

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Guilherme Duarte Toffoli

**Gamificação do Ensino de Pensamento
Computacional: Desenvolvimento de um Jogo
Sério Educacional baseado na prova *Bebras***

Uberlândia, Brasil

2026

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Guilherme Duarte Toffoli

**Gamificação do Ensino de Pensamento Computacional:
Desenvolvimento de um Jogo Sério Educacional baseado
na prova *Bebras***

Trabalho de conclusão de curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, como parte dos requi-
sitos exigidos para a obtenção título de Ba-
charel em Ciência da Computação.

Orientador: Prof. Dr. Renato de Aquino Lopes

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2026

Guilherme Duarte Toffoli

Gamificação do Ensino de Pensamento Computacional: Desenvolvimento de um Jogo SériO Educacional baseado na prova *Bebras*

Trabalho de conclusão de curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, como parte dos requi-
sitos exigidos para a obtenção título de Ba-
charel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 20 de março de 2026:

Prof. Dr. Renato de Aquino Lopes
Orientador

Prof. Dra. Ana Cláudia Martinez
Convidado 1

Prof. Dr. Diego Nunes Molinos
Convidado 2

Uberlândia, Brasil
2026

Agradecimentos

Agradecimentos à minha família, meus pais Mateus e Edna pelo apoio e amor incondicional desde o momento em que nasci, especialmente nesta etapa da vida, onde pude amadurecer e aprender com vocês como entrar na vida adulta com conselhos e firmeza. À minha irmã Giovana, por sempre estar comigo quando preciso, seja com brincadeiras ou com assuntos de família. E à eterna cachorra da família Meg, sua mistura de poodle e maltês que nos alegrou e nos deu amor por tantos anos de vida, agora descansa em paz.

À minha namorada Camilla pela paciência, compreensão ao longo desta jornada, sendo fundamental para que eu mantivesse foco e motivado, nossos momentos juntos após horas de faculdade, jornada de trabalho cansativas, e a escrita deste trabalho, foram importantes para mim para chegar neste momento, meu coração pertence a ti.

Aos meus amigos que considero irmãos de vida, Eduardo César, João Antônio Misson, Rafael Ramos, que me acompanham desde criança e compartilham a visão de um mundo mais leve, nossas horas jogando jogos de madrugada me inspiraram a realizar este trabalho, e mesmo a vida nos colocando em caminhos diferentes, sempre seremos capazes de nos encontrarmos para compartilhar histórias em uma tarde inteira.

Aos amigos de vida, Gabriel Oliveira, Guilherme Otoni, Isabela Barbosa, Lucas Boaventura e Maria Eduarda Koyama, obrigado por compartilharem comigo esta aventura de viver em Uberlândia em uma etapa tão importante na vida, pelas tardes de estudo e pelas risadas que tornaram a caminhada mais leve.

Aos professores da UFU pelos ensinamentos, e em especial meu orientador Dr. Renato de Aquino, obrigado pelos conselhos e orientação leve e tranquila, desde a iniciação científica.

À todos os amigos aqui não citados e à quem contribuiu diretamente ou indiretamente para a realização deste trabalho, deixo aqui meu sincero agradecimento.

Resumo

A inclusão da Computação na Base Nacional Comum Curricular (BNCC) estabeleceu o Pensamento Computacional como competência essencial na educação básica, criando a necessidade de ferramentas pedagógicas que auxiliem sua implementação em sala de aula. Nesse contexto, este trabalho tem como objetivo desenvolver um jogo sério digital baseado em desafios do *Bebras Challenge*, com foco no ensino de conceitos de Pensamento Computacional (PC) para estudantes do ensino fundamental. A metodologia adotada envolveu pesquisa bibliográfica sobre ensino de PC, jogos educativos e metodologias de desenvolvimento ágil, seguida da modelagem e implementação de um minijogo na *engine* Unity, utilizando linguagem C#. O minijogo desenvolvido baseia-se no desafio “Pontos de costura”, envolvendo a interpretação de uma Máquina de Estado Finita em um ambiente interativo tridimensional. Como resultado, obteve-se um artefato funcional que permite ao jogador experimentar conceitos como abstração, algoritmos, avaliação e reconhecimento de padrões por meio de interação lúdica, além de incorporar elementos de acessibilidade e *feedback* obtidos a partir de testes com usuários. Conclui-se que a ferramenta desenvolvida apresenta potencial como recurso de apoio ao educador para reforçar conteúdos de Pensamento Computacional, contribuindo para a transposição de atividades tradicionais em papel para experiências digitais interativas e motivadoras.

Palavras-chave: Pensamento Computacional, Jogo Sério, *Bebras Challenge*, Gamificação.

Lista de ilustrações

Figura 1 – Questão escolhida da prova <i>Bebras</i>	26
Figura 2 – Resposta da questão Pontos de Costura	27
Figura 3 – Primeira versão do minijogo da costura	28
Figura 4 – Interações na versão antiga do jogo	28
Figura 5 – Cenário antigo com o NPC Pato	29
Figura 6 – Interações na versão nova do jogo	30
Figura 7 – Nova versão após os testes	31
Figura 8 – Sistema de <i>feedback</i> do minijogo.	31
Figura 9 – Tela de confirmação	32
Figura 10 – Sistema de confirmação de resposta.	32
Figura 11 – Diagrama de caso de uso	33
Figura 12 – Documentação de um caso de uso	33
Figura 13 – Diagrama de classe	34

Lista de tabelas

Tabela 1 – Tabela Comparativa de Características dos Trabalhos Relacionados . .	21
Tabela 2 – Resultados da aplicação do SUS	30

Lista de Siglas

ABP Aprendizagem Baseada em Problemas. [9](#)

BNCC Base Nacional Comum Curricular. [9](#), [23](#), [36](#)

C/ES Sistema de Entidade e Componentes. [34](#)

GDD Game Design Document. [11](#), [15–17](#), [20](#), [23](#), [24](#), [27](#), [29](#), [35](#), [36](#)

IHC Interação Humano-Computador. [19](#)

MEC Ministério da Educação. [9](#)

MEF Máquina de Estado Finita. [11](#), [16](#), [25](#), [26](#), [36](#)

NPC Non-Playable Character. [27](#), [28](#)

PC Pensamento Computacional. [9](#), [11](#), [12](#), [14](#), [16](#), [17](#), [19](#), [20](#), [23](#), [25](#), [29](#)

PPGCE Programa de Pós-Graduação em Tecnologias, Comunicação e Educação. [29](#)

SGDD Serious Game Design Document. [15](#)

SUS System Usability Scale. [29](#)

UFU Universidade Federal de Uberlândia. [29](#)

XP Extreme Programming. [11](#), [23](#)

Sumário

1	INTRODUÇÃO	9
1.1	Justificativa	9
1.2	Objetivos	10
1.3	Contribuições	10
1.4	Organização	10
2	FUNDAMENTAÇÃO TEÓRICA	11
2.1	Desenvolvimento ágil	11
2.2	Pensamento Computacional	12
2.3	Jogos aplicados a educação	12
2.4	Jogo S�rio	13
2.5	Gamifica��o	14
2.6	<i>Bebras</i>	14
2.7	<i>Game Design Document</i>	15
2.8	M�quina de Estado Finita	16
3	TRABALHOS RELACIONADOS E METODOLOGIA	17
3.1	Trabalhos relacionados	17
3.2	Metodologia	21
4	DESENVOLVIMENTO	23
4.1	Pr�-desenvolvimento	23
4.2	Concep��o	23
4.3	Desenvolvimento T�cnico	27
5	CONCLUS�O	36
	REFER�NCIAS	37
	ANEXOS	40
	ANEXO A – GAME DESIGN DOCUMENT (GDD) GERAL	41
	ANEXO B – GAME DESIGN DOCUMENT (GDD) DO MINIJOGO DA COSTURA	54
	ANEXO C – DOCUMENTA��O DOS CASOS DE USO	59

1 Introdução

A inclusão da Computação na [Base Nacional Comum Curricular \(BNCC\)](#), por meio do parecer ([Conselho Nacional de Educação - Câmara de Educação Básica \(CNE/CEB\), 2022](#)), estabeleceu o Pensamento Computacional (PC) como uma competência essencial para a educação básica. Essa mudança normativa abriu uma lacuna estratégica no ensino fundamental e médio: a necessidade de trabalhar pilares como decomposição de problemas, reconhecimento de padrões, abstração e algoritmos de forma integrada ao currículo. Neste sentido, o artigo de ([GARCIA; BORGES, 2023](#)), discute a importância de todos adquirirem estes conhecimentos, pois envolvem a capacidade de modelar e sistematizar problemas reais, habilidades indispensáveis para o mercado de trabalho moderno.

Estudos recentes como ([SCHEFFEL; MOTTA, 2022](#)) exploram metodologias como a [Aprendizagem Baseada em Problemas \(ABP\)](#) e a Metodologia da Problemática para engajar alunos na construção de soluções computacionais, e pesquisas como a de ([ANUNCIÇÃO et al., 2025](#)) mostram que existem desafios como a de escassez de recursos tecnológicos e a recorrente visão restrita do [Pensamento Computacional \(PC\)](#) como conteúdo apenas de informática. Diante desse cenário, surge a oportunidade de desenvolver soluções tecnológicas que facilitem a transição entre a teoria normativa e a prática em sala de aula, além da utilização do [PC](#) como competência transversal, como exemplo um jogo sério plugado. Este trabalho propõe o desenvolvimento de um jogo sério baseado na iniciativa internacional *Bebras* ([Bebras Challenge, 2025](#)). Ao modelar essas questões em um ambiente 3D, busca-se fomentar a curiosidade de crianças e adolescentes, tornando o aprendizado do Pensamento Computacional uma experiência lúdica, intuitiva e acessível.

1.1 Justificativa

Neste sentido, o [PC](#) não é uma habilidade para se usar apenas na tecnologia ou na área da computação, é algo que a pessoa pode usufruir no cotidiano. Assim, a relevância deste estudo fundamenta-se no fato de que, embora o [Ministério da Educação \(MEC\)](#) tenha formalizado essas diretrizes, a implementação prática enfrenta desafios, e as principais dificuldades segundo ([ANUNCIÇÃO et al., 2025](#)) referem-se a formação de educadores em um meio digital, ausência de políticas educacionais que de fato implementam o [PC](#), e a principal dificuldade, recursos tecnológicos em falta, por tratar-se de uma área recente na educação básica, ainda há uma carência de ferramentas de apoio que auxiliem o educador e padronizem o ensino, especialmente na rede pública.

1.2 Objetivos

Assim, o objetivo deste trabalho é desenvolver um minijogo sério digital que sirva como ferramenta de apoio ao ensino de Pensamento Computacional. A proposta busca transformar desafios inspirados no *Bebras Challenge* em uma experiência interativa, contribuindo para o reforço de habilidades como abstração, decomposição e pensamento algorítmico em um contexto educacional lúdico.

1.3 Contribuições

O trabalho modela um artefato funcional tecnológico, onde o aluno interage com o ambiente 3D e realiza ações para reforçar conceitos que estão sendo abordados na sala de aula, assim deixando o aprendizado mais lúdico e interativo, transformando atividades baseadas em papel em experiências digitais, em que se espera ampliar a motivação e o engajamento de estudantes.

1.4 Organização

Para compreensão deste projeto, este trabalho está organizado em capítulos. A fundamentação teórica discorre sobre conceitos utilizados para melhor entendimento sobre o assunto. Complementando, trabalhos relacionados descreve uma série de pesquisas da utilização de conceitos como Pensamento Computacional, gamificação e a própria utilização do *Bebras* em salas de aulas e suas conclusões, comparadas com este trabalho. Além disso, o desenvolvimento descreve o processo de elaboração e criação do jogo, assim como o minijogo implementado. Com a conclusão, são apresentadas as considerações finais e sugestões para trabalhos futuros.

2 Fundamentação Teórica

Este capítulo apresenta os principais conceitos teóricos que fundamentam o desenvolvimento do jogo proposto. São abordados temas relacionados à engenharia de *software*, como metodologias ágeis, bem como conceitos educacionais e computacionais essenciais, incluindo o PC, jogos aplicados à educação, jogos sérios e gamificação. Além disso, são discutidos referenciais específicos utilizados no projeto, como o *Bebras Challenge*, a elaboração de documentos de design de jogos (*Game Design Document (GDD)*) e o modelo matemático de Máquina de Estado Finita (*Máquina de Estado Finita (MEF)*) utilizado no minijogo. Esses elementos fornecem a base conceitual necessária para compreender as decisões de design, implementação e abordagem pedagógica adotadas neste trabalho.

2.1 Desenvolvimento ágil

O desenvolvimento ágil é um tipo de abordagem diferente aos tradicionais de criação de *software*, decompondo seu processo em ciclos curtos e gerenciáveis, conhecidos como iterações ou *sprints*. Dentro deste paradigma, o *Scrum* é um dos *frameworks* mais utilizados, segundo (SCHWABER; SUTHERLAND, 2011), pois estrutura o trabalho em torno de um *Product Backlog*, lista priorizada de funcionalidades, do qual a equipe deve executar durante um *sprint*, o qual permite entregar um incremento funcional e testável do produto, assim gerando *feedback* contínuo e adaptação de mudanças, este modelo usado no projeto como um todo.

Neste sentido, de acordo com (SOARES, 2004), *Extreme Programming (XP)* é uma metodologia ágil utilizada por equipes menores, pois permite desenvolver o *software* de maneira simples e adaptá-lo rapidamente com base no retorno obtido tanto da equipe quanto dos clientes. Este *framework*, envolve também a cada *feedback* uma abordagem incremental, onde a comunicação entre as pessoas é encorajada. Ademais, o autor cita 12 práticas usadas nesta metodologia, onde a dupla que desenvolvia o exercício *Bebras* para o jogo utilizava a prática de Programação em pares, ou seja, o código era feito por uma pessoa, enquanto a outra revisa o que está sendo feito, pensando estrategicamente como identificar erros e sugerindo melhorias. Estes conceitos foram todos abordados na matéria de Engenharia de *Software*, que apresentou diversas maneiras de se organizar o fluxo de desenvolvimento de um projeto envolvendo uma equipe, e assim foi utilizada para a realização deste trabalho.

2.2 Pensamento Computacional

De acordo com (WING, 2006), o PC é como uma maneira de resolver problemas, fazer design de sistemas e compreender o comportamento humano por meio de usar como vantagem as concepções da ciência da computação, ou seja, é um jeito de pensar diferente à frente a um problema, o qual soluciona-se uma problemática com conceitos usados na computação. Ademais, ela ainda cita algumas noções da área, sendo decomposição, modelagem e representação de dados, busca binária, recursão e paralelização. Neste contexto, a autora diz que é uma habilidade que todos podem usufruir, não apenas as pessoas da área da tecnologia, desta maneira houve um grande interesse por parte das instituições de ensino usufruir do Pensamento Computacional para ajudar os alunos, já que é uma competência que pode ser utilizada nos problemas do dia a dia.

Nesse sentido, como também discutido em (GROVER; PEA, 2017), o PC não impõe o aprendizado de programação, mas trás estratégias essenciais para a resolução de problemas difíceis em várias áreas do conhecimento, pois permite que se interprete a situação sistematicamente, identificando padrões para desenvolver soluções eficientes, desde a resolução de desafios no cotidiano até decisões no contexto do trabalho. Por isso, o Pensamento Computacional é uma ferramenta essencial para o desenvolvimento da criatividade e do raciocínio lógico, e justamente aplicado ao ensino, análises revelam que alunos têm o desempenho acima do esperado quando se compara com métodos tradicionais (ZHONG-GEN, 2019), e que portanto, a união entre PC e jogos educativos está em crescimento na educação de hoje em dia, preparando estudantes para os desafios do século XXI, com pedagogia aplicada e ambiente motivador e interativo.

Em síntese, sua integração ao ambiente educativo, especialmente por meio de jogos, oferece uma alternativa aos métodos tradicionais ao colocar o estudante como sujeito ativo e motivado, capaz de decompor desafios e identificar padrões em diversas áreas do conhecimento. Assim, o minijogo desenvolvido neste trabalho aborda abstração, decomposição, algoritmos, avaliação e reconhecimento de padrões.

2.3 Jogos aplicados a educação

A utilização de jogos na educação é compreendida como uma manifestação cultural que, ao ser aplicada no ambiente escolar, funciona como uma estratégia para dinamizar o ensino e mediar conhecimentos de forma lúdica (ANTUNES; NASCIMENTO; QUEIROZ, 2020). Para além do entretenimento, os jogos educativos devem possuir um propósito explícito e objetivos pedagógicos bem delimitados, servindo como ferramentas que estimulam o pensamento estratégico e a resolução de situações-problema em um ambiente controlado (ANTUNES; NASCIMENTO; QUEIROZ, 2020). Nesse contexto, a pesquisa realizada para o desenvolvimento do jogo analógico Umuechem destaca que a criação de

um artefato educativo não deve ser um processo aleatório, mas sim pautado em uma metodologia científica rigorosa, como o Ciclo Criativo de Jogos Colaborativos, que envolve fases de pesquisa, definição de metas e sucessivos testes de usabilidade.

Complementando a discussão anterior, os jogos digitais potencializam esse processo ao oferecerem *feedbacks* instantâneos que permitem ao aluno-jogador corrigir seu curso de aprendizado em tempo real, superando a rigidez das avaliações periódicas tradicionais. Diferente dos métodos de ensino convencionais, centrados na transmissão passiva de informações, o uso dessas tecnologias coloca o estudante como sujeito ativo, permitindo que ele determine seu próprio ritmo e construa significados através da interação direta com o contexto do problema. Além da utilização de softwares prontos, a própria programação de jogos digitais é apontada como uma estratégia eficaz para o desenvolvimento de capacidades cognitivas superiores, como abstração, pensamento sistêmico e a decomposição de problemas complexos. Para que esses benefícios sejam plenamente alcançados, é fundamental que o professor atue como mediador, integrando os jogos de forma natural ao projeto pedagógico e promovendo ambientes de colaboração e debate crítico entre os participantes (MENEZES, 2015).

Em suma, este projeto se baseia na união entre Pensamento Computacional e as estratégias de design de jogo, pois as duas estabelecem um ecossistema de aprendizagem onde o raciocínio lógico e o pensamento lúdico se retroalimentam, onde é possível construir um ensino interativo e humanizado, que capacita o estudante a interpretar e solucionar desafios de forma sistemática, crítica e criativa.

2.4 Jogo Sérió

Diferente dos jogos tradicionais que o principal foco é o lazer, os jogos sérios envolvem tanto mecânicas envolventes quanto objetivos de ensino, os quais buscam promover o aprendizado de forma motivadora e interativa. Por isso, de acordo com (LAAMARTI; EID; SADDIK, 2014), o jogo sério tem a capacidade de envolver os jogadores por causa de cenários desafiadores e simulados, onde podem aplicar conhecimentos prévios e treinar habilidades em um ambiente controlado e seguro. Além disso, os jogos utilizam de elementos como *feedback* imediato, narrativa imersiva e desafios progressivos para desenvolver o aprendizado e melhorar a retenção de conhecimento.

Ademais, os jogos sérios são utilizados em propósitos como treinamento, saúde, conscientização social, porém são particularmente eficazes na educação, segundo (ZHONG-GEN, 2019), neste contexto educacional, tem mostrado impactos positivos no ensino, o qual melhorou o desempenho e promoveu habilidades cognitivas de estudantes em diversas disciplinas, como resolução de problemas e pensamento crítico. Um exemplo de jogos aplicados a educação, é o estudo de (SOUSA; LENCASTRE, 2014), que demonstrou como

o uso do *Scratch*, uma ferramenta de programação baseada em jogos, foi eficaz no desenvolvimento do PC e na competência de resolução de desafios em alunos do 8º ano, o qual evidencia o potencial dos jogos como facilitadores da aprendizagem ativa e colaborativa.

Dessa forma, desenvolvendo um jogo sério como este do trabalho, pode-se consolidar como uma ferramenta de ensino progressiva para engajar o estudante em um ambiente de experimentação seguro. Eficácia esta, potencializada por elementos de jogos como *feedback* imediato e narrativa, que ajudam a reter o conteúdo e tornar o exercício motivador e significativo para o aluno.

2.5 Gamificação

A gamificação pode ser definida como o uso de elementos e mecânicas de jogos em contextos que não são jogos, com o objetivo de motivar pessoas, ajudar na resolução de problemas e facilitar a aprendizagem (FARDO, 2013). Em vez de criar um jogo completo, essa técnica utiliza partes específicas, como sistemas de recompensas, narrativas e níveis, para gerar o mesmo engajamento que os jogadores sentem ao interagir com games comerciais (FARDO, 2013). Na educação, essa abordagem é útil porque as gerações atuais já estão acostumadas com essa linguagem, o que ajuda a combater o desinteresse pelos métodos tradicionais de ensino, segundo o autor.

Para que a gamificação funcione bem em um ambiente de aprendizagem, é preciso ir além de apenas dar pontos ou medalhas, o ideal é criar uma experiência que faça sentido para o aluno (FARDO, 2013). Algumas práticas incluem dar retornos rápidos (*feedback*), permitir que o estudante aprenda com o erro e dividir tarefas difíceis em missões menores e mais simples. Dessa forma, o ambiente de aula se torna mais dinâmico, ajudando o aluno a entender como cada pequena tarefa contribui para um objetivo maior (FARDO, 2013).

Assim, a gamificação atua neste projeto como uma ponte entre engajamentos dos jogos e os objetivos pedagógicos, a qual transforma a dinâmica da sala de aula ao aplicar mecânicas como recompensas e missões em conteúdos curriculares.

2.6 Bebras

O *Bebras Challenge* é um desafio feito em comunidades escolares proposto pela Prof. Valentina Dagienė da Universidade de Vilnius em 2004 na Lituânia, em que a tradução do nome *Bebras* em lituano é Castor, por sugerir ser um animal esforçado e inteligente. Este desafio, reúne mais de 60 países para discutir conceitos da informática na educação primária e secundária, os quais acreditam que a democracia seja tanto o objetivo, quanto o método para uma educação democrática e inclusiva para crianças e adolescentes desenvolverem seu potencial de criatividade (DAGIENĖ; SENTANCE; STUPURIENĖ, 2017),

onde cada país aplica sua prova com conceitos do Pensamento Computacional. Suas questões possuem grau definido em fácil, médio e difícil, e usam um sistema de categorização para atribuir conceitos de informática e habilidades do Pensamento Computacional.

Neste sentido, o projeto usufrui da complexidade pedagógica consolidada e revisada por uma comunidade internacional de professores e pessoas da área da computação, para fundamentar o desenvolvimento de minijogos em um ambiente 3D, transformando tarefas estáticas das provas em experiências imersivas, assim potencializando a atenção do aluno ao conteúdo e a visualização espacial.

2.7 *Game Design Document*

O Documento de Design de Jogo, conhecido como [GDD](#), funciona como um projeto detalhado e uma ferramenta de comunicação essencial para qualquer equipe de desenvolvimento de jogos. Ele é um documento vivo que centraliza todas as informações do projeto, descrevendo em detalhes os elementos que compõem o jogo, como o conceito central, a narrativa, os personagens, as mecânicas de jogabilidade, o estilo visual e sonoro, e a interface do usuário ([BETHKE, 2003](#)). O principal objetivo do [GDD](#) é garantir que todos os membros da equipe, de programadores a artistas, compartilhem uma visão unificada e compreendam o escopo e a direção do projeto, adicionando ordem e organização ao processo de desenvolvimento.

Existem diversas maneiras e modelos para se fazer um [GDD](#), inclusive focado para jogos sérios, pois inclui uma camada de complexidade de acordo com o alinhamento do entretenimento com propósitos pedagógicos. Essa necessidade criou a adaptação para um *Serious Game Design Document (SGDD)*, uma proposta específica para o desenvolvimento de jogos com fins educativos. Tais documentos comunicam não só mecânicas de jogo mas também o conteúdo educacional que o jogo sério deve ter, como saídas dos aprendizados, motivação do aprendizado, problemas resolvíveis entre outros ([IBRAHIM; JAAFAR, 2009](#)), de forma clara para uma equipe multidisciplinar, que inclui tanto especialistas em jogos quanto em pedagogia.

Em resumo, [GDD](#) e suas derivações como [SGDD](#) consolidam-se como uma estratégia de organização em uma equipe multidisciplinar. Neste sentido, o projeto também começou com um [GDD](#) geral, disponível em anexo [A](#), com todos da equipe participando da escrita, e um [GDD](#) para o minijogo feito, em anexo [B](#), permitindo um esclarecimento de conceitos e detalhamento de mecânicas, mantendo ao mesmo tempo um alinhamento com as diretrizes globais do projeto.

2.8 Máquina de Estado Finita

Segundo (GERSTING, 2001), a **MEF** é um modelo matemático que captura as características essenciais de um computador digital, indo além de simples sistemas aritméticos. Este modelo se baseia em um sistema que possui um número finito de estados internos possíveis, e a qualquer momento, a máquina se encontra em exatamente um desses estados. O estado atual funciona como uma espécie de memória dos dados de entrada que a máquina recebeu anteriormente. O funcionamento da **MEF** é determinístico e sincronizado por pulsos discretos, o que significa que suas ações em resposta a uma entrada são inteiramente previsíveis. Uma **MEF** é formalmente definida por como uma 5-tupla $M = [E, I, O, f_e, f_o]$, onde cada parâmetro representa:

- $[E]$ É um conjunto finito de estados.
- $[I]$ É um conjunto finito de símbolos de entrada (o alfabeto de entrada).
- $[O]$ É um conjunto finito de símbolos de saída (o alfabeto de saída).
- $[f_e]$ A função **próximo estado**, que leva um par (estado, dados de entrada) em um estado, onde $f_e : E \times I \rightarrow E$.
- $[f_o]$ A função **saída**, que é aplicada a um estado para obter a saída, onde $f_o : E \rightarrow O$.

A transição para um novo estado é definida por uma função "próximo estado"(f_e), que depende exclusivamente do estado atual e da entrada recebida. Adicionalmente, a máquina é capaz de gerar dados de saída, que são determinados por uma função "saída"(f_o) que se baseia unicamente no estado atual da máquina. Modelo este, estudado na matéria de Linguagens Formais e Autômatos, foi importante para compreender o modelo importante para a Ciência da computação. Ademais, modelo este que é a figura central da questão do *Bebras* Pontos de costura, onde é inserido em um contexto em que é representado por uma máquina de costura, que segue o padrão de pontos feitos pela **MEF**.

Assim, a fundamentação teórica apresentada evidencia a junção entre aspectos tecnológicos, pedagógicos e metodológicos necessários para o desenvolvimento de um jogo sério voltado ao ensino de **PC**. As metodologias ágeis contribuíram para a organização do processo de desenvolvimento, enquanto os conceitos de jogos educativos, gamificação e jogos sérios forneceram diretrizes para promover engajamento e aprendizagem significativa. O *Bebras Challenge* ofereceu a base pedagógica e os problemas estruturados, e o **GDD** possibilitou a organização sistemática das decisões de design. Por fim, a **MEF** serviu como modelo conceitual central para o minijogo desenvolvido, conectando a teoria da computação à prática lúdica. Dessa forma, os conceitos apresentados sustentam teoricamente as escolhas realizadas nas etapas de concepção e desenvolvimento descritas nos capítulos seguintes.

3 Trabalhos relacionados e Metodologia

3.1 Trabalhos relacionados

Esta seção apresenta trabalhos relacionados que fundamentam o uso de jogos e tarefas do *Bebras* no ensino de PC. São discutidos estudos que investigam a utilização de problemas do *Bebras* como base para design de jogos sérios e instrumentos de avaliação, abordagens digitais e não digitais para o desenvolvimento de habilidades de PC, e metodologias para documentação e planejamento, como o GDD. Assim, a análise desses trabalhos permite situar a proposta deste estudo no meio da literatura e evidenciar semelhanças, diferenças e lacunas que motivam o desenvolvimento do jogo sério em ambiente 3D.

O artigo de (LAISA; HENRIQUE, 2017) investigou se tarefas do *Bebras Challenge*, uma prova aplicada internacionalmente que promove o PC, podem ser usadas como base para o design de jogos sérios (jogos educativos) focados em PC, a partir disso foram traçados objetivos e hipóteses. E então, por meio de um experimento controlado com dois designers de jogos, foram criados 20 designs, 10 baseados em tarefas do *Bebras* (*Proposed Treatment*) e 10 em descrições convencionais de habilidades de PC (*Standard Treatment*). Os resultados, separados por tabela, avaliados por três designers de jogos e 8 professores, mostraram que os jogos baseados no *Bebras* foram considerados mais interessantes, (70% vs. 20%), criativos (80% vs. 60%) e alinhados pedagogicamente (60% vs. 40%) sendo *Proposed Treatment* à esquerda e *standard* à direita. Os participantes também relataram maior facilidade e eficácia no uso das tarefas do *Bebras*. Nesse sentido, o estudo mostra a eficiência pedagógica em termos de game design, além de contextualizar os conceitos de PC por meio de questões baseadas na prova *Bebras*, as quais também serão reproduzidas na ferramenta deste trabalho, porém com a proposta de ser um jogo sério 3D.

O objetivo do trabalho de (LOCKWOOD; MOONEY, 2018) foi desenvolver e validar um teste de PC utilizando problemas do desafio internacional *Bebras*, destinado a estudantes do ensino médio e universitários. Desta maneira, os autores criaram dois testes com questões de dificuldade nem tão fácil nem tão difícil, os quais foram avaliados por um painel diversificado de especialistas em Ciência da Computação, e aplicaram em escolas e em um curso de graduação com dificuldades proporcionais ao nível de ensino. Neste contexto, os resultados mostraram que os testes foram eficazes para medir habilidades de PC, com desempenho superior entre estudantes universitários e aqueles com experiência em programação ou matemática avançada, além de ambos os grupos, com diferentes níveis de conhecimento e idade, classificaram os exercícios com o mesmo nível de dificuldade. Ademais, os testes demonstraram potencial para prever sucesso acadêmico e ofereceram

uma ferramenta flexível, aplicável tanto online quanto em papel. O estudo sugere ainda a necessidade de desenvolver questões equivalentes para futuras aplicações. Portanto, o artigo complementa o trabalho em questão de argumentar, e mostrar resultados que o ensino de Pensamento Computacional, em qualquer área de conhecimento, serve como ferramenta de estudo para ampliar o conhecimento na área, mesmo aqueles que não são familiarizados com tecnologia ou matemática.

Já o trabalho de (ARAGÃO; AVELLAR; BARBOSA, 2023), realizou um Mapeamento Sistemático da Literatura (MLS) para estudar o uso de Realidade Virtual (RV), Realidade Aumentada (RA) e elementos de jogos no ensino de programação e PC. Neste sentido, os autores analisaram 20 estudos selecionados, identificando que a RA é a tecnologia mais utilizada (14 estudos), frequentemente combinada com ensino colaborativo, enquanto a RV apareceu em apenas 2 estudos. Por outro lado, os tópicos de programação abordados foram majoritariamente introdutórios (variáveis, condicionais, iterações), e jogos sérios foram a estratégia lúdica mais comum. Portanto, o artigo demonstra uma forte tendência entre o ensino do pensamento computacional e realidade aumentada, pois engaja os estudantes e beneficia na visualização, e prática de algum conceito aplicado, o qual busca-se reproduzir o benefício e prática do jogo baseado em *Bebras* que será desenvolvido neste trabalho.

O trabalho de (LYE; KOH, 2014) consistiu em uma revisão sistemática de 27 estudos empíricos com o objetivo de analisar como o Pensamento Computacional (PC) vem sendo ensinado e aprendido por meio da programação na educação básica (K-12) e no ensino superior. Os autores estruturaram a análise sob três dimensões fundamentais: conceitos, práticas e perspectivas computacionais, identificando que a maioria das intervenções utiliza linguagens de programação visual, como o *Scratch*, para reduzir a carga cognitiva e focar na lógica em detrimento da sintaxe. Nesse sentido, os resultados indicaram que a construção de programas baseada no construcionismo, aliada a suportes como scaffolding (andaime), reflexão e processamento de informações, promove resultados positivos no desempenho dos estudantes. O estudo destaca ainda que o PC desenvolve habilidades de resolução de problemas transferíveis para situações do cotidiano, tornando os alunos criadores ativos de tecnologia e não apenas consumidores passivos. Ademais, a pesquisa aponta uma lacuna em intervenções realizadas em contextos reais de sala de aula e sugere que o uso de ferramentas interativas 3D e o registro do processo de pensamento dos alunos são caminhos promissores para futuras pesquisas. Portanto, este artigo fundamenta a importância de se criar ambientes de aprendizagem baseados na resolução de problemas para consolidar o ensino do PC de forma dinâmica e lúdica.

Neste sentido, artigo de (ZHANG; NOURI, 2019) apresentou uma revisão sistemática de 55 estudos empíricos com o foco voltado especificamente para o ensino fundamental (K-9), investigando as competências de Pensamento Computacional (PC) desenvolvidas

por meio da ferramenta *Scratch*. Utilizando o referencial de Brennan e Resnick, os autores confirmaram que todas as dimensões originais — conceitos, práticas e perspectivas — são passíveis de desenvolvimento na educação básica, com maior frequência de êxito no ensino de conceitos como *loops*, sequências e condicionais. No entanto, a pesquisa identificou desafios cognitivos em tópicos mais complexos, como *loops* aninhados e o uso profundo de variáveis. Um diferencial relevante deste artigo foi a identificação de habilidades adicionais não previstas no *framework* original, como a compreensão de entrada/saída (*input/output*), o design multimodal, o pensamento preditivo e a [Interação Humano-Computador \(IHC\)](#). Os resultados sugerem que o desenvolvimento do PC acompanha a maturidade cognitiva dos estudantes, evoluindo de algoritmos simples de direção nos anos iniciais para sincronização complexa de mídias e lógica booleana nos anos finais do ensino fundamental. Ademais, o estudo ressalta a importância de ferramentas de avaliação automatizada, como o Dr. *Scratch*, para auxiliar professores na medição do progresso dos alunos. Portanto, o artigo contribui para a área ao detalhar "o que pode ser aprendido" em cada faixa etária, oferecendo um guia para o planejamento curricular e para a criação de ferramentas pedagógicas mais eficazes.

Ademais, o trabalho de ([RAIOL et al., 2023](#)) propôs e validou um modelo de avaliação do Pensamento Computacional (PC) na educação básica, estruturado a partir das competências estabelecidas pelo complemento à Base Nacional Comum Curricular (BNCC). A metodologia fundamenta-se na utilização da Taxonomia de Bloom para a definição de objetivos educacionais e na Taxonomia de Solo para a classificação dos níveis de compreensão atingidos pelos estudantes. Para validar a proposta, os autores realizaram um estudo de caso com uma oficina de "Computação Desplugada" utilizando o jogo "Piratas Desplugados", que introduz conceitos de algoritmos, variáveis e estruturas de repetição e decisão de forma lúdica.

Os resultados do estudo demonstraram que o modelo é eficaz para identificar a evolução da aprendizagem, permitindo observar que, após a intervenção, 82% dos alunos atingiram o nível Multiestrutural na Taxonomia de Solo, apresentando compreensão correta dos elementos do problema. Ademais, a pesquisa evidenciou que o uso de atividades desplugadas é uma estratégia viável para o ensino de PC em contextos com infraestrutura tecnológica limitada, garantindo o desenvolvimento de habilidades matemáticas e lógicas integradas. Portanto, o artigo oferece uma ferramenta prática e teoricamente embasada para auxiliar professores no planejamento e na mensuração do desenvolvimento do Pensamento Computacional conforme as novas diretrizes nacionais.

O artigo de ([CRUZ et al., 2023](#)) propõe um modelo estruturado de *Game Design Document (GDD)*, um documento que descreve todos os aspectos de um jogo, para acelerar o desenvolvimento de jogos educativos que ensinam PC. No mais, o modelo inclui elementos pedagógicos essenciais (como objetivos educacionais, mecânicas de jogo

alinhadas a habilidades de PC, narrativa e recursos para professores) e foi validado por especialistas em computação e desenvolvedores de jogos. Neste sentido, os autores destacam que o GDD proposto otimiza a integração entre conteúdo educacional e diversão, abordando desafios como engajamento e avaliação de aprendizagem. Como resultados, o modelo demonstra potencial para padronizar a criação de jogos de PC, embora sua eficácia prática ainda necessite de testes em ambientes reais de ensino, planejados como trabalhos futuros no próprio artigo. Portanto, o estudo enfatiza a importância de documentação rigorosa no design de jogos educativos para garantir qualidade pedagógica e técnica, pois é de suma importância para a criação de qualquer jogo, seja sério ou não, a criação de um documento como este para descrever todo o desenvolvimento, para que não tenha apenas os conceitos de PC aplicados de maneira correta, mas também elementos de gamificação para que o jogador se sinta imerso ao resolver um problema.

Em resumo, a análise dos trabalhos revela um campo de pesquisa que busca integrar o PC ao ensino por meio de tarefas direcionadas à tecnologia, sendo uma delas o *Bebras*, com ferramentas variadas que variam desde o papel até tecnologias como jogos e ambientes de programação virtual (*Scratch*). E assim, os trabalhos diferem quanto aos pilares de PC priorizados, assim como a maneira de se desenvolver. Portanto, justifica-se a criação de um jogo sério 3D, mantendo a pedagogia do *Bebras* e abordando os pilares do PC de cada questão da prova, a Tabela 1 apresenta uma comparação de características deste trabalho.

Tabela 1 – Tabela Comparativa de Características dos Trabalhos Relacionados

Referência	Utiliza <i>Bebras</i>	Desenv. 3D/Digital	Pilares do PC Abordados
(LAISA; HENRIQUE, 2017)	Sim	Não	Abstração, Decomposição e Pensamento Algorítmico
(LOCKWOOD; MOONEY, 2018)	Sim	Não	Abstração, Decomposição e Pensamento Algorítmico
(ARAGÃO; AVELLAR; BARBOSA, 2023)	Não	Analisa o uso	Programação (variáveis, condicionais, iterações)
(LYE; KOH, 2014)	Não	Sim (<i>Scratch</i>)	Resolução de Problemas, Criatividade, Raciocínio Lógico
(ZHANG; NOURI, 2019)	Não	Sim (<i>Scratch</i>)	Avaliação de habilidades de programação e Computação
(RAIOL et al., 2023)	Não	Não	Avaliação de PC (Geral)
Este Trabalho	Sim	Sim	Abstração, Decomposição, Algoritmos, Avaliação, Reconhecimento de Padrões

3.2 Metodologia

A metodologia deste trabalho é estruturada sob duas naturezas complementares: a pesquisa teórica e o desenvolvimento tecnológico. Inicialmente, realizou-se uma pesquisa bibliográfica pautada na investigação de artigos acadêmicos, com o objetivo de analisar as ações implementadas no contexto do ensino de Pensamento Computacional (PC) para os níveis fundamental e médio. Esta etapa focou, especificamente, nas adaptações ocorridas após a publicação do Parecer CNE/CEB nº 1/2022, bem como nas diversas aplicações práticas do PC no ambiente escolar identificadas na literatura recente, comparando também a utilização de desenvolvimento 3D ou digital nas pesquisas dos artigos com este trabalho.

No que tange ao desenvolvimento do jogo sério e no minijogo, a construção do artefato tecnológico foi fundamentada no uso de ferramentas e processos utilizadas na indústria de *software*. A plataforma Unity foi adotada como a *engine* de jogo principal, permitindo a criação e integração de componentes e ambientes tanto em 2D quanto em 3D. A lógica interna, que rege desde os controladores globais até o comportamento específico dos minijogos, foi implementada por meio da linguagem de programação C#.

Além disso, metodologias ágeis como *Scrum* e Extreme Programming (XP) foram úteis na organização da equipe e no desenvolvimento do projeto, os quais se beneficiaram com *sprints* quinzenais, permitindo a comunicação da etapa em que se encontrava o mini-jogo de cada equipe. Além disso, a equipe em que me encontrava, alternava-se entre quem escrevia o código e quem revisava para melhor distribuição de conhecimento e *feedbacks* imediatos.

Para assegurar a integridade e a organização do desenvolvimento, utilizou-se o Git como sistema de controle de versão. O fluxo de trabalho foi estruturado com base no modelo Gitflow, que possibilitou uma divisão clara entre as ramificações de desenvolvimento, funcionalidades e versões estáveis, garantindo uma metodologia de trabalho colaborativa e sistemática durante toda a evolução do projeto.

4 Desenvolvimento

4.1 Pré-desenvolvimento

O projeto começa com o propósito de ensinar Pensamento Computacional para crianças e adolescentes, assim como diz o [BNCC](#) Computação. Ao se ter conhecimento do *Bebras Challenge*, junta-se os dois objetivos em comum, que por meio de questões e quebra-cabeças, pode-se reforçar os conceitos fundamentais do [PC](#). Neste sentido, surge a ideia de desenvolver um conjunto de minijogos digitais tendo com base as questões *Bebras*, formando um ecossistema de jogos com objetivos de auxiliar o ensino do [PC](#). Ademais, o trabalho aqui desenvolvido, faz parte deste projeto colaborando o desenvolvimento do minijogo "Jogo da costura" do desafio *Bebras* "Pontos de costura" ([BEBRAS, 2019](#)). Para que assim, funcione como ferramenta ao educador mas que também haja engajamento por parte do público mais jovem ([LYE; KOH, 2014](#)) e complemente o que foi dado em sala de aula, ou seja, o educador leciona sobre certos temas do [PC](#) e monta um circuito de minijogos para reforçar os conceitos dados na sala de aula.

Assim, monta-se uma equipe formada por programadores e professores, e com o time de desenvolvedores, foi considerado utilizar o método de desenvolvimento ágil [XP](#), onde foi realizado trabalho em dupla para se codificar os exercícios da prova *Bebras* e transformá-los em uma fase do jogo sério. Ao se formar três equipes de desenvolvimento, as questões foram escolhidas de modo a serem equilibradas em nível fácil, médio e difícil, pois segundo ([BARBOSA et al., 2014](#)), os jogadores precisam resolver problemas por meio de progresso de jogo, os quais acumulam experiência e ferramentas em níveis mais fáceis para os níveis mais difíceis, além disso, como ([CARDOSO et al., 2025a](#)) explica, cada questão possui uma habilidade do [PC](#), Abstração, Pensamento Algorítmico, Decomposição, Avaliação e Generalização, que foram consideradas também na escolha das questões, para os minijogos desenvolvidos serem diversos nas habilidades reforçadas. Para a organização, o github foi usado para versionamento do código e cada dupla trabalhar em seu mini-jogo sem a necessidade de conflitos.

4.2 Concepção

Neste sentido, a equipe escreveu o [GDD](#) do jogo como um todo, e um [GDD](#) de cada um dos mini-jogos. Neste documento geral, foi descrito pontos essenciais, para que a equipe de programadores tenha uma direção a se seguir no desenvolvimento. Exemplo do que é descrito no tópico Mecânica do anexo [A](#).

O jogo será composto por uma série de ações padronizadas, que servirão como base para a resolução dos diferentes minijogos. Cada minijogo poderá combinar e utilizar essas ações conforme necessário para sua conclusão. Por isso, todas as ações disponíveis deverão ser apresentadas e explicadas ao jogador logo no início da experiência (CANABARRO et al, 2025).

Além de mecânicas de jogo, é apresentado como o jogo deve ocorrer, ou seja, o fluxo que o jogador deve percorrer pelo jogo também é formalizado neste documento, desde ações tomadas pelo jogador desde o educador que escolherá as fases, também contidos no anexo [A](#).

Todas as aplicações disponibilizadas pelo educador começarão com uma interface, contendo algumas configurações para acessibilidade, e a possibilidade de selecionar os mini jogos que serão incluídos naquela avaliação. [...]

Ao iniciar o jogo após a seleção das fases, o jogador será transportado para um cenário de floresta, onde encontrará um ambiente de piquenique que servirá como *lobby* central. Esse espaço terá uma função introdutória, apresentando as principais mecânicas do jogo e familiarizando o jogador com o sistema (CANABARRO et al, 2025).

Além destes pontos mais técnicos para jogos, existem pontos de design como a narrativa, que é uma excelente escolha para inspirar motivação para quem está jogando. Neste caso, a escolhida pela equipe no tópico de Narrativa e Estética foi a de fantasia, pois a prova em que se baseia o projeto é para crianças do sétimo e oitavo ano, assim a história fica mais receptível para o público-alvo. A história se baseia em um castor, animal figura da prova *Bebras* ([Bebras Challenge, 2025](#)), em que este castor, ao combinar um pique-nique com sua família em um dia, esquece de seu compromisso e terá que correr contra o tempo para conseguir itens para seu dia em família, assim terá que conversar com animais da floresta para resolver seus problemas, as questões da prova, para receber uma recompensa em troca, os quais são exatamente os itens necessários para concluir o pique-nique.

Assim, cada minijogo pode se encaixar nesta narrativa, sempre que o jogador resolver uma questão, acabará recebendo uma recompensa de acordo com itens relacionados a pique-nique, para que o castor conclua sua jornada. Ademais, para guiar a modelagem de cada questão da prova *Bebras* para o mundo virtual, cada dupla escreveu seu próprio [GDD](#) em conformidade com o documento geral do projeto. Estes documentos seguiram a estrutura proposta por ([CRUZ et al., 2023](#)) para jogos educacionais, mesclando elementos pedagógicos com componentes de jogos de entretenimento. Conforme exemplificado pelo [GDD](#) do minijogo da Costura em anexo [B](#), foram definidos:

- **Objetivos Educacionais**, como o desenvolvimento do pensamento algorítmico e a avaliação de máquinas de estado finitos.
- **Público-Alvo**, focado em estudantes do 7º ao 8º ano (12-13 anos) do ensino fundamental.
- **História e Narrativa**, na qual o jogador ajuda uma ovelha a consertar panos em troca de um item para seu piquenique
- **Mecânicas e Dinâmicas de Jogo**, que envolvem analisar a máquina de estados e interagir com sua interface para encontrar a sequência incorreta de costura.

(CANABARRO, TOFFOLI, 2025)

Essa abordagem alinha-se ao que defende (BARBOSA et al., 2014) sobre a importância da motivação no aprendizado baseado em problemas, utilizando a narrativa para criar um problema com significado e engajar o aluno na busca pela solução.

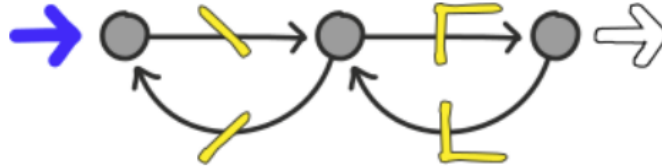
Desta maneira, foi desenvolvido o jogo da costura Figura 1, uma questão que consta na prova como fácil, e que se encontra no entendimento de MEF e a validação de mensagens, que podem ou não podem serem reconhecidas pela máquina. Pelo conceito do PC, o jogo se encontra em duas áreas, Pensamento Algorítmico, pois o aluno precisa pensar em sequências e regras, além de executar um algoritmo que seria a sequência correta da costura respeitando o autômato, e Avaliação, pois de acordo com as regras, precisa encontrar a melhor solução adequada. Pela narrativa, o castor encontra-se com um animal, uma ovelha, que não entendeu como se usa a máquina e costurou quatro panos, porém uma delas errado, cabe ao jogador compreender o uso da máquina de costura pela Máquina de Estados Finita (MEF) e descobrir qual a sequência errada que a ovelha costurou, recebendo como recompensa um pano de piquenique para o castor continuar sua jornada em busca de itens relacionados com a história.



Dificuldade: fácil | Origem:

2 – Pontos de Costura

Uma máquina de costura consegue fazer quatro tipos de pontos. As regras que a máquina segue para decidir qual o próximo ponto a fazer estão descritas no diagrama:



A máquina começa uma nova linha de pontos seguindo a seta que está mais à esquerda. Depois, a máquina vai de círculo em círculo, seguindo as setas. Cada vez que passa por uma seta, faz o ponto de costura indicado por essa seta. Cada vez que chega a um círculo e este permite seguir mais do que uma seta, a máquina pode escolher qual a seta que quer seguir. Os pontos terminam quando a máquina chega à seta que está mais à direita.

Pergunta

Qual dos seguintes pontos de costura não pode ser feito utilizando as regras acima?

Respostas Possíveis

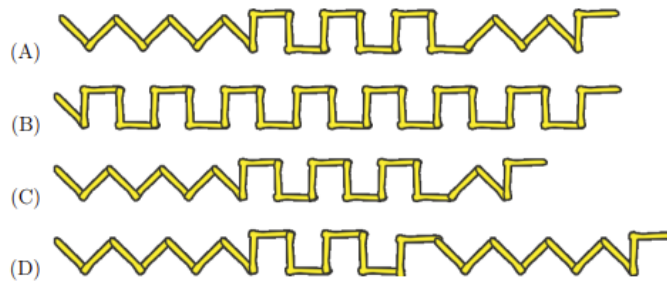


Figura 1 – Questão 2 da prova *Bebras*.

Fonte: (BEBRAS, 2019)

Em se tratando da questão, o aluno teria que avaliar a sequência das linhas de costura, e observar de acordo com as regras definidas, por meio da MEF, qual dos pontos de costuras são possíveis de fazer e quais não são. Neste sentido, o aluno pode observar e executar as respostas possíveis acompanhando as setas demonstradas no começo da questão, percebendo que a letra D, é uma sequência não permitida pela MEF, como mostra na Figura 2.

Respostas Possíveis

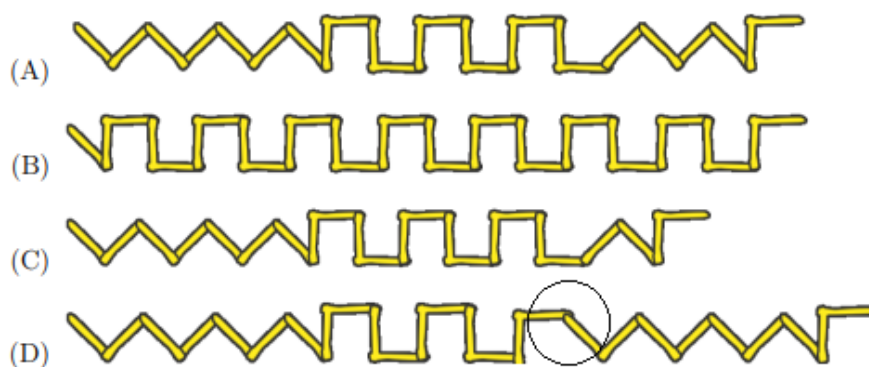


Figura 2 – Resposta da questão Pontos de Costura.

Fonte: (BEBRAS, 2019)

Ao se tratar de implementação, o progresso das duplas era apresentado e comentado pela equipe, que também aproveitava para alinhar padronização de interfaces utilizadas e atualização de todos com suas respectivas narrativas e implementação. Pelas duplas, cada uma combinaria reuniões entre si para discutirem como iriam implementar a prova feita no papel para o conceito de jogo sério, utilizando a *engine* de jogos Unity 3D e *scripts* em C#. E uma das reuniões da dupla, houve discussões da escrita do GDD do mini-jogo da costura, contido no anexo B, onde discutiu-se aspectos mais específicos das mecânicas e narrativas.

4.3 Desenvolvimento Técnico

Inicialmente, o jogo foi desenvolvido com navegação livre em primeira pessoa, permitindo ao jogador explorar o ambiente tridimensional de forma independente. Nesse modelo, o usuário poderia deslocar-se pelo cenário utilizando os comandos tradicionais de jogos do gênero, interagindo com **Non-Playable Character (NPC)**, ou personagens não jogáveis, para iniciar as missões.

Ao dialogar com o **NPC** responsável por cada fase, no caso do minijogo de costura, uma ovelha vide Figura 4a, a narrativa era apresentada por meio de caixas de diálogo, e o minijogo correspondente era liberado para interação. Assim, utilizando as teclas 'X', 'C', 'Z', 'V' do teclado para utilizar e interagir com o autômato, o jogador deveria completar três sequências de linhas geradas aleatoriamente, Figura 3, clicando na máquina de costura mostrado na Figura 4b para prosseguir para a etapa seguinte após a confirmação da resposta junto ao **NPC**.

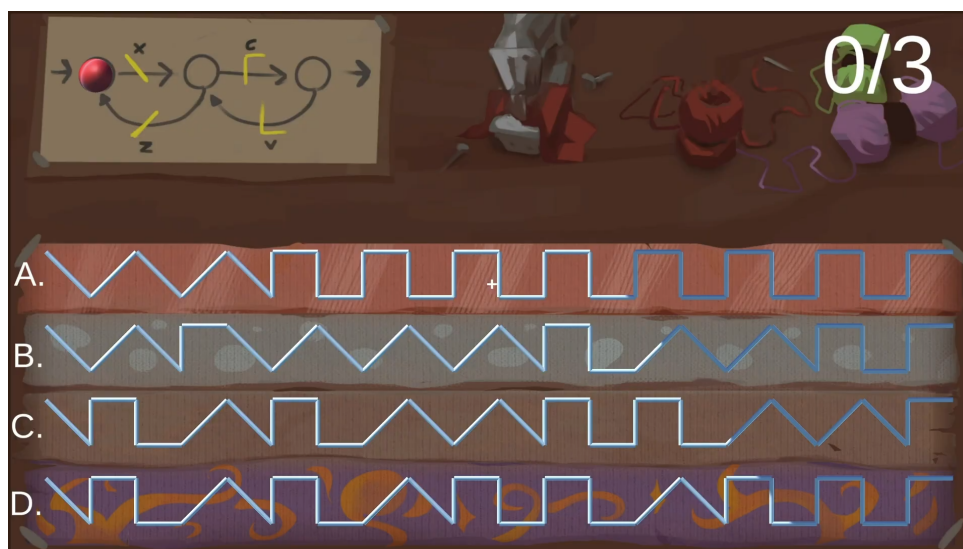


Figura 3 – Primeira versão do minijogo da costura.

Fonte: Captura de tela realizada pelo autor



(a) Interação com a ovelha na primeira versão.



(b) Interação com a máquina de costura.

Figura 4 – Interações na versão antiga do jogo

Fonte: Captura de tela realizada pelo autor

A movimentação do personagem era realizada por meio das teclas 'W', 'A', 'S' e 'D', enquanto a orientação da câmera era controlada pelo mouse, padrão amplamente utilizado em jogos de primeira pessoa. O jogador poderia percorrer a área inicial do jogo e escolher diferentes caminhos ao interagir com [NPC's](#) distribuídos pelo cenário, como mostrado na Figura 5.



Figura 5 – Cenário antigo com o NPC Pato.

Fonte: Captura de tela realizada pelo autor

Após a conclusão da primeira versão, foram realizados testes com usuários externos ao público-alvo definido no GDD, oito estudantes do mestrado profissional do [Programa de Pós-Graduação em Tecnologias, Comunicação e Educação \(PPGCE\)](#) da Faculdade de Educação da [Universidade Federal de Uberlândia \(UFU\)](#). O procedimento adotado consistiu na aplicação inicial de uma questão do Bebras em formato impresso, seguida da interação dos participantes com o jogo e seus minijogos, da aplicação de uma nova questão também em formato impresso e, por fim, aplicação do teste, conforme descrito em ([CARDOSO et al., 2025b](#)). Esse processo permitiu não apenas avaliar o desempenho dos usuários na resolução de problemas de pensamento computacional, mas também observar aspectos relacionados à usabilidade e à interação com o sistema.

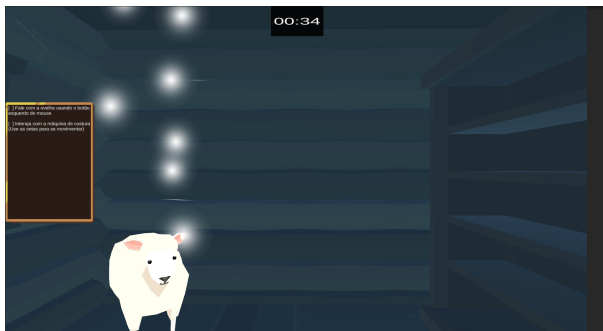
Os resultados mostrados na Tabela 2, segundo a avaliação [System Usability Scale \(SUS\)](#), indicaram dificuldades significativas de navegação, nas quais os jogadores frequentemente se perdiam no ambiente virtual, além de apresentarem limitações na coordenação necessária para o uso simultâneo de teclado e mouse. Observou-se também incerteza quanto aos objetivos do jogo e dificuldades na compreensão das mecânicas dos minijogos, especialmente no que se refere à interação por meio de teclas no minijogo de costura. Tais dificuldades também foram evidenciadas pelas respostas qualitativas dos participantes, onde os participantes opinaram sobre a narrativa do jogo e sobre os conceitos abordados de [PC](#) descritos em ([CARDOSO et al., 2025b](#)).

Tabela 2 – Resultados da aplicação do SUS

Participantes	Ímpares						Pares						SUS
	1	3	5	7	9	Soma	2	4	6	8	10	Soma	
Ficha 01	0	0	0	0	0	0	0	0	2	1	0	3	7,5
Ficha 02	4	2	4	3	2	15	4	3	4	4	1	16	77,5
Ficha 03	3	3	4	4	1	15	4	3	4	4	4	19	85
Ficha 04	3	3	3	4	3	16	4	2	4	4	1	15	77,5
Ficha 05	3	3	4	4	3	17	4	1	1	4	4	14	77,5
Ficha 06	4	4	4	3	4	19	4	1	4	4	2	15	85
Ficha 07	1	1	3	1	2	8	1	1	1	1	1	5	32,5
Ficha 08	3	2	4	2	3	14	3	3	4	4	4	18	80
Média SUS													65,3125

Fonte: Adaptado de (CARDOSO et al., 2025b)

Assim, optou-se por reformular a mecânica de exploração com foco na melhoria da usabilidade e na redução da carga cognitiva e motora exigida dos jogadores. Assim, a navegação livre foi substituída por um sistema guiado, no qual o personagem permanece em posição fixa e pode se mover apenas nas quatro direções cardeais por meio das teclas direcionais. Ademais, a interação com elementos do cenário passou a ocorrer por cliques do mouse em objetos destacados visualmente, tornando mais evidente o que é interativo no ambiente. Essa abordagem reduz tanto a complexidade de controle quanto a incerteza sobre as ações possíveis, contribuindo para uma experiência mais intuitiva. Para auxiliar na identificação dos elementos interativos, foram adicionados efeitos visuais como partículas indicativas (Figura 6a), contornos nos objetos selecionáveis (Figura 6b) e um painel de objetivos que atua como um guia contínuo de movimentação e interação, reforçando o aprendizado das mecânicas ao longo da experiência.



(a) Interação com a ovelha na versão nova.



(b) Interação com a máquina de costura

Figura 6 – Interações na versão nova do jogo

Fonte: Captura de tela realizada pelo autor

Assim, a mecânica do minijogo da costura também foi modificado para melhor clareza, com botões interativos com o mouse para criar e apagar as linhas de costura, Figura 7, além de um sistema de *feedback* ao completar as linhas de costura igual ao

diálogo, Figura 8, quando o jogador acerta uma sequência, uma caixa de diálogo aparece com a ovelha dizendo que é correta, 8a, se o jogador fizer uma sequência inexistente, vários diálogos de erro foram introduzidos para maior entendimento a quem está jogando 8b.

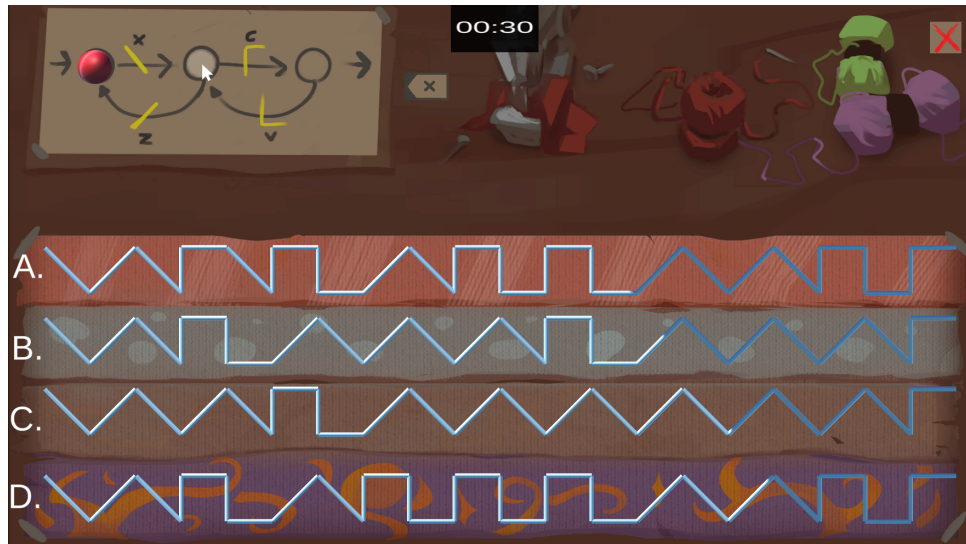


Figura 7 – Nova versão após os testes.
Fonte: Captura de tela realizada pelo autor



(a) *Feedback* do primeiro acerto.



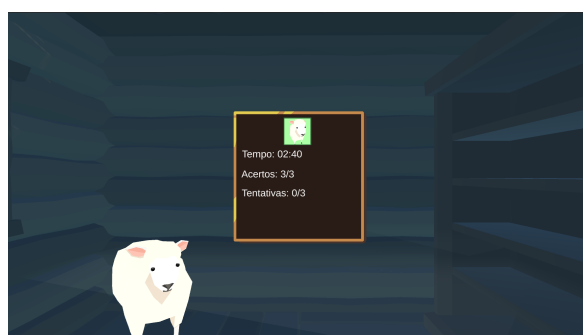
(b) *Feedback* de sequência inexistente.

Figura 8 – Sistema de *feedback* do minijogo.
Fonte: Captura de tela realizada pelo autor

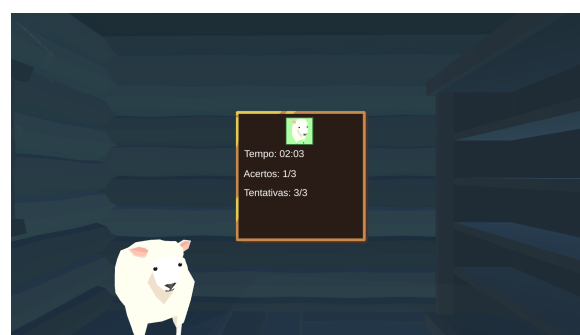
Neste sentido também, foi desenvolvido uma mecânica de confirmação de resposta, Figura 10, para que se tenha certeza que o usuário soube resolver o autômato de forma autêntica e não de forma aleatória, a qual se baseia em perguntar para o jogador qual é de todas as sequências que resolveu, qual é a incorreta, resultando em dois cenários. Um representado na Figura 10a, onde o jogador conseguiu realizar de uma a três sequências, com o seu tempo para realizar o minijogo, e em quantas tentativas, podendo assim ter três chances para acertar qual a sequência incorreta. O mesmo para o cenário de erro, na Figura 10b, o qual o jogador errou as três tentativas.



Figura 9 – Tela de confirmação.
Fonte: Captura de tela realizada pelo autor



(a) Cenário de acerto.



(b) Cenário de erro.

Figura 10 – Sistema de confirmação de resposta.
Fonte: Captura de tela realizada pelo autor

Para representar as funcionalidades disponíveis aos usuários, foi elaborado um diagrama de casos de uso, Figura 11, ilustrando as principais interações entre o jogador e o sistema. O ator principal é o jogador, responsável por executar ações como iniciar o minijogo interagindo com a máquina de costura, participar dos minijogos clicando nos estados do autômato, criando linhas de costura, e ainda interagir com o botão de apagar qualquer linha se preferir e receber *feedbacks*. Ademais, a documentação dos casos de uso, contidos no Anexo C, descrevem como funcionam os fluxos do diagrama, como no exemplo na Figura 12 do caso de uso de Iniciar o Minijogo, onde é descrito as condições iniciais e finais da ação do jogador, e os fluxos descritos com a resposta do sistema.

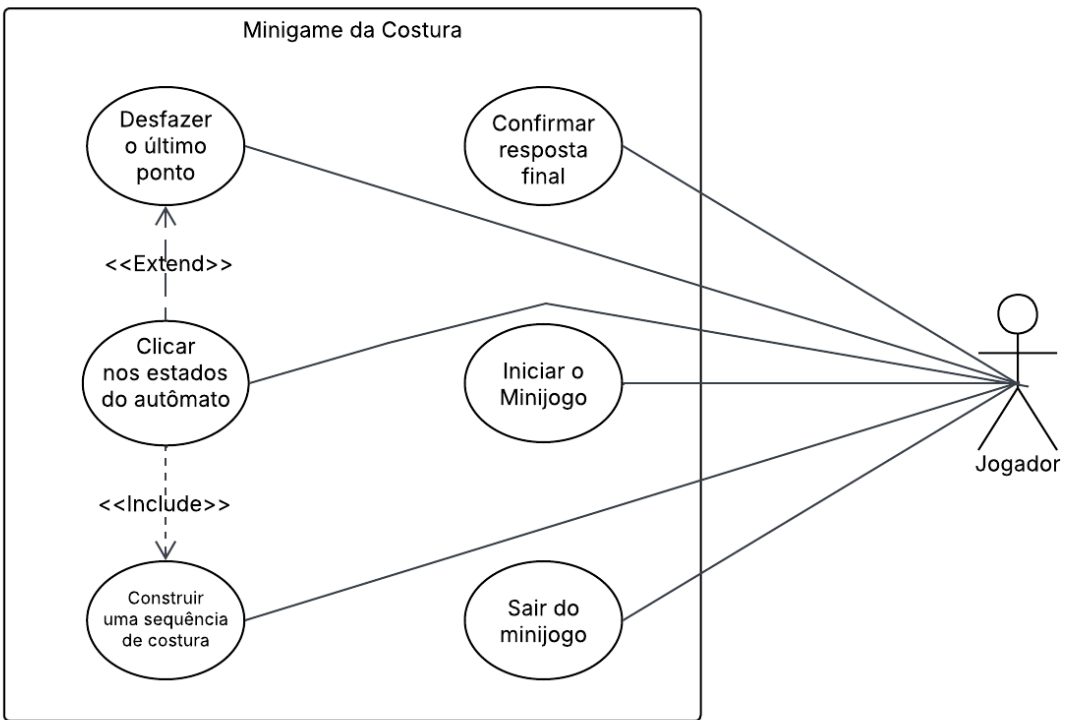


Figura 11 – Diagrama de caso de uso do jogo.
Fonte: Elaborado pelo autor

Nome	Iniciar o Minijogo
Ator Principal	Jogador
Resumo	Permite que o jogador inicie o minijogo da costura e visualize a interface do autômato e controles
Pré-condições	Conversar com a ovelha
Pós-condições	Minijogo carregado, estado inicial do autômato e UI exibidos
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Clica na máquina de costura	
	2. Carrega interfaces
	3. Inicia variáveis
Fluxo Alternativo	
1. Clica na máquina de costura	
	2. Carrega mensagem para conversar com a ovelha antes de iniciar

Figura 12 – Documentação do caso de uso Iniciar Minijogo.
Fonte: Elaborado pelo autor

A estrutura interna do *software* foi modelada por meio de um diagrama de classes, Figura 13, que apresenta os principais componentes responsáveis pelo funcionamento

do jogo. Destaca-se a classe *GameManager*, responsável pelo controle do fluxo do jogo, gerenciamento de estados, interação com os minijogos e comunicação com os sistemas de interface e diálogo. Ademais, a Unity foi usada como motor gráfico justamente por ser uma ferramenta com um padrão arquitetural conhecido como [Sistema de Entidade e Componentes \(C/ES\)](#) (ALMEIDA, 2016), o qual os objetos (*GameObjects*) que compõem o jogo são tratados como entidades, que por sua vez tem seu aspecto definido por componentes, ou seja, funcionalidades são implementadas por meio destes. Neste sentido, todas as classes da Unity herdam uma classe base chamada *MonoBehaviour*, tornando-as componentes, que permite acessar métodos específicos do chamado ciclo de vida do *script*, onde é possível acessar métodos como *Start*, que iniciam no primeiro *frame* do código, e *Update* que é monitorado pela Unity a cada *frame* do jogo.

Desta maneira, a classe desenvolvida *GameManager* é a mais importante do jogo, pois orquestra a interação entre usuário e o modelo matemático do autômato finito, onde foram usadas estruturas como listas para armazenar a sequência de linhas e pilhas para armazenar e permitir o retrocesso de linhas em caso de erro de percurso. Além disso, foi implementado no método *GenerateExercises* desta classe, um algoritmo de randomização que gera quatro alternativas distintas, três válidas e uma inválida, a cada nova partida, assim evitando aprendizado por memorização e forçando o jogador a interpretar as transições do autômato.

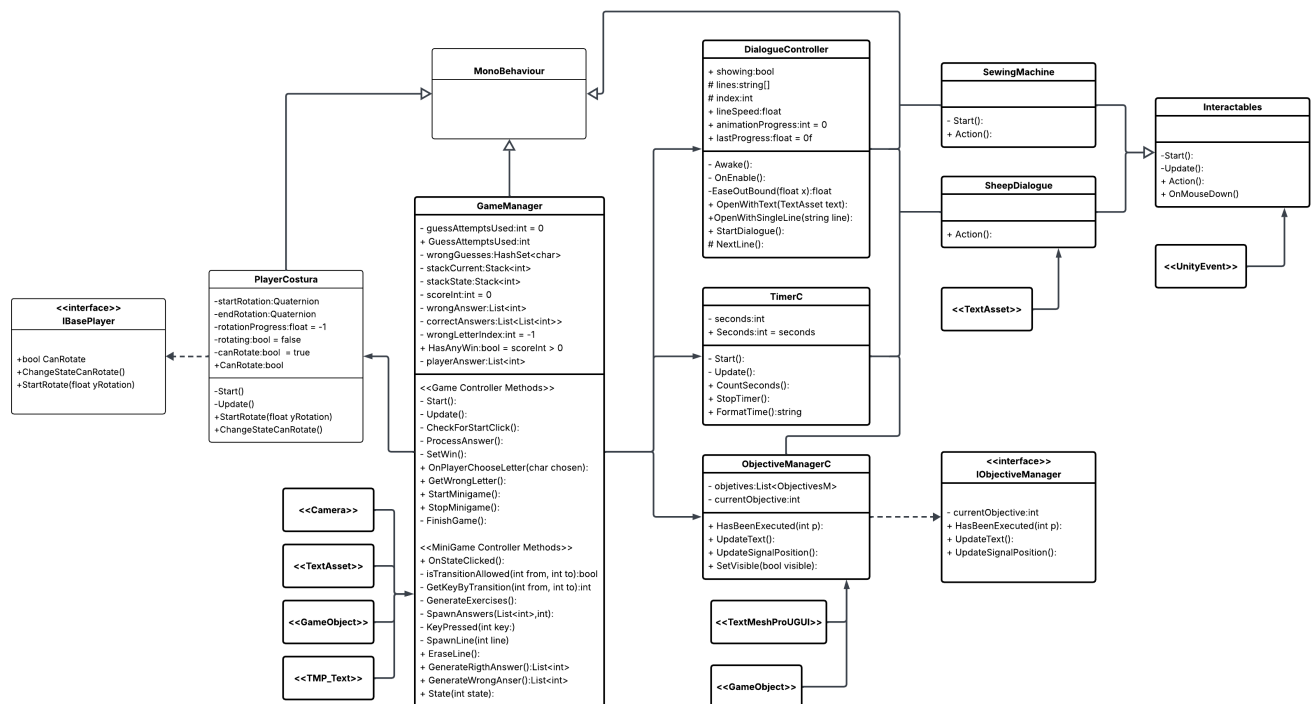


Figura 13 – Diagrama de classe.

Fonte: Elaborado pelo autor

Em síntese, este capítulo apresentou o processo de desenvolvimento do jogo, desde

a etapa de pré-desenvolvimento e concepção, embasada na elaboração do [GDD](#) e no planejamento do ecossistema de minijogos baseado no *Bebras*, até a implementação do minijogo da costura na Unity. Além disso, destacou-se a evolução técnica do projeto a partir de testes com usuários, que motivou a reformulação da navegação e da interação para reduzir a habilidade motora e tornar os objetivos mais explícitos, incorporando elementos de *feedback* visual e textual. Por fim, os diagramas de casos de uso e de classes foram utilizados para documentar, respectivamente, as funcionalidades oferecidas aos atores do sistema e a organização interna do *software*, consolidando a relação entre decisões de design e sua implementação.

5 Conclusão

Este trabalho apresentou o projeto e a implementação de um protótipo de jogo sério inspirado em tarefas do *Bebras*, visando oferecer uma alternativa digital para o fomento do Pensamento Computacional na educação básica. Pautada nas diretrizes da [BNCC](#), a proposta buscou transpor desafios tradicionalmente aplicados em papel para um ambiente 3D, explorando interações que visam tornar a experiência mais dinâmica. Como contribuição central, detalhou-se o desenvolvimento do minijogo da costura, fundamentado em [MEF](#), o qual integrou mecânicas de interação, randomização de estados e sistemas de *feedback*.

Os objetivos propostos foram alcançados por meio da entrega de um módulo funcional capaz de operacionalizar uma tarefa lógica em um contexto lúdico. Ao digitalizar o desafio, o sistema se apresenta como uma potencial ferramenta de apoio ao educador, podendo atuar como recurso complementar no desenvolvimento de habilidades como pensamento algorítmico e reconhecimento de padrões. Por fim, o projeto estabeleceu uma base técnica e metodológica que favorece a continuidade do desenvolvimento, permitindo a futura expansão do ecossistema de minijogos e a posterior validação de sua eficácia pedagógica em ambiente escolar.

Como continuidade deste projeto, recomenda-se a integração completa do minijogo desenvolvido ao ecossistema de minijogos previstos no [GDD](#) geral, incluindo um fluxo unificado de *lobby*, seleção de fases pelo educador e registro consolidado do desempenho do jogador. Além disso, um passo importante é evoluir a arquitetura do jogo para facilitar a adição de novos minijogos de forma modular, por exemplo, por meio de interfaces e contratos padronizados para inicialização, execução, finalização e envio de resultados ao gerenciador central. Essa abordagem reduziria acoplamento, evitaria duplicação de código e permitiria que novos minijogos fossem incorporados com menor esforço e maior consistência de interface.

Também é sugerida a ampliação de recursos de avaliação, como armazenamento de métricas (tempo, tentativas, tipos de erro) e exportação de relatórios simplificados para o educador, possibilitando acompanhamento pedagógico mais direto. Por fim, recomenda-se realizar novos testes com o público-alvo (alunos do ensino fundamental), comparando a efetividade do jogo em relação a atividades tradicionais, investigando aspectos como engajamento, compreensão dos conceitos e retenção do conteúdo, a fim de validar empiricamente os ganhos educacionais propostos.

Referências

- ALMEIDA, M. S. O. **Design de jogos baseado em componentes**. Tese (Doutorado) — Universidade de São Paulo, 2016. Citado na página 34.
- ANTUNES, J.; NASCIMENTO, V. S. d.; QUEIROZ, Z. F. d. CIENCIA DOS JOGOS APLICADA A EDUCACAO: UM ESTUDO DO PROCESSO CRIATIVO DE CONFECCAO DO JOGO ANALOGICO UMUECHEM. **Revista da FAEEBA: Educacao e Contemporaneidade**, scielo, v. 29, p. 377 – 402, 07 2020. ISSN 0104-7043. Disponível em: <http://educa.fcc.org.br/scielo.php?script=sci_arttext&pid=S0104-70432020000300377&nrm=iso>. Citado na página 12.
- ANUNCIAÇÃO, S. R. da F.; ARAÚJO, P. V. S. da S.; SANTOS, Y. Y. L.; MENEZES, G. O. de; LUZ, J. W. P. O desenvolvimento do pensamento computacional em crianças: Impactos e aplicações. **Congresso Nacional de Educação**, 2025. Citado na página 9.
- ARAGÃO, P. A. P.; AVELLAR, G. M. N.; BARBOSA, E. F. Ensino de programação e pensamento computacional utilizando realidade virtual, realidade aumentada e jogos: Um mapeamento sistemático da literatura. In: **Anais do Simpósio Brasileiro de Informática na Educação**. [S.l.]: SBC, 2023. p. 1–13. Citado 2 vezes nas páginas 18 e 21.
- BARBOSA, A. F. S.; PEREIRA, P. N. M.; DIAS, J. A. F. F.; SILVA, F. G. M. A new methodology of design and development of serious games. **International Journal of Computer Games Technology**, Hindawi Publishing Corporation, v. 2014, p. 1–9, 2014. Citado 2 vezes nas páginas 23 e 25.
- BEBRAS. **Castor Informático - O desafio internacional de Pensamento Computacional**. Portugal: [s.n.], 2019. Acesso em: 06 fev. 2025. Disponível em: <<https://bebras.pt/>>. Citado 3 vezes nas páginas 23, 26 e 27.
- Bebras Challenge. **Bebras – International Challenge on Informatics and Computational Thinking**. 2025. Acesso em: 15 mar. 2025. Disponível em: <<https://www.bebbras.org/>>. Citado 2 vezes nas páginas 9 e 24.
- BETHKE, E. **Game development and production**. [S.l.]: Wordware Publishing, Inc., 2003. Citado na página 15.
- CARDOSO, A.; GRAFFUNDER, F.; OLIVEIRA, G.; TOFFOLI, G.; OTONI, G.; GOMIDES, G.; OLIVEIRA, K.; BARBOSA, L.; LIMA, M.; LOPES, R.; DOMINGUES, R. Avaliação de um jogo sério com minijogos baseados no desafio bebras para apoio ao ensino do pensamento computacional. In: **Anais do XXXIII Workshop sobre Educação em Computação**. Porto Alegre, RS, Brasil: SBC, 2025. p. 837–848. ISSN 2595-6175. Disponível em: <<https://sol.sbc.org.br/index.php/wei/article/view/36221>>. Citado na página 23.
- _____. Avaliação de um jogo sério com minijogos baseados no desafio bebras para apoio ao ensino do pensamento computacional. In: **Anais do XXXIII Workshop sobre Educação em Computação**. Porto Alegre, RS, Brasil: SBC, 2025. p. 837–848. ISSN

2595-6175. Disponível em: <<https://sol.sbc.org.br/index.php/wei/article/view/36221>>. Citado 2 vezes nas páginas 29 e 30.

Conselho Nacional de Educação - Câmara de Educação Básica (CNE/CEB). **Parecer CNE/CEB nº 2, de 2022**. [S.l.], 2022. Acesso em: 15 mar. 2025. Disponível em: <https://normativasconselhos.mec.gov.br/normativa/view/CNE_PAR_CNECEB_02_2022.pdf>. Citado na página 9.

CRUZ, A. K. B. S. da; NETO, C. de S. S.; TEIXEIRA, M. M.; CRUZ, P. T. M. B. da; BARBOSA, K. B.; BRITO, C. P. L. Proposal for a game design document model for creating educational games to teach computational thinking. In: **Anais do Simpósio Brasileiro de Informática na Educação (SBIE)**. [S.l.]: Sociedade Brasileira de Computação (SBC), 2023. p. 1–13. Citado 2 vezes nas páginas 19 e 24.

DAGIENÈ, V.; SENTANCE, S.; STUPURIENÈ, G. Developing a two-dimensional categorization system for educational tasks in informatics. **Informatica**, v. 28, n. 1, p. 23–44, 2017. Disponível em: <<https://journals.sagepub.com/doi/abs/10.3233/INF-2017-1127>>. Citado na página 14.

FARDO, M. L. A gamificação aplicada em ambientes de aprendizagem. **RENOTE**, v. 11, n. 1, jul. 2013. Disponível em: <<https://seer.ufrgs.br/index.php/renote/article/view/41629>>. Citado na página 14.

GARCIA, S.; BORGES, M. Recomendações para grupos que fomentam o pensamento computacional. In: **Anais do II Workshop de Pensamento Computacional e Inclusão**. Porto Alegre, RS, Brasil: SBC, 2023. p. 135–144. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/wpci/article/view/26156>>. Citado na página 9.

GERSTING, J. L. **Fundamentos matemáticos para a ciência da computação**. [S.l.]: LTC, 2001. v. 3. 443 p. Citado na página 16.

GROVER, S.; PEA, R. Computational thinking: A competency whose time has come. In: _____. [S.l.]: ResearchGate, 2017. p. 21. ISBN 978-1-3500-5711-1. Citado na página 12.

IBRAHIM, R.; JAAFAR, A. Educational games (eg) design framework: Combination of game design, pedagogy and content modeling. In: **2009 International Conference on Electrical Engineering and Informatics**. [S.l.: s.n.], 2009. v. 01, p. 293–298. Citado na página 15.

LAAMARTI, F.; EID, M.; SADDIK, A. E. An overview of serious games. **International Journal of Computer Games Technology**, v. 2014, n. 1, p. 358152, 2014. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1155/2014/358152>>. Citado na página 13.

LAISA, J.; HENRIQUE, E. Computational thinking game design based on the bebras challenge: A controlled experiment. In: **Proceedings of the International Conference on Informatics in Schools**. [S.l.]: Springer, 2017. p. 1–11. Citado 2 vezes nas páginas 17 e 21.

LOCKWOOD, J.; MOONEY, A. Developing a computational thinking test using bebras problems. In: **Proceedings of the International Conference on Engaging Pedagogy (ICEP)**. [S.l.: s.n.], 2018. p. 1–12. Citado 2 vezes nas páginas 17 e 21.

- LYE, S. Y.; KOH, J. H. L. Review on teaching and learning of computational thinking through programming: What is next for k-12? **Computers in human behavior**, Elsevier, v. 41, p. 51–61, 2014. Citado 3 vezes nas páginas 18, 21 e 23.
- MENEZES, E. V. e Heraclito Junior e Patricia Rios e C. Aprendendo sobre o uso de jogos digitais na educação. **Anais do Workshop de Informática na Escola**, v. 21, n. 1, p. 444–453, 2015. ISSN 2316-6541. Disponível em: <<http://milanesa.ime.usp.br/rbie/index.php/wie/article/view/5072>>. Citado na página 13.
- RAIOL, A.; PORTELA, C.; SANTOS, I.; VIANA, J.; SANTOS, V.; SOUSA, D. Um modelo de avaliação do nível de aprendizagem do pensamento computacional aderente à bncc. In: **Anais do XXXIV Simpósio Brasileiro de Informática na Educação**. Porto Alegre, RS, Brasil: SBC, 2023. p. 1801–1812. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/sbie/article/view/26800>>. Citado na página 21.
- RAIOL, A. A. d. C. et al. Um modelo de avaliação do nível de aprendizagem do pensamento computacional aderente à bncc. **Congresso Brasileiro De informática na educação**, Universidade Federal do Pará, 2023. Citado na página 19.
- SCHEFFEL, E.; MOTTA, C. Desenvolvimento das competências de computação dispostas na bncc a partir da aprendizagem baseada em problemas com alunos do ensino fundamental. In: **Anais do XXXIII Simpósio Brasileiro de Informática na Educação**. Porto Alegre, RS, Brasil: SBC, 2022. p. 85–94. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/sbie/article/view/22398>>. Citado na página 9.
- SCHWABER, K.; SUTHERLAND, J. The scrum guide. **Scrum Alliance**, v. 21, n. 1, p. 1–38, 2011. Citado na página 11.
- SOARES, M. d. S. Metodologias Ágeis extreme programming e scrum para o desenvolvimento de software. **Revista Eletrônica de Sistemas de Informação**, v. 3, n. 1, 2004. Citado na página 11.
- SOUSA, R. M.; LENCASTRE, J. A. Scratch: uma opção válida para desenvolver o pensamento computacional e a competência de resolução de problemas. **RepositoriUM**, Universidade do Minho. Centro de Investigação em Educação (CIEd), 2014. Citado na página 13.
- WING, J. M. Computational thinking. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 49, n. 3, p. 33–35, mar. 2006. ISSN 0001-0782. Disponível em: <<https://doi.org/10.1145/1118178.1118215>>. Citado na página 12.
- ZHANG, L.; NOURI, J. A systematic review of learning computational thinking through scratch in k-9. **Computers & Education**, Elsevier, v. 141, p. 103607, 2019. Citado 2 vezes nas páginas 18 e 21.
- ZHONGGEN, Y. A meta-analysis of use of serious games in education over a decade. **International Journal of Computer Games Technology**, v. 2019, n. 1, p. 4797032, 2019. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1155/2019/4797032>>. Citado 2 vezes nas páginas 12 e 13.

Anexos

ANEXO A – Game Design Document (GDD) Geral

The Beaver's Odyssey

Game Design Document

Versão: 1.0

Autores:

Felipe Canabarro Graffunder

Gabriel Henrique de Oliveira

Guilherme Duarte Toffoli

Guilherme Henrique Andrade Otoni

Gustavo Duarte Gomides

Kiara de Oliveira

Luis Arthur Rabelo Barbosa

Uberlândia,
Fevereiro de 2025

Índice

1. Concepção e desenvolvimento.....	3
1.1 Categoria.....	3
1.2 Conceito.....	3
1.3 Plataforma.....	3
1.4 Cenário.....	3
1.5 Mecânica.....	4
1.6 Fluxo do jogo.....	4
2. Game Design.....	7
2.1 Narrativa e Estética.....	7
2.2 Padronização dos Mini-jogos.....	7
3. Produção.....	8
3.1 Interfaces.....	8
3.2 Ferramentas.....	12

1. Concepção e desenvolvimento

1.1 Categoria

Educacional; Jogo educacional para inteligência computacional; Pensamento lógico; Uso de métodos avaliativos; Aventura; Escape room;

1.2 Conceito

Para avaliar o conhecimento e as habilidades dos alunos em pensamento computacional, será utilizada a iniciativa Bebras. Seus desafios consistem em tarefas curtas, interativas e baseadas em imagens. O jogo consistirá em uma integração do desafio Bebras, utilizando elementos de jogo em atividades de aprendizagem, como níveis, pontos e recompensas. Portanto o projeto tem como papel, apresentar um jogo de fuga personalizável, projetado para praticar habilidades de pensamento computacional em diferentes níveis de educação para alunos do ensino básico. Os objetivos do projeto incluem a criação de um ambiente de jogo adaptável, no qual os professores possam selecionar manualmente os mini jogos mais adequados para suas turmas. A oferta de uma diversidade e complexidade de tarefas que abrangem vários domínios da informática e habilidades de Pensamento Computacional. Além disso, busca proporcionar uma experiência imersiva, aliada a um sistema de feedback e avaliação que permitirá aos professores monitorar o desempenho dos alunos em relação às competências de PC.

No jogo, o jogador será representado por um castor e deverá atravessar uma floresta, resolvendo desafios inspirados na prova Bebras. Cada desafio superado permitirá ao jogador desbloquear itens para compor um piquenique interativo no lobby, promovendo assim uma sensação de progresso e recompensa, enquanto estimula o desenvolvimento das habilidades cognitivas e de pensamento computacional dos alunos.

1.3 Plataforma

Inicialmente desenvolvido para PCs com sistema operacional Windows, com foco em acessibilidade para dispositivos de baixo desempenho e de futura adaptação para Realidade Virtual (VR).

1.4 Cenário

O jogo se passa em um ambiente de floresta, cercado por rios e envolto em uma atmosfera fantástica. A jornada tem início na entrada da floresta, onde o jogador, representado por um

castor, tem como objetivo coletar diversos itens para preparar um piquenique especial com sua família. Conforme o jogador avança, ele atravessa diferentes áreas da floresta, cada uma com sua própria temática e desafios únicos, mas sempre mantendo a essência do ambiente florestal.

1.5 Mecânica

O jogo será composto por uma série de ações padronizadas, que servirão como base para a resolução dos diferentes minijogos. Cada mini jogo poderá combinar e utilizar essas ações conforme necessário para sua conclusão. Por isso, todas as ações disponíveis deverão ser apresentadas e explicadas ao jogador logo no início da experiência. As ações que poderão ser utilizadas em cada mini jogo são:

- Rotacionar a câmera ao redor da posição
- Pegar objetos e arrastá-los até certa posição
- Abrir interfaces
- Clicar em objetos interativos
- ...

1.6 Fluxo do jogo

Todas as aplicações disponibilizadas pelo educador começarão com uma interface, contendo algumas configurações para acessibilidade, e a possibilidade de selecionar os mini jogos que serão incluídos naquela avaliação.

- Futuramente, será incorporada uma ferramenta que permitirá ao educador personalizar o fluxo de progressão das fases, definindo a ordem e o modo como os desafios serão apresentados aos alunos.

Ao iniciar o jogo após a seleção das fases, o jogador será transportado para um cenário de floresta, onde encontrará um ambiente de piquenique que servirá como lobby central. Esse espaço terá uma função introdutória, apresentando as principais mecânicas do jogo e familiarizando o jogador com o sistema. No lobby, haverá diversos espaços reservados para itens, representando os mini jogos selecionados anteriormente. Cada item terá uma posição fixa no cenário e, caso ainda não tenha sido obtido, será exibido com uma textura semi transparente, indicando sua ausência.

Ao passar o cursor sobre qualquer item — coletado ou não —, o jogador receberá informações adicionais, como:

- O nome do minijogo associado ao item.
- O que exatamente é o item

Apenas os itens correspondentes aos mini jogos escolhidos no seletor de fases serão carregados e exibidos no lobby. Dessa forma, o ambiente será personalizado de acordo com as escolhas no início da aplicação da prova.

Após compreender os controles e se familiarizar com o ambiente, o jogador perceberá que, ao apontar o cursor do mouse para uma determinada área, poderá acessar a seleção de fases.

Essa seleção ocorrerá por meio de um mapa interativo, permitindo que o jogador navegue por ele e escolha qual desafio deseja enfrentar. O mapa será intuitivo e integrará visualmente o ambiente da floresta.

- Em versões futuras, caso o jogo adote uma progressão multilinear, o mapa evoluirá para o formato de uma árvore ramificada. Permitindo que o jogador escolha o caminho possível que desejar.

Ao selecionar o mini jogo desejado, o jogador será transportado diretamente para a respectiva cena. Cada minijogo contará com uma mecânica e ambientação própria, conforme os padrões definidos na Seção 1.6.

Durante todo o processo de realização da atividade. Ou seja, a partir do momento que o jogador comece os mini jogos, um contador daria início, sendo assim o jogador teria um tempo limite para terminar a avaliação, O tempo será gerado a partir dos mini jogos escolhidos para compor a avaliação, onde cada mini jogo tem um tempo limite definido, e o tempo total da prova seria o tempo somado de cada um.

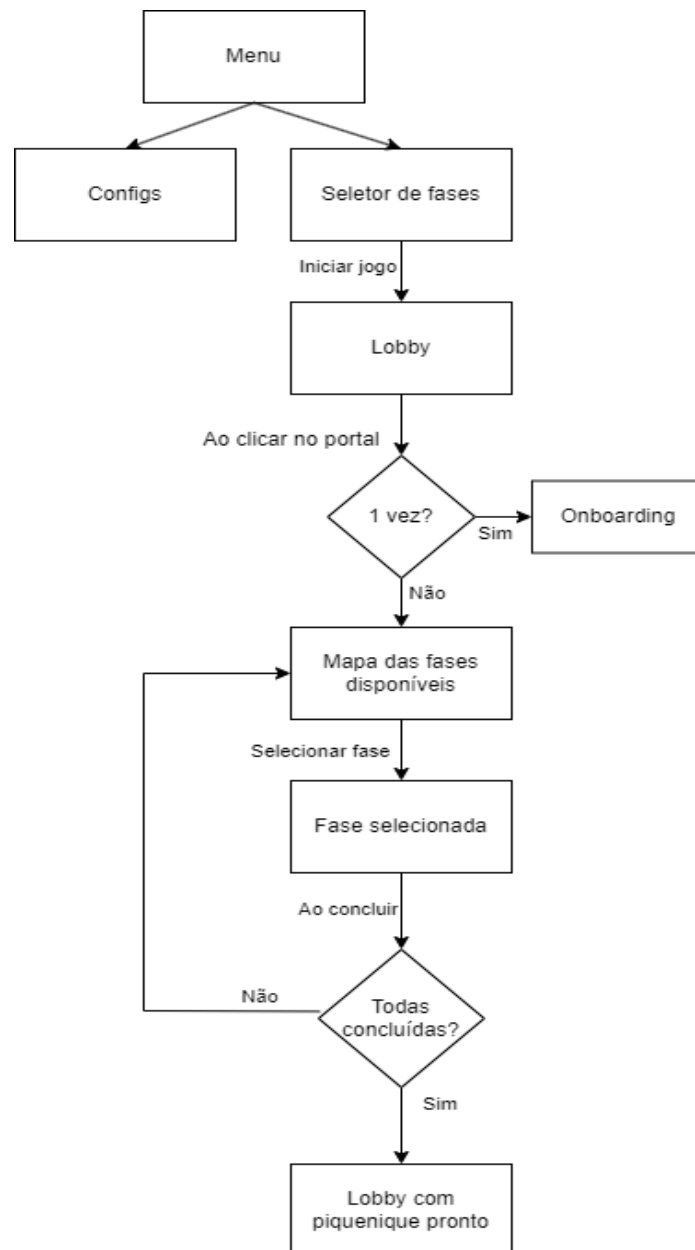
Após concluir o desafio proposto, o jogador retornará ao mapa interativo, podendo então escolher uma nova fase para prosseguir. No entanto, o mini jogo já resolvido será marcado como **bloqueado**, impedindo que ele seja acessado novamente, forçando o jogador a avançar por novos desafios.

- Caso o sistema de progressão multilinear esteja implementado, a conclusão de determinados minijogos também desbloqueará caminhos alternativos e novos desafios no mapa.
- A partir do momento em que o jogador iniciar o seu primeiro mini jogo, o acesso ao lobby do piquenique será temporariamente bloqueado. O jogador deverá seguir avançando pelos desafios. Apenas após concluir todos que poderá retornar.

Ao retornar ao lobby, o jogador poderá visualizar os itens vinculados aos desafios. Os itens referentes aos mini jogos concluídos com sucesso serão exibidos de forma completa e no cenário, enquanto aqueles cujos desafios não foram resolvidos corretamente permanecerão com a textura semi transparente, indicando que o desafio não foi concluído de forma correta.

Além disso, o jogador terá acesso a um painel de informações sobre seu desempenho na prova, incluindo:

- Tempo de resolução de cada minijogo.
- Arquivo de dados das respostas corretas e incorretas em cada desafio.



2. Game Design

2.1 Narrativa e Estética

O jogo se passa em uma floresta inspiradas em fábulas onde deve usar lógica de pensamento computacional para ajudar outros personagens em suas tarefas ou desbloquear quebra cabeças. Cada desafio superado recompensa o jogador com um item especial, que será utilizado para montar um piquenique em família no final da jornada. Ao concluir todas as tarefas e coletar os itens, o jogador poderá celebrar o piquenique com os personagens e sua família, usando os itens conquistados ao longo do percurso como forma simbólica de sua evolução e aprendizado.

2.2 Padronização dos Mini Jogos

Ao iniciar o desafio, um marcador visual indicará um Non Playable Character (NPC), indicando que ele deve interagir com o personagem. Ao clicar no NPC, este apresentará a situação do problema que precisa ser resolvida e apresentará o ambiente assim como os objetos necessários para sua resolução. Para completar o desafio, o jogador deverá utilizar os recursos e objetos disponíveis, aplicando raciocínio e experimentação para encontrar a solução adequada. Após resolver o problema, o jogador poderá interagir novamente com o NPC. Caso a solução seja uma possível solução (**não necessariamente correta**), a fase será concluída com sucesso.

Ao finalizar o desafio, o mapa de seleção de fases será exibido, permitindo que o jogador escolha o próximo mini jogo e dê continuidade à sua jornada.

Todos os mini jogos terão algumas categorias a serem consideradas, essas que serão mostradas tanto antes de iniciar a prova, ao selecionar os mini jogos presente no percurso, quanto antes de escolher as fases da floresta. As categorias serão divididas em: Conhecimento exigido em pensamento computacional, Dificuldade, Tempo médio de resolução, Nome, Exercício referente a prova Bebras

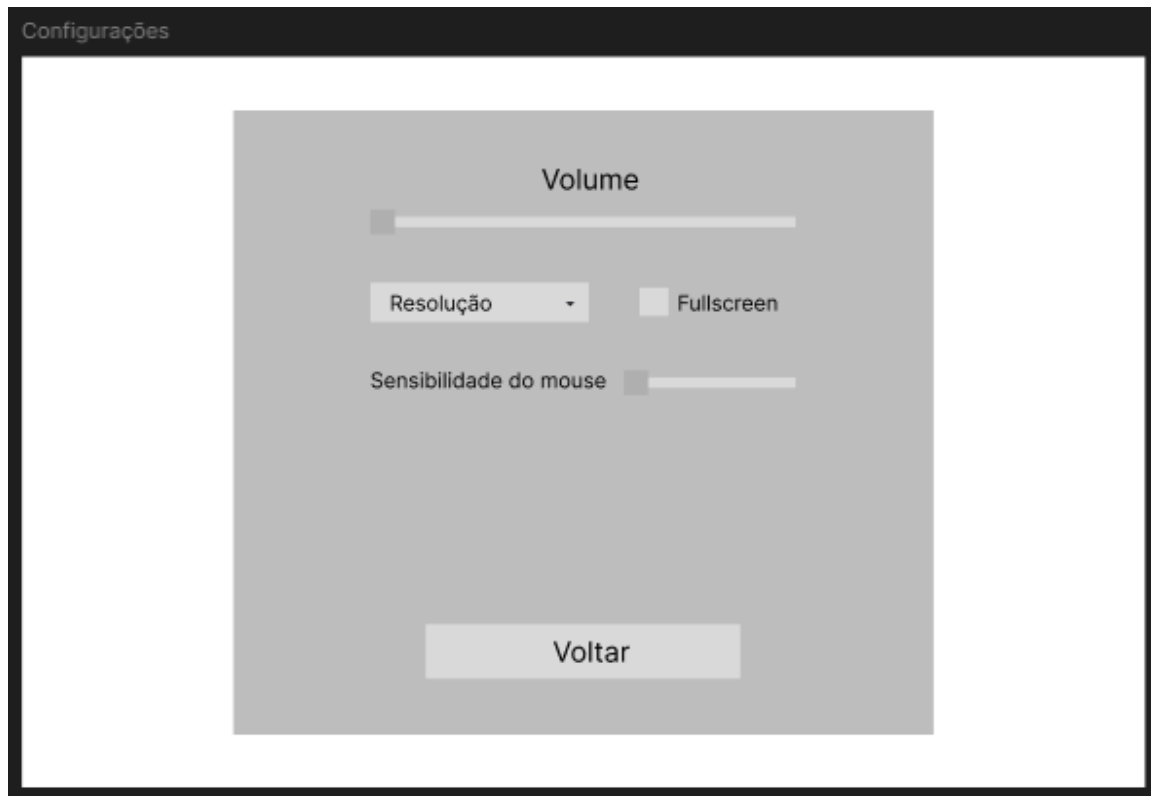
3. Produção

3.1 Interfaces

O jogo iniciaria com um tela contendo três botões, **iniciar**, que direciona o jogador a uma nova tela para selecionar os mini jogos que estarão contidos na avaliação, **configurações**, que direciona-o para uma nova tela contendo alguns ajustes de acessibilidade e vídeo , e **sair**, que fechará o executável, considerando a possibilidade do jogo ser executado fora do Web

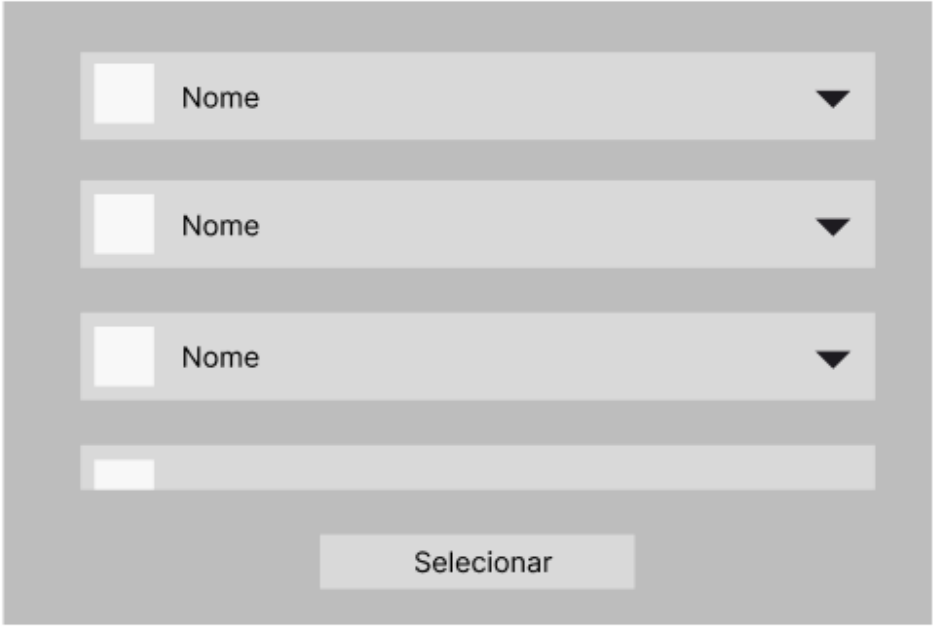


A tela de configurações conteria ajustes como: volume, resolução da tela, tela cheia, configuração da sensibilidade do mouse. (Pode ser adicionado mais ajustes).



Após a inicialização da prova, uma nova tela será exibida, permitindo que o jogador selecione os mini jogos que farão parte da avaliação. Essa abordagem será adotada na primeira versão, com a possibilidade de futuras melhorias para simplificar o processo. O objetivo é que o usuário que realizará a prova possa inserir um código ou ter acesso direto aos minijogos previamente definidos, sem a necessidade de selecioná-los manualmente.

Seleção de fases



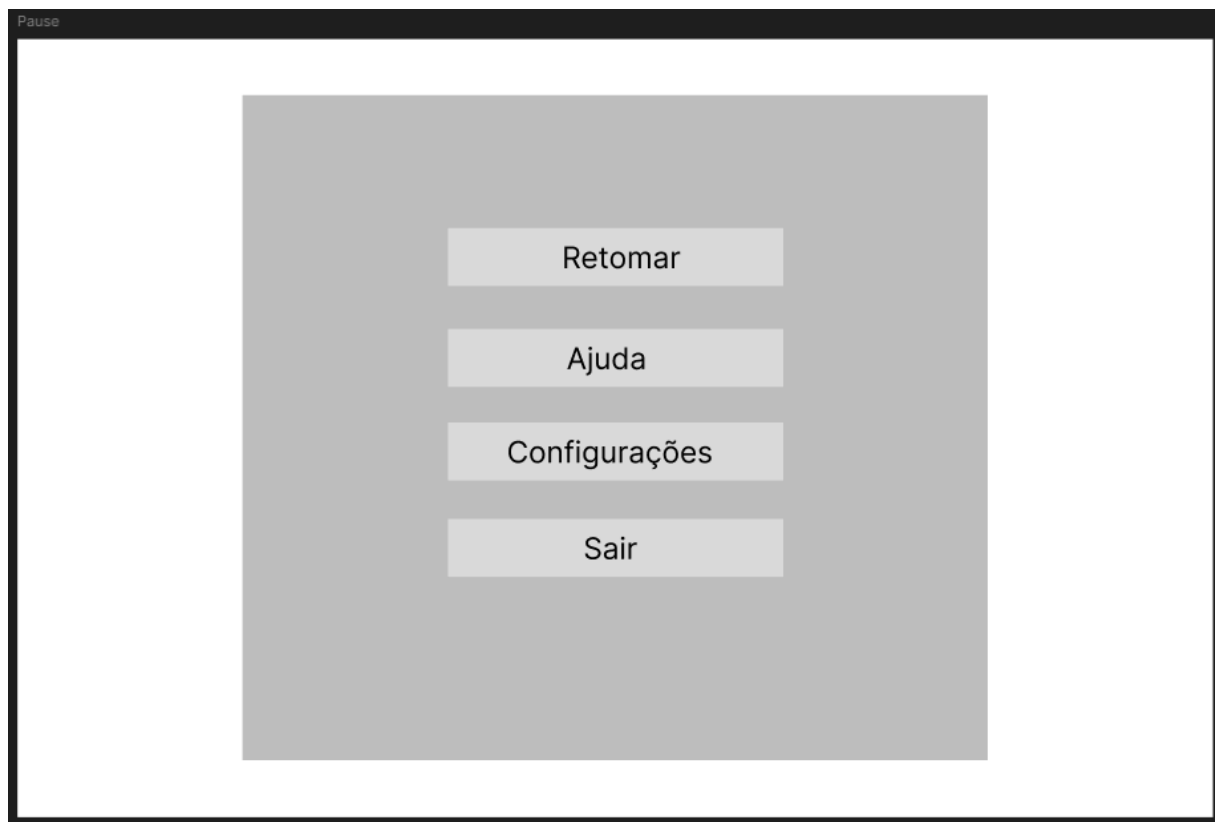
Nome

Nome

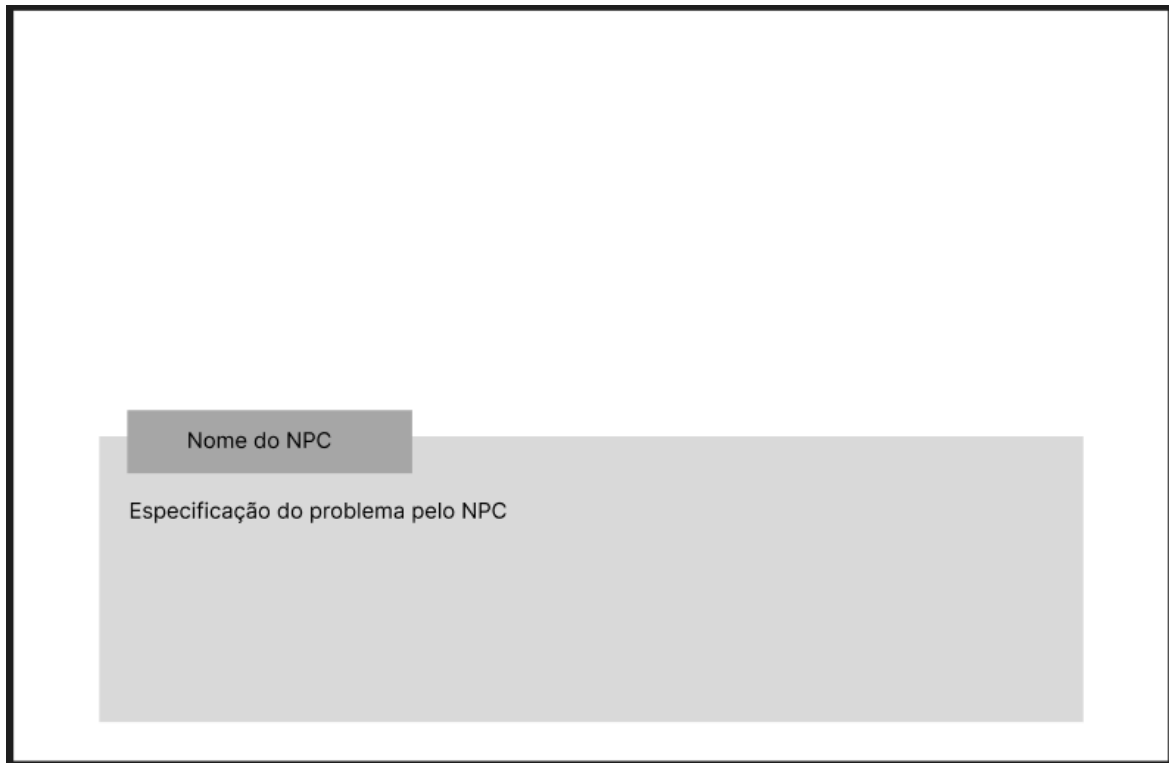
Nome

Selecionar

Durante todo o processo da prova o jogador poderá por uma tecla pausar o jogo, O contador seria interrompido e Além disso seria disponibilizado uma interface de ajuda, contendo controles e as ações disponíveis dentro daquele mini jogo.



A caixa de diálogo seria apresentada ao jogador assim que o mesmo iniciar algum mini jogo, já apresentando o problema e o que ele deve fazer para resolvê-lo. Após terminar o diálogo, o jogador pode interagir com o NPC que será aberto o diálogo novamente.



3.2 Ferramentas

Para o desenvolvimento do projeto, será utilizada a **Unity**, uma engine gratuita com ótimo suporte ao desenvolvimento de jogos 3D, sendo a principal plataforma para a implementação do jogo. A linguagem de programação adotada será **C#**, por sua integração nativa com a Unity. Para a criação e edição dos elementos gráficos, como HUD, botões, ícones e interfaces, serão usados softwares como **Photoshop e Krita** ou ferramentas alternativas como **Canva**, conforme a necessidade e disponibilidade.

ANEXO B – Game Design Document (GDD) do minijogo da Costura

Minijogo da Costura

Game Design Document

Versão: 1.0

Autores:

Felipe Canabarro Graffunder

Guilherme Duarte Toffoli

Uberlândia,
Fevereiro de 2025

Educational Objectives of the Game

Habilidade Computacional: Pensamento algorítmico e avaliação
Palavras chaves: Dados, estruturas de dados e máquina de estados finita

Target audience and Educational Level

Faixa etária: 12-13 anos
Nível acadêmico: fundamental

O minijogo faz parte de um exercício da categoria Cadetes do Bebras Portugal. Fará parte de uma das opções disponíveis para o professor que queira aplicar um teste a estudantes do 7º ao 8º ano.

Game Mechanics and Dynamics

No jogo da costura, o jogo se baseia em uma máquina de estados finitos (MEF), a qual tem-se que analisar e compreender quais sequências de linhas podem ser feitas ou quais não podem, ou seja, quais sequências o autômato reconhece ou não.

Ao adentrar a casa, o jogador acha uma máquina de costura que, ao interagir com ela, encontra-se com a interface do minijogo. Nessa tela, o jogador terá de interpretar a máquina de estados desenhada, apertando os botões certos para assim mudar de estados e processar as linhas de costuras corretas, a fim de processar todas as costuras.

Encontrando a costura errada, o jogador sai da interface de costura e conversa com um personagem fora da casa para dar seu palpite final. Se correto, poderá seguir.

Story and Narrative

Recompensa: Pano de piquenique

O jogador está em busca de completar sua jornada com um pano de piquenique. Entrando na fase da Costura, o jogador interage com um personagem em frente a uma casa. O personagem diz que possui 4 tipos de panos que precisam ser consertados, mas que a máquina de costura é estranha e só veio com um tutorial de uso. Juntando o útil ao agradável, o personagem oferece um pano para o jogador se ele conseguir consertar os 4 panos.

Characters and Environments

Teremos como personagem principal desse minijogo e o único, a Ovelha, ela terá o papel de explicar ao jogador o problema do exercício e o que deve se fazer para resolvê-lo.

Ao finalizar a fase, a Ovelha localizada em frente a casa estará esperando o jogador para receber o palpite final para continuar a jornada do Castor.

O ambiente do jogo é dado em um ambiente 3D, representando uma floresta com muitas árvores, o local é cercado por montanhas ao fundo do cenário e água ao redor da ilha abrindo espaço para apenas dois caminhos: a entrada para o minijogo e a saída do minijogo. Na fase de Costura não é diferente, uma floresta densa ao redor e uma casa de madeira, a qual contém a interface para o minijogo.

Game Interface

Dentro da casa de madeira, o jogador interage com a máquina de costura, o jogador visualiza a regra de funcionamento da máquina finita e como criar as linhas, juntamente com as questões (linhas).

Challenges and Puzzle

O desafio do minijogo se encontra no entendimento de autômatos finitos e a validação de mensagens, que podem ou não podem ser reconhecidas.

O objetivo da questão é compreender o padrão de reconhecimento da estrutura e achar qual sequência não é possível fazer pelo autômato.

Para resolver o exercício, tem-se de analisar cada linha passo-a-passo (ponto de costura a ponto de costura).



Com o exemplo dado, o jogador terá de perceber que não é possível fazer essa costura verificando as regras de funcionamento do autômato.

Teacher Support Resources

O minijogo fará parte de um conjunto de vários outros onde o Professor que estiver utilizando da ferramenta para fazer o seu teste pode utilizá-lo em seu teste.

Evaluation and Feedback

(Vazio)

ANEXO C – Documentação dos casos de uso

Nome	Iniciar o Minijogo
Ator Principal	Jogador
Resumo	Permite que o jogador inicie o minijogo da costura e visualize a interface do autômato e controles
Pré-condições	Conversar com a ovelha
Pós-condições	Minijogo carregado, estado inicial do autômato e UI exibidos
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Clica na máquina de costura	
	2. Carrega interfaces
	3. Inicia variáveis
Fluxo Alternativo	
1. Clica na máquina de costura	
	2. Carrega mensagem para conversar com a ovelha antes de iniciar

Nome	Clicar nos estados do autômato
Ator Principal	Jogador
Resumo	Permite que o jogador interaja com o autômato selecionando estados/transições para formar uma costura
Pré-condições	Minijogo iniciado
Pós-condições	Um novo ponto/transição é registrado na sequência, se a ação for válida
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Clica em um estado do autômato (Se for permitido)	
	2. Atualiza o estado atual do autômato na interface
	3. Constrói uma linha de costura de acordo com o estado

Nome	Construir uma sequência de costura
Ator Principal	Jogador
Resumo	Registra e exibe a sequência de pontos construída pelo jogador a partir das interações com o autômato
Pré-condições	Jogador realiza uma interação válida com o autômato
Pós-condições	Sequência atual atualizada e exibida na interface; histórico pronto para desfazer
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Clica em um estado do autômato	
	2. Adiciona uma nova linha e a registra
	3. Repita o passo 1
Fluxo Alternativo	
4. Clica no autômato para finalizar a sequência	
	5. Valida se a sequência construída é válida ou não
	6. Exibe o <i>feedback</i> se a sequência foi válida ou não

Nome	Desfazer o último ponto
Ator Principal	Jogador
Resumo	Permite remover o último ponto adicionado e retornar o autômato ao estado anterior
Pré-condições	Existir pelo menos 1 ponto de costura na sequência
Pós-condições	Último ponto removido; sequência e estado atual atualizados
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Aperta o botão de remover linha	
	2. Remove o último ponto da sequência
	3. Atualiza a interface

Nome	Confirmar resposta final
Ator Principal	Jogador
Resumo	Permite ao jogador indicar qual alternativa/sequência é incorreta e receber o resultado (acerto/erro)
Pré-condições	Ter acertado pelo menos uma sequência
Pós-condições	Confirma a resposta e atualiza o progresso se sucesso (acerto) ou falha (utilizou as três tentativas)
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Seleciona uma das quatro alternativas	
	2. Valida a alternativa
	3. Atualiza o número de tentativas realizadas
Fluxo de Resposta Correta	
	4. Exibe a tela final de sucesso com as métricas
Fluxo de Resposta Errada com tentativas restantes	
	5. Exibe mensagem da ovelha dizendo que a resposta está errada
	6. Retira opção tentada e permite outra chance
Fluxo de Resposta Errada sem tentativas restantes	
	7. Exibe mensagem dizendo qual era a correta e que acabou as tentativas
	8. Exibe a tela final de falha com as métricas

Nome	Sair do minijogo
Ator Principal	Jogador
Resumo	Permite encerrar o minijogo
Pré-condições	Minijogo em execução
Pós-condições	Jogador volta a controlar o personagem
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Aperta o botão de sair	
	2. Fecha a tela do minijogo