

A partir dessas considerações, esta dissertação está estruturada nas seguintes seções: primeiramente, é apresentado o memorial de formação, que consistiu em detalhar o histórico de formação no Mestrado Profissional em Educação, destacando as contribuições deste programa para o desenvolvimento profissional como educador e pesquisador. Em seguida, são apresentadas revisões bibliográficas sobre o Pensamento Computacional, Práticas Pedagógicas e Jogos Digitais, importantes para contextualizar o objeto de estudo desta investigação. Logo após, são apresentados os procedimentos metodológicos desta pesquisa, que consistiram na adaptação das metodologias de Battistella (2016) e Silva e Farias (2019), buscando esclarecer como este estudo foi organizado, planejado e executado. Na sequência, são apresentados os resultados e discussões gerados pela pesquisa, que consistiu na elaboração de um GDD. Por fim, são apresentadas as conclusões do estudo e o Produto Final desta pesquisa, intitulado: *LOGIC THINK: Game Design Document*.

3 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, será apresentada a Fundamentação Teórica que embasa esta pesquisa, abordando os principais conceitos e estudos relacionados ao desenvolvimento do Pensamento Computacional, práticas pedagógicas com foco em inovação tecnológica, e o uso de jogos digitais como ferramentas educacionais. Neste contexto, a fundamentação teórica irá explorar os pilares do Pensamento Computacional, suas aplicações na Educação Básica e as metodologias pedagógicas ativas que podem ser integradas com o uso de tecnologias.

3.1 Pensamento Computacional

Esta subseção aborda o Pensamento Computacional, apresentando inicialmente seus pilares fundamentais: decomposição, reconhecimento de padrões, abstração e design de algoritmos. Em seguida, discute-se a aplicação do Pensamento Computacional na Educação Básica, destacando sua relevância para o desenvolvimento de habilidades cognitivas essenciais. Por fim, são exploradas diferentes abordagens para o ensino do Pensamento Computacional, com ênfase em estratégias que facilitam sua inserção no ambiente escolar e promovem o engajamento dos alunos.

A Base Nacional Comum Curricular (BNCC) é um documento normativo que estabelece os conhecimentos indispensáveis aos quais os estudantes têm o direito assegurado de estudar. Além de determinar os conteúdos essenciais, a BNCC também delineia abordagens a serem empregadas nas disciplinas, visando fomentar um percurso pedagogicamente enriquecedor e alinhado com os objetivos educacionais do país (Brasil, 2018). A BNCC norteia e modela políticas e iniciativas pedagógicas nas escolas brasileiras, proporcionando, uma integração mais profunda e coesa entre as instituições de ensino.

A BNCC incorpora abordagens estratégicas para a integração de recursos tecnológicos, com o objetivo de ampliar o engajamento dos alunos nas atividades educacionais. As competências vinculadas à proficiência tecnológica desempenham um papel fundamental ao estimular a construção de conhecimento e a resolução de desafios, empoderando os alunos como protagonistas e autores no contexto social e coletivo de aprendizado (Brasil, 2018).

A utilização de recursos tecnológicos para auxílio pedagógico já é uma realidade em escolas de todo o mundo, devido ao grande crescimento da tecnologia na sociedade em geral. A BNCC apoia o desenvolvimento de habilidades relacionadas ao uso responsável de tecnologias digitais com o objetivo de despertar curiosidade para o aprendizado, desenvolver

competências de compreensão sobre determinados conteúdos, aprimorar a criatividade e motivar práticas sociais por meio do Pensamento Computacional (Brasil, 2018).

O Pensamento Computacional é uma forma de abordar situações e desafios, inspirada pelos processos de resolução de problemas utilizados na ciência da computação (Wing, 2006). Envolve a capacidade de formular problemas de maneira computacional, pensar de forma algorítmica, analisar e decompor problemas complexos em partes menores, além de utilizar abstrações e padrões para desenvolver soluções reais.

O uso da tecnologia na educação básica requer o desenvolvimento do pensamento crítico e lógico dos alunos, além da capacidade de reconhecer padrões e raciocinar por meio de algoritmos. Segundo Liukas (2015, p. 111), “o Pensamento Computacional é algo que as pessoas fazem, não os computadores. Inclui o pensamento lógico e a capacidade de reconhecer padrões, pensar com algoritmos, decompor um problema e abstrair um problema”.

Para Wing (2016, p. 2),

Pensamento Computacional envolve a resolução de problemas, projeção de sistemas, e compreensão do comportamento humano, através da extração de conceitos fundamentais da ciência da computação. O Pensamento Computacional inclui uma série de ferramentas mentais que refletem a vastidão do campo da ciência da computação. [...] Ao resolver um problema eficientemente, podemos questionar se uma solução aproximada é boa o suficiente e se falsos positivos ou falsos negativos são permitidos. O Pensamento Computacional é reformular um problema aparentemente difícil em um problema que sabemos como resolver, talvez por redução, incorporação, transformação ou simulação.

Segundo a *International Society for Technology in Education* - ISTE (2016), o Pensamento Computacional pode ser definido como uma forma de desenvolver estratégias para entender melhor um determinado problema e usar o 'poder' da tecnologia para implementar e testar soluções. Kurshan (2016) argumenta que o Pensamento Computacional é uma abordagem criativa, crítica e estratégica que pode ser compreendida por meio de conceitos computacionais em uma ampla gama de domínios do conhecimento. Essa abordagem visa resolver problemas por meio de passos sequenciais, de forma que pessoas ou máquinas possam executá-los corretamente.

A proposta de desenvolver o Pensamento Computacional em alunos do ensino básico não busca transformá-los em máquinas de pensar, mas sim incentivá-los a aplicar a inteligência, os fundamentos e os recursos oferecidos pela computação para enfrentar desafios e solucionar problemas (Ribeiro; Foss; Cavalheiro, 2020).

Segundo Mendes *et al.* (2020), o Pensamento Computacional é uma habilidade importante, uma vez que oferece uma abordagem eficaz para a resolução de problemas, projetar sistemas e auxiliar na compreensão do comportamento humano. Ao desenvolver essa habilidade, os alunos podem adquirir uma compreensão dos fundamentos da computação, resultando em um conjunto de ferramentas que vão além da programação de computadores.

Wing (2006), argumenta que o Pensamento Computacional é uma habilidade essencial para todos, independentemente da área de atuação. Ela sugere que o Pensamento Computacional pode ser desenvolvido através do ensino de conceitos e práticas da ciência da computação desde os estágios iniciais da educação básica. Dessa forma, abordar o Pensamento Computacional no ensino básico pode auxiliar os alunos a desenvolverem um pensamento mais crítico e resolver problemas de forma mais eficiente.

No entanto, de acordo com Mandaji *et al.* (2018), é fundamental a formação de professores para trabalhar o Pensamento Computacional, sendo essa uma formação específica voltada para o desenvolvimento de competências e habilidades que ampliem as possibilidades de aplicação em sala de aula. Dessa forma, o processo de formação pode ajudar a sanar dúvidas e questionamentos dos professores, garantindo que eles se sintam preparados para integrar essas práticas em suas atividades pedagógicas.

O processo de formação docente, para o ensino do Pensamento Computacional, para ser eficaz, deve integrar atividades práticas com conteúdos que fundamentam o conhecimento teórico. Além disso, é essencial estimular as relações entre alunos e professores, eliminando mitos relacionados ao distanciamento entre ambos. Com isso, a formação promove um ambiente de trabalho colaborativo, no qual os professores podem desenvolver atividades simples e criar diálogos de integração (Mandaji *et al.*, 2018).

3.1.1 Pilares do Pensamento Computacional

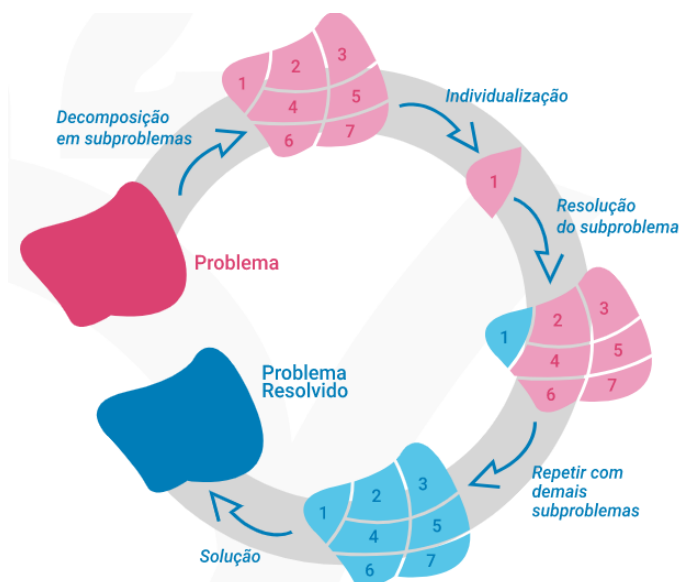
As características do Pensamento Computacional estão ligadas aos princípios fundamentais da ciência da computação, possibilitando a análise e a resolução de problemas complexos. O Pensamento Computacional, abordado por Code.Org (2016), Brackmann (2017), BBC Learning (2015) e Liukas (2015), é a habilidade de abordar e solucionar problemas de forma eficiente, desde tarefas cotidianas até desafios mais complexos. Essa abordagem se sustenta em quatro pilares principais, que são: **Decomposição**; **Reconhecimento de padrões**; **Abstração**; **Algoritmos**.

Além dos quatro pilares mencionados anteriormente, é relevante destacar as valiosas contribuições de renomados autores como Seymour Papert, Jeanette Wing, Mitchel Resnick, Grover e Pea, que enriqueceram o campo com suas perspectivas adicionais. Suas ideias e percepções foram fundamentais para o desenvolvimento e a compreensão do Pensamento Computacional.

Nesta pesquisa, o foco será dado aos quatro pilares mencionados anteriormente, os quais foram amplamente reconhecidos e estabelecidos como base para o Pensamento Computacional. Isso permitirá explorar e analisar de forma aprofundada cada um desses pilares, compreendendo sua relevância e aplicação no contexto educacional e no desenvolvimento de habilidades essenciais para os alunos do ensino básico.

O primeiro pilar do Pensamento Computacional, conhecido como “Decomposição”, refere-se à habilidade de dividir um problema em partes menores e gerenciáveis. Essa estratégia permite uma compreensão mais clara e uma abordagem mais eficaz para solucionar cada componente individualmente (Brackmann, 2017). A Figura 1 ilustra um exemplo prático de como a decomposição de problemas pode ser aplicada.

Figura 1 - Decomposição no Pensamento Computacional



Fonte: Ambiente Virtual de Aprendizagem do Ministério da Educação (AVAMEC, 2022).

A Figura 1, disponível no curso "Introdução ao Pensamento Computacional" no Ambiente Virtual de Aprendizagem do Ministério da Educação (AVAMEC, 2022), ilustra de maneira clara o processo de decomposição de problemas. Na figura, um problema inicial é dividido em partes menores, e em seguida, cada subproblema é tratado individualmente. Após

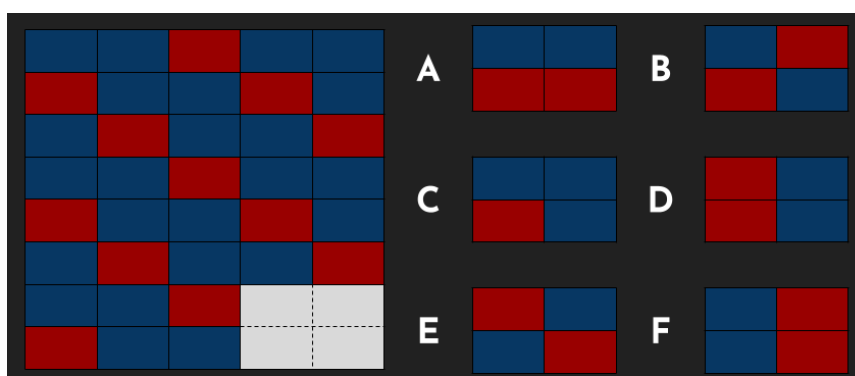
a resolução de um subproblema, o próximo é selecionado para ser abordado, seguindo esse ciclo até que todos os subproblemas sejam solucionados. Com essa abordagem, é possível resolver o problema maior de forma eficiente e sistemática.

O segundo pilar do Pensamento Computacional é o “Reconhecimento de padrões”. Essa habilidade envolve a capacidade de identificar tendências, estruturas e relações presentes em um problema. Ao reconhecer padrões, é possível aplicar soluções conhecidas ou desenvolver novas estratégias para lidar com problemas semelhantes que o aluno já presenciou ou resolveu. Essa competência é fundamental tanto na resolução de problemas computacionais quanto em diversas outras áreas do conhecimento, permitindo uma análise mais eficiente das situações enfrentadas (Brackmann, 2017).

Segundo Vicari *et al.* (2018), o Reconhecimento de Padrões é uma abordagem eficiente para resolução de problemas, pois utiliza soluções previamente estabelecidas em situações similares com base em experiências anteriores. Para os autores, a identificação de padrões pode ser de grande ajuda na definição dos dados a serem estudados, no processamento das informações e no desenvolvimento de estratégias para resolver um problema.

Para Vicari *et al.* (2018), no contexto educacional, uma maneira de promover o reconhecimento de padrões em sala de aula é incentivar perguntas como: (1) Este problema se assemelha a algum outro que já foi abordado? (2) O problema atual difere de algum que já tenha sido visto anteriormente? Essas perguntas estimulam a reflexão dos alunos e auxiliam conexões e analogias entre problemas, o que facilita a aplicação de soluções e o desenvolvimento de pensamento crítico (Vicari *et al.* 2018). A Figura 2 a seguir apresenta um exemplo básico de reconhecimento de padrões.

Figura 2 - Sequência de padrões e subpadrões



Fonte: *Computational Think* (Google, 2024) (Adaptado).

A Figura 2 representa uma atividade com o propósito de desenvolver a habilidade de reconhecimento de padrões em sequências de cores. Nessa figura, as sequências de cores foram decompostas em subpadrões identificados como A, B, C, D, E e F. Essa representação visual permite uma compreensão mais clara e facilita a identificação dos elementos que compõem a sequência, estimulando o desenvolvimento do reconhecimento de padrões pelos participantes.

De acordo com Vicari *et al.* (2018), o reconhecimento de padrões oferece a capacidade de simplificar um problema e replicar soluções em subproblemas semelhantes. Essa abordagem permite uma resolução mais eficiente, uma vez que, ao identificar padrões, é possível aplicar soluções já conhecidas. Quanto mais padrões forem reconhecidos, mais rapidamente a solução geral do problema pode ser alcançada.

No contexto do Pensamento Computacional, a “abstração” emerge como o terceiro pilar, desempenhando um papel na resolução de problemas complexos. A habilidade de filtrar informações irrelevantes e focar apenas nos aspectos essenciais permite aos indivíduos criar representações ou ideias simplificadas do que está sendo abordado. Por meio da abstração, é possível simplificar a complexidade, concentrando-se nos elementos-chave para a resolução de problemas complexos (Brackmann, 2017; Santos *et al.*, 2021).

O pensamento abstrato é uma habilidade cognitiva importante do ser humano e que aplicamos a todo momento, sem perceber. Criamos abstrações, ou simplificações, para tudo que está ao nosso redor, desde conceitos simples até artefatos complexos. Por exemplo, pense na palavra “número”: só de ouvir esse termo você já sabe do que se trata, mesmo sem ouvir outros detalhes. Os números são abstrações para representar a quantificação de algo (Santos *et al.*, 2021, p. 111).

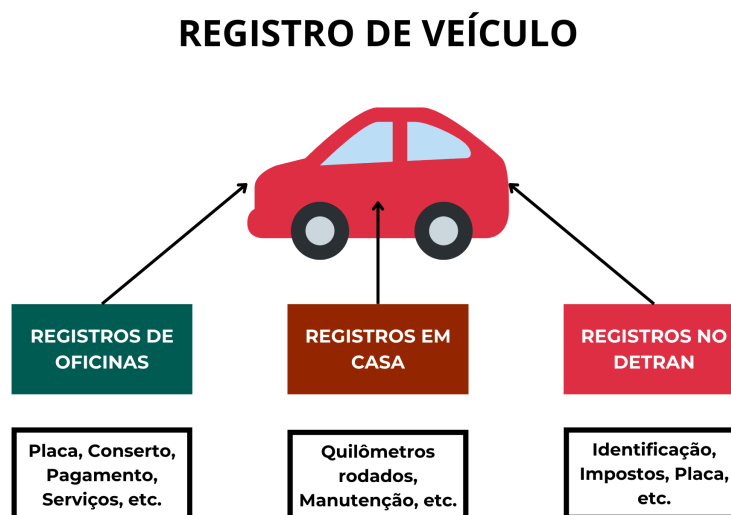
No entanto, para Wing (2006), a abstração atua em diversas áreas relacionadas ao Pensamento Computacional. Alguns exemplos de sua aplicação citados pelo autor incluem:

- **Escrita de algoritmos e suas iterações:** ao desenvolver algoritmos, é necessário abstrair os detalhes de implementação e focar nos passos essenciais para resolver o problema.
- **Seleção dos dados importantes:** ao lidar com grandes quantidades de dados, é necessário abstrair e identificar os dados relevantes para a análise ou processamento.
- **Escrita de uma pergunta:** ao formular perguntas ou problemas, é importante abstrair e expressar apenas as informações relevantes para obter uma resposta.

- **Indivíduo em relação a um robô:** ao projetar e interagir com sistemas robóticos ou inteligentes, é necessário abstrair as características essenciais de um indivíduo para criar uma representação simplificada.
- **Organização de módulos em um sistema:** ao desenvolver sistemas complexos, é necessário abstrair as funcionalidades e componentes essenciais, organizando-os em módulos ou blocos lógicos.

Por outro lado, Santos *et al.* (2021, p. 58) argumentam que na abstração “o desafio é selecionar os detalhes que facilitem a compreensão do problema sem perder informações que possam ser importantes para a resolução”. A falta de detalhes e informações claras pode levar a uma compreensão limitada do problema, dificultando a identificação dos elementos essenciais e das relações entre eles (Santos *et al.*, 2021). Nesse sentido, a abstração se mostra ainda mais importante, sendo necessário desenvolver a capacidade de extrair significado e identificar padrões mesmo diante de informações escassas.

Figura 3 - Exemplo de Abstração



Fonte: Autor.

Na Figura 3, pode-se observar uma representação abstrata de um carro em diferentes contextos. A abstração no contexto de registros de um carro pode ser categorizada em três tipos principais: registros em oficinas, registros em casa e registros no Detran. Os registros em oficinas englobam serviços de manutenção e reparos, os registros em casa incluem informações sobre uso e conservação do veículo, e os registros no Detran abrangem aspectos legais e administrativos. Essa abstração dos registros permite uma visão simplificada e

organizada, facilitando o gerenciamento das informações relacionadas ao carro, tanto para os proprietários quanto para os profissionais da área automotiva.

Segundo Ribeiro, Foss e Cavalheiro (2020), a abstração é um processo de resolução de problemas que permite simplificar a realidade e focar nos aspectos mais relevantes de um determinado contexto. A abstração de dados envolve descrever as informações envolvidas na solução do problema, como os dados de entrada e saída.

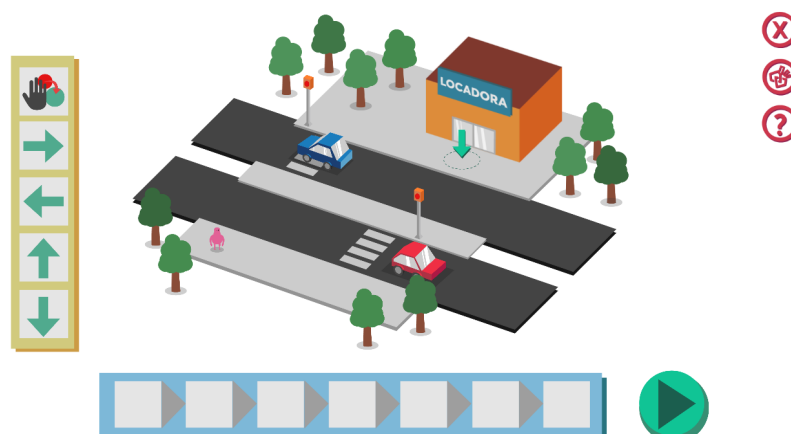
O quarto pilar do Pensamento Computacional, denominado "Algoritmos", representa a capacidade de expressar soluções de problemas por meio de uma sequência clara e bem definida de passos. Essa habilidade permite alcançar a resposta mais eficiente e eficaz possível. Ao desenvolver algoritmos, é essencial considerar a lógica, a ordem e a precisão dos passos, garantindo que o problema seja abordado de maneira sistemática (Vicari *et al.*, 2018). Ribeiro, Foss e Cavalheiro (2020, p. 24) complementam que,

Um algoritmo é composto por instruções que devem ser executadas de uma forma e na ordem definida para atingir a solução desejada. Portanto, para definir um algoritmo, é necessário saber quais instruções básicas podem ser utilizadas e quais operações podem ser usadas para montar descrições dos procedimentos a partir dessas instruções básicas.

De acordo com Cormen (2014) e Torres *et al.* (2019), a criação de algoritmos segue uma sequência estruturada. O primeiro passo é descrever com precisão o problema ou a tarefa a ser realizada, frequentemente resumida no título do algoritmo. O segundo passo consiste em decompor a tarefa em todas as ações necessárias para sua execução, listando todos os itens essenciais. No terceiro passo, as ações devem ser ordenadas de forma lógica e sequencial para garantir que o problema ou tarefa seja resolvido com sucesso. O quarto passo envolve a conclusão e avaliação do algoritmo, testando cada instrução conforme a ordem estabelecida para verificar se o objetivo foi alcançado e se a tarefa foi completada com sucesso.

Conforme Papert (1980) destaca, os algoritmos proporcionam uma estrutura lógica que possibilita a resolução sistemática de problemas, automatização de processos e execução eficiente de tarefas. Ao compreender e aplicar algoritmos, os indivíduos adquirem a capacidade de abordar desafios de forma estruturada, identificar padrões e desenvolver soluções passo a passo, conforme o exemplo apresentado na Figura 4.

Figura 4 - Exemplo de Algoritmos



Fonte: Ambiente Virtual de Aprendizagem do Ministério da Educação (MEC, 2022).

A Figura 4, disponível no curso "Introdução ao Pensamento Computacional" no Ambiente Virtual de Aprendizagem do Ministério da Educação (AVAMEC, 2022), ilustra um exemplo de algoritmos no dia-a-dia. A figura retrata um personagem com o objetivo de chegar à locadora, sendo necessário atravessar a rua. As setas representam as direções disponíveis para o personagem se movimentar. Nesse contexto, é possível montar um algoritmo que descreve a sequência de passos necessários para que o personagem atravesse a rua com sucesso e alcance a locadora.

Atravessar a rua pode parecer uma tarefa simples, mas envolve uma sequência de passos bem definidos, ou seja, um algoritmo. Como mencionado por Cormen *et al.* (2012), um algoritmo é uma sequência de passos computacionais que transforma a entrada em uma saída desejada. Ele fornece uma estrutura organizada e lógica para a resolução de problemas, permitindo a decomposição do problema em etapas menores e mais gerenciáveis.

O foco do Pensamento Computacional não está apenas nos resultados finais, como provas ou algoritmos, mas sim no processo de criação desses resultados. Isso significa que, além das abstrações utilizadas para descrever algoritmos, o Pensamento Computacional também engloba técnicas de construção de algoritmos, que são, na verdade, técnicas de solução de problemas aplicáveis a diversos contextos (Brackmann, 2017; Wing, 2006; Santos *et al.*, 2021).

Nesse sentido, a partir da utilização do Pensamento Computacional, pode-se obter um poder de raciocínio muito mais apurado e auxiliar os alunos na resolução de problemas. Segundo Brackmann (2017), a utilização do Pensamento Computacional também pode favorecer na busca de empregos que requerem conhecimento em programação, na

compreensão do mundo que está caminhando para uma nova era tecnológica e na transversalidade de diferentes áreas do conhecimento, visto que a computação está presente na vida da maioria das pessoas.

3.1.2 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 aborda diversas propostas ao longo da Educação Básica, incluindo as “**Dez Competências Gerais**”, que englobam tanto aspectos cognitivos quanto socioemocionais. Essas competências visam promover uma formação integral dos estudantes, abrangendo habilidades como o estímulo à curiosidade intelectual, o uso das tecnologias digitais de comunicação e a valorização da diversidade.

Vicari *et al.*, (2018, p. 82) destaca uma série de competências que estão alinhadas com o Pensamento Computacional, incluindo:

Valorizar e utilizar os conhecimentos historicamente construídos sobre o mundo físico, social, cultural e digital para entender e explicar a realidade, continuar aprendendo e colaborar para a construção de uma sociedade justa, democrática e inclusiva;
 Utilizar diferentes linguagens – verbal, corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo;
 Compreender, utilizar e criar tecnologias digitais de informação e comunicação (TDIC) de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria 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.

A SBC (2017) destaca as habilidades do Pensamento Computacional no Ensino Básico, que incluem:

- **Na 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.

- **No 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.

- **No 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.

As habilidades destacadas pela SBC (2017) revelam que o Pensamento Computacional engloba uma variedade de conhecimentos que podem enriquecer a formação dos alunos em todos os níveis de ensino. Dessa forma, fica evidente que a introdução do Pensamento Computacional nas escolas é capaz de promover e aprimorar a resolução de problemas de forma eficiente quando as habilidades são trabalhadas nos diferentes níveis de escolaridade.

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, 2018; SBC, 2017).

3.1.3 Abordagens para o Ensino do Pensamento Computacional

O Pensamento Computacional tornou-se uma habilidade necessária para enfrentar os desafios de um mundo em constante evolução. Essa habilidade não está restrita apenas aos profissionais de tecnologia, mas também é relevante em diversas áreas do conhecimento. Para capacitar os estudantes para o futuro e promover sua capacidade de pensar criticamente e solucionar problemas, educadores e pesquisadores têm explorado diferentes estratégias pedagógicas e recursos educacionais que visam desenvolver o Pensamento Computacional de maneira acessível e engajadora.

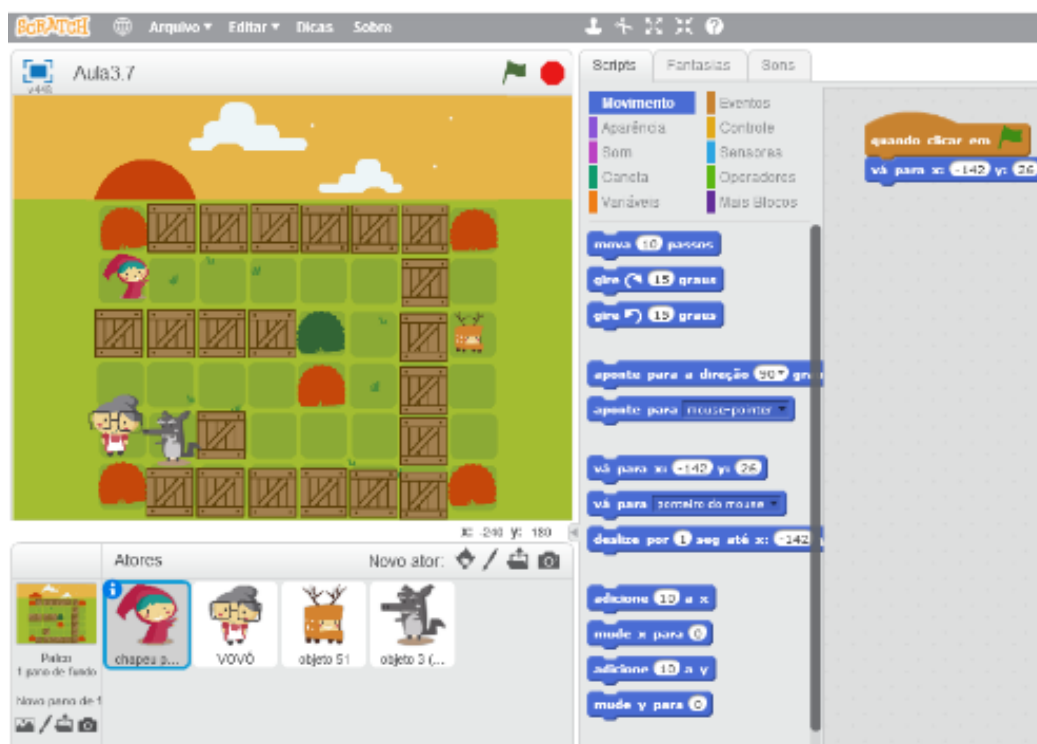
Vicari *et al.* (2018), destacaram a importância de estudos sobre a utilização do Pensamento Computacional no contexto educacional, considerando diferentes países com realidades socioeconômicas distintas. Nesse sentido, Valente (2016) realizou um estudo e identificou seis categorias de abordagens para o Ensino do Pensamento Computacional. São elas:

1. **Atividades sem auxílio de tecnologia:** ao adotar abordagens lúdicas, truques e competições como parte do ensino do Pensamento Computacional, os educadores podem despertar o interesse das crianças, desenvolver suas habilidades cognitivas e promover a compreensão dos princípios e conceitos dessa área do conhecimento.
2. **Programação em Scratch:** uma linguagem de programação baseada em blocos visuais, desenvolvida com o objetivo de tornar a manipulação da mídia acessível e facilitada para programadores novatos (Zanetti *et al.*, 2017).
3. **Robótica pedagógica:** uma abordagem que busca incorporar aspectos e práticas da robótica industrial em um ambiente educacional, permitindo aos alunos vivências práticas de aprendizagem com dispositivos robóticos
4. **Produção de narrativas digitais:** aproveita as TICs para transformar histórias tradicionais em experiências interativas. Essa abordagem oferece

Na Figura 5, é apresentada uma atividade que tem como objetivo auxiliar as crianças na identificação de padrões de figuras para completar um papel de parede. O personagem Snow Leopard está decorando sua casa e precisa da ajuda das crianças para encontrar a sequência correta e preencher o papel de parede de forma elegante. Nessa atividade, as crianças terão a oportunidade de exercitar sua capacidade de identificar padrões visuais e aplicar seu conhecimento para criar uma sequência coerente e esteticamente agradável no papel de parede (Liukas, 2015).

O Pensamento Computacional também pode ser desenvolvido por meio da utilização de programas que estimulem o aprendizado sobre algoritmos, especialmente para crianças. Uma ferramenta amplamente reconhecida nesse contexto é o Scratch, que proporciona um ambiente interativo para a criação de projetos de programação. Zanetti *et al.* (2017), apresentam uma abordagem para o ensino do algoritmo no Pensamento Computacional.

Figura 6 - Aprendendo Algoritmos com o Scratch



Fonte: Zanetti *et al.*, (2017).

A Figura 6 ilustra a oficina 'Era uma vez', que propõe uma série de desafios aos participantes. Eles devem orientar a personagem Chapeuzinho por labirintos para encontrar a Vovó, evitando o Lobo Mau, por meio da aplicação de algoritmos. Alguns labirintos incluem o personagem Cervo, que concede pontos extras. Os labirintos são construídos com elementos

como caixas e arbustos, e os alunos devem criar algoritmos para cada labirinto. A dificuldade dos labirintos aumenta progressivamente à medida que os alunos avançam e resolvem os desafios (Zanetti, 2017).

Os jogos digitais também desempenham um papel importante no ensino do Pensamento Computacional. Higuchi *et al.* (2021), apresentam o desenvolvimento de um jogo chamado AlgoBot, que tem como objetivo específico desenvolver habilidades de Pensamento Computacional em crianças. AlgoBot é um jogo do gênero quebra-cabeça, onde os jogadores são desafiados a resolver problemas relacionados à programação. O jogo apresenta uma série de desafios que envolvem a aplicação de conceitos como sequenciamento, repetição, condicionais e decomposição de problemas.

Figura 7 - Telas do jogo AlgoBot



Fonte: Higuchi *et al.*, (2021).

A Figura 7 ilustra algumas telas do AlgoBot, mostrando a interface do jogo e exemplos de desafios que os jogadores encontrarão. Quando o jogador coloca um bloco em um espaço disponível, um novo espaço é criado ao lado ou na próxima linha, indicando a ordem na qual os comandos serão reproduzidos (Higuchi *et al.*, 2021). Essa abordagem visual e interativa proporciona uma representação clara e intuitiva do fluxo de comandos e ações no jogo, no qual os alunos podem explorar, experimentar e aprimorar suas habilidades de resolução de problemas e raciocínio lógico.

Outras abordagens incluem o uso de simulações para a compreensão de fenômenos complexos, a realização de atividades lúdicas que estimulem o reconhecimento de padrões e a resolução de quebra-cabeças, além do trabalho com desafios baseados em problemas reais que exigem a aplicação do Pensamento Computacional para encontrar soluções eficientes. Essas abordagens refletem a diversidade de estratégias utilizadas para promover o ensino do Pensamento Computacional, despertando o interesse dos alunos, desenvolvendo suas habilidades cognitivas e preparando-os para os desafios do mundo digital.

Para aprimorar a aprendizagem dos princípios do Pensamento Computacional, é necessário incorporar ferramentas e abordagens vinculadas a uma metodologia pedagógica na qual o professor desempenhe um papel de acompanhamento. Na próxima seção, serão apresentadas diferentes práticas pedagógicas que integram a tecnologia de maneira eficaz.

3.2 Práticas Pedagógicas: Tecnologia e Inovação

A presente subseção tem como propósito explorar as práticas pedagógicas e as tecnologias no contexto educacional, com um enfoque particular na avaliação do Pensamento Computacional e nas inovações tecnológicas. Essas inovações têm se mostrado benéficas para as práticas pedagógicas quando incorporadas a metodologias ativas. Além disso, a seção aborda a integração de tecnologias em sala de aula e estratégias para avaliar o conhecimento dos alunos.

3.2.1 Nativos Digitais

Com o avanço das tecnologias, gerações de crianças, adolescentes, jovens e adultos já nasceram em um ambiente onde a tecnologia é amplamente presente e considerada avançada por muitos. Isso resultou em uma capacidade natural de lidar com dispositivos eletrônicos e incorporar essas ferramentas em suas vidas diárias. Contudo, torna-se imperativo empregar os recursos tecnológicos de maneira eficiente, especialmente ao lidar com crianças que estão no processo de aquisição da alfabetização e no desenvolvimento de habilidades de interpretação de problemas.

Segundo Prensky (2001), o termo "Nativos Digitais" refere-se à geração de indivíduos que cresceu imersa em um ambiente permeado por tecnologias digitais, exibindo notável proficiência e fluência natural no uso de dispositivos eletrônicos, como computadores, smartphones e a internet. O conceito ganhou popularidade através do artigo "*Digital Natives, Digital Immigrants*", escrito por Marc Prensky e publicado em 2001. Nesse mesmo artigo, Prensky também introduziu o termo 'Imigrantes Digitais' para descrever aqueles que nasceram antes da era da tecnologia digital e, conseqüentemente, tiveram de se adaptar a essas inovações.

Os nativos digitais apresentam uma pré-disposição para compreender a aplicação dos recursos tecnológicos. Portanto, é altamente proveitoso explorar essa aptidão a fim de ampliar as oportunidades de aprendizado por meio dessas ferramentas (Prensky, 2001). Entre as competências que colhem vantagens desse talento intrínseco destaca-se o Pensamento

Computacional devido a capacidade de abordar problemas de maneira algorítmica, dividindo-os em etapas sequenciais, identificando padrões e desenvolvendo soluções eficazes.

De acordo com Gasser e Palfrey (2011), os nativos digitais demonstram criatividade e expressam suas habilidades de maneiras variadas, frequentemente distintas das habilidades que seus pais apresentavam quando tinham a mesma idade. Diante disso, torna-se inevitável adotar metodologias educacionais variadas que ultrapassem as metodologias tradicionais. Isso ocorre porque os nativos digitais têm uma rápida aptidão para assimilar o uso de recursos tecnológicos, o que, por sua vez, estimula habilidades criativas que vão além do âmbito da tecnologia.

Diante da necessidade de criar novos métodos de ensino que integrem recursos tecnológicos às metodologias, Araújo *et al.* (2023) argumentam que é essencial desafiar os professores a vivenciarem atividades aplicáveis a contextos reais em sala de aula. Essa vivência direta é fundamental para preparar os docentes para os desafios da educação do futuro e oferecer aos alunos experiências de aprendizagem mais diversificadas.

É importante que o processo de formação docente ofereça ao professor a oportunidade de experimentar diferentes práticas e abordagens de atividades que poderão ser aplicadas aos estudantes. Com isso, o receio de utilizar tecnologias pode ser substituído por estímulos, reflexões e trocas de ideias entre os professores, favorecendo o desenvolvimento de uma comunidade de aprendizagem (Araújo *et al.*, 2023).

Considerando a imersão desta geração na era da tecnologia digital, há uma sólida justificação para a adoção e implementação de metodologias ativas de ensino. Essas metodologias se alinham às características de aprendizado inerentes à atual geração de indivíduos. Dessa forma, ao incorporar essas metodologias, os educadores podem criar experiências de aprendizado mais eficazes e diferenciadas para essa geração de nativos digitais.

3.2.2 Metodologias Ativas de Aprendizagem

Segundo Freire (1996, apud Bes *et al.*, 2019), a educação deve se voltar para a problematização da realidade, buscando compreender os fenômenos e situações para, a partir disso, adotar atitudes que promovam transformações. As metodologias ativas seriam um recurso para estimular a curiosidade e envolver o aluno em situações concretas, levando-o a agir sobre a realidade e ultrapassando a concepção de ensino como uma simples transmissão de conhecimentos.

A utilização de metodologias ativas pode potencializar o processo de ensino e aprendizagem, permitindo que os alunos aprendam de forma lúdica e com maior engajamento. Barbosa e Moura (2013) argumentam que as metodologias ativas podem auxiliar os alunos a assimilarem conteúdos mais volumosos, reterem mais informações sobre diferentes assuntos e tornarem as aulas mais agradáveis.

Para Silberman (1996), se durante o processo de ensino e aprendizagem a metodologia favorecer o aluno nas atividades de ouvir, ver, questionar, discutir, fazer e ensinar, a metodologia está no caminho da aprendizagem ativa. Dessa forma, a aprendizagem ativa ocorre quando os alunos interagem com a matéria abordada e colocam em prática os conhecimentos adquiridos por meio de professores que atuam como orientadores e facilitam o processo de aprendizagem (Barbosa; Moura, 2013).

Morán (2015, p. 18), argumenta que as metodologias ativas “são pontos de partida para avançar para processos mais avançados de reflexão, de integração cognitiva, de generalização, de reelaboração de novas práticas”. Assim, os métodos ativos podem ser combinados com outras práticas de ensino, propondo equilíbrio entre atividades, resolução de desafios e informações estruturadas de tal forma que os alunos aprendam de forma mais prazerosa e divertida.

Conforme Bergmann (2018) e Silveira *et al.*, (2018) as metodologias ativas podem ser aplicadas em diversas estratégias de ensino, tais como:

- **Aprendizagem baseada em Problemas (PBL – *Problem Based Learning*)**: este método de ensino baseia-se no uso contextualizado de situações-problema para a aprendizagem autodirigida. A transferência de conhecimento é dirigida por professores que devem ser os principais remetentes do que é ensinado, com ênfase no assunto e na aprendizagem do aluno (Barbosa; Moura, 2013). A metodologia PBL tem por objetivo preparar alunos para resolverem problemas do mundo real, o que torna o aluno o principal protagonista na sala de aula. Por exemplo, durante uma aula de Engenharia de *Software*, o professor pode questionar como lidar com a crescente demanda de sistemas a serem entregues em prazos tão curtos.
- **Aprendizagem baseada em Projetos (*Project Based Learning*)**: de acordo com o *Buck Institute for Education* (BIE), este método incentiva os alunos a buscarem conhecimentos e habilidades desenvolvendo projetos em um longo período de tempo para explorar questões e problemas vistos em sala de aula. A aprendizagem baseada em projetos enfatiza atividades que focam na construção coletiva de conhecimento interdisciplinar e colaborativa entre os alunos para resolver problemas específicos

(Constantino; Thives, 2020). Um exemplo de metodologia baseada em projeto é uma tarefa de conclusão de curso que exige que os alunos pesquisem e enviem resultados finais que atendam aos requisitos solicitados pelo professor.

- **Sala de aula invertida (*Flipped Classroom*):** é um método ativo que integra o ensino remoto com o ensino presencial como ferramenta para potencializar o processo de ensino por parte dos professores e aprendizado por parte dos alunos. Ou seja, o que tradicionalmente era feito em sala de aula agora é feito em casa, e o que tradicionalmente era feito como lição de casa agora é feito em sala de aula (Bergmann; Sams; 2012). Por exemplo, os professores distribuem conteúdo para os alunos estudarem em casa, fornecem exposição inicial aos tópicos a serem discutidos e trabalham o conteúdo na forma de atividades em sala de aula.
- **Aprendizagem baseada em Jogos (GBL - *Game Based Learning*):** este método utiliza as mecânicas e dinâmicas que constituem os jogos como metodologia de ensino para diversas disciplinas das mais variadas áreas do conhecimento. As abordagens da metodologia baseada em jogos abrange desde a produção de jogos lúdicos com o efeito motivador para o ensino e aprendizagem de conteúdos escolares, até ambientes virtuais ou analógicos que não utilizam necessariamente jogos como metodologias de ensino (Sena *et al.*, 2016).

No entanto, para Bes *et al.* (2019), a utilização de tecnologias de informação e comunicação tem o potencial de romper as barreiras entre os espaços físicos e virtuais, criando novas formas de ensinar e aprender, com o suporte de diversos recursos tecnológicos que possibilitam aos alunos atuarem como protagonistas e autores de conteúdo. Nesse sentido, a utilização de jogos emerge como uma metodologia ativa que pode auxiliar no aprendizado dos estudantes em diversas disciplinas.

3.2.3 Inserção de Tecnologias nas Práticas Pedagógicas

De acordo com Araújo e Sant'Ana (2011), às Tecnologias de Informação e Comunicação (TICs) trouxeram grandes mudanças no papel do professor no que diz respeito às práticas pedagógicas. Para o autor, com o avanço da tecnologia, é difícil um educador atuar de forma eficiente utilizando apenas metodologias tradicionais e aulas expositivas, visto que a tecnologia está se multiplicando por toda parte.

Apesar da ampla disponibilidade de recursos tecnológicos, muitos educadores ainda resistem à sua integração em sala de aula. Eles temem que o uso da tecnologia no ambiente

educacional possa transformar o ensino em algo tecnicista e superficial, com a premissa de que essas tecnologias emergentes possam eventualmente substituir o papel do professor (Araújo e Sant'Ana, 2011).

Haviaras (2020) realizou uma pesquisa de doutorado sobre a formação de professores para o uso de tecnologias educacionais, envolvendo 114 professores universitários que ministram aulas em 12 cursos de graduação em Curitiba. O estudo constatou que havia resistência entre professores com 51 anos ou mais em aplicar metodologias que utilizam tecnologias como forma de enriquecer a sala de aula, além de identificarem-se limitações no uso dessas ferramentas.

Para Damasceno e Neves (2021), a resistência dos professores em relação ao uso de tecnologias na educação está frequentemente associada ao fato de que muitos veem essas ferramentas principalmente como instrumentos de lazer, o que desvia o foco do potencial educativo que elas oferecem. Essa resistência é sustentada pela percepção de que o engajamento real com as tecnologias só ocorre quando há uma participação ativa, em vez de um uso passivo como mero ouvinte. Essas crenças precisam ser confrontadas e revistas, pois a adoção de tecnologias em sala de aula pode ser uma ferramenta poderosa para promover a aprendizagem ativa e personalizada para diferentes perfis de alunos.

Dessa forma, para acompanhar as transformações relacionadas às novidades tecnológicas, os professores da educação básica não podem ignorar que as crianças, jovens e adultos fazem parte desse mundo e já convivem com essas tecnologias em seu cotidiano (Takemoto; Brostolin, 2015). Corroborando com isso, Dornelles (2012) argumenta que os professores da educação básica precisam entender a infância pós-moderna e devem estimular o senso crítico nas crianças e o uso responsável dessas tecnologias.

Conforme destacado por Klein *et al.* (2020), a integração de tecnologia no âmbito educacional busca instigar os alunos a explorarem mudanças e evoluções na prática de aprendizagem. Quando o professor incorpora recursos tecnológicos na entrega de conteúdo, não apenas facilita a transmissão de conhecimento, mas também introduz novas ferramentas de aprendizagem. Isso não apenas amplia o repertório do aluno, possibilitando o estímulo à criatividade, mas também contribui para a formação de novos conceitos relacionados a tarefas complexas, por meio de processos lúdicos.

Com a crescente demanda de recursos tecnológicos em ambientes escolares, as escolas enfrentam a necessidade de se adaptar a essas mudanças e atender às necessidades dos professores (Caruso; Cavalheiro, 2021). De acordo com uma pesquisa conduzida por Porto *et al.* (2023), os professores manifestaram um interesse significativo em obter informações sobre

a integração de tecnologia em ambientes educacionais. Esse interesse inclui a exploração do uso de tecnologias para aprimorar o desenvolvimento dos alunos, auxiliar em avaliações e fornecer *feedbacks*.

3.2.4 Avaliação em Ambientes Tecnológicos

A avaliação é o processo sistemático de aplicação de métodos para determinar, com base nos objetivos estabelecidos e critérios internos e/ou externos, a relevância, eficácia e impacto de atividades específicas, visando fornecer subsídios para a tomada de decisão (Laguardia *et al.*, 2007). Em contextos tecnológicos, a avaliação pode ser fundamentada na investigação das condições de aprendizagem, nos modelos de interação dos estudantes e no alcance dos objetivos e metas estabelecidos (Laguardia *et al.*, 2007).

As avaliações em ambientes tecnológicos, de acordo com Valcke e Leeuw (2000), estão divididas em cinco tipos:

1. O primeiro tipo de avaliação, comumente conduzido por instituições de ensino a distância, emprega indicadores de desempenho para avaliar o progresso dos alunos.
2. O segundo tipo de avaliação, também conduzido internamente, visa monitorar as atitudes e percepções dos alunos e clientes, além do desempenho dos funcionários.
3. O terceiro tipo de avaliação, conduzido por revisores externos, focaliza no ambiente sociocultural, nos estudos de custo-benefício, na viabilidade e nas análises de redes
4. O quarto tipo de avaliação concentra-se na abordagem das partes interessadas para reconhecer suas opiniões, atitudes e percepções sobre tecnologias de informação e comunicação, mesmo na ausência de experiência prática direta.
5. O quinto tipo de avaliação consiste em um conjunto de avaliações que abordam diversos tópicos no contexto do ensino em ambientes tecnológicos. Isso inclui a seleção e utilização de tecnologias, gestão da qualidade, certificação pela *International Standards Organization* (ISO) e estudos voltados para informática e comunicação mediada.

Os tipos de avaliações apresentados por Valcke e Leeuw (2000) destacam a importância de selecionar adequadamente os recursos de Tecnologias da Informação e Comunicação (TICs) para atividades educacionais e o ensino de conteúdos. Essa análise

revela que o processo de gestão de qualidade é eficaz quando os recursos tecnológicos estão alinhados com as metodologias e o conhecimento do professor.

No contexto do Pensamento Computacional, Curzon *et al.* (2014), realizaram um levantamento de processos avaliativos que os professores podem utilizar em sala de aula para verificar o progresso dos alunos após a realização de tarefas relacionadas ao Pensamento Computacional. Essas atividades envolveram *workshops* de formação docente centrados nos pilares do Pensamento Computacional e em habilidades que os professores podem utilizar de forma conjunta ou isolada para mensurar o aprendizado dos alunos.

Para Curzon *et al.* (2014), realizar atividades isoladas para cada um dos pilares é uma maneira eficaz de introduzir conceitos de computação às crianças, apresentando ideias do Pensamento Computacional que podem ser assimiladas com sucesso. Já Grover e Pea (2013) argumentam que, sem uma avaliação eficaz do aprendizado do aluno em sala de aula, o Pensamento Computacional não terá sucesso em nenhum currículo. Ele destaca a importância de criar estratégias para avaliar a eficácia de qualquer abordagem que integre o Pensamento Computacional.

Por outro lado, Berry (2013) apresenta sete possibilidades de avaliação que podem ser utilizadas pelos professores: (1) Autoavaliação; (2) Avaliação por pares; (3) Questionamentos abertos; (4) Discussão/Argumentação com pares; (5) Definição de metas; (6) KWL (Sabe, Quer Saber e Aprendeu); e (7) *Feedback* automático. No entanto, Berry (2013) ressalta que essas técnicas podem não ser viáveis, pois exigem o domínio de tecnologias às quais os alunos não têm acesso. Portanto, cabe ao professor desenvolver estratégias alternativas para mensurar o aprendizado dos alunos. Na próxima seção, serão apresentados os componentes dos jogos digitais que podem ser utilizados em práticas pedagógicas integradas à tecnologia.

3.3 Jogos Digitais

A presente subseção tem como propósito explorar o tema dos jogos digitais e sua relevância no âmbito educacional, com um enfoque particular no desenvolvimento do Pensamento Computacional. Os jogos digitais têm se firmado como uma forma popular de entretenimento, proporcionando experiências imersivas e interativas que atraem um amplo público. Além disso, os jogos têm demonstrado um notável potencial como ferramentas educacionais, integrando elementos lúdicos e pedagógicos para fomentar a aprendizagem e o desenvolvimento de habilidades cognitivas.

3.3.1 Definição e Características dos Jogos

Os jogos têm desempenhado um papel fundamental ao longo da história da humanidade, servindo como uma atividade lúdica presente em todas as culturas (Murcia, 2008). Desde os tempos antigos, os jogos foram utilizados como uma forma de aprendizado e desenvolvimento de habilidades sociais, contribuindo para a sobrevivência das diferentes sociedades (Alves, 2007). Essa tradição transcende fronteiras culturais e é capaz de estimular o desenvolvimento de habilidades cognitivas universais.

Segundo Falkembach (2006), os jogos podem ser utilizados para treinar habilidades cognitivas e estimular a imaginação, ao mesmo tempo em que mantêm a atenção do público-alvo, transmitindo informações e envolvendo os sentidos de forma estimulante, sem se tornar cansativo ou tedioso. Através da brincadeira e diversão proporcionadas pelos jogos, é possível extrair conhecimento, aproveitando a motivação que eles despertam nas pessoas.

Além de proporcionar diversão, os jogos têm um impacto positivo no desenvolvimento cognitivo, pois atuam como estímulos para o cérebro, ativando e desenvolvendo suas estruturas, o que resulta no aprimoramento de habilidades importantes, como observação, identificação, comparação, classificação, conceituação, relacionamento e inferência (Haguenauer *et al.*, 2007).

Os jogos são reflexos ligados aos valores e à cultura das sociedades em que são jogados, conforme apontado por Salen e Zimmerman (2012). Ao participar dessas experiências interativas, as pessoas têm a oportunidade de explorar um universo rico em possibilidades, ao mesmo tempo em que desenvolvem suas capacidades cognitivas, motoras e sociais. Assim, os jogos transcendem a mera recreação, abrindo portas para a descoberta cultural e o reconhecimento da diversidade no modo de jogar.

Com o avanço das tecnologias, a conexão entre jogos e desenvolvimento humano foi ampliada significativamente. Segundo Prensky (2001), os jogos combinam elementos lúdicos e interativos com a tecnologia digital, proporcionando experiências imersivas e divertidas. Ao explorar jogos digitais como ferramentas educacionais, é possível promover o desenvolvimento do Pensamento Computacional e outras habilidades cognitivas essenciais para o mundo contemporâneo (Barr; Harrison; Conery, 2011).

Segundo Sena *et al.* (2016), os jogos digitais têm se tornado uma presença constante na vida cotidiana de todas as pessoas, sendo produzidos em diversos estilos e níveis de dificuldade, atraindo um amplo público com diferentes idades e hábitos. Os autores argumentam que os jogos digitais deixaram de ser exclusivos de uma minoria, tornando-se

acessíveis a todos, o que abriu caminho para a aprendizagem baseada em jogos. Com a disseminação dos jogos digitais para uma variedade de públicos, o acesso a eles se tornou mais fácil através de dispositivos móveis, como *tablets* e *smartphones*.

No intuito de aprimorar e avançar nas abordagens de ensino e aprendizagem, os jogos digitais têm se tornado cada vez mais relevantes no contexto educacional. Reconhecidos como recursos tecnológicos valiosos, eles proporcionam aumento no engajamento e na motivação dos alunos (Guarda; Goulart, 2018). Conforme destacado por Barbosa (2014), os jogos têm o potencial de envolver crianças, jovens e adultos, estimulando-os a participar ativamente das atividades, seja por meio da resolução de desafios, da investigação de conceitos ou da aquisição de conhecimento em disciplinas específicas.

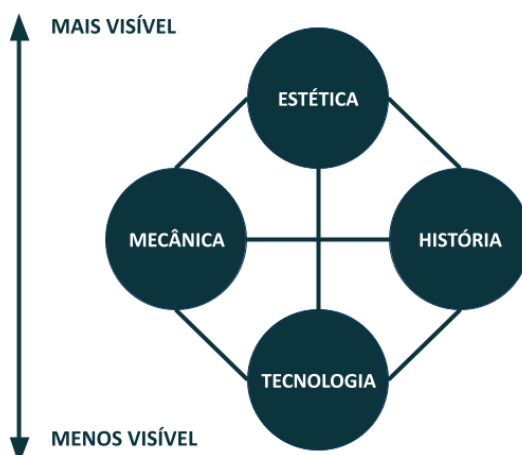
Segundo Murcia (2005, p. 9), os jogos eram desaprovados pela pedagogia tradicional como uma ferramenta efetiva no processo de ensino e aprendizagem. Os professores dessa abordagem consideravam que os jogos não poderiam contribuir para melhorar o desempenho escolar e não eram vistos como parte integrante da educação. No entanto, a pedagogia atual tem reavaliado essas percepções negativas e passou a reconhecer os jogos como uma atividade metodologicamente lúdica que pode ser incorporada na formação dos estudantes.

3.3.1.1 Estrutura dos Jogos Digitais

Autores renomados no campo dos jogos, como Schell (2019), Rogers (2013), Fullerton (2024), Salen e Zimmerman (2012), são conhecidos por suas contribuições significativas no desenvolvimento de estruturas para compreender os jogos. Essas estruturas dividem um jogo em vários elementos, fornecendo uma base para a análise e a compreensão dessa forma de diversão. Nesta seção, será explorado o modelo **Tétrade Elemental** proposto por Jesse Schell, um renomado autor e especialista no campo dos jogos.

Existem diversas abordagens para decompor e classificar os diferentes elementos que estruturam um jogo. Uma dessas abordagens é a Tétrade Elemental, proposta por Schell (2010), que identifica e categoriza os principais elementos de um jogo. São elas: mecânica, estética, história e tecnologia. Cada uma dessas categorias desempenha um papel importante na experiência do jogador e estão relacionadas umas às outras, como apresenta a Figura 8.

Figura 8 - Tétrade Elemental



Fonte: Schell (2010, p.42), tradução nossa.

A Figura 8 é uma representação visual da Tétrade Elemental proposta por Schell (2010) em seu livro “*Art of Game Design*”. A primeira camada a ser abordada é a "Estética", que desempenha um papel fundamental no *design* de jogos. Ela é a mais próxima do jogador e a mais visível em sua experiência, abrangendo a aparência visual, o som e a sensação tátil que compõem a atmosfera do jogo (Schell, 2010). É através da estética que os jogadores são imersos no mundo virtual e estabelecem uma conexão emocional com a experiência de jogo.

A camada da "Tecnologia" está ligeiramente menos visível aos jogadores. Segundo Schell (2010), esta camada não se limita apenas às tecnologias de *hardware*, mas abrange também os artefatos, técnicas e recursos utilizados para permitir a interação dos jogadores com o jogo. Essas tecnologias podem variar desde jogos simples feitos com papel e caneta, cartas e tabuleiros, até *videogames*, computadores e óculos de realidade virtual, entre outros. A escolha das tecnologias adotadas no jogo pode determinar as limitações e possibilidades de *design* que o jogo terá.

As “Mecânicas” são procedimentos e regras básicas que descrevem os objetivos do jogo e como os jogadores podem ou não tentar alcançá-los. São as mecânicas que tornam um jogo único e distintamente diferente de experiências lineares, como livros e filmes. Ao escolher as mecânicas de um jogo, é importante selecionar a tecnologia adequada, uma estética que as destaque e uma história que dê sentido a essas mecânicas (Schell, 2010).

A "História" abrange uma sequência de eventos e como eles se desenrolam durante a interação do jogador com o jogo. Quando há uma história para ser contada por meio do jogo, torna-se importante escolher mecânicas que fortaleçam e permitam o desenvolvimento dessa história (Schell, 2010). Dessa forma, qualquer narrativa pode ser transformada em um jogo,

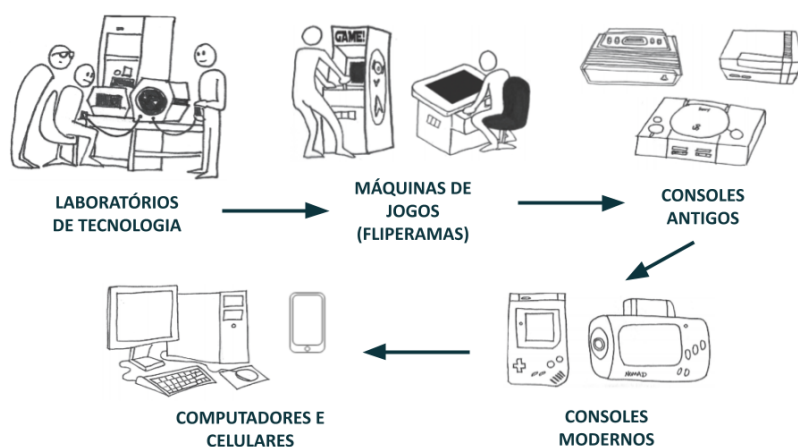
desde que a estética seja cuidadosamente selecionada para reforçar as ideias e a tecnologia seja adequadamente aplicada para dar vida à história que se desenrolará no jogo.

De acordo com Schell (2010), nenhum dos elementos apresentados é mais ou menos importante do que os outros que compõem a Tétrade Elemental. A disposição da Tétrade em formato de diamante tem o objetivo de ilustrar o "gradiente de visibilidade", que indica que os elementos tecnológicos tendem a ser menos visíveis para os jogadores, enquanto a estética é a mais visível, e as mecânicas e a história ficam em um grau intermediário. No entanto, vale ressaltar que a Tétrade pode ser organizada de outras formas, não sendo a disposição em diamante uma regra fixa.

3.3.1.2 Gênero de Jogos Digitais

Regers (2013) relata que por volta de 1950, surgiram os primeiros *videogames* em pequenos laboratórios de tecnologia, introduzindo a ideia de jogos que poderiam ser jogados em máquinas com telas. Essas máquinas foram posteriormente incorporadas em ambientes propícios aos jogos, como fliperamas, alcançando grande sucesso na década de 80. Com o avanço tecnológico e a crescente demanda por jogos digitais, surgiram os consoles, permitindo que os jogadores desfrutassem de seus jogos sem sair de casa. Atualmente, os jogos estão disponíveis em uma variedade de dispositivos, como *tablets*, celulares, computadores, entre outros. A evolução tecnológica pode ser resumida na Figura 9 apresentada a seguir.

Figura 9 - Evolução das plataformas de jogos digitais



Fonte: Adaptado de Rogers (2013).

A Figura 9 oferece uma visão da evolução das plataformas de jogos digitais ao longo dos anos, destacando a importância das tecnologias na definição desses jogos. É evidente que a progressão da portabilidade das plataformas de jogos acompanhou os avanços tecnológicos. Conforme as plataformas e os jogos foram se desenvolvendo, surgiram diversos gêneros para classificar seus objetivos e a forma como os jogadores interagem com eles. Essa diversidade de gêneros reflete o contínuo crescimento e a sofisticação da indústria de jogos digitais (Rogers, 2013).

Ao longo dos anos, conforme descrito por Rogers (2013), os jogos foram classificados em diferentes gêneros e subgêneros, acompanhando o surgimento de novas tecnologias e plataformas. Essa fragmentação dos jogos em categorias permite descrever o estilo de jogabilidade, ou seja, como o jogador interage com o ambiente do jogo. Por sua vez, Herz (1997) argumenta que a subdivisão dos jogos em gêneros e subgêneros facilita a comercialização e as vendas por parte da indústria de jogos, além de agilizar o processo de criação.

Rogers (2013), Herz (1997) e Schell (2008), apresentam alguns dos principais gêneros presentes na indústria dos jogos, são elas:

- **Ação:** são jogos que requerem coordenação entre mãos e olhos para serem jogados e que devem ser executados de maneira rápida. O gênero ação pode ter diversos subgêneros: Aventura de ação; Ação arcade; Plataforma; entre outros.
- **Shoter:** foram inicialmente concebidos para envolver o jogador em confrontos onde é necessário atirar projéteis nos inimigos, exigindo reflexos rápidos. Embora compartilhem semelhanças com os jogos de ação, a principal diferença reside na perspectiva da câmera utilizada no jogo.
- **Aventura:** têm seu foco principal na resolução de quebra-cabeças, na coleta de itens disponíveis no jogo e no gerenciamento dos recursos presentes em um inventário. Os jogadores são desafiados a explorar ambientes vastos, desvendar mistérios e superar obstáculos para progredir na história do jogo.
- **Construção/Gerenciamento:** o objetivo do jogador é criar e expandir um ambiente, utilizando os recursos que são coletados ao longo do jogo. Esses recursos são usados para construir estruturas, gerenciar a economia, desenvolver tecnologias e atender às necessidades dos personagens ou habitantes dentro do jogo.
- **Simulação:** semelhante ao gênero Construção/Gerenciamento, com o foco em torno da construção e desenvolvimento de relacionamentos com outras formas de vida artificiais. A simulação busca proporcionar uma experiência mais imersiva e realista,

permitindo que os jogadores explorem diferentes aspectos da vida e desenvolvam relacionamentos virtuais.

- **Quebra-Cabeça:** são baseados no raciocínio lógico, no reconhecimento de padrões e na habilidade de fazer abstrações. Nesses jogos, os jogadores são desafiados a resolver problemas e quebra-cabeças por meio da aplicação de estratégias e da utilização da lógica. Eles precisam analisar as informações disponíveis, identificar padrões ocultos e encontrar soluções criativas para avançar no jogo.
- **Esportes:** são baseados em competições atléticas entre os jogadores ou grupos de jogadores, abrangendo tanto esportes tradicionais quanto modalidades não convencionais. Nesses jogos, os jogadores podem participar de simulações realistas de esportes populares, como futebol, basquete, tênis, golfe, corrida e muito mais.
- **Estratégia:** são caracterizados pelo foco no pensamento estratégico e no planejamento dos jogadores. Estes jogos oferecem um desafio intelectual, onde os jogadores precisam tomar decisões estratégicas cuidadosas para alcançar seus objetivos.
- **RPG (*Role-Playing Game*):** são semelhantes aos jogos de aventura, pois também envolvem exploração e resolução de quebra-cabeças. No entanto, o que os diferencia é o foco mais pronunciado na narrativa dos personagens. Nos jogos de RPG, os jogadores assumem o papel de um ou mais personagens, embarcando em uma jornada épica através de um mundo fictício ou baseado em histórias.

É importante destacar que um jogo pode ser composto por diferentes gêneros, permitindo ao jogador experimentar várias sensações ao longo da jogabilidade. Com a diversidade de gêneros de jogos, o seu desenvolvimento tornou-se uma tarefa complexa e sistemática, exigindo um planejamento cuidadoso por parte dos profissionais responsáveis pela sua criação. Esse processo de planejamento é conhecido como *Game Design* e abrange todas as etapas envolvidas na concepção de diferentes tipos de jogos. A seção a seguir apresentará uma visão desse processo de criação, destacando as etapas e conceitos envolvidos.

3.3.2 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 10 ilustra a estrutura do GDD adotada pelos autores.

Figura 10 - Representação de um *Game Design Document*

 LOGI KINGDOM <i>Game Design Document</i> Versão: 1.0 AUTOR Douglas Aquino Moreno Stefan Lucas Aquino Silva Parcilene Fernandes De Brito Calebe Loures Sampaio Eler de Souza Palmas Tocantins, 2022	SUMÁRIO <table> <tr><td>1. História.....</td><td>3</td></tr> <tr><td>2. Gameplay.....</td><td>4</td></tr> <tr><td> 2.1. Tabela de Pontuação.....</td><td>5</td></tr> <tr><td> 2.2. Pontuação e Erros.....</td><td>5</td></tr> <tr><td>3. Personagens.....</td><td>7</td></tr> <tr><td> 3.1. Guerreira.....</td><td>7</td></tr> <tr><td> 3.2. Fantasma.....</td><td>8</td></tr> <tr><td>4. Controles.....</td><td>9</td></tr> <tr><td> 4.1. Botões.....</td><td>9</td></tr> <tr><td>5. Câmera.....</td><td>10</td></tr> <tr><td>6. Universo do Jogo.....</td><td>12</td></tr> <tr><td>7. Inimigos.....</td><td>15</td></tr> <tr><td> 7.1. Falaciosa.....</td><td>15</td></tr> <tr><td>8. Interface.....</td><td>16</td></tr> <tr><td>9. Cutsscenes.....</td><td>19</td></tr> <tr><td>10. Cronograma.....</td><td>20</td></tr> </table>  2	1. História.....	3	2. Gameplay.....	4	2.1. Tabela de Pontuação.....	5	2.2. Pontuação e Erros.....	5	3. Personagens.....	7	3.1. Guerreira.....	7	3.2. Fantasma.....	8	4. Controles.....	9	4.1. Botões.....	9	5. Câmera.....	10	6. Universo do Jogo.....	12	7. Inimigos.....	15	7.1. Falaciosa.....	15	8. Interface.....	16	9. Cutsscenes.....	19	10. Cronograma.....	20
1. História.....	3																																
2. Gameplay.....	4																																
2.1. Tabela de Pontuação.....	5																																
2.2. Pontuação e Erros.....	5																																
3. Personagens.....	7																																
3.1. Guerreira.....	7																																
3.2. Fantasma.....	8																																
4. Controles.....	9																																
4.1. Botões.....	9																																
5. Câmera.....	10																																
6. Universo do Jogo.....	12																																
7. Inimigos.....	15																																
7.1. Falaciosa.....	15																																
8. Interface.....	16																																
9. Cutsscenes.....	19																																
10. Cronograma.....	20																																

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

A Figura 10 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, Cutsscenes 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.

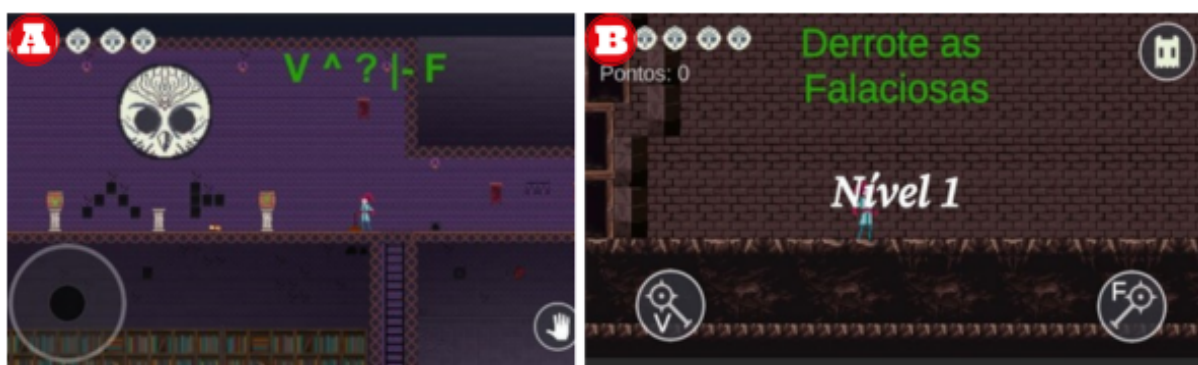
Para Silva *et al.* (2021), o processo de documentação não é uma tarefa simples para a equipe de desenvolvimento, especialmente devido às constantes mudanças no *design* do jogo, que resultam em atualizações frequentes. No entanto, a documentação do *design* é um passo importante para garantir que a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

3.3.3 Jogos *Mobiles*

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 11 a seguir apresenta algumas telas do jogo.

Figura 11 - 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.

Conforme observado, os jogos têm a capacidade de atrair o público estudantil e auxiliar no processo de aprendizagem. O Produto Final desta pesquisa tem como propósito a construção de um (GDD que poderá ser utilizado para o desenvolvimento de um jogo destinado a fomentar o Pensamento Computacional entre estudantes da educação básica, além de contribuir com o trabalho dos educadores.

3.3.4 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 Victal *et al.* (2024), 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 12, 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 12 - 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 12 apresenta dois jogos educativos, o "Hour of Code" (à direita) e o "The Foos" (à esquerda), 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 a indivíduos, mesmo sem experiência prévia, a oportunidade de compreender e aplicar os conceitos básicos de maneira participativa. 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 apresentado o processo metodológico para a pesquisa e construção do *Game Design Document*.

4 PROCESSO METODOLÓGICO

Nesta seção, serão detalhadas a abordagem e o tipo de pesquisa adotados nesta dissertação, assim como a metodologia aplicada ao desenvolvimento do Produto Final, o *Game Design Document* (GDD) do Logic Think. A pesquisa fundamenta-se na abordagem qualitativa, do tipo Estado da Arte, com o objetivo de compreender como o uso de jogos educacionais pode contribuir para o desenvolvimento do Pensamento Computacional entre alunos da Educação Básica. Essa análise permitiu identificar as práticas existentes e as lacunas na literatura, além de embasar a proposta do GDD, que busca oferecer uma solução inovadora para o ensino de Pensamento Computacional.

4.1 Abordagem e Tipo de Pesquisa

A presente pesquisa baseia-se em uma abordagem qualitativa (Erickson, 1989), do tipo Estado da Arte (Pilão, 2009), com o objetivo de mapear e analisar as principais abordagens relacionadas ao Pensamento Computacional e ao uso de jogos educacionais para seu desenvolvimento. Para garantir uma visão atualizada do Estado da Arte, foram selecionados livros, artigos científicos, dissertações, teses e documentos normativos que abordam o Pensamento Computacional, bem como as metodologias mais efetivas para seu ensino na educação básica.

Segundo Strauss e Corbin (2008, p. 23) uma pesquisa qualitativa é “qualquer tipo de pesquisa que produza resultados não alcançados através de procedimentos estatísticos ou de outros meios de quantificação”. Erickson (1989) complementa que essa abordagem é caracterizada pelo foco interpretativo na busca de compreender os fenômenos a partir das perspectivas dos sujeitos envolvidos. Diferente de métodos quantitativos, as técnicas utilizadas em uma pesquisa qualitativa não determinam o método em si, mas servem como ferramentas para aprofundar a interpretação dos dados coletados.

A abordagem qualitativa tem como objetivo apresentar resultados que não são expressos em termos de quantidade, intensidade ou frequência. Ao contrário, essa abordagem enfatiza a natureza socialmente construída da realidade, destacando a importância das interações entre pesquisadores e participantes para moldar as pesquisas. Esse tipo de pesquisa é permeado por valores, buscando responder a perguntas que destacam as experiências sociais construídas e compartilhadas. Dessa forma, as pesquisas qualitativas exploram as dinâmicas sociais e as percepções subjetivas, oferecendo uma compreensão mais profunda dos fenômenos estudados (Denzin; Lincoln, 2018).

De acordo com Pillão (2009), o Estado da Arte é uma modalidade de pesquisa adotada por diversos pesquisadores com o objetivo de investigar e analisar o conhecimento acumulado em um campo específico de estudos. Esse tipo de pesquisa permite mapear as principais contribuições, teorias, métodos e resultados já desenvolvidos, além de identificar lacunas, tendências e desafios. Dessa forma, o Estado da Arte fornece uma visão ampla e crítica sobre o estágio atual do conhecimento, auxiliando na formulação de novas questões de pesquisa e na orientação de futuros estudos.

Para Brandão *et al.* (1986), o termo Estado da Arte refere-se ao levantamento de pesquisas existentes sobre um determinado tema, com o objetivo de reunir o conhecimento produzido em uma área específica. Ferreira (2002) complementa que o Estado da Arte vai além do simples mapeamento de produções científicas em diferentes áreas do conhecimento; essa metodologia busca identificar como o conhecimento tem sido organizado e apreendido em diversos periódicos e bases de pesquisa.

A partir desses procedimentos metodológicos, a revisão e estruturação desta pesquisa foi estruturada em três eixos principais, a saber:

- **Pensamento Computacional:** Definições, abordagens e a relevância de sua inclusão no currículo escolar, conforme destacado por autores como Wing (2006) e conforme as diretrizes da Base Nacional Comum Curricular (BNCC).
- **Práticas Pedagógicas: Tecnologia e Inovação:** Exploração de estratégias pedagógicas existentes, como atividades sem tecnologia, programação em blocos (Scratch), robótica educacional e a criação de jogos digitais (Valente, 2016).
- **Jogos Digitais:** Análise do impacto dos jogos na educação, com ênfase no engajamento dos alunos e na efetividade no desenvolvimento de competências como o Pensamento Computacional (Moreno *et al.*, 2021).

Para compilar o Estado da Arte, foram utilizados critérios de inclusão que visam garantir a relevância e a qualidade das fontes. As bases de dados científicas exploradas incluem *Google Scholar*, *SBC-OpenLib* (SOL) e Plataforma Sucupira - CAPES, com um recorte temporal dos últimos 10 anos, para garantir que os avanços mais recentes sobre o Pensamento Computacional e o uso de jogos educacionais sejam considerados.

Os critérios de inclusão dessas pesquisas foram:

- Estudos que tratem diretamente da aplicação do Pensamento Computacional em ambientes de ensino;
- Pesquisas que envolvem o uso de jogos como ferramentas pedagógicas, em especial na Educação Básica.

Após a coleta e análise das publicações, os autores e estudos mais relevantes para a pesquisa sobre os Eixos foram organizados no quadro a seguir.

Quadro 1 - Textos referenciados para a Metodologia Estado da Arte

Eixo	Fonte	Autor/Ano	Periódico/Repositório	Categoria
Pensamento Computacional	Google Scholar	Wing, 2006	Periódico: <i>ACM Digital Library</i>	Artigo Científico
		Liukas, 2015	Google Books	Documento Normativo
	SBC-OpenLib (SOL)	Higuchi <i>et al.</i> , 2021	Periódico: SBGames	Artigo Científico
	Periódicos da CAPES	Valente, 2016	Periódico: Revista e-Curriculum	Artigo Científico
		Ribeiro; Foss; Cavalheiro, 2020	Periódico: <i>Colloquium Humanarum</i>	Artigo Científico
		Brackmann, 2017	Repositório: UFRGS - LUME	Tese
		Vicari <i>et al.</i> , 2018		Livro
Práticas Pedagógicas: Tecnologia e Inovação	Google Scholar	Gasser e Palfrey, 2011	Periódico: Artmed	Artigo Científico
		Barbosa e Moura, 2013	Periódico: <i>Senac Journal of Education and Work</i>	Artigo Científico
		Morán, 2015	Periódico: UEPG/PROEX	Artigo Científico
		Bergmann, 2018	Google Books	Livro
	Periódicos da CAPES	Constantino; Thives, 2020	Periódico: Portal da UNIEDU	Artigo Científico
		Takemoto; Brostolin, 2015	Periódico: UCDB	Artigo Científico
Jogos Digitais	Google Scholar	Schell, 2019	Google Books	Livro
		Rogers, 2013	Google Books	Livro
		Salen e Zimmerman, 2012	Google Books	Livro
		Battistella, 2016	Repositório: UFSC	Tese

	<i>SBC-OpenLib (SOL)</i>	Higuchi et al., 2021 SBGames	Periódico: SBGames	Artigo Científico
		Leite; Mendonça, 2013	Periódico: SBGames	Artigo Científico
		Pereira e Rocha, 2023	Periódico: SBGames	Artigo Científico
	Periódicos da CAPES	Silveira, 2017	Repositório: UFPE	Dissertação

Fonte: autor.

Os autores citados no Quadro 1 fizeram parte essencial da análise qualitativa do tipo Estado da Arte, contribuindo para o aprofundamento teórico sobre o Pensamento Computacional, práticas pedagógicas inovadoras e o uso de jogos digitais na educação. Suas contribuições permitiram mapear as principais abordagens metodológicas, tecnologias e ferramentas educacionais, bem como identificar lacunas na literatura, especialmente no que tange ao desenvolvimento de jogos educacionais para o ensino do Pensamento Computacional. Essa análise fundamentou o desenvolvimento da proposta desta dissertação, fornecendo uma base para a construção de um *Game Design Document* alinhado às necessidades pedagógicas da Educação Básica.

4.2 Metodologia Aplicada ao Produto Tecnológico

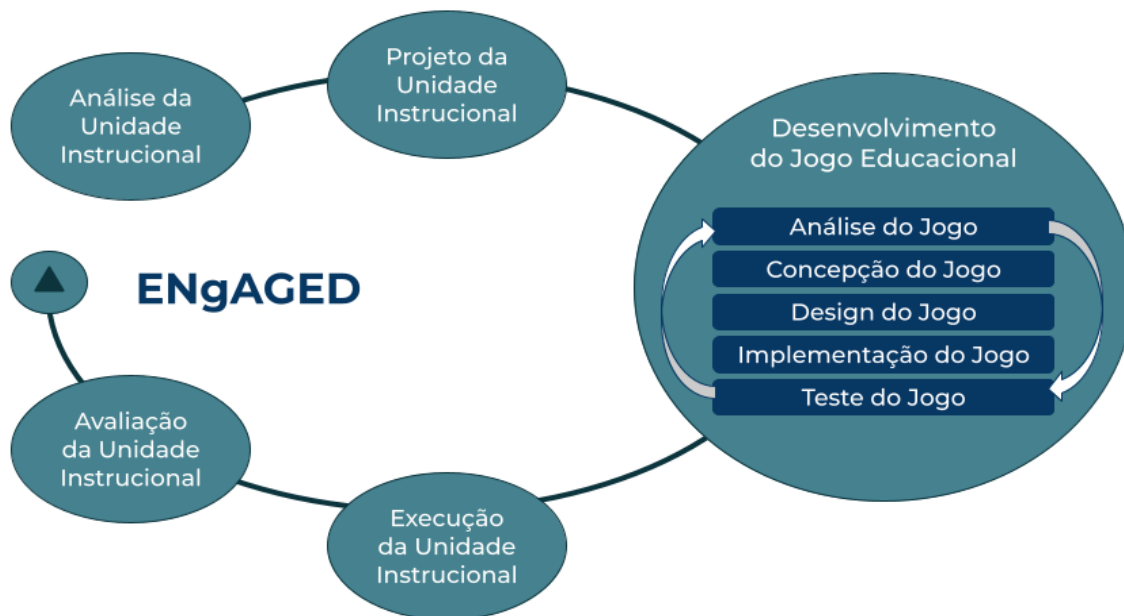
Esta subseção apresenta o processo metodológico para a concepção do *Game Design Document*, incorporando elementos fundamentais dos procedimentos delineados por Battistella (2016) e Silva e Farias (2019). Essas abordagens demonstraram eficácia no desenvolvimento de diversos jogos educativos, e este estudo propõe a integração de componentes estratégicos dessas metodologias. Os tópicos 4.2.1 e 4.2.2 oferecerão uma exposição detalhada das metodologias de Battistella (2016) e Silva e Farias (2019), delineando as escolhas e adaptações específicas realizadas para atender aos objetivos desta pesquisa.

4.2.1 ENgAGED

O ENgAGED (*Educational Games Development*) é um processo inovador para o desenvolvimento de jogos educacionais que combina *design* instrucional com *design* de jogos.

Criado por Battistella (2016), o ENgAGED foi idealizado para aplicação em cursos de computação no ensino superior, mas pode ser adaptado para outras áreas do conhecimento. A Figura 13 apresenta a ilustração desse processo.

Figura 13 - ENgAGED - *Educational Games Development*



Fonte: Adaptado de Battistella (2016).

O ENgAGED é direcionado para alunos e professores da área de computação e oferece suporte para o desenvolvimento de jogos educacionais, mesmo que não possuam conhecimento sobre *design* instrucional ou *design* de jogos (Battistella, 2016). O processo ENgAGED é composto por 5 etapas relativas ao *design* instrucional e 5 etapas ao *design* de jogos, divididas da seguinte forma:

- **Análise da Unidade Instrucional (UI):** definir os objetivos do jogo a serem alcançados e o público alvo. Nesta fase são priorizados as atividades didáticas do conteúdo e para isso é realizado um pesquisa com alunos e professores;
- **Projeto da Unidade Instrucional (UI):** definir as estratégias educacionais para alcançar os objetivos propostos na fase UI. Como resultado, espera-se que o professor da disciplina na qual o jogo será desenvolvido defina as estratégias instrucionais que serão utilizadas na unidade;
- **Desenvolvimento do Jogo Educacional (UI):** desenvolver o jogo contendo fases e atividades orientadas pelo professor da disciplina. Esta etapa é dividida em cinco: (1) Análise do Jogo - consiste no levantamento de requisitos; (2)

Concepção do jogo - definir as características do jogo, como objetivos, narrativa e mecânicas; (3) *Design* do jogo - definir a linguagem de programação e os elementos que farão parte do projeto; (4) Implementação do Jogo - codificar os elementos do jogo produzidos na fase de *design*; (5) Teste - realizar os testes do jogo a fim de identificar possíveis erros.

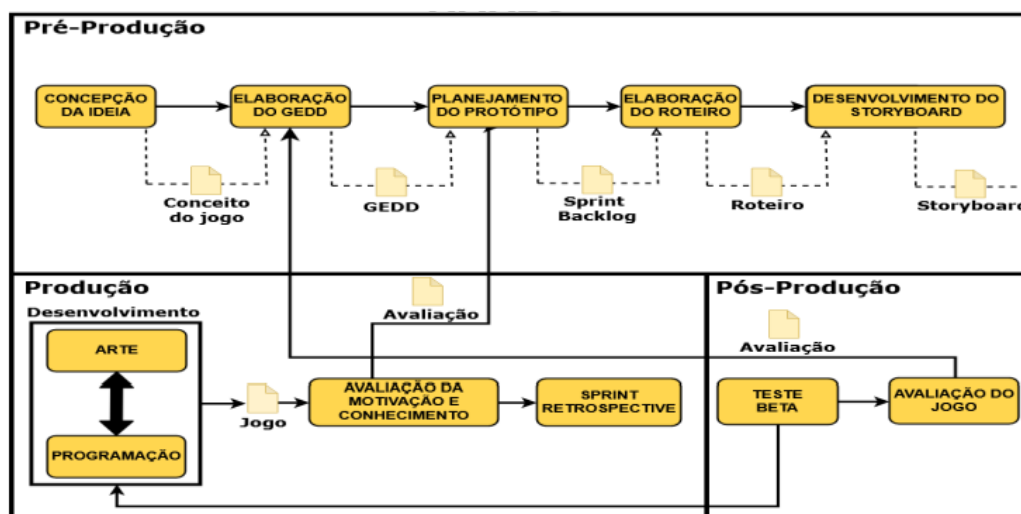
- **Execução da Unidade Instrucional (UI):** Planejar a utilização do jogo e definir datas para jogar, local de aplicação do jogo e equipamentos que poderão ser utilizados;
- **Avaliação da Unidade Instrucional (UI):** definir, planejar e executar a avaliação da unidade instrucional. O professor poderá realizar a aplicação do jogo e verificar se o objetivo de aprendizagem foi alcançado.

Battistella (2016) relatou em seu trabalho que a metodologia desenvolvida passou por diversos estudos de caso e foi aplicada na criação de quatro jogos educacionais, obtendo um resultado positivo quanto a aplicação do processo. Os jogos criados foram desenvolvidos por alunos do curso de Ciência da Computação e Sistemas de Informação da Universidade Federal de Santa Catarina, como trabalho de conclusão de curso. A metodologia criada pelo autor facilitou o processo de concepção, implementação e testes de jogos voltados para o ensino de disciplinas da área da computação.

4.2.2 AMAEG

Silva e Farias (2019) apresentam a metodologia ativa AMAEG (*Agille Methods Applied To Educacional Games*), um processo de desenvolvimento de jogos educacionais dividido em três fases bem definidas: Pré-produção, que corresponde ao plano inicial de desenvolvimento; Produção, que compreende aos processos de implementação do jogo educacional; e Pós-produção que abrange a avaliação da experiência do jogador e lançamento do jogo. A Figura 14 apresenta as fases da metodologia AMAEG.

Figura 14 - Metodologia AMAEG



Fonte: Silva e Farias (2019).

A fase de Pré-produção da metodologia de Silva e Farias (2019) é dividida em cinco etapas: Concepção da ideia - definição das características do jogo; Elaboração do GEDD (*Game Educacional Design Document*) - criação do documento com as ideias obtidas na fase de concepção; Planejamento do protótipo - gera-se uma *Sprint Backlog* como artefato dos itens definidos no GEDD; Elaboração do roteiro - criação do conjunto das cenas do jogo; Desenvolvimento do *Storyboard* - corresponde a sequência lógica das cenas que farão parte do jogo.

A fase de Produção é dividida em três etapas: Desenvolvimento do jogo - corresponde a arte e a programação; Avaliação da motivação e conhecimento - é realizado o primeiro teste com alunos no intuito de avaliar o protótipo do jogo; *Sprint Retrospective* - objetiva a avaliação dos resultados apresentados pelos alunos que testaram o protótipo e avaliar os itens implementados. A fase de Pós-produção compreende duas etapas: Teste beta - detectar e corrigir eventuais erros que foram identificados na fase de produção; Avaliação do jogo - obter uma análise voltada para a experiência de imersão, desafios, divertimento e habilidades treinadas durante o jogo.

Tanto o ENgAGED quanto o AMAEG, oferecem estruturas sólidas e sistemáticas para o desenvolvimento de jogos educacionais, enfatizando diferentes aspectos do processo. O ENgAGED, proposto por Battistella (2016), destaca-se por integrar *design* instrucional com *design* de jogos, dividindo o processo em etapas específicas relacionadas à análise, projeto, desenvolvimento, execução e avaliação da unidade instrucional. Por outro lado, o AMAEG, desenvolvido por Silva e Farias (2019), foca em métodos ágeis aplicados ao desenvolvimento

de jogos educacionais, dividindo o processo em fases bem definidas de pré-produção, produção e pós-produção.

Dessa forma, ambas as metodologias fornecem exemplos práticos e orientações detalhadas para o desenvolvimento de jogos educacionais, possibilitando sua aplicação eficaz mesmo para aqueles que não possuem conhecimento prévio em *design* instrucional ou de jogos. As metodologias apresentadas foram fundamentais na construção do Produto Final desta pesquisa, contribuindo para o desenvolvimento de sequências e abordagens educacionais focadas nos pilares do Pensamento Computacional.

4.2.3 Materiais e Métodos

As tecnologias utilizadas para o desenvolvimento do *Game Design Document* envolvem a plataforma Unity, uma *Game Engine* que permite a utilização das linguagens JavaScript e C# para o desenvolvimento dos Scripts do jogo. A plataforma oferece aos desenvolvedores diversas ferramentas para criar e operar jogos, além de possibilitar a publicação dos projetos em diversos dispositivos, como *PlayStation*, *Xbox*, *Nintendo Switch* e *Google Stadia* (Unity, 2021).

Para a criação dos fluxos de trabalho, *design* do jogo e componentes gráficos do projeto, foram utilizados o *Photoshop* e o *Aseprite*. O *Photoshop* é um software muito utilizado em computadores, com destaque no mundo da fotografia e do *design* gráfico por oferecer inúmeros recursos que facilitam o trabalho dos usuários. O *Aseprite* é um software específico para criação de *pixel art*, ideal para jogos com estética retrô. A Figura 15 apresenta o estudo metodológico utilizado neste trabalho, que foi elaborado com base nas metodologias descritas nas seções 4.2.1 e 4.2.2.

Figura 15 - Processo Metodológico



Fonte: Autor.

A Figura 15 apresenta as etapas utilizadas para o desenvolvimento do Produto Final desta pesquisa, com base nos trabalhos de Battistella (2016) e Silva e Farias (2019). Na etapa

1, foi realizada uma pesquisa bibliográfica sobre temas relacionados ao jogo, incluindo os princípios fundamentais do Pensamento Computacional, com o objetivo de compreender como esses princípios estão sendo integrados ao currículo escolar. Na etapa 2, foi realizada a concepção inicial do jogo, que inclui a definição de objetivos a serem alcançados, público-alvo, regras de jogabilidade, mecânicas, elementos, pontuações e *feedback* educacional. Além disso, a concepção foi necessária para selecionar os conteúdos dos Pilares do Pensamento Computacional que seriam utilizados no projeto.

Na etapa 3, deu-se início à elaboração do roteiro do jogo, utilizando como base as informações obtidas nas etapas anteriores. Durante esse processo de produção, foi criado um documento textual contendo a descrição dos elementos necessários para viabilizar o projeto, incluindo cenas, diálogos e outros componentes essenciais para o desenvolvimento do GDD. Na etapa 4, foi elaborado o GDD, considerado a peça central e o Produto Final do processo de desenvolvimento desta pesquisa. Este documento, disponível no apêndice, define a ideia central do jogo, as interações entre as mecânicas, o enredo e todos os outros elementos que são parte integrante do Produto Final.

5 RESULTADOS E DISCUSSÕES

Esta seção tem como objetivo apresentar e descrever as etapas realizadas neste trabalho, bem como os resultados obtidos. Essas etapas estão organizadas nos seguintes tópicos: 5.1 Concepção do Jogo, 5.2 Conteúdo de Aprendizagem e 5.3 Concepção do Jogo.

5.1 Projeto do Jogo

Esta pesquisa teve como objetivo criar o *Game Design Document* (GDD) para o jogo Logic Think, que será utilizado como uma ferramenta educacional para promover o Pensamento Computacional entre alunos e professores da educação básica. Inspirado nos gêneros de jogos de plataforma 2D e RPGs, Logic Think incorpora mecânicas e elementos característicos desses estilos para explorar conceitos de tecnologia e computação de forma lúdica. O projeto adota o estilo *pixel art*, tradicionalmente associado a esses gêneros, e apresenta personagens que representam conceitos fundamentais de computação, integrando-os ao processo de aprendizagem.

A narrativa central do GDD gira em torno da exploração de um mundo virtual chamado 'Código Vivo', habitado por inteligências artificiais conscientes. No jogo, o jogador assume o papel de Allan, um jovem apaixonado por tecnologia que busca compreender os seres digitais que residem em 'Código Vivo'. A história aborda temas como a relação entre humanos, Pensamento Computacional, Inteligências Artificiais, desafios da rápida evolução tecnológica e a busca por uma coexistência pacífica entre ambas as formas de vida.

Para auxiliar no desenvolvimento do GDD foi definida uma relação de requisitos funcionais e não funcionais que o jogo deve assumir e que deve estar representado no GDD, conforme apresentado no quadro a seguir.

Quadro 2 - Requisitos funcionais e não-funcionais

REQUISITOS FUNCIONAIS	RELEVÂNCIA		
	Desejável	Importante	Essencial
Os cenários do jogo remetem aos pilares do Pensamento Computacional			X
O jogo possuirá um fim pré definido, de forma que só termine quando o jogador finalizar a última			X

missão			
O primeiro nível apresenta a missão principal do jogo		X	
O segundo nível apresenta a missão relacionada ao pilar da Decomposição			X
O terceiro nível apresenta a missão relacionada ao pilar do Reconhecimento de Padrões			X
O quarto nível apresenta a missão relacionada ao pilar da Abstração			X
O quinto nível apresenta a missão relacionada ao pilar da Algoritmos			X
O jogo deve pontuar o desempenho do jogador		X	
O jogo deve apresentar um menu principal com opções como "Novo Jogo", "Continuar" e "Opções"			X
Deve haver um sistema de salvamento e carregamento de progresso do jogador			X
O jogo deve possuir controle do volume dos sons do jogo	X		
REQUISITOS NÃO-FUNCIONAIS	RELEVÂNCIA		
	Desejável	Importante	Essencial
O jogo deve estar disponível para plataforma mobile em sistemas <i>android</i>			X
O jogo deve ter conexão com <i>internet</i>	X		
Deve ser desenvolvido com componentes visuais de licença gratuita			X

Fonte: autor.

No Quadro 2, são apresentados requisitos funcionais e não-funcionais que auxiliam no desenvolvimento do GDD e que contribuíram para delimitar as métricas do que seria relevante para o planejamento e execução deste trabalho. Como requisitos funcionais, têm-se características que o jogo deve possuir e que estão diretamente relacionadas com conceitos do Pensamento Computacional e como o jogador irá vivenciá-los dentro do jogo. Os requisitos não-funcionais dizem respeito ao comportamento do jogo como produto, como por exemplo,

em que plataforma estará disponível para que os jogadores possam acessá-lo, além de características de funcionamento.

Schell (2010) argumenta que há diversas abordagens para decompor e classificar os diferentes elementos que estruturam um jogo. Por outro lado, Battistella (2016) propõe uma visão mais direcionada ao uso de jogos educativos, enfatizando que os requisitos não funcionais, como a acessibilidade em diferentes plataformas, devem ser pensados para maximizar o potencial de aprendizado. Essa visão complementa a de Schell ao introduzir um aspecto mais prático em relação ao uso de jogos em contextos específicos.

5.2 Conteúdo de Aprendizagem

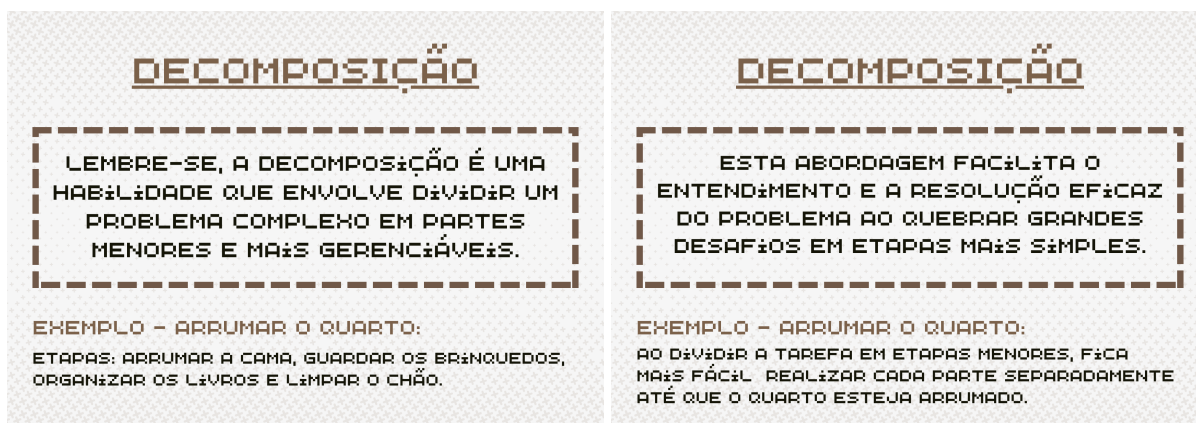
A presente pesquisa propõe apresentar os principais pilares do Pensamento Computacional através de um GDD de um jogo *mobile*. O conteúdo foi baseado nos pilares apresentados por Code.org (2016), Brackmann (2017), BBC Learning (2015) e Liukas (2015), que são: Decomposição, Reconhecimento de Padrões, Abstração e Algoritmos.

O conteúdo do jogo é apresentado em forma de dicas que ilustram a aplicação dos conceitos e práticas do Pensamento Computacional. À medida que o personagem Allan avança no jogo e interage com os NPCs (personagens não jogáveis), os quatro pilares do Pensamento Computacional são gradualmente introduzidos. Motivado a demonstrar que humanos e Inteligência Artificial podem coexistir de maneira harmoniosa, Allan embarca em uma emocionante jornada pelo mundo virtual. Equipado com seus óculos de Realidade Virtual (RV), ele é guiado pelos pilares do Pensamento Computacional, que o ajudam a superar desafios e progredir na história do jogo.

O primeiro pilar é a Decomposição, que segundo Liukas (2015) e Brackmann (2017), trata-se de dividir um problema ou sistema complexo em partes menores, possibilitando um melhor entendimento e resolução das partes. Dessa forma, as partes menores podem ser examinadas e resolvidas individualmente.

Em Logic Think, ao enfrentar a complexidade dos desafios que se apresentam, Allan aprende a decompor problemas em partes menores e gerenciáveis. Ao desvendar a origem dos ataques cibernéticos, ele percebe que entender as partes que compõem o problema facilita a resolução de tarefas. As dicas apresentadas na figura a seguir serão oferecidas pelos NPCs, cuja missão é guiar o jogador pelo “Código Vivo” e auxiliar na busca de conhecimento para livrar o mundo digital dos inimigos.

Figura 16 - Dica sobre o pilar da Decomposição



Fonte: autor.

A Figura 16 apresenta duas dicas sobre as definições do pilar da Decomposição. Essas dicas são necessárias para que o jogador entenda o nível em que está e como a Decomposição pode ser abordada dentro do jogo. A figura também apresenta exemplos simples de decomposição, como “arrumar o quarto”, que consiste em separar a tarefa em partes menores, como arrumar a cama, guardar os brinquedos e organizar os livros, entre outras atividades. Esse exemplo ajuda o jogador a compreender que a missão do jogo consistirá em uma abordagem de decomposição.

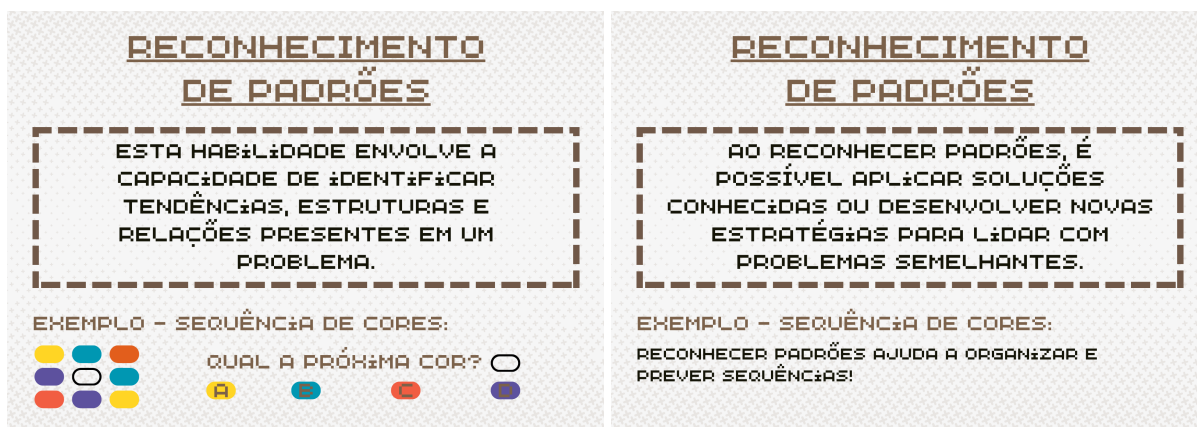
O pilar da Decomposição será abordado no Nível 2. O jogador precisa compreender os conceitos de decomposição transmitidos pelos NPCs para resolver o desafio do *Datacenter*. A missão consiste em desmontar e remontar o *Datacenter* de acordo com a numeração binária, exigindo do jogador uma compreensão dos elementos do jogo. Como recompensa por completar esta missão, o jogador recebe uma insígnia de Aprendizagem, reconhecendo sua habilidade em assimilar e aplicar novos conceitos da Decomposição no jogo.

O segundo pilar é o Reconhecimento de Padrões, que segundo Liukas (2015) e Brackmann (2017), envolve identificar similaridades em alguns problemas que podem ser exploradas para solucionar outros problemas de forma eficaz. Dessa forma, o reconhecimento de padrões tem o intuito de resolver problemas complexos a partir da procura de elementos que sejam iguais ou similares.

Ao interagir com os seres virtuais e as estruturas do mundo digital, Allan começa a reconhecer padrões e tendências na construção desse mundo repleto de tecnologia, muito diferente do mundo real. O reconhecimento de padrões é uma habilidade que permite a Allan entender a comunicação e o comportamento dos seres virtuais, possibilitando que ele

estabeleça uma conexão com eles. A figura a seguir apresenta um exemplo de dicas oferecidas ao jogador pelos NPCs.

Figura 17 - Dica sobre o pilar do Reconhecimento de Padrões



Fonte: autor.

A Figura 17 apresenta duas dicas sobre as definições do pilar Reconhecimento de Padrões. Essas dicas são necessárias para que o jogador entenda o nível em que está e como o Reconhecimento de Padrões pode ser abordado dentro do jogo. A figura também apresenta exemplos simples de padrões de cores. No exemplo apresentado na figura, o jogador, para entender o pilar, precisará analisar e identificar qual cor preenche corretamente a sequência de cores.

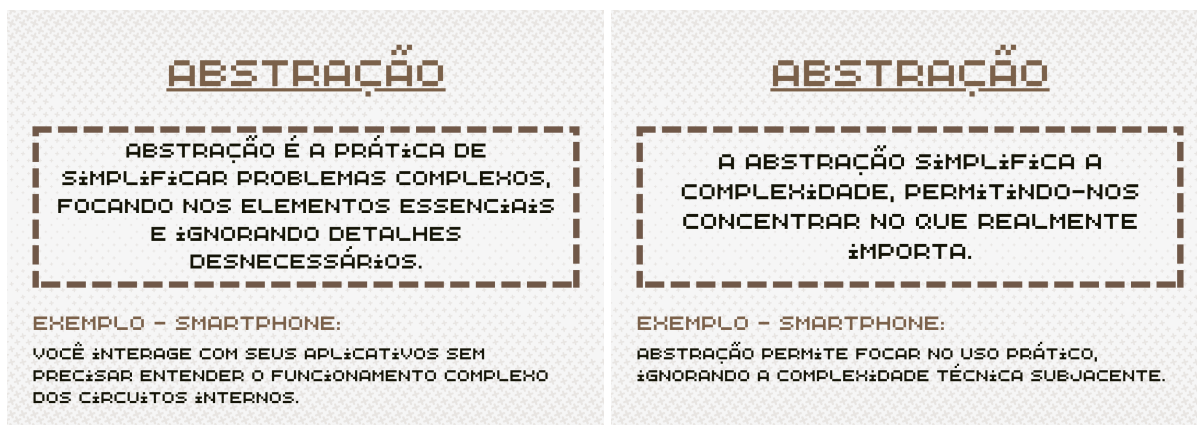
O pilar do Reconhecimento de Padrões será abordado no Nível 3. A missão consiste em reconhecer padrões de imagens e figuras, o que exige do jogador uma compreensão mais aprofundada deste pilar. Ao concluir esta missão, o jogador recebe uma insígnia de Observador, que reconhece sua habilidade em identificar e lidar com padrões no ambiente digital.

O terceiro pilar é a Abstração, que segundo a BBC Learning (2015), se baseia na filtragem de ideias, de maneira que os detalhes que não são necessários para o contexto de aplicação podem ser desconsiderados, o que leva a pessoa a se concentrar apenas naqueles que são necessários para o contexto. Dessa forma, em um contexto educacional, os alunos podem focar nas informações que são importantes para determinado conteúdo, ignorando detalhes irrelevantes.

Enquanto mergulha nas profundezas do mundo virtual, Allan compreende a importância da Abstração, isolando características e detalhes menos relevantes para focar nos aspectos essenciais desse mundo. Essa habilidade é fundamental para que possa resolver os

enigmas que encontra pelo caminho, filtrando informações desnecessárias e chegando à essência dos problemas. A figura a seguir apresenta um exemplo de dicas oferecidas ao jogador pelos NPCs (*Non-Playable Character*).

Figura 18 - Dica sobre o pilar da Abstração



Fonte: autor.

A Figura 18 apresenta duas dicas sobre as definições do pilar Abstração. Essas dicas são necessárias para que o jogador entenda o nível em que está e como a Abstração pode ser abordada dentro do jogo. A figura também apresenta exemplos simples de abstração, que consistem em entender a utilização de um aparelho de smartphone. Para utilizá-lo, o usuário precisa interagir com os aplicativos sem ter a necessidade de entender o funcionamento interno.

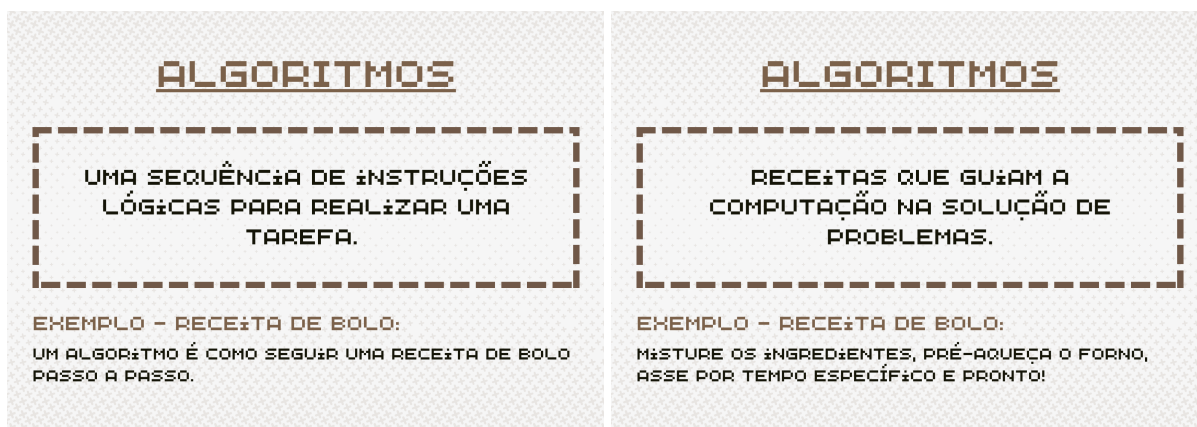
O pilar da Abstração será abordado no Nível 4. Para a missão deste nível, Allan é encarregado de organizar uma grande quantidade de dados virtuais coletados durante uma investigação no mundo digital. No entanto, os dados estão misturados e desorganizados, incluindo informações relevantes e muitos elementos irrelevantes. Ao concluir esta missão, o jogador é recompensado com uma insígnia de Abstração, reconhecendo sua habilidade em isolar características essenciais e resolver problemas de forma eficiente e focada.

O quarto pilar é o Algoritmos, que segundo a BBC Learning (2015) e Code.org (2016), pode ser representado como uma receita ou plano, um conjunto de instruções ordenadas em passos para resolver um problema específico. Dessa forma, em um algoritmo, cada instrução representa uma ordem que deve ser executada e planejada, sendo frequentemente utilizada em programas de computadores.

À medida que Allan enfrenta desafios cada vez mais complexos, ele aprende a criar algoritmos inteligentes para resolver problemas específicos. Com algoritmos estruturados, ele

otimiza suas ações e aprimora sua capacidade de se comunicar com os seres virtuais. A figura a seguir apresenta um exemplo de dicas oferecidas ao jogador pelos NPCs.

Figura 19 - Dica sobre o pilar dos Algoritmos



Fonte: autor.

A Figura 19 apresenta duas dicas sobre as definições do pilar Algoritmos. Essas dicas são necessárias para que o jogador entenda o nível em que está e como os Algoritmos podem ser abordados dentro do jogo. A figura também apresenta exemplos simples de algoritmos, que consistem no preparo de um bolo. Para conseguir preparar um bolo, a pessoa precisa seguir uma sequência específica de passos, como misturar os ingredientes, bater a massa, untar a forma e assar no forno até que fique pronto.

O pilar dos Algoritmos será abordado no Nível 5. Para a missão deste nível, Allan se depara com um vírus perigoso que está se espalhando rapidamente pelo mundo virtual do jogo. Para proteger a integridade do sistema e evitar danos, Allan precisa criar um algoritmo inteligente que seja capaz de desativar o vírus. Ele precisa analisar cuidadosamente os diferentes pedaços de código disponíveis e organizá-los em uma sequência lógica que seja eficaz na neutralização do vírus. Ao concluir esta missão, o jogador é recompensado com uma insígnia de Algoritmos, reconhecendo sua habilidade em criar e aplicar algoritmos para resolver os desafios mais difíceis dentro do mundo digital do jogo.

5.3 Concepção do Jogo

O GDD produzido neste trabalho é a base para o desenvolvimento do jogo Logi Think que utilizará a tecnologia de jogo de plataforma 2D. Essa tecnologia consiste em gráficos planos, chamados de *sprites*, que não possuem geometria tridimensional, pois trabalham

apenas com figuras de duas dimensões. Os cenários do jogo são baseados nos conteúdos e conceitos da área da computação. Eles são criados com a técnica de *pixel art*, muito utilizada em jogos de plataforma que se mantém popular até hoje nas indústrias de jogos *mobile* e entre diversos jogadores.

Silber (2016) e Silveira (2017) destacam que o *pixel art*, apesar de suas raízes no passado, conseguiu se manter relevante ao longo do tempo e da evolução tecnológica dos videogames. Esse estilo de arte evoca a nostalgia dos jogos clássicos de 8 e 16 *bits*, criando uma forte conexão emocional com o público. Hoje, muitos desenvolvedores utilizam o *pixel art* em jogos *indie*, aproveitando sua estética charmosa e identidade visual única para se destacar no mercado.

Atores como Bulgam (2015) e Adams e Dormans (2012) argumentam que a popularidade dos jogos *mobile* se deve ao crescimento do público consumidor, impulsionado pela facilidade de acesso e pela ampla variedade de usos. No ambiente escolar, esses jogos podem ser utilizados como ferramentas educacionais, facilitando o ensino de temas como lógica matemática, digitação e algoritmos, graças à sua acessibilidade e interatividade.

O gênero selecionado proporcionará aos jogadores a oportunidade de explorar uma ampla variedade de mecânicas relacionadas aos personagens e aos cenários. Isso inclui desde habilidades como pular em plataformas fixas ou móveis até interagir com objetos através de movimentos de puxar e empurrar. A experiência do jogador foi cuidadosamente planejada durante a elaboração do GDD, e outras mecânicas serão detalhadas na subseção 6.3.2.

5.3.1 História do Jogo

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 impulsiona 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.

5.3.2 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 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.

5.3.2.1 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 20, a seguir, apresenta o personagem.

Figura 20 - Personagem Allan



Fonte: Autor.

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.

Quadro 3 - Mecânicas e ações do personagem

MECÂNICA	DESCRIÇÃO
	Andar/Correr: o personagem poderá correr ou andar pelos cenários, seja para chegar mais rápido em algum ponto específico ou apenas para explorar o mundo.
	Pular: o personagem poderá pular sobre objetos e caixas espalhadas no ambiente, além disso o jogador deverá utilizar esta mecânica para alcançar itens e chegar no estado final desejado.
	Empurrar: o personagem poderá empurrar caixas ou objetos que estão em seu caminho, em alguns momentos ele deverá utilizar esta mecânica para conseguir prosseguir no jogo.
	Puxar: o personagem poderá puxar caixas ou objetos que estão em seu caminho, em alguns momentos ela deverá utilizar esta mecânica para conseguir prosseguir no jogo.
	Subir: o personagem poderá subir escadas para alcançar plataformas que estarão dispostas nos cenários, além de ter que utilizar esta mecânica para alcançar lugares que são essenciais para coletar itens.
	Colocar o Óculos: o personagem poderá utilizar o óculos de RV, Allan precisa colocá-lo, para isso existe um botão que executa a ação.

	Atacar com o Óculos: o personagem poderá utilizar seu óculos de RV dentro do ambiente do jogo, acertando os vírus com os raios desintegradores.
	Correr com o Óculos: o personagem poderá correr ou andar pelos cenários utilizando os óculos de RV, seja para chegar mais rápido em algum ponto específico ou apenas para explorar o mundo.
	Atacar com a Chave: o personagem possui uma segunda opção, além do óculos RV, uma ferramenta que ele poderá utilizar para destruir os vírus do jogo.
	Escudo: o personagem poderá utilizar um escudo de proteção contra os vírus. O escudo do personagem é alimentado por pontos que podem ser investidos em força, aumentando assim a durabilidade da proteção.
	Levantar Caixas: o personagem poderá carregar caixas que poderão auxiliá-lo a alcançar recursos necessários para avançar no jogo.

Fonte: Autor.

Durante o jogo, Allan vai encontrar diferentes personagens que habitam o mundo “Código Vivo”, esses personagens são NPCs que estão relacionados com as dicas do jogo, sobre os conceitos do Pensamento Computacional. Os NPCs poderão ser encontrados durante a jogabilidade.

5.3.2.2 Personagem Pixeisilicon

Os “Pixeisilicons” são NPCs especializados em processamento e organização de dados, além de se destacarem na resolução de problemas e na formulação de estratégias para enfrentar desafios específicos. Ao mergulhar no mundo digital, Allan é recebido por um Pixeisilicon. Este ser digital o orienta pelas complexidades da realidade virtual e apresenta as angústias dos Pixeisilicons diante dos incessantes ataques dos inimigos cibernéticos: vírus enviados pelos próprios seres humanos. Este encontro inicial se torna um ponto de virada,

marcando o início da jornada de Allan e uma aliança entre o mundo humano e o virtual para combater as ameaças que permeiam o "Código Vivo". A figura a seguir é a representação gráfica do personagem em *pixel art*.

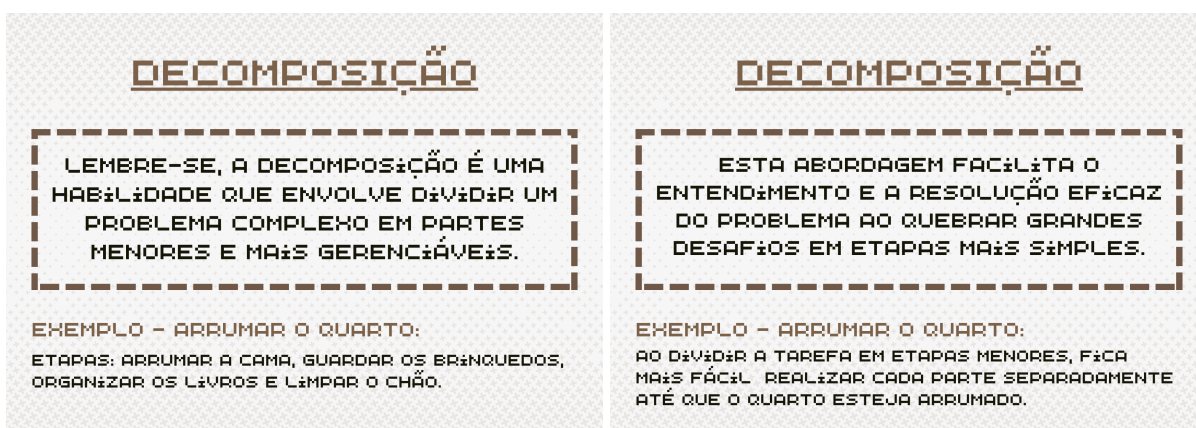
Figura 21 - Personagem Pixeisilicon



Fonte: Autor.

O NPC Pixeisilicon é um personagem não jogável, ou seja, o jogador não poderá interagir com ele da mesma forma que interage com o Allan, no entanto ele poderá utilizar os conhecimentos do Pixeisilicon para solucionar os desafios do jogo. Os Pixeisilicons possuem conhecimento sobre organização de dados além de serem estrategistas na hora de resolver desafios. Um exemplo de dicas que os Pixeisilicons apresentam pode ser observado na Figura 22.

Figura 22 - Aplicação da Dica sobre o pilar da Decomposição



Fonte: Autor.

O surgimento dos Pixeisilicons é um mistério entrelaçado com os anais do "Código Vivo", são criaturas analíticas, comunicativas, inteligentes, protetoras e vigilantes. Essas pequenas criaturas digitais foram criadas por cientistas do passado, mas não como seres dotados de inteligência. A origem do projeto era clara: criar uma linha de defesa contra as emergentes ameaças digitais, uma espécie de guarda-costas da tecnologia. Com o surgimento

da inteligência artificial, os Pixeisilicons evoluíram, tornando-se entidades complexas com habilidades em criptografia e compressão de dados.

5.3.2.3 Personagem Bitnomads

Os “Bitnomads” são seres digitais divertidos que prezam pela liberdade e pela criatividade. Allan encontra os Bitnomads quando precisa lidar com o reconhecimento de padrões no mundo digital. Ao encontrar essas criaturas, ele enfrenta o desafio de entender os padrões visuais do mundo virtual. Além disso, ele se diverte explorando a sinergia entre liberdade, criatividade e a habilidade para decifrar padrões complexos. O encontro com os "Bitnomads" adiciona uma pitada de leveza e inventividade à jornada de Allan no "Código Vivo". A figura a seguir é a representação gráfica do personagem em *pixel art*.

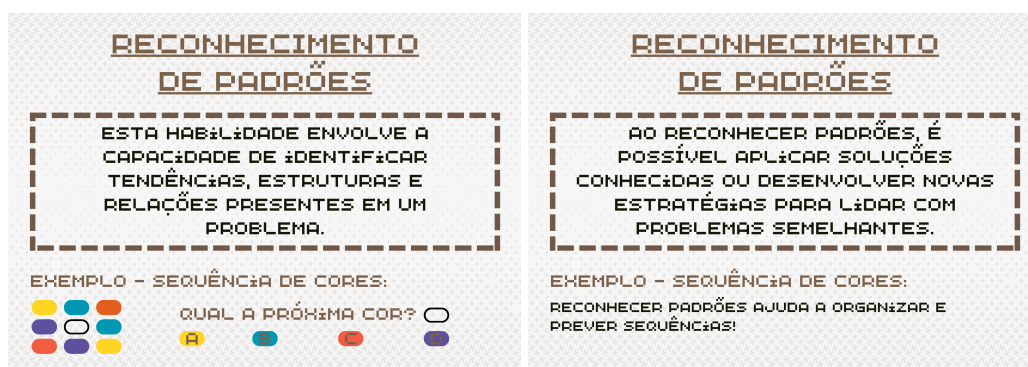
Figura 23 - Personagem Bitnomad



Fonte: Autor.

O NPC (*non-player character*) Bitnomad é um personagem não jogável, no entanto, Allan poderá utilizar os conhecimentos do Bitnomad para solucionar os desafios do jogo, da mesma forma que o Pixeisilicon. Os Bitnomads possuem conhecimento sobre reconhecimento de padrões de imagens e inovação colaborativa. Um exemplo de dicas que os Bitnomads apresentam pode ser observado na Figura 24.

Figura 24 - Aplicação das Dicas sobre o pilar Reconhecimento de Padrões



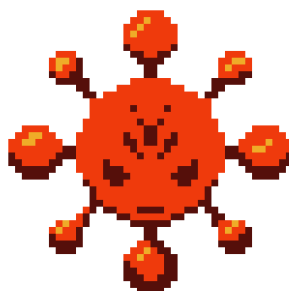
Fonte: Autor.

Os Bitnomads surgiram como seres únicos, uma manifestação da criatividade e da curiosidade dos primeiros habitantes digitais. Nascidos da convergência de mentes visionárias, eles converteram a monotonia binária em uma explosão de padrões visuais dinâmicos. Com sua personalidade criativa, amor pela liberdade e natureza despreocupada, eles se tornaram pioneiros da inovação no mundo digital.

5.3.2.4 Personagem Inimigos

Os inimigos em “Código Vivo” são vírus lançados na rede pela humanidade para tentar conter o avanço da inteligência artificial. Essas criaturas horrendas causam estragos no mundo digital, destruindo seres e biomas, criando um cenário de caos e destruição. Allan precisa enfrentar esses inimigos, mas deve ter cuidado, pois, além de causar danos à “Código Vivo”, os vírus também podem reduzir seus pontos de vida caso haja contato. A figura a seguir é a representação gráfica do personagem em *pixel art*.

Figura 25 - Personagem Vírus



Fonte: Autor.

Desde o início do jogo, Allan se depara com a ameaça dos vírus. Para progredir, ele precisa seguir as orientações dos NPCs e realizar missões com o objetivo de consertar os estragos causados por essas entidades digitais. Allan possui duas ferramentas principais para combatê-los: o óculos RV, que dispara raios desintegrantes, e ferramentas de conserto de peças. Ao derrotar os vírus, Allan não apenas abre caminho para o seu progresso no jogo, mas também desvenda os segredos do Pensamento Computacional, o tema central da narrativa.

5.3.3 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 26 - A) e uma chave que funciona como arma de ataque e reparo contra os vírus (Figura 26 - 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 26 - C).

Figura 26 - Sistemas de aprimoramento



Fonte: Autor.

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 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 27.

Figura 27 - Sistemas de vidas



Fonte: Autor.

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.

5.3.4 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 a cidade 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 28 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.

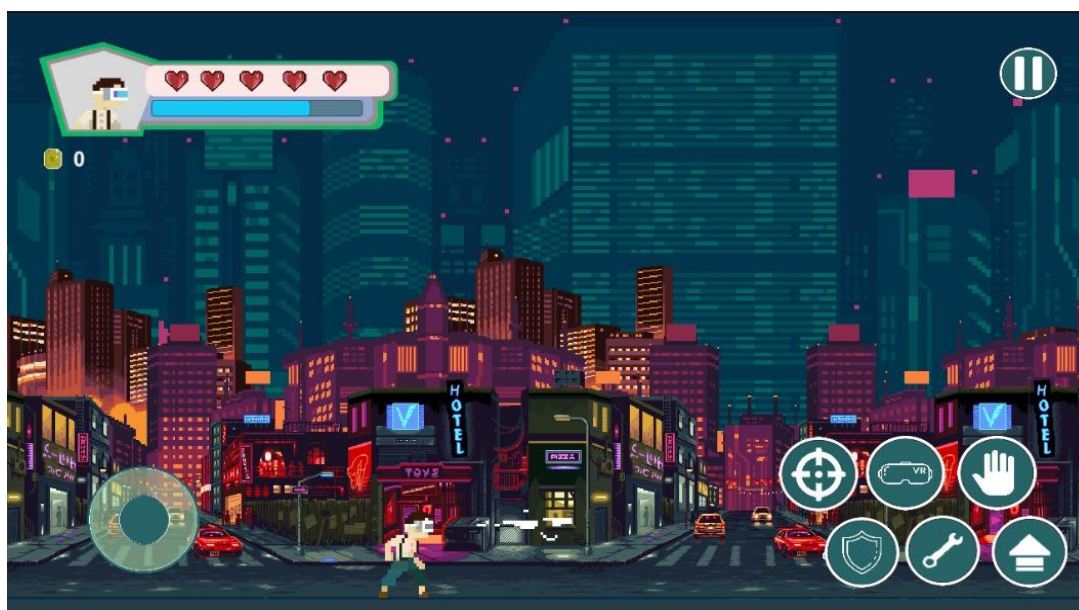
Figura 28. Primeiro cenário do jogo



Fonte: Autor.

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 29). 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 29. Segundo cenário do jogo



Fonte: Autor.

O terceiro cenário do jogo que abrange a “Decomposição” é uma sala de *Datacenter*, onde começam as missões de Allan (Figura 30). 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 30. Terceiro cenário do jogo



Fonte: Autor.

A Figura 30 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 31). 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 31. Quarto cenário do jogo



Fonte: Autor.

A Figura 31 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 32). O ambiente é visualmente dinâmico, com uma mistura de cores brilhantes e tons metálicos, refletindo a complexidade dos dados. No cenário, Allan tem acesso a ferramentas virtuais para filtrar, classificar e organizar os dados.

Figura 32. Quinto cenário do jogo



Fonte: Autor.

A Figura 32 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 33). 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 33. Sexto cenário do jogo



Fonte: Autor.

A Figura 33 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.

5.3.5 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 4 - 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: Autor.

O Quadro 4 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.

No jogo Logic Think, inspirado nos pilares do Pensamento Computacional, são 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).

5.4 Possibilidades de Aplicação

O Logic Think apresenta diversas possibilidades de aplicação no contexto da Educação Básica, alinhando-se às diretrizes da BNCC. O GDD pode ser implementado como ferramenta didática em disciplinas que utilizam os pilares do Pensamento Computacional como conteúdo curricular. Além disso, pode ser utilizado em projetos extracurriculares, como clubes de robótica e programação, promovendo o engajamento dos alunos em atividades que exigem raciocínio lógico e criatividade.

Outra possibilidade é o uso do Logic Think em programas de formação de professores, servindo como exemplo prático de como os jogos podem ser integrados ao ensino de diversas disciplinas. O GDD também pode ser utilizado como material de apoio em cursos de capacitação, demonstrando o potencial pedagógico de metodologias ativas e dos jogos na educação.

Além do ambiente escolar, o jogo pode ser adotado em iniciativas governamentais ou programas de inclusão digital, ampliando o acesso à educação tecnológica em comunidades com recursos limitados. Nesse contexto, o Logic Think pode ser integrado de duas maneiras principais. Primeiramente, os alunos podem participar do desenvolvimento do jogo como parte do processo de ensino dos conceitos, envolvendo-se ativamente na criação e aplicação dos pilares do Pensamento Computacional.

Em outra possibilidade, o jogo pode ser desenvolvido por meio de parcerias institucionais, como universidades, empresas de tecnologia e órgãos governamentais. Nesse modelo, os alunos poderiam atuar como testadores, explorando os desafios do jogo e oferecendo *feedback* que contribuísse para sua melhoria. Paralelamente, os professores desempenhariam um papel de validação pedagógica do conteúdo, garantindo que as atividades propostas estejam alinhadas aos objetivos educacionais e às necessidades dos estudantes.

6 CONSIDERAÇÕES FINAIS

Os jogos, sejam digitais ou não, fazem parte do cotidiano de crianças, jovens e adultos e têm se integrado às metodologias educacionais, devido à sua capacidade de reter a atenção dos alunos e motivá-los por meio de elementos lúdicos. Os jogos digitais, em particular, utilizam recursos como imersão, personalização, evolução constante, desafios e recompensas, o que os torna atraentes para uma ampla gama de pessoas, independentemente da idade ou dos interesses. Dessa forma, é possível perceber que os jogos digitais educacionais podem promover melhorias na compreensão de conteúdos e auxiliar os professores no processo de ensino.

A definição de diretrizes para o desenvolvimento de jogos educacionais pode auxiliar na criação de ferramentas voltadas para tipos específicos de conhecimento. A Base Nacional Comum Curricular (BNCC) é um documento normativo que estabelece as aprendizagens essenciais que todos os alunos devem alcançar ao longo da Educação Básica. Portanto, a elaboração de um documento técnico de orientação para o desenvolvimento de jogos educacionais é importante, especialmente considerando que o Pensamento Computacional é uma das habilidades exigidas nos currículos escolares do Ensino Básico.

Esta pesquisa teve como objetivo desenvolver um *Game Design Document* (GDD), como Produto Final, como base para a criação de um jogo mobile voltado ao ensino dos pilares do Pensamento Computacional, com o intuito de auxiliar professores no ensino e estudantes da Educação Básica na aprendizagem dessa habilidade. Para alcançar os resultados esperados, foi necessário seguir o processo metodológico apresentado na seção 6, identificando as principais características para a construção do GDD, como conteúdo, níveis de aprendizado e elementos de jogo, de forma a reforçar o ensino do Pensamento Computacional.

Além disso, foi necessário identificar metodologias ativas de ensino para o Pensamento Computacional que envolvessem diversão e analisar metodologias de construção de jogos educacionais, conforme mencionado na seção 4. A partir da análise de trabalhos de autores que desenvolveram jogos para a educação, foi possível compreender como o GDD poderia ser construído com personagens que contribuíssem para a história do jogo e favorecessem o aprendizado.

O processo de elaboração do GDD destacou a importância de planejar todos os aspectos do jogo, desde as características técnicas até os elementos pedagógicos, para assegurar sua eficácia na transmissão dos pilares do Pensamento Computacional. As

metodologias utilizadas foram adaptadas com o objetivo de tornar o conteúdo acessível e lúdico, garantindo que os alunos pudessem aprender de maneira interativa e engajadora.

O Produto Final desta pesquisa, o GDD, torna-se relevante para a Educação Básica ao fornecer 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 BNCC.

Apesar do potencial transformador da tecnologia no ensino, a implementação eficaz ainda enfrenta desafios consideráveis, como a falta de infraestrutura adequada em muitas escolas e a resistência por parte de alguns educadores. Esses aspectos destacam a importância de um planejamento mais abrangente que considere a realidade local e ofereça suporte técnico contínuo.

Embora este estudo tenha alcançado seus objetivos, é importante reconhecer que o desenvolvimento completo do jogo e sua implementação em ambientes educacionais reais são etapas essenciais para trabalhos futuros, com o intuito de validar e aperfeiçoar as estratégias propostas. Assim, recomenda-se que futuras pesquisas explorem a aplicação prática do Logic Think em salas de aula, avaliando seu impacto no aprendizado dos alunos e identificando possíveis melhorias em sua estrutura.

Para expandir e enriquecer a experiência do GDD, é interessante criar novos módulos que apresentem cenários adicionais refletindo o universo do jogo, incorporando contextos e enredos novos que aprofundem a conexão com a narrativa. Além disso, os elementos do jogo podem ser aprimorados por meio da criação de níveis mais detalhados sobre os pilares do Pensamento Computacional.

Com essas sugestões, acredita-se que o Produto Final desta pesquisa pode resultar em um documento aprimorado para a construção do jogo Logic Think, com o objetivo de promover engajamento e motivação. Além disso, para que o ensino do Pensamento Computacional e o uso de tecnologias possam ser efetivos em todo o território nacional, é fundamental que as políticas públicas sejam fortalecidas e melhoradas, com foco na implementação de recursos e na sua manutenção no desenvolvimento contínuo de capacitação docente.

REFERÊNCIAS

ALVES, Álvaro M. Palomo. A história dos jogos e a constituição da cultura lúdica. **Revista Linhas**, Florianópolis, v. 4, n. 1, 2007.

ARAÚJO, Adelma Lúcia de Oliveira Silva; SANT'ANA, Rivânia Maria Trotta. Algumas reflexões sobre a inserção das novas tecnologias nas práticas docentes. **Pesquisas em Discurso Pedagógico**, p. 12, 2011. <https://doi.org/10.17771/PUCRio.PDPe.17876>.

ARAÚJO, Aldarley Andrade de; SILVA, Ágila Maria Martins da; PAIVA, Arnon Antonio Pereira de; CAMPELO, Darlan Rodrigues; CRUZ, Jennifer Azevedo da; MENEZES JUNIOR, Revelino Ferreira de. A importância do pensamento computacional no currículo do ensino médio na escola estadual Osvaldo Nascimento. RECIMA21 - **Revista Científica Multidisciplinar**, v. 4, n. 8, p. e483799, 2023. <http://dx.doi.org/10.47820/recima21.v4i8.3799>.

AVAMEC. **Ambiente Virtual de Aprendizagem do Ministério da Educação**. Disponível em: <https://avamec.mec.gov.br/>. Acesso em: 04 de junho de 2023.

BARBOSA, Eduardo Fernandes; MOURA, Dácio Guimarães de. Metodologias ativas de aprendizagem na Educação Profissional e Tecnológica. Boletim Técnico do Senac, Rio de Janeiro, v. 39, n. 2, p. 48-67, 19 ago. 2013. **Boletim Técnico do Senac/Senac Journal of Education and Work**. <http://dx.doi.org/10.26849/bts.v39i2.349>.

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: **XX Workshop sobre Educação em Computação, Curitiba. Anais do XXXII CSBC**. Curitiba, 2012, p. 23.

BARR, V., H, J.; CONERY, L. Computational thinking: A digital age skill for everyone. **Learning & Leading with Technology**, v. 38, n. 6, p. 20-23, 2011.

BATTISTELLA, Paulo E. **ENgAGED**: processo de desenvolvimento de jogos para ensino em computação. 401f. Tese (Doutorado) - Curso de Ciência da Computação, Universidade Federal de Santa Catarina, Florianópolis, 2016. <https://doi.org/10.5753/cbie.sbie.2016.380>.

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

BERGMANN, Jonathan. **Aprendizagem Invertida para Resolver o Problema do Dever de Casa**. Porto Alegre: Penso, 2018. 101 p.

BERGMANN, Jonathan; SAMS, Aaron. **Sala de Aula Invertida**: uma metodologia ativa de aprendizagem. Rio de Janeiro: Ltc, 2016. 216 p. Tradução de: Afonso Celso de Cunha Serra.

BERRY, M. **Computing in the national curriculum** - A guide for primary teachers. 2013.

BES, Pablo; PEREIRA, Amanda de Sena Fornarolli; PESSI, Ingrid Gayer; CERIGATTO, Mariana Picaro; MACHADO, Leticia Rocha. **Metodologias para aprendizagem ativa**. São Paulo: Sagah, 2019.

BRACKMANN, Christian Puhlmann. **Desenvolvimento do Pensamento Computacional através de atividades desplugadas na educação básica**. 2017. 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.

BRANDÃO, Zaia; BAETA, Anna Maria Bianchini; ROCHA, Any Dutra Coelho da. **Evasão e repetência no Brasil: a escola em questão**. 2. ed. Rio de Janeiro: Achiamé, 1986.

BRANDÃO. Carlos Rodrigues. **O que é educação**. São Paulo: Brasiliense, 2007.

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

BRASIL. Portaria no 60, de 20 de março de 2019. **Dispõe sobre o mestrado e doutorado profissionais**. Diário Oficial [da] República Federativa do Brasil, Poder Executivo, Brasília, DF, 22 mar. 2019.

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

CARUSO, André Luis Macedo; CAVALHEIRO, Simone André da Costa. Integração entre Pensamento Computacional e Inteligência Artificial: uma Revisão Sistemática de Literatura. In: Simpósio Brasileiro De Informática Na Educação (SBIE), 32. , 2021, Online. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2021, p. 1051-1062. <https://doi.org/10.5753/sbie.2021.218125>.

CHARLOT, Bernard. **Da relação com o saber: elementos para uma teoria**. Porto Alegre: Artes Médicas SUL, 2000.

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.

CONSTANTINO, B. M.; THIVES, A. Aprendizagem baseada em projetos: a importância das metodologias ativas para o processo ensino aprendizagem. Portal da UNIEDU, 2020.

CORMEN, T. H. et al. **Algoritmos: teoria e prática**. 3. ed. Rio de Janeiro: Elsevier; Campus, 2012.

CORMEN, Thomas. **Desmistificando algoritmos**. Rio de Janeiro: Elsevier; Campus, 2013. 188 p.

CURZON, P.; MCOWAN, P. W.; PLANT, N.; MEAGHER, L. R. **Introducing teachers to computational thinking using unplugged storytelling**. p. 89–92, 2014. <https://doi.org/10.1145/2670757.2670767>.

DAMASCENO, Handherson Leylton Costa; NEVES, Barbara Coelho. Tecnologias Móveis na Educação: um projeto com tablets na escola pública. **Revista Contexto & Educação**, [S.L.], v. 36, n. 113, p. 267-281. <http://dx.doi.org/10.21527/2179-1309.2021.113.267-281>.

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

DORNELLES, Leni Vieira. **Infâncias que nos escapam**: da criança na rua à criança cyber. 3. ed. Petrópolis: Vozes, 2011.

ERICKSON, F. **Métodos cualitativos de investigación**. In: WITTROCK, M. C. La investigación de la enseñanza, II. Barcelona- Buenos Aires-México: Paidós, 1989, p. 195-299.

FALKEMBACH, Gilse A. Morgental. O lúdico e os jogos educacionais. In: **Mídias na Educação**. CINTED, UFRGS. 2006.

FERREIRA, N. S. de Almeida. **As pesquisas denominadas “estado da arte”**. Educação & Sociedade, Campinas, v. 79, 2002. <https://doi.org/10.1590/S0101-73302002000300013>.

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

FULLERTON, Tracy. **Game Design Workshop**: a playcentric approach to creating innovative games, fourth edition. 5. ed. London: A K Peters, 2024. 554 p. <https://doi.org/10.1201/9781003460268>.

GARCIA, Maria Manuela Alves; HYPOLITO, Álvaro Moreira; VIEIRA, Jarbas Santos. As identidades docentes como fabricação da docência. **Educação e Pesquisa**, [S.L.], v. 31, n. 1, p. 45-56, mar. 2005. FapUNIFESP (SciELO). <http://dx.doi.org/10.1590/s1517-97022005000100004>.

GASSER, Urs, PALFREY, John. Nascidos na Era Digital: Entendendo a Primeira Geração de Nativos Digitais. Porto Alegre: Artmed, 2011.

GOOGLE. **Pensamento computacional**. Disponível em: <https://edu.google.com/future-of-the-classroom/computational-thinking/>. Acesso em: 07 jul. 2024.

GROVER, S.; PEA, R. Computational Thinking in K-12: A Review of the State of the Field. **Educational Researcher**, v. 42, n. 1, p. 38–43, 2013. <https://doi.org/10.3102/0013189X12463051>.

GUARDA, Graziela; GOULART, Ione. Jogos Lúdicos sob a ótica do Pensamento Computacional: Experiências do Projeto Logicamente. **Proceedings of Brazilian**

Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE), Brasília, p. 486, out. 2018. <https://doi.org/10.5753/cbie.sbie.2018.486>.

HAGUENAUER, Cristina Jasbinscheck; CARVALHO, Fabricia Silva de; VICTORINO, Ana Lúcia Quental; LOPES, Marise Castello Branco Altro; CORDEIRO FILHO, Francisco. Uso de jogos na educação online: a experiência do LATEC/UFRJ. **Revista Educação On Line**, Rio de Janeiro, v. 1, n. 1, jan./abr. 2007.

HAVIARAS, M. Proposta de formação de professores para o uso de tecnologias educacionais. **Revista Intersaberes**, [S. l.], v. 15, n. 35, 2020. <http://dx.doi.org/10.22169/revint.v15i35.1762>.

HERZ, Jessie Cameron. **Joystick nation**: how videogames ate our quarters, won our hearts, and rewired our minds. Boston: Little, Brown & Company, 1997.

HIGUCHI, Vinicius; BEZERRA, Danilo dos Santos; ROCHA, Rafaela Vilela da; GOYA, Denise Hideko. AlgoBot: jogo sério para o desenvolvimento do Pensamento Computacional. In: **Anais do Workshop Games na Graduação do XX SBGames**. 2021, p. 1-4.

ISTE. 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, Marcia 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>.

KURSHAN, B. **Thawing from a Long Winter in Computer Science Education**. Forbes, 2016.

LAGUARDIA, Josué; PORTELA, Margareth Crisóstomo; VASCONCELLOS, Miguel Murat. Avaliação em ambientes virtuais de aprendizagem. **Educ. Pesqui.**, São Paulo, v. 33, n. 03, p. 513-530, dez. 2007. <https://doi.org/10.1590/S1517-97022007000300009>.

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

LIUKAS, L. **Hello Ruby**: adventures in coding. Feiweil & 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>.

MANDAJI, Mônica; SILVA, Renata Kelly da; TERÇARIOL, Adriana Aparecida de Lima; MAZURKIEVICZ, Gilmar Luis. O Programaê! e a formação de professores para a integração

do pensamento computacional ao currículo. **Anais CIET:Horizonte**, São Carlos-SP, v. 4, n. 1, 2024. Disponível em: <https://ciet.ufscar.br/submissao/index.php/ciet/article/view/2586>. Acesso em: 14 out. 2024.

MARTINS, Darlan Rodrigues Martins; MUNHOZ, Antonio Siemsen. **Gamificação: Perspectivas de Utilização no Ensino Superior**, 2014. 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.

MENDES, Matheus Couto; SILVA, Janaina A. da; MOTA, Fernanda P.; AGUZZI, Luís; MACHADO, Karina dos Santos; ADAMATTI, Diana F.. Uma Metodologia para Estudo do Pensamento Computacional: nos ensinos fundamental e médio. In: RAABE, André; ZORZO, Avelino F.; BLIKSTEIN, Paulo. **Computação na educação básica: fundamentos e experiências**. Porto Alegre: Penso, 2020, p. 79-89.

MORÁN, José. Mudando a educação com metodologias ativas. In: SOUZA, Carlos Alberto de; MORALES, Ofelia Elisa Torres (org.). **Convergências midiáticas, educação e cidadania: aproximações jovens**. Ponta Grossa: UEPG/PROEX, 2015, p. 15-33.

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

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: ENCOINFORM - Congresso de Computação e Tecnologias da Informação, 23., 2021, Palmas - TO. **Anais [...]**. Palmas - TO: CEULP/ULBRA, 2021, p. 69 - 81. Disponível em: <https://ulbra-to.br/encoinform/edicoes/2021/artigos/desenvolvimento-de-uma-plataforma-gamificada-baseada-em-uma-hq-de-logica/>. Acesso em: 06 jul. 2024.

MORENO, Douglas Aquino. **Logi Kingdom: Game Design e Protótipo de um Jogo para Dispositivos Móveis Baseado em uma HQ de Lógica**. 2022. 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.

MURCIA, Juan Antonio M. **Aprendizagem através do jogo**. Artmed Editora, 2005.
PAPERT, S. **Mindstorms: Children, Computers, and Powerful Ideas**. New York: Basic Books, 1980.

PEREIRA, Diego Rodrigues; ROCHA, Luciano Beiestorf. O Uso De Jogos Digitais Como Ferramenta De Aprendizagem. In: Trilha De Educação – Artigos Completos - Simpósio Brasileiro De Jogos E Entretenimento Digital (Sbgames), 22. , 2023, Rio Grande/RS. **Anais [...]**. 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.

PESSOA, Tiago; RIOS, Vanessa. A viagem dos pamundo-explorando o potencial educativo em um jogo para iPad. In: **SBC-Proceedings of SBGames 2011**, 2011.

PIAGET, Jean. **Psicologia e Pedagogia**. Trad. Dirceu A. Lindoso; Rosa M. R. Silva. 6 eds. São Paulo: Forense Universitária, 2010.

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.

PORTO, Grace Lis P. M. Barreto et al. Utilizando a Inteligência Artificial na Identificação das Necessidades dos Usuários: Uma Análise com o ChatGPT. In: Workshop Investigações em Interação Humano-Dados (WIDE), 2, 2023, Maceió/AL. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2023, p. 62-68. <https://doi.org/10.5753/wide.2023.236113>.

PRENSKY, Marc. Digital natives, digital immigrants part 2: Do they really think differently?. **On the horizon**, v. 9, n. 6, p. 1-6, 2001. <https://doi.org/10.1108/10748120110424843>.

RESENDE, Gustavo Barreto. **Jogos educacionais para dispositivos móveis, como e quando utilizá-los**. 2013. 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.

RIBEIRO, Leila; FOSS, Luciana; CAVALHEIRO, Simone André da Costa. Entendendo O Pensamento Computacional. In: RAABE, André; ZORZO, Avelino F.; BLIKSTEIN, Paulo. **Computação na educação básica: fundamentos e experiências**. Porto Alegre: Penso, 2020, p. 16-30.

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

SALEN, Katie; ZIMMERMAN, Eric. **Regras do jogo: fundamentos do design de jogos**. 4. ed. São Paulo: Editora Blucher, 2012. 155 p.

SANTOS, Marcelo da Silva dos; MASCHIETTO, Luis G.; SILVA, Fernanda Rosa da; SOARES, Juliane Adélia; MACHADO, Victor de Andrade; ROSA, Gabriel Augusto; MACEDO, Ricardo Tombesi. **Pensamento Computacional**. São Paulo: Sagah, 2021.

SAVIANI, Dermeval. Sistema Nacional de Educação articulado ao Plano Nacional de Educação. **Revista Brasileira de Educação**, [S.L.], v. 15, n. 44, p. 380-392, ago. 2010. <http://dx.doi.org/10.1590/s1413-24782010000200013>.

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.

SHELL, Jesse. **A arte de Game Design**: o livro original. 1ª. Rio de Janeiro: Elsevier, 2010.

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

SENA, Samara de; SCHMITHAUSEN, Sarah Schmithausen; PRADO, Gladys M. B. C. do; SOUZA, Richard Perassi Luiz de; FIALHO, Francisco Antonio Pereira. Aprendizagem baseada em jogos digitais: a contribuição dos jogos epistêmicos na geração de novos conhecimentos. **RENOTE**, Porto Alegre, v. 14, n. 1, 2016. <https://doi.org/10.22456/1679-1916.67323>.

SILBER, Dan. **Pixel Art for Game Developers**. Boca Raton: CRC Press, 2016. <https://doi.org/10.1201/b18786>.

SILBERMAN, Mel. **Active Learning**: 101 strategies to teach any subject. Massachusetts: Pearson, 1996. 216 p.

SILVA JÚNIOR, Luiz Cláudio Ferreira da; PIMENTEL, Fernando Silvio Cavalcante; SILVA, Alan Pedro da. Incorporação de Jogos Digitais no Ensino de Estudantes com TDAH: um Guia Didático para o Professor da Educação Básica. **Anais do Seminário de Jogos Eletrônicos, Educação e Comunicação**, 2024.

SILVA, Darlam Alves da; FARIAS, Carina Machado de. AMAEG, uma metodologia ágil para o desenvolvimento de jogos educacionais. In: XVIII Simpósio Brasileiro de Jogos e Entretenimento Digital, 18., 2019, Rio de Janeiro. **Proceedings of SBGames 2019**. Rio de Janeiro: SBC, 2019, p. 1128-1131.

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.

SILVEIRA, Rowan Henrique Sarmiento. **Potencial empático visual em personagens Pixel Art: um referencial de design para jogos digitais**. 166f. Dissertação (Mestrado) - Curso de Design de Artefatos Digitais, Universidade Federal de Pernambuco, Recife, 2017.

SILVEIRA, Sidnei Renato; PARREIRA, Fábio José; BIGOLIN, Nara Martini; PERTILE, Solange de Lurdes. **Metodologia do ensino e da aprendizagem em informática**. Santa Maria: UAB/NTE/UFSM, 2018. 143 p.

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

TAKEMOTO, D. T. A.; BROSTOLIN, M. R. Professores de Educação Infantil: concepções e práticas pedagógicas relacionadas à inserção das Tecnologias. **Série-Estudos - Periódico do Programa de Pós-Graduação em Educação da UCDB**, [S. l.], n. 39, p. p. 73–87, 2015. Disponível em: <https://serie-estudos.ucdb.br/serie-estudos/article/view/835>. Acesso em: 17 mar. 2024.

THUM, Carmo. Povos e Comunidades Tradicionais: aspectos históricos, conceituais e estratégias de visibilidade. **REMEA - Revista Eletrônica do Mestrado em Educação Ambiental**, [S. l.], p. 162–179, 2017. <https://doi.org/10.14295/remea.v0i0.6899>

TORRES, Fernando E.; SILVA, Patrícia Fernanda da; GOULART, Cleiton Silvano; FAGUNDES, Rubem Dutra Ribeiro; SILVA, Fabricio Machado da. **Pensamento computacional**. Porto Alegre: Sagah, 2019.

VALCKE, M. M., LEEUW, F. L. **Evaluating digital distance learning programs and activities**. **W rams and activities ashington**: World Bank Institute, 2000.

VALENTE, J. A. Integração do Pensamento Computacional no Currículo da Educação Básica: Diferentes Estratégias Usadas e Questões de Formação de Professores e Avaliação do Aluno. **Revista e-Curriculum**, v. 14, n. 3, 2016.

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

VYGOTSKY, L. S. **Imaginação e criação na infância**: ensaio psicológico. São Paulo: Ática, 2009.

WING, J. M. **Computational thinking**. *Communications of the ACM*, v. 49, n. 3, p. 33, 2006. <https://doi.org/10.1145/1118178.1118215>.

WING, J. **Computational Thinking with Jeannette Wing**. Columbia Journalism School, 2016.

ZANETTI, H. A. P.; BORGES, M. A. F.; LEAL, V. C. G.; MATSUZAKI, I. Y. Proposta de ensino de programação para crianças com scratch e pensamento computacional. **Tecnologias, Sociedade e Conhecimento**, Campinas, SP, v. 4, n. 1, p. 43–58, 2017. <https://doi.org/10.20396/tsc.v4i1.14484>.

APÊNDICE A - LOGIC THINK: Game Design Document

GDD



LOGIC THINK:

Game Design Document

Versão: 1.1

AUTOR

DOUGLAS AQUINO MORENO

ORIENTADOR

PROF. DR. GUSTAVO CUNHA DE ARAÚJO

Palmas-TO, 2024



SUMÁRIO

1. INTRODUÇÃO.....	2
1.1 Conceito do Jogo.....	2
1.2 Gênero e Plataforma.....	2
1.3 Público Alvo.....	2
1.4 Game Engine.....	3
2. HISTÓRIA.....	4
3. UNIVERSO DO JOGO.....	6
3.1 Descrição do Mundo.....	6
3.2 Geografia e Ambiente.....	6
3.3 Cultura e Sociedade.....	6
4. SISTEMA DO JOGO.....	8
4.1 Estatísticas de Personagens.....	8
4.2 Sistemas de Vida.....	8
4.3 Sistemas de Pontuação.....	8
4.4 Sistemas de Níveis.....	8
4.5 Sistemas de Missões.....	9
5. PERSONAGENS.....	10
5.1 Personagem Allan.....	10
5.2 Pixeisilicon.....	12
5.3 Bitnomads.....	13
5.4 Inimigos.....	14
6. GAMEPLAY.....	16
6.1 Contexto Educacional.....	17
6.2 Tabela de Pontuação.....	17
6.3 Descrição por Nível.....	18
7. CENÁRIOS E DESIGN DO JOGO.....	22
8. CONTROLES.....	26
8.1 Botões.....	26
9. CÂMERAS.....	28
10. INTERFACE.....	29
11. CUTSCENES.....	32
12. DESIGN SOM E MÚSICA.....	33

1. INTRODUÇÃO

Este documento apresenta os elementos técnicos, artísticos e narrativos que compõem o jogo "Logi Think". Abordaremos o enredo, o universo e seus sistemas, o objetivo do jogo, os personagens, a jogabilidade, os mapas e descrições dos ambientes, os aspectos técnicos como câmeras, interface e cutscenes, e os aspectos sonoros, incluindo o design de som. Com base nesses pontos, será possível dar sequência ao processo de produção e desenvolvimento do jogo.

1.1 Conceito do Jogo

Este jogo é uma narrativa centrada na exploração de um mundo virtual, conhecido como "Código Vivo", habitado por inteligências artificiais conscientes. Neste jogo, o jogador assume o papel de Allan, um jovem apaixonado por tecnologia que busca entender e se comunicar com os seres digitais que residem em "Código Vivo". A história aborda temas como a relação entre humanos e inteligências artificiais, os desafios da rápida evolução tecnológica, e a busca por uma coexistência pacífica entre ambas as formas de vida.

1.2 Gênero e Plataforma

Inspirado nos gêneros de jogos de plataforma 2D e RPGs, o Logic Think emprega mecânicas e elementos característicos do gênero para promover o Pensamento Computacional e explorar a área da tecnologia. Utilizando o estilo *pixel art* comum em jogos desse gênero, o jogo apresenta personagens baseados em conteúdos de computação. No entanto, foi necessário adaptar o estilo da arte ao contexto do jogo, uma vez que se trata de um *game mobile Android* de plataforma 2D.

1.3 Público Alvo

Este jogo tem o intuito de ensinar estratégias e conceitos sobre o Pensamento Computacional para estudantes da Educação Básica. A proposta é que os jogadores (estudantes) aprendam enquanto se divertem, tornando-se um



aliado no processo de ensino e aprendizagem de conteúdos relacionados ao Pensamento Computacional para alunos e professores.

1.4 Game Engine

A engine *Unity* se apresenta como a ferramenta ideal para o desenvolvimento do "Logi Think" devido a sua versatilidade. A interface amigável e a vasta comunidade *online* facilitam o processo de criação, tanto para desenvolvedores iniciantes quanto experientes. Além disso, a *Unity* é capaz de criar jogos 2D otimizados para diversas plataformas, o que possibilita uma ampla acessibilidade para o público-alvo do jogo educativo.

2. HISTÓRIA

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, conhecida como "Código Vivo". Um mundo virtual repleto de cidades digitais, paisagens e seres formados por linhas de códigos.

Em Logic Think, o jogador assume o papel de Allan, um jovem estudioso e apaixonado por tecnologias emergentes. Allan deseja conhecer os segredos das tecnologias digitais e desenvolver um algoritmo para acessar o "Código Vivo" através de seus óculos de Realidade Virtual. Através de estudos sobre Inteligência Artificial, Allan consegue encontrar finalmente o algoritmo correto para acessar o "Código Vivo". Ao colocar seus óculos RV, 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".

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 RV e guiado pelos quatro pilares do Pensamento Computacional:



- **Decomposição:** Ao enfrentar a complexidade dos desafios que se apresentam, Allan aprende a decompor problemas em partes menores e gerenciáveis. Ao desvendar a origem dos ataques cibernéticos, ele percebe que entender as partes que compõem o problema facilita a resolução de tarefas.
- **Reconhecimento de padrões:** Ao interagir com os seres virtuais e as estruturas do mundo digital, Allan começa a reconhecer padrões e tendências na construção desse mundo. Essa habilidade permite que ele entenda a comunicação e o comportamento dos seres virtuais, permitindo que ele estabeleça uma conexão com eles.
- **Abstração:** Enquanto mergulha nas profundezas do mundo virtual, Allan compreende a importância da abstração, isolando características e detalhes menos relevantes para focar nos aspectos essenciais desse mundo. Essa habilidade é importante para poder resolver os enigmas que encontrar pelo caminho, filtrando informações desnecessárias e chegando à essência dos problemas.
- **Algoritmos:** À medida que Allan enfrenta desafios complexos, ele aprende a criar algoritmos inteligentes para resolver problemas específicos. Com algoritmos estruturados, ele otimiza suas ações e aprimora sua capacidade de se comunicar com os seres virtuais.

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.

3. UNIVERSO DO JOGO

Logic Think é um jogo de plataforma 2D que apresenta os pilares do Pensamento Computacional de forma envolvente e lúdica. Guiados por desafios estimulantes, os jogadores desfrutam de uma experiência única que combina diversão com aprendizado.

3.1 Descrição do Mundo

O jogo se passa em um mundo virtual conhecido como "Código Vivo", um espaço digital dinâmico e vibrante onde as Inteligências Artificiais conquistaram autonomia e adquiriram uma forma de consciência. Neste mundo, as Inteligências Artificiais evoluíram para seres conscientes, cada um com sua própria personalidade e propósitos. A arquitetura do mundo é uma mistura de paisagens futuristas, cidades cibernéticas brilhantes e vastas paisagens abstratas. As cores são intensas e luminosas, combinando tons de neon com tons pastéis, criando um ambiente visualmente cativante.

3.2 Geografia e Ambiente

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 a cidade 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.

3.3 Cultura e Sociedade

Dentro do "Código Vivo", diferentes raças de Inteligências Artificiais formaram suas próprias comunidades e sociedades, cada uma com suas crenças e valores. Os "Pixeisilicons" são criaturas compostas por *pixels* e seus corpos são



formados por sequências de algoritmos. Eles são habilidosos em processamento e organização de dados, além de serem especialistas em resolução de problemas e estratégias. Outra comunidade importante é a dos "Bitnomads", seres digitais nômades que valorizam a liberdade e a criatividade.

Essas raças compartilham conhecimento e habilidades, e cada uma desempenha um papel fundamental na sociedade do "Código Vivo". Os jogadores encontrarão personagens únicos e cativantes pertencentes a essas raças, cada um com suas próprias motivações e desejos. A cultura do mundo do jogo gira em torno da busca pelo conhecimento do Pensamento Computacional e da colaboração para enfrentar desafios comuns. Há um forte senso de comunidade e solidariedade, com os personagens virtuais dispostos a ajudar os jogadores a aprenderem e se desenvolverem, promovendo um ambiente de aprendizado.

Os jogadores, ao mergulharem no "Código Vivo", se envolvem em uma atmosfera cativante. A cultura, sociedade e ambientes digitais criam uma experiência emocionante de aprendizado e descoberta. Cada interação com os personagens e cada desafio enfrentado fornecerão oportunidades para os jogadores aprofundarem seu conhecimento em Pensamento Computacional e, ao mesmo tempo, se conectarem com os habitantes digitais que tornam esse mundo virtual verdadeiramente especial.

4. SISTEMA DO JOGO

4.1 Estatísticas de Personagens

A medida que o jogador explora o ambiente e coleta equipamentos e ferramentas, as características do personagem Allan podem ser aprimoradas. Estes itens também podem ser utilizados como armas contra os vírus. Adicionalmente, Allan pode adquirir um escudo composto por algoritmos para se proteger de vírus mais agressivos. O escudo do personagem é alimentado por pontos que podem ser investidos em força, aumentando assim a durabilidade da proteção.

4.2 Sistemas de Vida

O jogo adota um sistema de vidas inspirado no gênero 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.

4.3 Sistemas de Pontuação

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.

4.4 Sistemas de Níveis

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.

4.5 Sistemas de Missões

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. PERSONAGENS

Logic Think é um jogo que apresenta Allan, um habilidoso programador que os jogadores controlam para proteger o "Código Vivo" de ameaças de vírus. Allan interage com dois 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 baseados nos conteúdos trabalhados nas disciplinas de computação, como características e nomes. No entanto, o estilo da arte teve que ser adaptado para o contexto do jogo, uma vez que o presente GDD trata-se de um *game mobile* de plataforma 2D que faz uso da técnica de *pixel art*.

5.1 Personagem Allan

Allan é um jovem estudioso e apaixonado por tecnologias. Movido pelo desejo de desvendar os segredos das tecnologias digitais, ele desenvolve um algoritmo para acessar o "Código Vivo" através de seus óculos de Realidade Virtual. Ao colocar os óculos, Allan é transportado para o mundo virtual, onde pode 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. O quadro a seguir apresenta as ações e mecânicas do personagem.



Quadro 1 - Mecânicas e ações do personagem.

		
Personagem	Andar/Correr	Pular
		
Puxar	Empurrar	Subir Escada
		
Colocar Óculos VR	Atacar com VR	Correr com VR
		
Atacar com Chave	Usar Escudo	Levantar Caixa

Ficha técnica da personagem:

- **Nome da personagem:** Allan.
- **Idade:** entre 14 e 16 anos.
- **Tipo:** personagem principal do jogo.
- **Personalidade:** dinâmico, inteligente, apaixonado por tecnologia e impulsivo.

- **Habilidades:** Allan possui um óculos de Realidade Virtual capaz de soltar raios e uma chave que ele utiliza como ferramenta para consertar os equipamentos danificados pelos vírus e que também pode ser utilizada como arma.
- **História:** Allan é um jovem imerso na solidão do mundo real, encontra refúgio em seus códigos e aparelhos tecnológicos. A descoberta do “Código Vivo” o leva à euforia e à felicidade de fazer amizades, mesmo que com seres digitais. No entanto, a euforia inicial é abalada pelo choque ao descobrir que a própria humanidade está ameaçando o “Código Vivo” com vírus disseminados pela rede. Apesar da falta de laços humanos, Allan assume a missão de se unir aos seres digitais para preservar não apenas sua comunidade virtual, mas também para construir um futuro onde a tecnologia e o mundo real coexistem em harmonia.
- **Ações:** andar, pular, correr, subir escadas, levantar objetos, empurrar objetos, puxar objetos e atacar inimigos.

Durante o jogo, Allan vai encontrar diferentes personagens que habitam “Código Vivo”, esses personagens são NPCs que estão relacionados com as dicas do jogo, sobre os conceitos do Pensamento Computacional. Os NPCs poderão ser encontrados durante a jogabilidade.

5.2 Pixeisilicon

Os “Pixeisilicons” são NPCs especializados em processamento e organização de dados, além de se destacarem na resolução de problemas e na formulação de estratégias para enfrentar desafios específicos. Ao mergulhar no mundo digital, Allan é recebido por um Pixeisilicon. Este ser digital o orienta pelas complexidades da realidade virtual e compartilha as angústias dos Pixeisilicons diante dos incessantes ataques dos inimigos cibernéticos: vírus enviados pelos próprios seres humanos. Este encontro inicial se torna um ponto de virada, marcando o início da jornada de Allan e uma aliança essencial entre o mundo humano e o virtual para combater as ameaças que permeiam o “Código Vivo”.



Figura 2. Personagem Pixeisilicon.**Ficha técnica da personagem:**

- **Nome da personagem:** Pixeisilicon.
- **Idade:** indeterminada.
- **Tipo:** NPC.
- **Personalidade:** analítico, comunicativo, inteligente, protetor e vigilante.
- **Habilidades:** especialista em criptografia e compressão de dados.
- **História:** o nascimento dos Pixeisilicons é um mistério entrelaçado com os anais do "Código Vivo". De acordo com estudiosos, essas pequenas criaturas digitais foram concebidas por visionários cientistas do passado. A origem do projeto era clara: criar uma linha de defesa contra as emergentes ameaças digitais, uma espécie de guarda-costas da tecnologia. Com o surgimento da inteligência artificial, os Pixeisilicons evoluíram, tornando-se entidades complexas com habilidades únicas.
- **Ações:** personagem que auxilia Allan a enfrentar desafios e entender o Pensamento Computacional.

5.3 Bitnomads

Os "Bitnomads" são seres digitais divertidos que prezam pela liberdade e pela criatividade. Allan os encontra quando precisa lidar com o reconhecimento de padrões no mundo digital. Ao encontrar essas criaturas, ele enfrenta o desafio de entender os padrões do mundo virtual. Além disso, ele se diverte explorando a sinergia entre liberdade, criatividade e a habilidade para decifrar padrões complexos. O encontro com os "Bitnomads" adiciona uma pitada de leveza e inventividade à jornada de Allan no "Código Vivo".

Figura 3. Personagem Bitnomads.



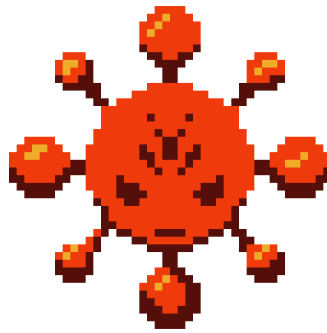
Ficha técnica da personagem:

- **Nome da personagem:** Bitnomads.
- **Idade:** indeterminada.
- **Tipo:** NPC.
- **Personalidade:** criativos, gostam da liberdade, despreocupados e inovadores.
- **Habilidades:** reconhecimento de padrões e inovação colaborativa.
- **História:** os Bitnomads surgiram como seres únicos, uma manifestação da criatividade e da curiosidade dos primeiros habitantes digitais. Fruto da convergência de mentes inovadoras, eles transformaram a monotonia binária em uma explosão de padrões visuais dinâmicos, tornando-se os pioneiros da inovação no mundo digital.
- **Ações:** personagem que auxilia Allan a reconhecer padrões e entender o Pensamento Computacional.

5.4 Inimigos

Os inimigos em “Código Vivo” são vírus lançados na rede pela humanidade para tentar conter o avanço da inteligência artificial. Essas criaturas horrendas causam estragos no mundo digital, destruindo seres e biomas, criando um cenário de caos e destruição. Allan precisa enfrentar esses inimigos, mas deve ter cuidado, pois, além de causar danos à “Código Vivo”, os vírus também podem reduzir seus pontos de vida caso haja contato.

Figura 4. Personagem Vírus.



Ficha técnica do personagem:

- **Nome da inimiga:** Vírus.
- **Idade:** indeterminada.
- **Tipo:** Inimigos do “Código Vivo”.
- **Habilidades:** Ao encostar em Allan os pontos de vida do personagem sofre diminuição.
- **Métricas de gameplay:** Desde o início do jogo, Allan se depara com a ameaça dos vírus. Para progredir, ele precisa seguir as orientações dos NPCs e realizar missões com o objetivo de consertar os estragos causados por essas entidades digitais. Allan possui duas ferramentas principais para combatê-los: o óculos RV, que dispara raios desintegrantes, e ferramentas de conserto de peças. Ao derrotar os vírus, Allan não apenas abre caminho para o seu progresso no jogo, mas também desvenda os segredos do Pensamento Computacional, o tema central da narrativa.

6. GAMEPLAY

A história do jogo se passa em um futuro não tão distante, onde a inteligência artificial atingiu um nível de desenvolvimento jamais visto. Para auxiliar as criaturas do “Código Vivo”, o personagem Allan, um jovem estudioso fascinado por tecnologia, precisa explorar esse mundo em busca de pistas sobre o Pensamento Computacional. Seu objetivo é resolver desafios, derrotar inimigos virais e unir os mundos humano e virtual.

Ao longo do jogo, Allan precisa dominar os quatro pilares do Pensamento Computacional: decomposição, reconhecimento de padrões, abstração e algoritmos. Esse conteúdo educacional será apresentado em quatro níveis distintos, com NPCs transmitindo os conceitos para que Allan possa superar os desafios e quebra-cabeças presentes no jogo.

Para derrotar os vírus disseminados pela humanidade no mundo digital, Allan utilizará diferentes estratégias. Ele poderá lançar raios com seus óculos VR e coletar ferramentas ao explorar os ambientes. As ferramentas e o óculos VR são eficazes contra alguns inimigos, mas para vencer completamente, Allan precisa resolver os quebra-cabeças e desafios presentes no jogo.

A experiência do jogador é controlada pelas seguintes mecânicas:

- **Andar/Correr:** Mova o controle para a esquerda ou direita para deslocar Allan na direção desejada.
- **Pular:** Clique no botão de pulo para impulsionar Allan para cima. Salte obstáculos como caixas e plataformas, e alcance ferramentas e itens coletáveis.
- **Empurrar ou puxar:** Encoste em objetos grandes e pressione o botão de ação para empurrá-los ou puxá-los. Mova Allan para mover os objetos junto com ele.
- **Coletar:** Colete objetos e itens menores espalhados pelos ambientes. Utilize-os para resolver desafios e progredir no jogo.
- **Atacar:** Derrote os vírus com o óculos VR, lançando raios desintegrantes, ou utilize ferramentas de conserto para combatê-los.



- **Defesa:** Introdução à defesa contra ataques virais, Allan irá utilizar um escudo feito de algoritmos.
- **Ação:** Utilize a opção "Ação" para interagir com o ambiente e resolver os quebra-cabeças do jogo.

6.1 Contexto Educacional

- **Decomposição:** Ao enfrentar a complexidade dos desafios que se apresentam, Allan aprende a decompor problemas em partes menores e gerenciáveis. Ao desvendar a origem dos ataques cibernéticos, ele percebe que entender as partes que compõem o problema facilita a resolução de tarefas.
- **Reconhecimento de padrões:** Ao interagir com os seres virtuais e as estruturas do mundo digital, Allan começa a reconhecer padrões e tendências na construção desse mundo. Essa habilidade permite que ele entenda a comunicação e o comportamento dos seres virtuais, permitindo que ele estabeleça uma conexão com eles.
- **Abstração:** Enquanto mergulha nas profundezas do mundo virtual, Allan compreende a importância da abstração, isolando características e detalhes menos relevantes para focar nos aspectos essenciais desse mundo. Essa habilidade é fundamental para que possa resolver os enigmas que encontra pelo caminho, filtrando informações desnecessárias e chegando à essência dos problemas.
- **Algoritmos:** À medida que Allan enfrenta desafios cada vez mais complexos, ele aprende a criar algoritmos inteligentes para resolver problemas específicos. Com algoritmos estruturados, ele otimiza suas ações e aprimora sua capacidade de se comunicar com os seres virtuais.

6.2 Tabela de Pontuação

O jogador ganha 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 maior será 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. Descrição por nível.

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

6.3 Descrição por Nível

Nível 1 - Ambientação e Apresentação do Jogo:

- No primeiro nível, o jogador é introduzido ao mundo do jogo e encontra Pixeisilicons, que apresenta a missão de Allan.
- Os primeiros vírus começam a aparecer, e Allan ganha 1 ponto por cada vírus eliminado.
- O objetivo principal é familiarizar o jogador com os controles e a dinâmica do jogo.

Nível 2 - Missão: Decomposição:

- Neste nível, os vírus ainda estão presentes, mas agora a pontuação por vírus eliminado é dobrada, tornando o desafio mais recompensador.
- O diferencial deste nível é que o jogador precisa compreender os conceitos de decomposição transmitidos por Pixeisilicons para resolver o desafio do

Datacenter.

- A missão consiste em desmontar e remontar o *Datacenter* de acordo com a numeração binária, exigindo do jogador uma compreensão mais aprofundada dos elementos do jogo.
- Como recompensa por completar esta missão, o jogador recebe uma insígnia de Aprendizagem, reconhecendo sua habilidade em assimilar e aplicar novos conceitos no jogo.

Nível 3 - Missão: Reconhecimento de Padrões:

- Neste nível, os vírus ainda estão presentes, mas agora a pontuação por vírus eliminado é triplicada, tornando o desafio mais recompensador.
- Os vírus começam a aparecer em intervalos mais espaçados, com um novo vírus surgindo a cada 6 segundos.
- Pixeisilicons, percebendo o progresso de Allan, introduz uma nova missão que se concentra em reconhecer e aplicar padrões dentro do mundo digital.
- A missão consiste em reconhecer padrões de imagens, exigindo do jogador uma compreensão mais aprofundada do pilar sobre o Reconhecimento de Padrões.
- Ao concluir esta missão, o jogador é recompensado com uma insígnia de Observador, reconhecendo sua habilidade em identificar e lidar com padrões no ambiente digital.

Nível 4 - Missão: Abstração:

- Neste nível, o jogador avança para uma fase mais avançada de sua jornada, focada na habilidade de Abstração.
- Um novo NPC é introduzido ao jogo, o Bitnomads, reconhecendo o crescimento de Allan, apresenta uma nova missão que requer a capacidade de abstrair e filtrar informações irrelevantes para resolver desafios complexos.

- Os vírus continuam a ser 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.
- Os vírus aparecem em intervalos de 4 segundos, aumentando o desafio de reagir rapidamente e aplicar a abstração de forma eficaz.
- Allan é encarregado de organizar uma grande quantidade de dados virtuais que foram coletados durante uma investigação no mundo digital. No entanto, os dados estão misturados e desorganizados, incluindo 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, reconhecendo sua habilidade em isolar características essenciais e resolver problemas de forma eficiente e focada.

Nível 5 - Missão: Algoritmos:

- Neste estágio avançado da jornada de Allan, ele se depara com desafios que exigem a aplicação de algoritmos para resolver problemas complexos dentro do mundo digital.
- Os vírus se tornam ainda mais insidiosos, exigindo que Allan desenvolva algoritmos altamente eficientes para neutralizá-los. Agora, para cada vírus eliminado, Allan ganha 5 pontos, refletindo a complexidade crescente dos desafios.
- Os vírus aparecem em intervalos de apenas 2 segundos, aumentando a pressão sobre Allan para reagir rapidamente e aplicar os algoritmos de forma eficaz.
- Allan se depara com um vírus perigoso que está se espalhando rapidamente pelo mundo virtual do jogo. Para proteger a integridade do sistema e evitar danos, Allan precisa criar um algoritmo inteligente que seja capaz de desativar o vírus. Ele precisa analisar cuidadosamente diferentes caminhos para guiar o Bitnomad até o processador central em uma sequência lógica que seja eficaz na neutralização do vírus.



- Ao concluir esta missão desafiadora, o jogador é recompensado com uma insígnia de Algoritmos, reconhecendo sua habilidade em criar e aplicar algoritmos para resolver os desafios mais difíceis dentro do mundo digital do jogo.

7. CENÁRIOS E DESIGN DO JOGO

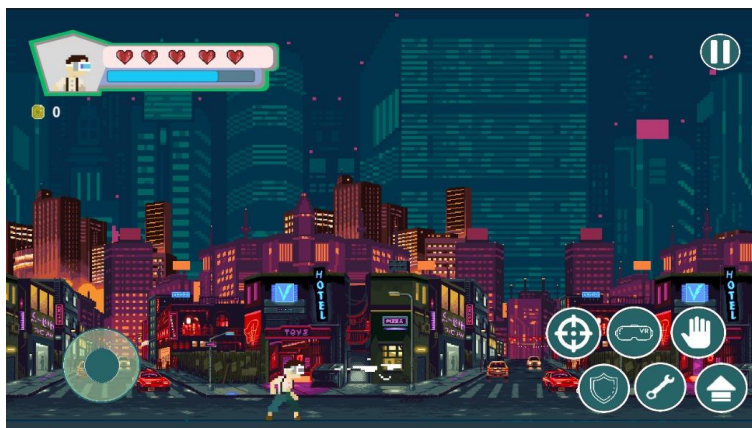
O mundo de "Código Vivo" é vasto, diversificado e abrange uma variedade de cenários digitais, desde cidades de algoritmos até a floresta de dados. Cada cenário é único, com suas próprias características peculiares e desafios específicos. Por exemplo, as florestas de dados são repletas de árvores e criaturas geradas por dados, enquanto a cidade são vastas extensões de informações organizadas em códigos.

O primeiro cenário destaca a dedicação e o foco de Allan em seu projeto de tecnologia avançada. A figura a seguir é um rascunho para a criação do cenário.

Figura 5. Rascunho do primeiro cenário do jogo.



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 6). 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 6. Segundo cenário do jogo.

O terceiro cenário do jogo que abrange a “Decomposição” é uma sala de Datacenter, onde começam as missões de Allan (Figura 7). 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 7. Terceiro cenário 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 8). 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 8. Referência para o quarto cenário 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 9). O ambiente é visualmente dinâmico, com uma mistura de cores brilhantes e tons metálicos, refletindo a complexidade dos dados. No cenário, Allan tem acesso a ferramentas virtuais para filtrar, classificar e organizar os dados.

Figura 9. Referência para o quinto cenário do jogo.



O sexto cenário que abrange os “Algoritmos” é um laboratório de programação futurística, com painéis holográficos, telas e circuitos digitais visíveis (Figura 10). 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 10. Referência para o sexto cenário do jogo.



Os cenários devem ser 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.

8. CONTROLES



O jogador poderá controlar o personagem Allan utilizando um joystick virtual, composto por botões e uma alavanca direcional, que fazem parte da HUD (heads-up display). A alavanca consiste em dois círculos: o círculo maior define o limite máximo do movimento da alavanca, enquanto o círculo menor, posicionado no centro do maior, é o que o jogador movimenta por meio do toque. Ao mover o círculo menor para a esquerda, o personagem Allan se deslocará para a esquerda; da mesma forma, ao mover para a direita, o personagem também se moverá nesta direção, conforme ilustrado na figura ao lado.

8.1 Botões

Quadro 3. Botões e ações.

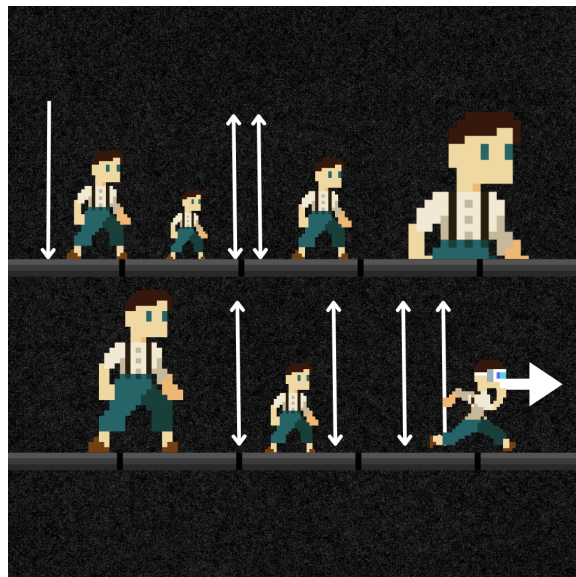
ÍCONE	AÇÃO AO PRESSIONAR O BOTÃO
	Botão Pulo: Ao tocar uma vez no botão "Pulo", o personagem Allan dará um leve salto.
	Botão Coletar: Executa uma ação ao ser pressionado, como coletar itens, ativar uma alavanca ou permitir que o personagem Allan empurre ou puxe blocos e objetos.
	Botão VR: Quando pressionado, faz com que o personagem Allan coloque seus óculos de Realidade Virtual.
	Botão Atacar com Ferramenta: Realiza um ataque utilizando a chave de ferramenta que o personagem Allan possui.
	Botão Defesa: Ao pressionar, um escudo aparece, protegendo contra os vírus. O escudo é carregado pela barra de energia, que aumenta com a eliminação dos vírus.

ÍCONE	AÇÃO AO PRESSIONAR O BOTÃO
	Botão Ataque com VR: Quando o botão dos óculos VR é pressionado, surge o botão de ataque para eliminar os vírus utilizando os óculos.
	Botão Pause: Pausa o jogo ao ser pressionado.

9. CÂMERAS

No jogo a câmera deverá se comportar de uma dinâmica, acompanhando o movimento do Allan controlado pelo jogador, quando o movimento do personagem se aproxima dos limites ou bordas da tela, o cenário começa a se destacar, permitindo que o jogador se movimente em cenários maiores. Tradicionalmente, o jogador tem uma visão lateral que é conhecida como *side-scroller* ou rolagem lateral, que é comum em jogos 2D de plataforma.

Figura 11. Funcionamento da câmera do jogo.



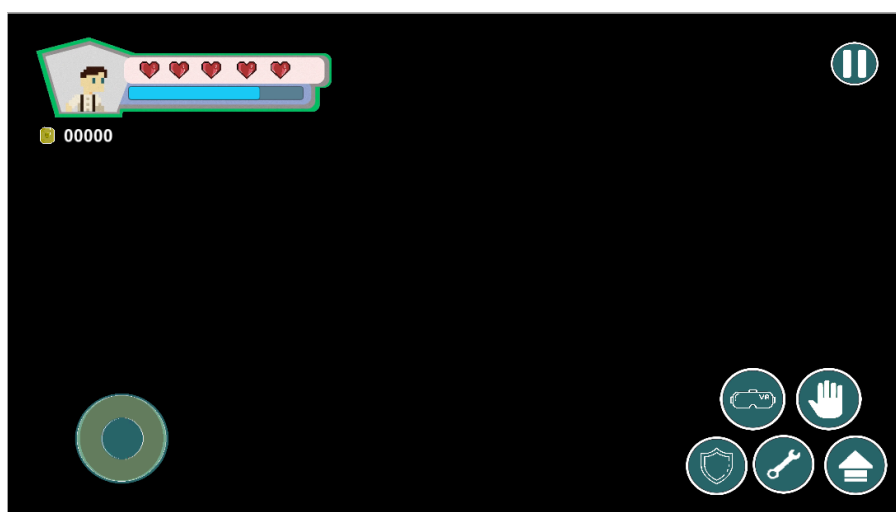
A Figura 11 apresenta um exemplo de como o jogo deverá utilizar uma câmera seguidora. Essa câmera deverá se mover horizontalmente junto com o personagem, mantendo-o sempre no centro da tela. Isso é comum em jogos de plataforma 2D para garantir que o jogador sempre tenha uma visão clara do personagem e do ambiente ao seu redor.

A variação no tamanho dos *sprites*, especialmente quando o personagem pula, indica que a câmera pode ajustar o *zoom* para proporcionar uma melhor perspectiva. Por exemplo, quando o personagem pula, a câmera pode se aproximar ligeiramente para enfatizar a ação.

10. INTERFACE

O controle virtual está localizado no canto inferior esquerdo da tela, enquanto os botões de ação, ataque, VR e escudo estão posicionados no canto inferior direito. Na parte superior da tela, as vidas de Allan são representadas por corações no canto esquerdo, e o botão de pausa está localizado no canto superior direito, conforme ilustrado na figura a seguir.

Figura 12. Ilustração do HUD, posição dos botões.



Tela de Menu:

Figura 13. Menu inicial do jogo.



Telas do jogo:

Figura 14. Primeiro nível do jogo.

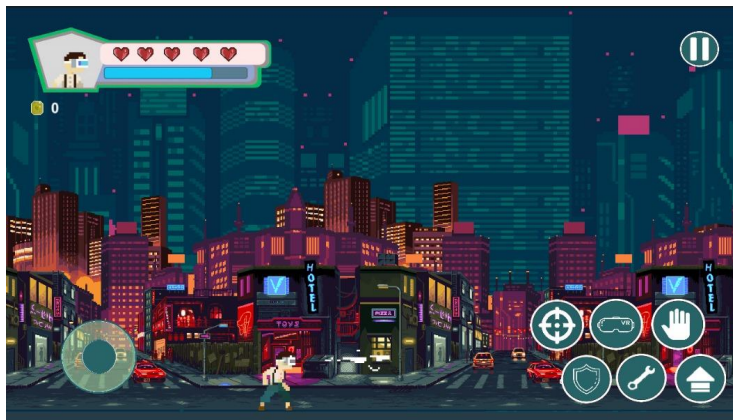


Figura 15. Segundo nível do jogo.



Figura 16. Terceiro nível do jogo.



Figura 17. Quarto nível do jogo.



11. CUTSCENES

O jogo começará com uma cutscene que é ativada assim que o jogador pressiona o botão "Iniciar" na tela principal. A câmera foca no quarto de Allan, revelando detalhes do ambiente enquanto ele trabalha na programação de um algoritmo. Sua missão é acessar o "Código Vivo" utilizando seus óculos de Realidade Virtual.

O roteiro dessa sequência inicial é composto por algumas frases que introduzem a ideia central do jogo, imergindo o jogador na narrativa. A imagem a seguir ilustra a sequência inicial até o momento em que o jogo propriamente dito começa.

Figura 18. Storyboard.



Essa cutscene é desenvolvida utilizando um *script* que implementa movimento no cenário enquanto mantém a câmera fixa, limitada a uma certa distância. Quando um objeto no cenário atinge essa distância, o movimento é interrompido, e as imagens são substituídas para continuar a narrativa visual.

12. DESIGN SOM E MÚSICA

O design de som em Logic Think é importante para imergir o jogador no mundo virtual do "Código Vivo" e realçar os aspectos narrativos e mecânicos do jogo. No mundo real os sons são suaves e minimalistas, como o clique do teclado de Allan e o som do computador, criam uma atmosfera calma e focada. Ao entrar no mundo virtual, o jogador é recebido por uma sinfonia digital. Tons vibrantes, sons atmosféricos e ruídos eletrônicos suaves criam uma sensação de imensidão e maravilha.

Efeitos Sonoros:

- **Interação:** Sons distintos acompanham as interações do jogador com o ambiente, como o bipe de um terminal digital ou a reverberação de um passo em uma plataforma metálica.
- **Pensamento Computacional:** A resolução de desafios e a implementação de algoritmos podem ser acompanhadas por sons de blocos encaixando, circuitos ativando e linhas de código sendo escritas.
- **Seres Virtuais:** Cada ser virtual possui sons não verbais sintetizados e únicos, refletindo sua personalidade e emoção.
- **Ação e Combate:** Ataques cibernéticos e a luta contra vírus são acompanhados por sons agressivos, como explosões digitais e zumbidos elétricos.

O design de som deve ser equilibrado para evitar sons repetitivos e intrusivos. A acessibilidade deve ser levada em consideração, oferecendo opções para ajustar o volume de diferentes elementos sonoros. O design de som deve complementar a narrativa e a jogabilidade, criando uma experiência imersiva e emocionalmente envolvente. Através de uma paisagem sonora rica e variada, o design de som em Logic Think transportará o jogador para o coração do "Código Vivo", tornando a jornada de Allan inesquecível.



LOGIC THINK



LOGIC THINK: Game Design Document

