

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Gustavo Duarte Gomides

**O problema do baú: jogo sério para o auxílio do
ensino de pensamento computacional**

Uberlândia, Brasil

2025

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Gustavo Duarte Gomides

**O problema do baú: jogo sério para o auxílio do ensino de
pensamento computacional**

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: Renato de Aquino Lopes

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2025

Gustavo Duarte Gomides

O problema do baú: jogo sério para o auxílio do ensino de pensamento computacional

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 12 de setembro de 2025:

Renato de Aquino Lopes
Orientador

Fernanda Maria da Cunha Santos

Roberto Júnio Silva Caetano

Uberlândia, Brasil

2025

Resumo

O Ministério da Educação definiu a computação como uma matéria a ser ensinada nas escolas, dentre as temáticas dessa matéria está o pensamento computacional, cujos conhecimentos não se aplicam somente a ciência da computação. Diante disso, houve a oportunidade de criação de ferramentas para auxiliar os professores nesta tarefa. Neste contexto, o presente trabalho tem a finalidade de criar um protótipo de jogo sério como uma possível ferramenta de aprendizado para crianças, além de apresentar os métodos e recursos usados para confeccioná-lo, para que possa posteriormente implementado em um projeto desenvolvido na Faculdade de Computação com o objetivo de desenvolver um ecossistema de minijogos para auxiliar o ensino de pensamento computacional.

Palavras-chave: Jogos sérios, Bebras, educação, pensamento computacional.

Abstract

The ministry of education defined computing as a subject that should be teached in schools, between the themes of this subject, computacional thinking is one of them, of which concepts doesn't need to be applied only to computer science. Therefore, emerged the oportunity of creating tools to help teachers with this task. In light of this, this work has the purpose of creating the prototype of a serious game as a possible tool for teaching children, also show the methods and resources used to make it, for it to be integrated in a project developed by the Faculdade de Computação with the objective to create a ecosystem of minigames to assist in teaching computational thinking.

Keywords: Serious games, Bebras, education, computational thinking.

Lista de ilustrações

Figura 1 – Tabela de símbolos	13
Figura 2 – Representação dos antigos controles	21
Figura 3 – Representação dos novos controles	21
Figura 4 – Tabela de códigos	21
Figura 5 – Mensagem cifrada	22
Figura 6 – Diagrama de caso de uso	24
Figura 7 – Diagrama de classes.	24
Figura 8 – Diagrama de classes de Interactables.	25
Figura 9 – Diagrama de classes de ObjectManager.	25
Figura 10 – Diagrama de classes de ChestUI.	26
Figura 11 – Diagrama de classes de Dialogue.	27
Figura 12 – Diagrama de classes de Timer.	27
Figura 13 – Diagrama de classes de Player.	28
Figura 14 – Diagrama de classes de OutlineSelection.	29
Figura 15 – Imagem do jogo mostrando a coruja e tabela de simbolos	29
Figura 16 – Imagem do jogo mostrando o baú	30
Figura 17 – Imagem do jogo mostrando o dialogo	30
Figura 18 – Imagem do jogo mostrando a interface do baú	30
Figura 19 – Imagem do jogo mostrando os resultados	31
Figura 20 – Diagrama de classe	36

Lista de tabelas

Tabela 1 – Comparaço entre os trabalhos 19

Lista de abreviaturas e siglas

GDD	Game Design Document
MEC	Ministério da Educação
NPC	Non-Player Character
POO	Programação Orientada a Objeto
TCC	Trabalho de Conclusão de Curso
UI	User Interface
UML	Unified Modeling Language

Sumário

1	INTRODUÇÃO	10
1.1	Justificativa	10
1.2	Objetivo Geral	11
1.3	Objetivos específicos	11
2	FUNDAMENTAÇÃO TEÓRICA	12
2.1	Pensamento computacional	12
2.2	Gamificação	12
2.3	Criptografia	12
2.4	Bebras	13
2.5	Unity	14
2.6	Programação orientada a objeto	14
2.7	Unified Modeling Language	15
2.8	Diagrama de Classes	15
2.9	Diagrama de caso de uso	15
2.10	Game Design Document	16
2.11	Considerações finais	16
3	TRABALHOS RELACIONADOS	17
4	DESENVOLVIMENTO	20
4.1	Ferramentas Usadas	20
4.2	Base do jogo	20
4.3	Questão do Bebras	21
4.4	Tema Abordado	22
4.5	Game Design Document	22
4.6	Requisitos funcionais	23
4.7	Estruturação das classes	24
4.8	Gameplay	29
5	CONCLUSÃO	32
	REFERÊNCIAS	33

APÊNDICES

35

APÊNDICE A – DIAGRAMA DE CLASSES COMPLETO 36

1 Introdução

No Brasil, a Base Comum Curricular (BNCC), documento que define o conjunto de aprendizagens essenciais que todos os alunos devem desenvolver ao longo das etapas e modalidades da educação básica, aponta a importância da integração do pensamento computacional na educação básica, como descrito em ([Ministério da Educação, 2022](#)). Isso reforça a necessidade de ferramentas para preparar e auxiliar os educadores para que eles estejam preparados para ensinar esse tema.

Pensamento computacional é um conjunto de técnicas de resolução de problemas que envolve expressar e solucionar um problema de forma que um computador seja capaz de executar. Ele ajuda com a resolução de problemas complexos por meio do uso de conceitos da ciência da computação e exercita o uso de pensamento crítico e lógico.

Observa-se que a aplicação deste conjunto de habilidades não precisa se limitar a área da ciência da computação, mas podem ser usados em diversas situações, como por exemplo na educação. Desta forma este trabalho propõem a gamificação de uma questão da prova Bebras como forma de auxiliar o ensino do pensamento computacional nas escolas.

1.1 Justificativa

A importância do tema vem, inicialmente, do fato que, no Brasil, o parecer CNE/CEB de Nº2/2022 aprovou a inserção do ensino de Computação na Educação Básica, estabelecendo as normas e o complemento à Base Nacional Comum Curricular ([Ministério da Educação, 2022](#)). Além disso, as competências e habilidades específicas foram definidas em três eixos principais: Pensamento Computacional, Cultura Digital e Mundo Digital, incidindo sobre a educação infantil, o ensino fundamental e o ensino médio, como apontado em ([Ministério da Educação, 2022](#)).

Também é importante ressaltar que pensamento computacional é um conjunto de técnicas que ajudam a formulação e resolução de problemas e podem ser usadas em outros contextos além da ciência da computação como leitura, escrita e aritmética, ([WING, 2006](#)). Isso ressalta a necessidade de incentivar o uso dessa habilidade.

Ademais, o uso de jogos no auxílio do ensino é relatado em alguns artigos como por exemplo em ([AGBO et al., 2023](#)) onde é indicado os efeitos positivos desse meio no pensamento crítico e compreensão dos indivíduos. Assim espera-se que o jogo produzido neste trabalho possa auxiliar os professores durante o ensino de pensamento computacional.

1.2 Objetivo Geral

O objetivo principal deste trabalho é o desenvolvimento do protótipo de um jogo sério para auxiliar o ensino do pensamento computacional por meio da gamificação de uma questão da prova Bebras de Portugal da edição de 2019.

1.3 Objetivos específicos

Os objetivos específicos desse trabalho são:

- Analisar e escolher uma questão Bebras;
- Desenvolver o jogo seguindo os parâmetros do projeto usando a engine de jogos Unity;
- Fazer um game design document

2 Fundamentação Teórica

Esse capítulo apresenta uma explicação das ferramentas e conceitos usados durante a produção deste trabalho.

2.1 Pensamento computacional

Pensamento computacional é o uso de conceitos fundamentais da ciência da computação, como o pensamento recursivo, abstração e decomposição, para resolver problemas, projetar sistemas e compreender o comportamento humano de acordo com ([WING, 2006](#)).

De acordo com ([HSU; CHANG; HUNG, 2018](#)) houve um crescimento no número de artigos criados usando o pensamento computacional, o número de artigos publicados foi de 21 para 44 entre os anos de 2016 e 2017. Dado o aumento de pesquisas envolvidas e a importância do assunto no cenário nacional, o MEC decidiu aplicar o ensino de computação nas escolas, onde um dos eixos é o pensamento computacional.

2.2 Gamificação

O processo de gamificação usa elementos presentes em jogos, como a narrativa, sistema de feedback, sistema de recompensa, competição, objetivo e regras claras, interatividade, entre outros, com um dos objetivos sendo obter o mesmo nível de motivação e envolvimento encontrados em jogadores quando interagindo com jogos bons ([FARDO, 2013](#)), isso é um indicativo que esse processo pode auxiliar o ensino de pensamento computacional.

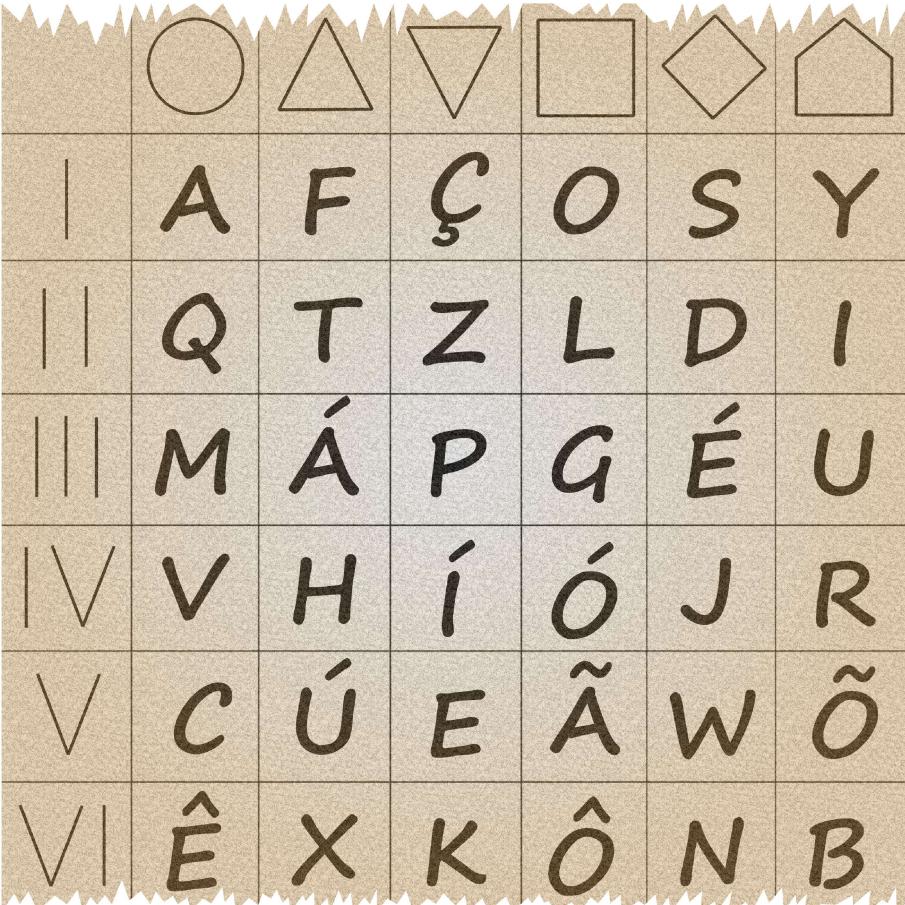
Esse tipo de técnica já foi usada para conduzir aulas de game design descrito no livro Multiplayer Classroom: Designing Coursework as a Game, do professor norte-americano Lee Sheldon (2012). Ao decorrer do livro o professor apontou que essa experiência contribui para o aumento do interesse, participação e motivação, além do aumento da interatividade entre os alunos ([FARDO, 2013](#)), demonstrando os efeitos benéficos que a gamificação pode trazer para o processo de ensino.

2.3 Criptografia

Com a chegada da internet a disseminação de informações foi facilitada, junto disso veio a necessidade de impedir que indivíduos mal-intencionados leiam e alterem as mensagens transmitidas, por conta disso surgiu a criptografia. De acordo com ([TERADA,](#)

2008) os algoritmos criptográficos objetivam "esconder" informações sigilosas de qualquer pessoa desautorizada a lê-las. Em virtude do objetivo da criptografia, a questão do Bebras usada para a criação do jogo aborda o conceito.

Neste trabalho foi usado uma cifra de substituição para a criação do jogo, nesse método um símbolo é substituído por outro símbolo usando um determinado método, no caso desta monografia a combinação de dois símbolos resulta em uma letra do alfabeto, incluindo algumas letras com acento e o "ç", como visto na Figura 1. Esses conhecimentos foram usados na matéria de segurança da dados.



A tabela de símbolos Bebras é construída em uma grade de 6 linhas por 6 colunas. As linhas são rotuladas com números romanos (I a VI) e as colunas com traços verticais. No topo da grade, seis símbolos geométricos são exibidos: círculo, triângulo, triângulo invertido, quadrado, losango e pentágono. Cada célula da grade contém uma letra maiúscula, formando uma matriz de 36 caracteres. A grade é bordada com um padrão de serrilhos.

	Círculo	Triângulo	Triângulo Invertido	Quadrado	Losango	Pentágono
I	A	F	Ç	O	S	Y
II	Q	T	Z	L	D	I
III	M	Á	P	G	É	U
IV	V	H	Í	Ó	J	R
V \	C	Ú	E	Ã	W	Ö
VI \	Ê	X	K	Ô	N	B

Figura 1 – Tabela de símbolos

2.4 Bebras

Bebras é uma iniciativa internacional que surgiu na Lituânia com o intuito de ensinar informática e pensamento computacional, (BEBRAS, 2025). A segunda semana de novembro é considerada a semana do Bebras para a solução de tarefas, porém muitos países realizam atividades relacionadas ao longo do ano.

Um desafio Bebras é uma competição que é composta por um conjunto de 12 *Tasks* (questões) que trabalham com decomposição, abstração, reconhecimento de padrões, pen-

samento algorítmico e avaliação. Esses conceitos representam a quebra de problemas maiores e difíceis de lidar em questões menores e mais fáceis de lidar, o foco apenas nas informações importantes, a busca por similaridades e características em comum dentro de um problema, o desenvolvimento da solução passo a passo e a checagem se a solução obtida é correta.

A utilização do Bebras neste trabalho contribui com um arcabouço de questões elaboradas e fundamentadas em conceitos pedagógicos relacionados ao pensamento computacional, auxiliando com toda a metodologia para a elaboração de questões.

2.5 Unity

Unity é uma game engine para o desenvolvimento de jogos 2D e 3D para múltiplas plataformas, como computadores, celulares , consoles, realidade aumentada e realidade virtual, lançada em junho de 2005 pela Unity Technologies, ([UNITY, 2025](#)). Essa ferramenta foi usada em jogos como Cuphead, Hollow Knight, The Forest e Subnautica.

Uma game engine é um framework com a função de simplificar e abstrair o desenvolvimento de jogos eletrônicos. Essa ferramenta é composta por um motor gráfico para a renderização de gráficos 2D e/ou 3D, motor de física para simular a física, suporte para uma linguagem script, suporte de animações e sons.

A Unity foi escolhida para o desenvolvimento deste trabalho pelas suas ferramentas e por ter sido a engine escolhida no desenvolvimento projeto na Faculdade de Computação com o objetivo de desenvolver um ecossistema de minijogos para auxiliar o ensino de pensamento computacional.

2.6 Programação orientada a objeto

Segundo ([FARINELLI, 2007](#)), programação orientada a objeto (POO) é um paradigma de programação no qual o código é organizado em objetos, estruturas que combinam atributos e métodos, e que interagem entre si. Para definir as características e comportamentos de um conjunto de objeto são usados o conceito de classe. Além de objetos e classes, POO também apresenta conceitos como o encapsulamento, herança, interface e polimorfismo.

O encapsulamento métodos e atributos impedem o acesso de seus valores por outra classe. Na herança, métodos e atributos podem ser herdados por outras classes, mas ela não é limitada a esses componentes.

Interfaces são "contratos": uma classe que implementa uma interface deve ter uma implementação de todos os métodos da interface, com isso a classe é considerada do mesmo

tipo da interface. Polimorfismo significa que múltiplas classes com o mesmo método podem ter implementações diferentes desse mesmo método. Os conceitos aplicados durante o trabalho foram abordados na matéria de Programação orientada a objeto 1 e 2.

2.7 Unified Modeling Language

A *Unified Modeling Language* (UML), segundo ([GUEDES, 2018](#)), é uma linguagem visual de utilizada para modelar softwares usando o paradigma de orientação a objeto, sendo o padrão adotado nas internacionalmente. O objetivo da UML é auxiliar os engenheiros de software a definirem as características do sistema, como seus requisitos, seu comportamento, sua estrutura lógica, a dinâmica de seus processos e até mesmo suas necessidades físicas.

Além do diagrama de caso e o de classes, que foram usados nessa monografia, a UML é composta pelo diagrama de objetos, de pacotes, de sequência, de comunicação, de máquina de estados, de atividade, de visão geral de interação, de componentes, de implantação, de estrutura composta, de tempo e de perfil. Os conhecimentos sobre esse sistema foram obtidos durante as aulas de modelagem de software.

2.8 Diagrama de Classes

De acordo com ([GUEDES, 2018](#)), um diagrama de classe é usado para modelar as classes de um sistema, exibir o relacionamento entre elas e descrever o funcionamento e serviços oferecidos por eles. Diante disso, esse diagrama demonstra o funcionamento de um sistema, permitindo o planejamento da implementação de seus componentes.

Alguns dos elementos de um diagrama de classe são as classes, representações de um objeto ou conjunto de objetos que compartilham uma estrutura e um comportamento comum, e relacionamentos, indicando a conexão entre os elementos do modelo, demonstrando o comportamento entre eles. Todos esses conhecimentos foram abordados na matéria modelagem de software.

2.9 Diagrama de caso de uso

Um diagrama de caso de uso tem como objetivo apresentar uma visão geral das funcionalidades que o sistema deverá oferecer aos usuários, ([GUEDES, 2018](#)). Este diagrama ajuda na representação cenários que o sistema interage com as pessoas, metas e escopo do sistema. Os elementos que compõem esse diagrama são os atores, sistemas e metas.

Os atores são os usuários que interagem com o sistema, podendo ser uma pessoa, organização ou sistema externo. O sistema é uma sequência específica de ações e interações entre os atores e o sistema. Já as metas é o resultado final do diagrama, caso ele seja feito corretamente descreve as atividades e variantes usadas. O diagrama foi estudado nas aulas de modelagem de software.

2.10 Game Design Document

Um *Game Design Document* (GDD) é uma ferramenta textual que descreve e descreve todas as características de um jogo, desde conceitos simples como a premissa ou elementos mais detalhados como projeto de níveis, (MOTTA; JUNIOR, 2013). Isso permite que todas as funcionalidades e necessidades do jogo sejam documentadas, permitindo um guia para a criação do jogo, facilitando o processo de criação e permitindo que todos os envolvidos estejam ligados ao mesmo objetivo.

Neste projeto o GDD é composto pelos capítulos: objetivos educacionais, público-alvo e nível de escolaridade; narrativa; personagens e ambientação; mecânicas do jogo e jogabilidade; e questão da prova do Bebras 2019. Com esses tópicos é possível descrever todas as funcionalidades, o que está sendo ensinado, a temática e os objetivos do jogo.

2.11 Considerações finais

Para o desenvolvimento de um jogo inicialmente é necessário a documentação e formulação de suas funcionalidades e requisitos, por isso se usa um GDD, servido para definir elementos do jogo como as mecânicas usadas ou as ferramentas usadas garantindo que todos os envolvidos sigam o mesmo caminho. Além disso, com o uso dos diagramas da UML é possível indicar as funcionalidades de um software, além que ele seja estruturado de maneira ordenada.

Após a definição desses elementos é necessário programar usando as ferramentas anteriormente descritas, como a Unity usando a programação orientada a objeto. Caso seja usado o processo de gamificação precisa-se levar em conta os elementos que compõem o conteúdo que se deseja gamificar para que seja feita uma boa adaptação.

3 Trabalhos Relacionados

Neste capítulo será apresentado outros trabalhos que apresentam uma temática similar a este projeto e apresentam elementos que contribuem para a relevância deste projeto.

(HSU; CHANG; HUNG, 2018) apresenta o estado da arte sobre o ensino e a aprendizagem sobre pensamento computacional. Neste artigo foi apresentando que com o passar dos anos os números de artigos relacionados a pensamento computacional aumentou consideravelmente, em 2017 o número de publicações com esse tema foi maior que o dobro do número do ano passado, indicando que a relevância desse tema aumentou e um maior estudo nessa área foi necessário.

Ademais, foi apontado durante o artigo descrito no parágrafo anterior que alguns possíveis tipos de ensino de pensamento computacional e a quantidade de artigos que apontaram esses métodos. Dentre eles a aprendizagem baseada em jogos ficou em 4º lugar com 12 artigos feitos sobre referido assunto, ficando abaixo apenas de aprendizagem baseada em problemas, aprendizagem baseada em projetos e aprendizagem colaborativa com 22, 22 e 16 artigos feitos respectivamente.

Diante disso, é possível notar que os artigos apresentaram maior interesse na aprendizagem baseada em jogos, apresentando a relevância deste projeto e a necessidade de experimentar sobre essa área, mesmo que os jogos sejam apenas um elemento auxiliar para os professores.

Ademais o artigo (CAPECCHI et al., 2022) demonstra a criação de uma aplicação com o intuito de ensinar arquitetura e os efeitos dela nos estudantes. Essa aplicação é um jogo em realidade virtual e usa design de Leonardo Da Vinci com o intuito de permitir a aprendizagem. Para avaliar a eficiência desse jogo o artigo realizou um teste com um conjunto de 35 estudantes entre 14 e 16 anos usando quatro escalas, telepresença, affordance, ergonomia cognitiva e perceptiva e autoeficácia geral.

Esses elementos representam a sensação de estar no ambiente virtual, o nível de imersão, adequação do ambiente virtual às capacidades cognitivas e perceptivas do usuário e a percepção que o indivíduo tem na capacidade de enfrentar novos desafios. Além disso, usando essas métricas foi observado que esse jogo sério permitiu a imersão dos indivíduos e mostrou-se intuitivo, comprehensível, não causou desconforto aos usuários e os estudantes acharam que a sua autoeficácia foi afetada positivamente.

De acordo com o que foi observado, a utilização de jogos sérios no ensino é eficaz, incentivando o uso de jogos sérios para o auxílio de pensamento computacional, mesmo

que esse artigo não aborde a temática.

O artigo ([AGBO et al., 2023](#)), demonstra um experimento realizado com 49 alunos de ciência da computação na Nigéria usando três minijogos do aplicativo de realidade virtual iThinkSmart a fim de nutrir o pensamento computacional. Com base nisso foram avaliados 8 hipóteses dentro do artigo, sendo elas:

- O desafio do jogo afeta positivamente o feedback;
- A clareza do objetivo influencia de maneira positiva o feedback;
- A clareza influencia positivamente o pensamento reflexivo;
- O desafio do jogo afeta positivamente no pensamento reflexivo;
- A imersão favorece a compreensão;
- A interação favorece a compreensão;
- Pensamento reflexivo prediz a compreensão;
- Adequação dos conteúdos de aprendizagem modera positivamente o efeito do pensamento reflexivo sobre a compreensão;

Dentre essas hipóteses, apenas a quarta não foi confirmada, indicando que o nível de desafio do jogo e a clareza afetam a experiência do usuário, apenas a clareza favorece o pensamento reflexivo, a imersão e o nível de interação do usuário afeta a compreensão, o pensamento reflexivo é um indicativo da compreensão do jogo e a qualidade de adaptação dos conteúdos controla o efeito do pensamento reflexivo.

Todos esses fatores demonstram os efeitos positivos que um jogo sério tem sobre a aprendizagem, além de demonstrar fatores importantes para a criação de um. Dessa forma detalhes que se deve levar em conta, como a clareza e o nível de desafio do jogo, e demonstrando a sua eficiência, mesmo que este não use a realidade virtual.

O projeto ([CARDOSO et al., 2025](#)) apresenta o ecossistema de minijogos ao qual o jogo produzido neste trabalho será posteriormente adicionado. Nesse artigo foi apresentado um conjunto de 3 minijogos, na qual o programa desenvolvido nesta monografia não estava incluso, que usavam como base questões Bebras a fim de garantir que os conceitos trabalhados estivessem alinhados com práticas já consolidadas na educação em pensamento computacional e realizar um teste com eles.

Durante esse teste foi observado que os indivíduos tinham a familiaridade limitada com o assunto. Além disso, durante a jogatina foi observado que os jogadores tiveram dificuldades na navegação do ambiente, localização, direcionamento, entendimento da tarefa a ser realizada e manusear o controle do jogo.

Artigos	Realidade Virtual	Pensamento Computacional	Testes	Bebras
(HSU; CHANG; HUNG, 2018)		X		
(CAPECCHI et al., 2022)	X		X	
(AGBO et al., 2023)	X	X	X	
(CARDOSO et al., 2025)		X	X	X
Este TCC		X		X

Tabela 1 – Comparação entre os trabalhos

Esses elementos negativos aliados as boas avaliações nos aspectos relacionados a narrativa, mecânica e funções dos jogos indicam que existe ainda desafios a serem resolvidos a fim de tornar o jogo mais intuitivo e acessível.

Diante às características de cada um desses trabalhos, vistas na Tabela 1, é possível observar que entre eles (HSU; CHANG; HUNG, 2018) é o único que não desenvolve nenhum jogo, logo não preenche a maioria das categorias, porém ele apresenta o estado da arte de artigos sobre pensamento computacional, dividindo o tema com (AGBO et al., 2023), (CARDOSO et al., 2025) e está monografia.

Além disso, destaca-se o que dentre os artigos que desenvolveram um jogo, apenas (CAPECCHI et al., 2022) e (AGBO et al., 2023) usaram um ambiente de realidade virtual. Também observa-se que esses artigos não usaram o Bebras como uma fonte de inspiração para o desenvolvimento de seus jogos, diferentemente deste Trabalho de Conclusão de Curso (TCC) e do projeto (CARDOSO et al., 2025). Ademais, dentre os artigos que desenvolveram um jogo apenas esta monografia não realizou um teste para verificar a eficácia do programa desenvolvido, visto que não houve tempo para integrar o protótipo no projeto (CARDOSO et al., 2025) antes que o teste fosse realizado.

4 Desenvolvimento

Este capítulo apresenta o processo para desenvolvimento do jogo sério, além dos elementos que o compõem.

4.1 Ferramentas Usadas

Para o desenvolvimento do jogo foi usado a engine de jogos Unity e a linguagem de programação orientada a objeto C# para a codificação. Os modelos 3D usados foram retirados de um projeto da Faculdade de Computação para o auxílio do ensino de pensamento computacional ou foram feitos usando a plataforma de modelagem 3D Blender. As figuras das senhas cifradas e da tabela de símbolos foram feitas na ferramenta de criação de ilustrações Krita

4.2 Base do jogo

Os processos e elementos que compõem o jogo vieram de um projeto da Faculdade de Computação para o auxílio do ensino de pensamento computacional, isso inclui a plataforma de criação, motivo pelo qual o jogo é feito por meio da gameficação de uma prova do Bebras, a narrativa e as mecânicas inclusas dentro do jogo.

Antes da produção deste jogo, o projeto passou por um teste utilizando 3 minijogos desenvolvidos com um conjunto de indivíduos que apresentou a dificuldade dos participantes de manusear os controles. Esses minijogos usavam o mouse para movimentação da câmera e usava as letras "W", "A", "S"e "D"para movimentar o personagem para frente, para trás e para os lados em um ambiente de livre movimentação, representado na Figura 2. Por conta disso, foi decidido que o sistema de controles seria alterado para a movimentação da câmera usando as setas do teclado e o mouse seria usado para interagir com objetos, ilustrados na Figura 3.

O jogo desenvolvido neste trabalho foi feito usando as decisões do teste realizado pelo projeto (CARDOSO et al., 2025), além do game design document que foi modificado após os resultados obtidos, usando o feedback fornecido pelos participantes para permitir oferecer uma melhor experiência para outros jogadores.

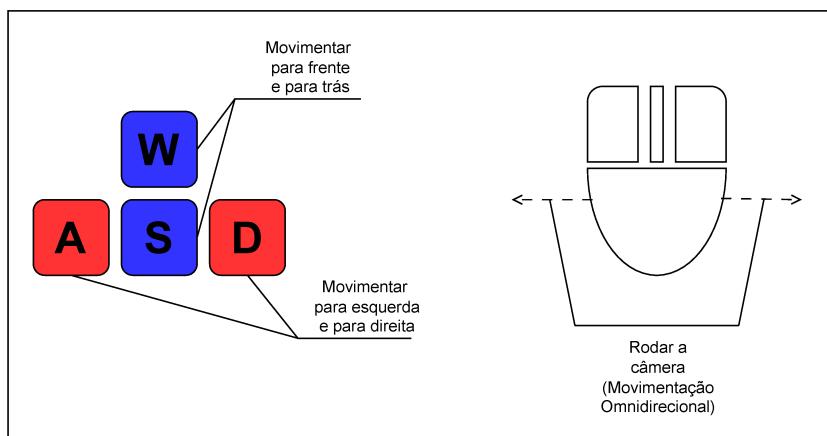


Figura 2 – Representação dos antigos controles

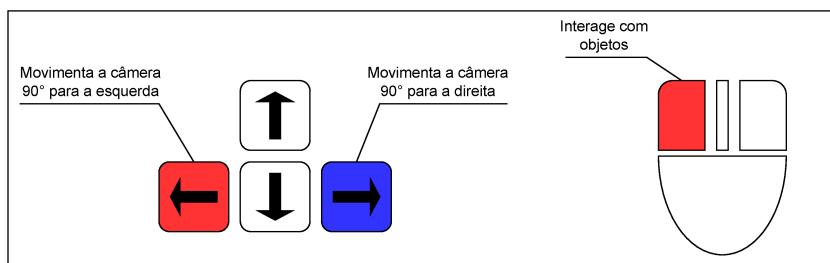


Figura 3 – Representação dos novos controles

4.3 Questão do Bebras

O jogo foi feito por meio da gamificação da questão Mensagem do Castor Ancião da edição de 2019 da prova do Bebras de Portugal para alunos do 7º e 8º ano de escolaridade e tem como tema o processo de criptografia. Resumidamente, o enunciado da questão quer que o indivíduo descubra uma frase por meio do deciframento de um conjunto de símbolos, observado na Figura 5, usando uma tabela de códigos, demonstrado na Figura 4, disponibilizada para o exercício.

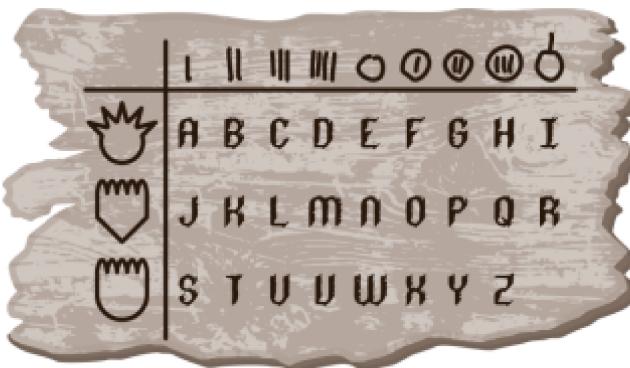


Figura 4 – Tabela de códigos



Figura 5 – Mensagem cifrada

Essa questão foi escolhida devido à importância que a segurança de dados apresenta nos dias atuais por conta do constante fluxo de dados que trafegam pela internet e a necessidade de mantê-los em segredo. Um dos métodos para isso é a criptografia, a temática abordada nessa questão, logo o processo de gamificação permitiria o auxílio do ensino desse conteúdo nas escolas.

4.4 Tema Abordado

O principal tema abordado durante o jogo é a criptografia, de acordo com ([TERADA, 2008](#)) criptografia é o conceito de esconder uma informação sigilosa de forma que indivíduos não autorizados não tenham acesso a ela. A cifragem de informação dentro do jogo foi feita por meio de uma tabela de substituição simples, a qual consiste na troca das letras da informação sigilosa por outros símbolos.

Por meio do jogo, o jogador é apresentado a maneira pela qual os computadores cifram e decifram os dados, além de compreender o que um criptoanalista precisaria para decifrar um texto e a importância de determinadas partes do processo de cifragem e decifragem. Como por exemplo a tabela ou senha usada durante a cifragem.

Ademais, o tema proposto incentiva os jogadores a observarem e analisarem a maneira como as mensagens cifradas são organizadas e começar a entender como um criptoanalista tentam decifrar um informação sigilosa sem ter acesso a informações importantes.

4.5 Game Design Document

O Game Design Document (GDD) foi feito com o intuito de descrever as mecânicas implementadas, o contexto narrativo e a gameplay do jogo. As mecânicas apresentadas e a narrativa não devem desviar ou comprometer o GDD principal desse projeto.

Assim, o contexto do jogo tem o jogador realizando desafios e problemas em uma floresta e ganhando como recompensa elementos para um piquenique. Ademais o jogador tem a capacidade de movimentar a câmera, abrir interfaces, clicar em objetos que podem ser interagidos e pegar objetos e arrastá-los para certas posições. Essa última mecânica não foi usada devido a falta de necessidade.

Para descrever o jogo da melhor maneira possível o GDD apresenta a descrição dos

objetivos educacionais, o público-alvo, escolaridade, narrativa, personagem e ambientação, mecânicas do jogo, jogabilidade e a questão sendo usada, os detalhes de cada tópico podendo ser visto em seguida.

Objetivos educacionais: Criptografia, identificação de padrões.

Público-alvo e escolaridade: Indivíduos da faixa etária entre 12-13 anos, cursando o ensino fundamental.

Narrativa: Uma coruja percebeu que o jogador fará um pique-nique e gostaria de oferecer a ele geléia, porém ela está guardada em um baú trancado, cuja a senha a coruja esqueceu. Entretanto a coruja possui a senha cifrada do baú e uma tabela de símbolos que foi usado para a cifragem dessa senha, logo caberá ao jogador decifrar essa senha e abrir o baú.

Personagem e ambientação: O jogo estará localizado em uma cabine em um floresta, na cabana estará uma coruja (o personagem ao qual deve ser interagido), um baú e uma tabela de símbolos fixada na parede da cabine.

Mecânicas do jogo e Jogabilidade: O jogador é capaz de movimentar a câmera com as setas do teclado, interagir com alguns objetos com o botão esquerdo do mouse. Interagindo com o baú uma interface aparece com uma mensagem cifrada mais um conjunto de botões e letras, será apresentada onde o usuário poderá alterar as letras usando os botões para formar a versão decifrada da mensagem.

Questão Bebras: Será usado a questão 6, Mensagem do Castor Ancião, da edição de 2019 para o 7º e 8º ano de escolaridade de Portugal.

4.6 Requisitos funcionais

Para determinar os requisitos funcionais do sistema e as interações do sistema foi definido um diagrama de casos de uso, observado na Figura 6. Diante disso é possível indicar como agente principal do sistema os jogadores, sendo aquele que vai usar todas as funções que o jogo pode executar, sendo elas movimentar a câmera, interagir com o baú e Non-Player Characters (NPCs), responder a questão.

Quando o baú for utilizado uma interface será apresentada na tela do jogador para que ele possa dar uma resposta. Quando houver uma interação com um NPC a caixa de dialogo será apresentada e se o jogador já tiver conversado com o NPC a partir de agora ele irá verificar a resposta dada, essas ações sendo indicadas pelo "include". Caso o jogador tenha acertado a resposta ou tenha extrapolado o limite de tentativas o sistema apresentará a tela com o resultado final, indicado pelo "extends".

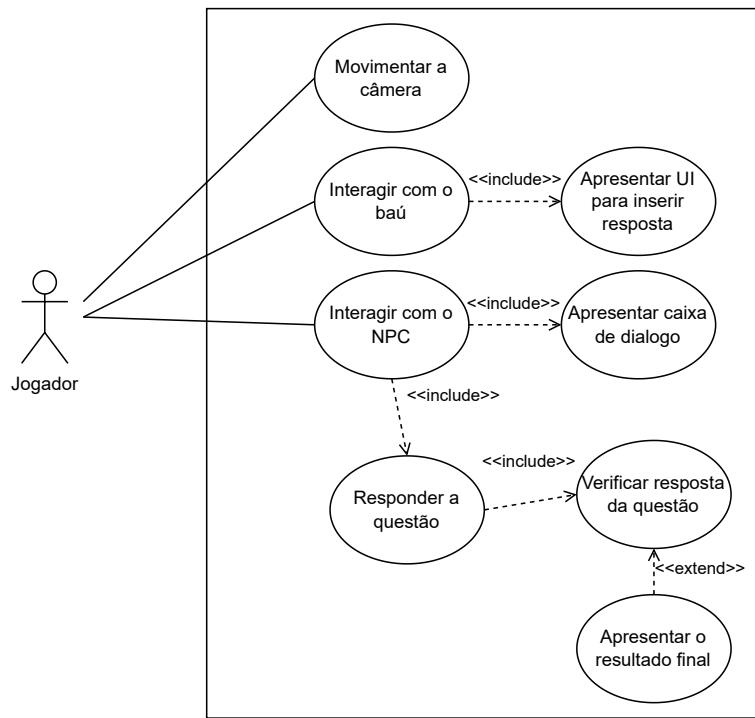


Figura 6 – Diagrama de caso de uso

4.7 Estruturação das classes

Durante o desenvolvimento do jogo a estruturação e relação das classes foi analisado na forma de um diagrama de classes, podendo ser visto de maneira simplificada na Figura 7. O diagrama foi simplificado para reduzir o tamanho da imagem e permitir a visualização por meio da remoção dos atributos e funções deixando apenas as relações, o diagrama pode ser visto por completo no apêndice A.

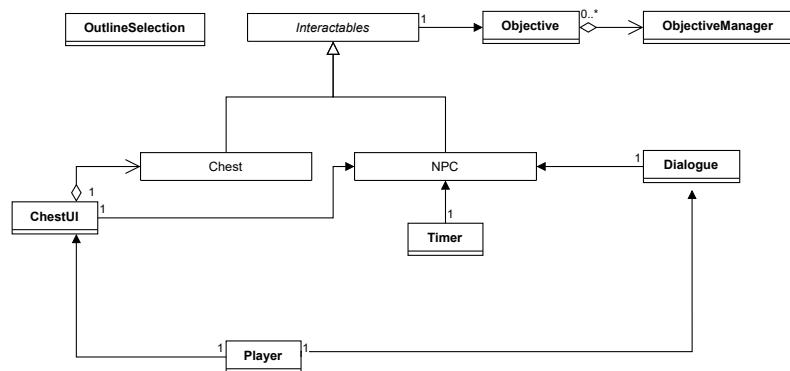


Figura 7 – Diagrama de classes simplificado.

Para os objetos que serão interagidos pelos jogadores representados pelas classes **Chest** e **NPC**, ambos herdam da classe abstrata **Interactables**, demonstrado na Figura 8,

devido o fato que a função `OnMouseDown`, responsável pela detecção do botão esquerdo do mouse, deve ser a mesma entre as duas funções, porém `Action`, ações realizadas pelo objeto, deve ser diferente entre eles.

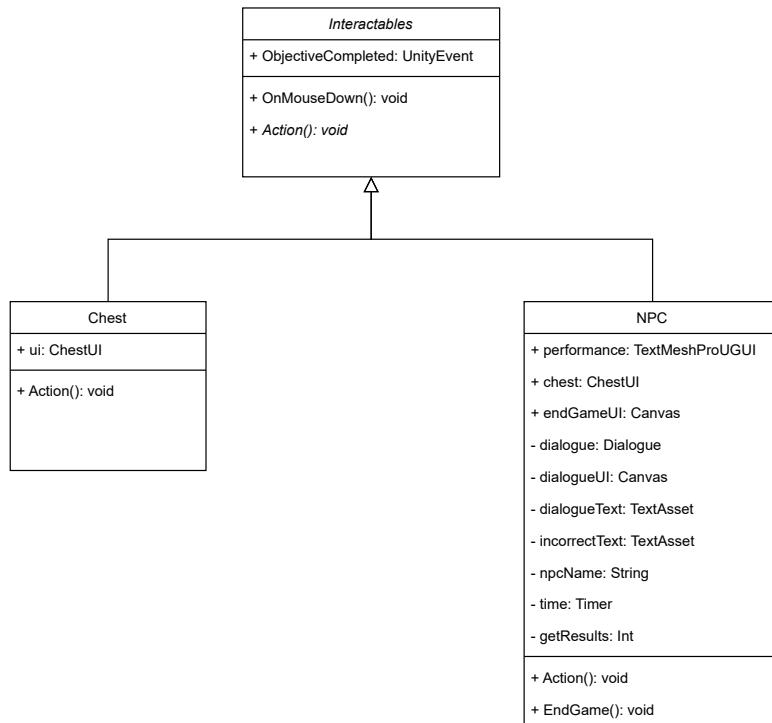


Figura 8 – Diagrama de classes da classe `Interactables`.

Ademais, a classe `ObjectManager` é responsável por gerenciar e apresentar na tela os objetivos, sendo composto por múltiplas classes `Objective`, responsável por armazenar os dados base do objetivo. Esta classe está associado a um objeto da classe `Interactables` pois deve perceber quando a ação para sua conclusão for executada, sendo demonstrado na Figura 9.

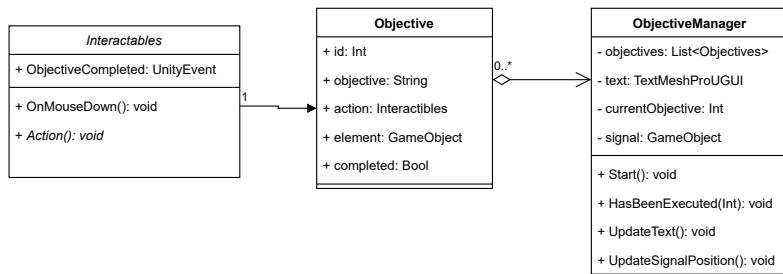


Figura 9 – Diagrama de classes da classe `ObjectManager`.

Além da classe `ObjectManager`, existe outras classes que afetam a User Interface (UI), sendo elas `ChestUI`, `Dialogue` e `Timer` responsáveis por apresentar o sistema de

inserção de respostas e a senha do baú cifrada, apresentar o diálogo de um determinado non-player character (NPC) e mostrar o tempo decorrido da jogatina respectivamente.

ChestUI é um componente da classe Chest, visto que ela oferece a interatividade que Chest necessita para executar suas funções e é referenciado por NPC já que ele precisa da resposta para apresentar os resultados, sendo demonstrado na Figura 10. Dialogue é referenciado por NPC, para permitir que Dialogue possa ser manipulado, porém não deve estar atrelado a um único NPC, demonstrado na Figura 11. Finalizando, Timer é referenciado por NPC para que seus dados possam ser usados na apresentação dos resultados, como observado na Figura 12.

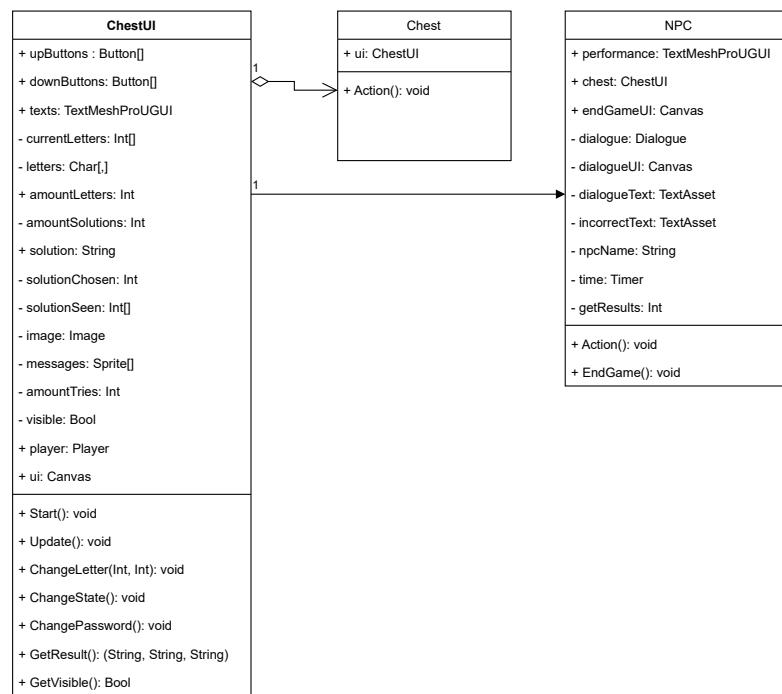


Figura 10 – Diagrama de classes da classe ChestUI.

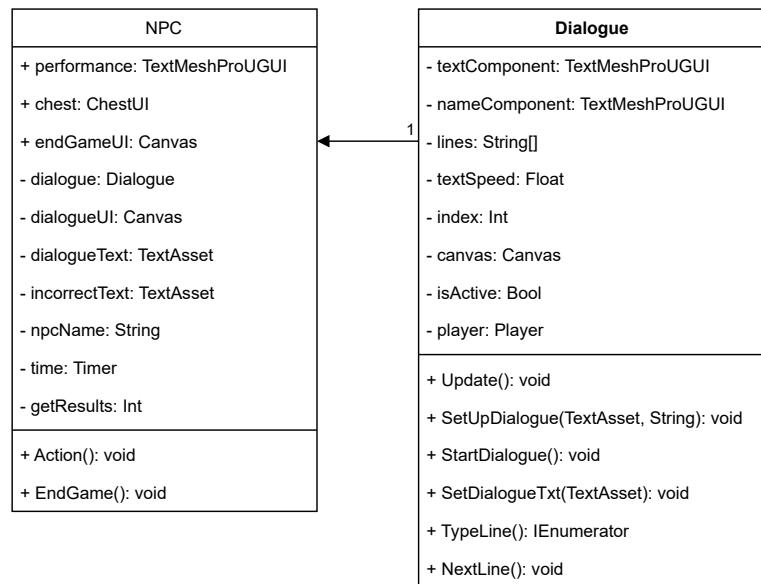


Figura 11 – Diagrama de classes da classe Dialogue.

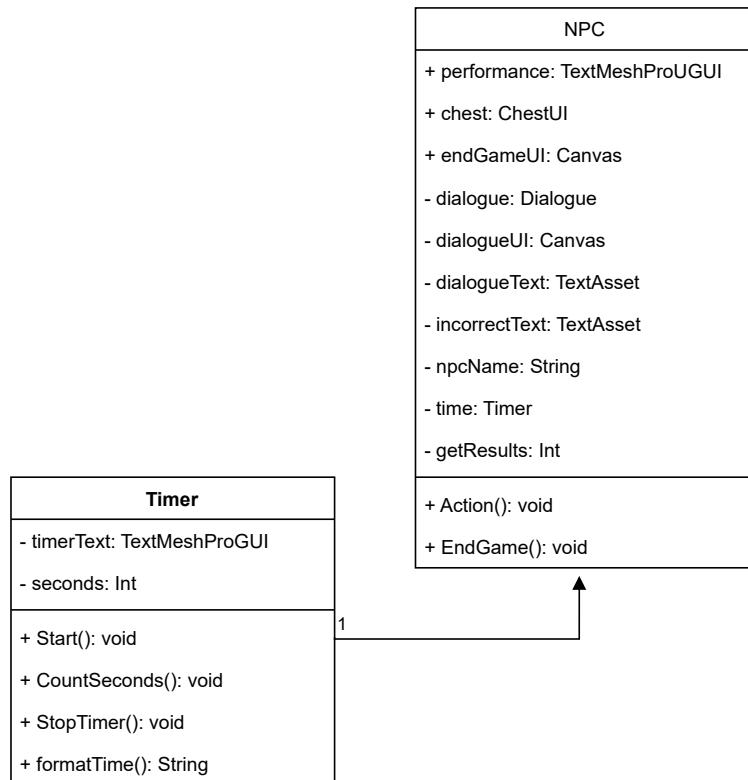


Figura 12 – Diagrama de classes da classe Timer.

Concluindo, tem a classe Player responsável pela rotação da câmera e é referenciado

pelas classes ChestUI e Dialogue, observado na Figura 13, visto que eles tem que desativar a rotação da câmera quando ativado, e a classe OutlineSelection que destaca o contorno dos objetos que podem ser interagidos usando o sistema de tags da Unity, sendo possível observar a classe em maior detalhes na Figura 14.

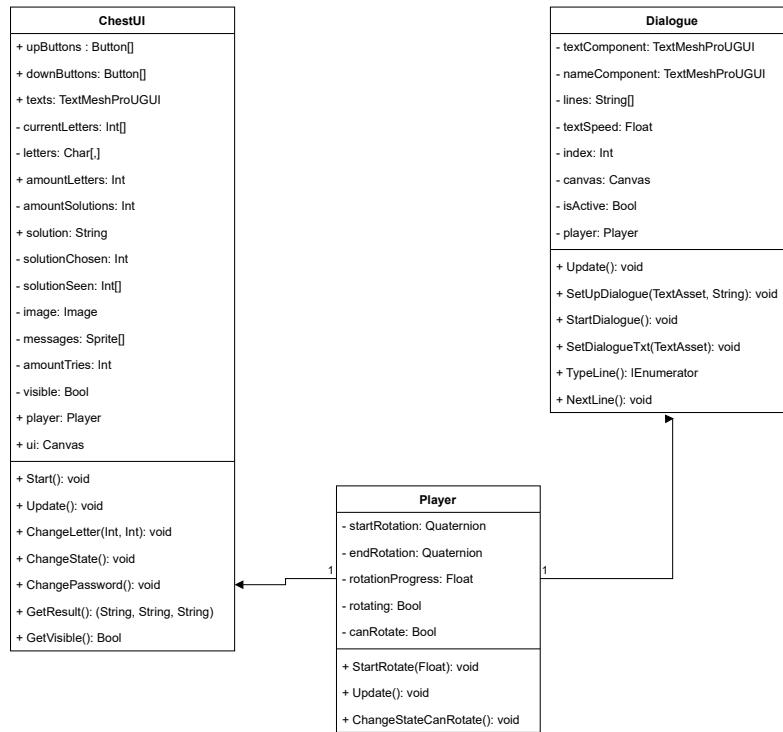


Figura 13 – Diagrama de classes da classe Player.

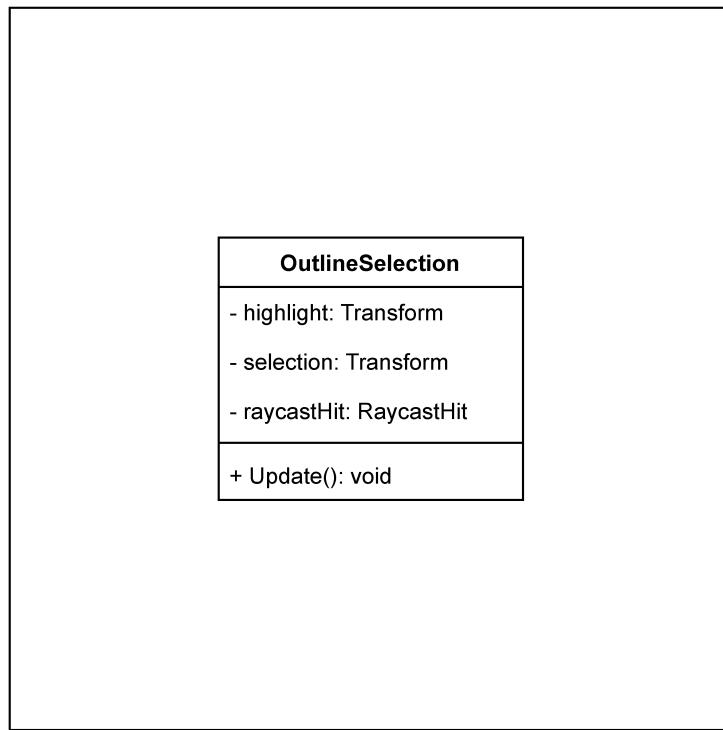


Figura 14 – Diagrama de classes da classe OutlineSelection.

4.8 Gameplay

Durante a jogatina o jogador será capaz de movimentar a câmera em 90° para direita ou para a esquerda usando as setas do teclado e capaz de interagir com alguns objetos do cenário usando o botão esquerdo do mouse. O objetivo principal é decifrar a senha de um baú usando uma tabela de símbolos que foi disponibilizada.



Figura 15 – Imagem do jogo mostrando a coruja e tabela de simbolos

Primeiramente, o jogador deverá interagir com o NPC o qual apresentará a historia e o objetivo desse cenário, sendo observado na Figura 15 sem o dialogo e na Figura 17

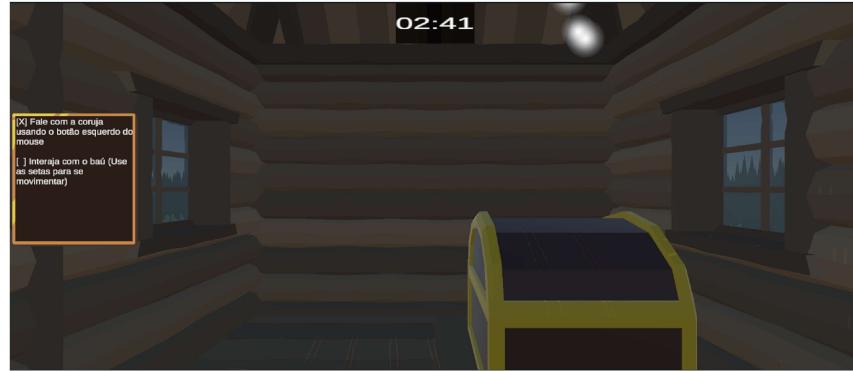


Figura 16 – Imagem do jogo mostrando o baú

com dialogo. Em seguida, o jogador deverá interagir com o baú, demonstrado na Figura 16, isso apresentará ao jogador com a interface responsável por inserir a resposta e por apresentar a senha cifrada, apresentado na Figura 18. Os botões em são responsáveis de trocar as letras disponibilizadas de maneira circular.

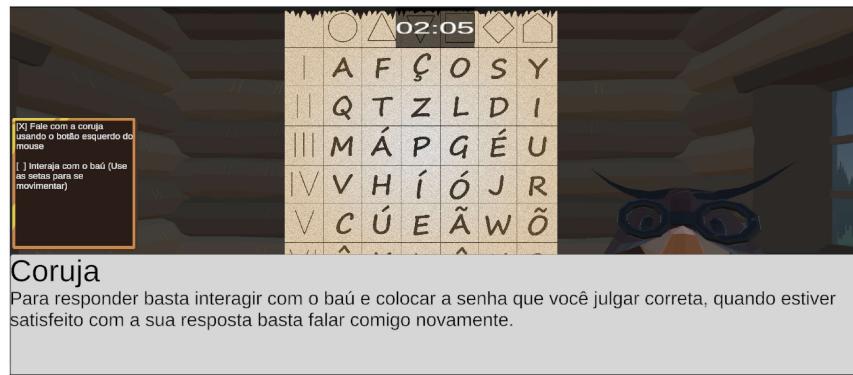


Figura 17 – Imagem do jogo mostrando o dialogo



Figura 18 – Imagem do jogo mostrando a interface do baú

Após, o jogador definir a resposta que julgar correta, ele deve retornar ao NPC, o qual dirá se a resposta está correta ou errada. Caso o usuário esteja equivocado a senha será alterada e o jogador terá outra chance, com um máximo de três tentativas, caso acerte

será apresentado as estatísticas, como tempo, decifragem de todas as senhas apresentadas e o item do piquenique recebido, encerrando a sessão, podendo ser visto na Figura 19.



Figura 19 – Imagem do jogo mostrando os resultados

5 Conclusão

Durante a produção do jogo, o processo para a produção de um programa, como o desenvolvimento de uma documentação, modelagem do software e a codificação, foram aprendidas. Para o desenvolvimento de uma documentação foi criado um game design document a fim de informar todos os componentes do jogo, estabelecendo uma visão inicial de todas as funções do jogo, isso foi feito para o projeto da Faculdade de Computação para o auxílio do ensino de pensamento computacional.

Na modelagem do software a criação de um diagrama de classes permitiu a estruturação e a interação das classes antes da sua implementação e com diagrama de caso de uso foi possível estabelecer as interações entre o sistema e os usuários, além de definir requisitos principais do sistema, esses diagramas foram feitos especificamente para este trabalho.

No transcorrer da codificação foi observado as características de uma linguagem orientada a objeto com o uso da linguagem de codificação C#, com o uso de heranças, além das características da ferramenta de desenvolvimento de jogos Unity, como a criação de objetos, organização do ambiente, o sistema de tipagem e como os objetos interagem entre si.

No decorrer do desenvolvimento, houve dificuldades durante a elaboração da estruturas das classes, visto que o futuro desse jogo é ser parte de um conjunto maior, logo a elaboração das classes foi pensada para poderem atender à funcionalidades gerais.

Em conclusão, neste trabalho foi possível desenvolver um jogo sério para auxiliar o ensino de pensamento computacional, por meio da gameficação da questão mensagem do castor ancião do Bebras, além de ter sido criado um GDD para servir de documentação para o jogo.

Futuramente, planeja-se adicionar o jogo sério como um mini-jogo dentro do projeto da Faculdade de Computação para o auxílio do ensino de pensamento computacional e executar neste conjunto testes com indivíduos diferentes e relacionados a temática. Além disso, a parte visual do jogo será incrementada.

Referências

- AGBO, F. J.; OLALEYE, S. A.; BOWER, M.; OYELERE, S. S. Examining the relationships between students' perceptions of technology, pedagogy, and cognition: the case of immersive virtual reality mini games to foster computational thinking in higher education. **Smart Learning Environments**, Springer, v. 10, n. 1, p. 16, 2023. Citado 3 vezes nas páginas 10, 18 e 19.
- BEBRAS. Bebras. 2025. <<https://www.bebras.org/>>. Acesso em 06 de Maio de 2025. Citado na página 13.
- CAPECCHI, I.; BORGHINI, T.; BARBIERATO, E.; GUAZZINI, A.; SERRITELLA, E.; RAIMONDI, T.; SARAGOSA, C.; BERNETTI, I. The combination of serious gaming and immersive virtual reality through the constructivist approach: an application to teaching architecture. **Education Sciences**, MDPI, v. 12, n. 8, p. 536, 2022. Citado 2 vezes nas páginas 17 e 19.
- 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 3 vezes nas páginas 18, 19 e 20.
- FARDO, M. L. A gamificação aplicada em ambientes de aprendizagem. **Renote**, v. 11, n. 1, 2013. Citado na página 12.
- FARINELLI, F. Conceitos básicos de programação orientada a objetos. **Instituto Federal Sudeste de Minas Gerais**, 2007. Citado na página 14.
- GUEDES, G. T. **UML 2-Uma abordagem prática**. [S.l.]: Novatec Editora, 2018. Citado na página 15.
- HSU, T.-C.; CHANG, S.-C.; HUNG, Y.-T. How to learn and how to teach computational thinking: Suggestions based on a review of the literature. **Computers & Education**, Elsevier, v. 126, p. 296–310, 2018. Citado 3 vezes nas páginas 12, 17 e 19.
- Ministério da Educação. **Normas sobre computação na educação básica - complemento à Base Nacional Comum Curricular**. 2022. Acesso em 14 de Julho de 2025. Disponível em: <http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=235511-pceb002-22&category_slug=fevereiro-2022-pdf&Itemid=30192>. Citado na página 10.
- MOTTA, R. L.; JUNIOR, J. T. Short game design document (sgdd). **Proceedings of SBGames**, v. 2013, p. 115–121, 2013. Citado na página 16.
- TERADA, R. **Segurança de dados: criptografia em rede de computador**. [S.l.]: Editora Blucher, 2008. Citado 2 vezes nas páginas 13 e 22.

UNITY. 2025. Acesso em 02 em Julho de 2025. Disponível em: <<https://unity.com>>. Citado na página 14.

WING, J. M. Computational thinking. **Communications of the ACM**, ACM New York, NY, USA, v. 49, n. 3, p. 33–35, 2006. Citado 2 vezes nas páginas 10 e 12.

Apêndices

APÊNDICE A – Diagrama de classes completo

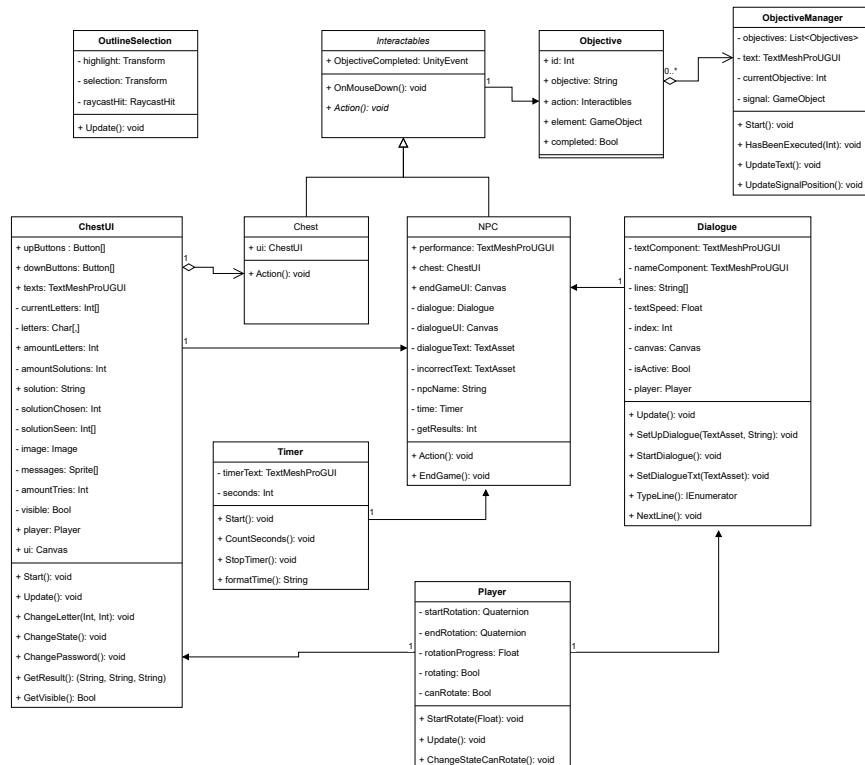


Figura 20 – Diagrama de classe