

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

MATHEUS GONÇALVES RIBEIRO

**DESENVOLVIMENTO DE UM JOGO SÉRIO EM REALIDADE
VIRTUAL: UMA APLICAÇÃO PRÁTICA NA EDUCAÇÃO**

Ouro Preto
2026

MATHEUS GONÇALVES RIBEIRO

**DESENVOLVIMENTO DE UM JOGO SÉRIO EM REALIDADE VIRTUAL: UMA
APLICAÇÃO PRÁTICA NA EDUCAÇÃO**

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Reinaldo Silva Fortes

Coorientador: Prof. Dr. Saul Emanuel Delabrida Silva

Ouro Preto, MG
2026



FOLHA DE APROVAÇÃO

Matheus Gonçalves Ribeiro

Desenvolvimento de um Jogo Sério em Realidade Virtual: Uma aplicação prática na educação

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 27 de Fevereiro de 2026

Membros da banca

Reinaldo Silva Fortes (Orientador) - Doutor - Universidade Federal de Ouro Preto
Saul Emanuel Delabrida Silva (Coorientador) - Doutor - Universidade Federal de Ouro Preto
Rone Ilídio da Silva (Examinador) - Doutor - Universidade Federal de São João del Rey
Laura Martins da Costa Coura Marinho (Examinadora) - Mestre - Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Ouro Preto

Reinaldo Silva Fortes, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 27/02/2026



Documento assinado eletronicamente por **Reinaldo Silva Fortes, PROFESSOR DE MAGISTERIO SUPERIOR**, em 03/03/2026, às 13:38, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1062472** e o código CRC **EEA1CEC3**.

Por todo o apoio incondicional, dedico este trabalho aos meus pais, Mirtes Gonçalves Silva
Ribeiro e Paulo Cesar Ribeiro, e ao meu irmão Diego Gonçalves Ribeiro.

Agradecimentos

Meu profundo agradecimento a todos aqueles que me influenciaram de qualquer forma na minha trajetória acadêmica, mas dedico este espaço de folha para aqueles que foram fundamentais em minha jornada, em especial aos meus familiares. Portanto, nada mais justo do que começar com um agradecimento especial para com aqueles que abriram mão de tanto para acreditar no meu sonho e que sempre estiveram do meu lado em cada decisão, minha mãe, Mirtes Gonçalves Silva Ribeiro, e ao meu pai, Paulo Cesar Ribeiro. Agradeço também ao meu irmão Diego Gonçalves Ribeiro por todos os conselhos valiosos e principalmente por ser a imagem de inspiração durante toda a minha trajetória acadêmica, profissional e pessoal. Por fim, gostaria de agradecer a todos os amigos que me apoiaram nesta jornada, em especial minha companheira e amiga, Luiza Mayra dos Anjos Caldeira, e meu amigo Luiz Henrique Pereira Paes, por também acreditarem e me aconselharem durante este projeto e graduação.

Agradeço também aos meus orientadores, Prof. Dr. Reinaldo Silva Fortes e Prof. Dr. Saul Emanuel Delabrida Silva, por terem acreditado no meu trabalho e por toda a valiosa orientação durante seu desenvolvimento.

Agradeço também às instituições que apoiaram e viabilizaram este projeto, a Universidade Federal de Ouro Preto (UFOP) e a Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) - Códigos de Financiamento APQ-03665-22 e APQ-06466-24.

“The combined use technologies such as 3D scanners, Augmented Reality and Virtual Reality and wearable devices open a wide range of opportunities to the creation of new applications for education, training and analysis purposes.”

(SILVA, 2018)

Resumo

Este trabalho apresenta o desenvolvimento de um jogo sério em *Realidade Virtual (RV)*, voltado ao ensino de noções introdutórias de geolocalização a estudantes do ensino fundamental, de 9 a 10 anos de idade. A concepção inicial da proposta teve origem em um dos projetos elaborados no contexto da aplicação da metodologia apresentada por [Marinho \(2024\)](#) em sua dissertação *EducaMinasVR: qualificação de professores em design de interfaces imersivas educativas em realidade virtual*. Diante do cenário educacional que demanda metodologias mais dinâmicas e imersivas ([SOUZA; SOUSA; JUCÁ, 2025](#)), a proposta consiste na criação de um ambiente virtual que integra exploração prática por meio de instrumentos de orientação espacial, como mapa e bússola virtuais, a conteúdos pedagógicos disponibilizados por um *Sistema de Gestão de Aprendizagem (LMS)*. Metodologicamente, adotou-se o *Canvas de Design de Jogos Educativos (MARINHO, 2024) (CDJE)* para a idealização do projeto, traçando objetivos de aprendizagem, mecânicas e *feedbacks*, estruturando o projeto em blocos de base, pedagógico, aprendizado e imersão, sendo utilizado também durante as etapas de reformulação do *design*. A implementação foi realizada na ferramenta *Unity*, contemplando a construção de um ambiente virtual explorável, com terreno gerado proceduralmente e interface interativa adaptada para dispositivos de realidade virtual. O *LMS* escolhido foi o *Moodle* e a integração foi contruída com base nas estruturas do pacote *SeriousGameComponents*, em conjunto com adaptações específicas para o carregamento dinâmico de conteúdos dentro do jogo, permitindo a inserção de materiais pedagógicos no jogo. Testes de usabilidade nos dispositivos *Meta Quest 2*, *Meta Quest 3* e *Meta Quest Pro* validaram a viabilidade das interações e o fluxo geral do projeto. Em síntese, o jogo sério implementado evidencia a viabilidade da proposta e indicam o potencial do uso de jogos sérios em *RV* como ferramenta de apoio ao ensino de geolocalização. Além disso, este projeto gerou um caso de uso da aplicação do *CDJE* sob a perspectiva do desenvolvedor.

Palavras-chave: Realidade Virtual. Jogos Sérios. Geolocalização. *Unity*. *Moodle*.

Abstract

This work presents the development of a serious game in *Virtual Reality* (VR), aimed at teaching introductory notions of geolocation to elementary school students aged 9 to 10 years. The initial conception of the proposal originated from one of the projects developed in the context of the application of the methodology presented by [Marinho \(2024\)](#) in her master's dissertation *EducaMinasVR: qualification of teachers in the design of immersive educational interfaces in virtual reality*. Given the educational scenario that demands more dynamic and immersive methodologies ([SOUZA; SOUSA; JUCÁ, 2025](#)), the proposal consists of creating a virtual environment that integrates practical exploration through spatial orientation instruments, such as virtual maps and compasses, with pedagogical content provided through an LMS. Methodologically, the CDJE was adopted for the conception of the project, defining learning objectives, mechanics, and *feedback*, structuring the project into base, pedagogical, learning, and immersion blocks, and also being used during the stages of *design* reformulation. The implementation was carried out using the *Unity* engine, including the development of an explorable virtual environment with procedurally generated terrain and an interactive interface adapted for virtual reality devices. The chosen LMS was *Moodle*, and the integration was built based on the structures of the *SeriousGameComponents* package, together with specific adaptations for the dynamic loading of content within the game, allowing the insertion of pedagogical materials into the game. Usability tests conducted on the *Meta Quest 2*, *Meta Quest 3*, and *Meta Quest Pro* devices validated the feasibility of the interactions and the overall flow of the project. In summary, the implemented serious game demonstrates the feasibility of the proposal and indicates the potential of using serious games in VR as a tool to support the teaching of geolocation. Furthermore, this project generated a use case of the application of CDJE from the developer's perspective.

Keywords: Virtual Reality. Serious Games. Geolocation. Unity. Moodle.

Lista de Figuras

Figura 2.1 – Amostra de <i>Perlin Noise</i> , a escala de cinza representa valores entre 0 e 1. Fonte: Unity (2025).	7
Figura 2.2 – Canvas de Design de Jogos Educativos. Fonte: (MARINHO, 2024).	10
Figura 2.3 – Utilização dos componentes de slides e do reprodutor de áudio do pacote proposto por (PIMENTA, 2022) durante a fase de prototipação. Fonte: Autor (2026).	11
Figura 3.1 – Versão final do CDJE preenchido para este projeto. Fonte: do autor (2026).	12
Figura 3.2 – Cenas do ambiente do jogo. Fonte: do autor (2025).	15
Figura 3.3 – Áreas predeterminadas para instanciar objetos de interesse. Fonte: do autor (2025).	15
Figura 3.4 – Terreno gerado proceduralmente. Fonte: do autor (2026).	16
Figura 3.5 – Etapas da geração do mapa de alturas, dispostos como textura em um <i>GameObject</i> , onde tons escuros representam valores próximos a 0 e tons claros valores próximos a 1. Fonte: do autor (2026).	17
Figura 3.6 – Objeto Programável responsável por armazenar informações sobre as regiões. Fonte: do autor (2026).	17
Figura 3.7 – Malha do terreno com altura dos vértices baseada no mapa de ruído. Fonte: do autor (2026).	18
Figura 3.8 – Classe <i>MapData</i> e seus atributos. Fonte: do autor (2026).	19
Figura 3.9 – Objetos interativos do protótipo inicial. Fonte: do autor (2026).	19
Figura 3.10–Modelos 3D do <i>tablet</i> e do relógio inteligente. Fonte: do autor (2026).	20
Figura 3.11–Modelo 3D da tela de bússola. Fonte: do autor (2026).	22
Figura 3.12– <i>Script NeedleController</i> . Fonte: do autor (2026).	22
Figura 3.13– <i>Script NeedleController</i> no <i>inspector</i> . Fonte: do autor (2025).	22
Figura 3.14–Modelo 3D da tela do mapa. Fonte: do autor (2026).	23
Figura 3.15–Correção de cores para marcação do X. Fonte: do autor (2026)	24
Figura 3.16–Modelo 3D da tela do Temporizador. Fonte: do autor (2026).	25
Figura 3.17–Tela de configurações. Fonte: do autor (2026).	25
Figura 3.18–Modelo 3D da tela do Radar. Fonte: do autor (2026).	26
Figura 3.19–Cálculo de distância realizado no <i>script WatchController</i> . Fonte: do autor (2026).	26
Figura 3.20–Modelos dos componentes de recursos educacionais mostrando conteúdos baixados do <i>Moodle</i> . Fonte: do autor (2026).	27
Figura 3.21–Injeção dos slides no <i>LmsLoader</i> após o download do conteúdo. Fonte: do autor (2026)	28

Figura 3.22–Inicialização e recuperação dos slides a partir do LmsLoader para exibição no sistema. Fonte: do autor (2026)	29
Figura 3.23–Diagrama de fluxo do uso do pacote <i>SeriousGameComponent</i> neste projeto. Fonte: do autor (2026).	30
Figura 4.1 – Jogador dentro da experiência imersiva. Fonte: do autor (2026).	32

Lista de Tabelas

Tabela 2.1 – Comparativo entre os trabalhos relacionados e a proposta atual. Fonte: O autor (2025). 9

Lista de Abreviaturas e Siglas

API *Application Programming Interface*. 4–6, 29

CDJE *Canvas de Design de Jogos Educativos* (MARINHO, 2024). v–vii, 1, 9, 10, 12, 14, 18, 20, 31–35

HMD *Head Mounted Display*. 4

LMS *Sistema de Gestão de Aprendizagem*. v, vi, 9, 11, 14, 29, 35

LRS *Sistema de Registro de Aprendizagem*. 35

RV *Realidade Virtual*. v, 1, 2, 4, 6–9, 31, 34

UI *Interface de Usuário*. 6

VR *Virtual Reality*. vi

Sumário

1	Introdução	1
1.1	Justificativa	2
1.2	Objetivos	2
1.3	Organização do Trabalho	2
2	Revisão Bibliográfica	4
2.1	Fundamentação Teórica	4
2.1.1	Realidade Virtual (RV)	4
2.1.2	Jogos Sérios	4
2.1.3	C#	5
2.1.4	Application Programming Interface	5
2.1.5	Unity	5
2.1.5.1	Pacotes	5
2.1.5.2	Objetos de Jogo	5
2.1.5.3	Transform	6
2.1.5.4	Prefabs	6
2.1.5.5	Cenas	6
2.1.5.6	Objeto Programável	6
2.1.5.7	OpenXR	6
2.1.5.8	XR Interaction Toolkit	6
2.1.6	Perlin Noise	7
2.2	Trabalhos Relacionados	7
3	Desenvolvimento	12
3.1	Idealização e Metodologia	12
3.2	Implementação e Decisões de Design	14
3.2.1	Configuração do Ambiente de Desenvolvimento	14
3.2.2	Ambiente Virtual	15
3.2.3	Objetos Interativos	18
3.2.3.1	Decisões de <i>Design</i> e Reformulação Arquitetural	18
3.2.3.2	Bússola	21
3.2.3.3	Mapa	23
3.2.3.4	Temporizador	24
3.2.3.5	Radar	25
3.2.3.6	<i>Slides</i> e <i>Player</i> de Áudio	27
3.2.4	Observações Durante o Desenvolvimento	30
4	Estudo de Caso	31
4.1	Visão Geral do Funcionamento do Jogo	31

4.2	Evolução do Protótipo ao Sistema Final	32
4.3	Uso do Canvas no Processo de Desenvolvimento	33
5	Considerações Finais	34
5.1	Conclusão	34
5.2	Trabalhos Futuros	35
 Referências		 36

1 Introdução

O atual cenário educacional tem enfrentado desafios constantes para se adaptar às novas demandas de ensino. A crescente presença dos recursos digitais no cotidiano dos estudantes exigem metodologias mais dinâmicas, que despertem o interesse e motivação destes alunos (SOUZA; SOUSA; JUCÁ, 2025). Diante disto, *Realidade Virtual (RV)* se apresenta como uma ferramenta promissora, por permitir a imersão e interatividade a partir da criação de ambientes tridimensionais. Tais ambientes oferecem ao estudante a possibilidade de vivenciar a teoria na prática, favorecendo a motivação e a absorção do aprendizado (TORI; KIRNER; SISCOUITO, 2006; FREITAS; GODINHO-PAIVA, 2021). Unido a isto, existem os jogos sérios, que, por definição, o propósito vai além do entretenimento, sendo utilizados para ensinar, treinar ou informar seus usuários (MICHAEL; CHEN, 2006).

Um dos conteúdos que mais apresentam desafios no ensino fundamental é a geolocalização. Estudos como o de Silva et al. (2018) apontam que muitos alunos encontram dificuldades em compreender conceitos como pontos cardeais, uso de mapas e manipulação de bússolas. Tais limitações sugerem a necessidade de abordagens inovadoras, que aproximem a teoria da prática de forma envolvente.

Sendo assim, este trabalho propõe o desenvolvimento de um jogo sério em *RV* voltado ao ensino de geolocalização. A proposta surge da dificuldade observada no ensino desse conteúdo por meio de abordagens tradicionais, que muitas vezes não engajam os alunos e não proporcionam experiências práticas suficientes para a compreensão de conceitos espaciais. Neste contexto, o projeto consiste em um ambiente imersivo onde os estudantes exploram cenários virtuais com auxílio de objetos como mapa e bússola, aplicando os conceitos de orientação espacial em uma atividade prática. Adicionalmente, este trabalho propõe uma integração à plataforma Moodle¹, escolhida por ser amplamente utilizada em instituições de ensino e por oferecer recursos para gerenciamento de conteúdos educacionais e acompanhamento do progresso dos alunos. Essa integração permite ao professor disponibilizar materiais complementares e acompanhar o desempenho dos alunos. Além disso, o trabalho busca analisar o processo de desenvolvimento do jogo sério a partir da aplicação do *Canvas de Design de Jogos Educativos (MARINHO, 2024) (CDJE)*, observando sua utilidade como ferramenta de apoio ao design de jogos educacionais em realidade virtual.

Para as próximas seções, este capítulo apresenta a justificativa que embasa a proposta do estudo (Seção 1.1), destacando sua relevância no contexto educacional. Em seguida, são definidos o objetivo geral e os objetivos específicos do trabalho (Seção 1.2). Por fim, é apresentada a organização do documento (Seção 1.3), indicando a forma como os capítulos estão estruturados.

¹ Moodle é um sistema de gerenciamento de aprendizagem de código aberto, amplamente utilizado para criação de ambientes virtuais de ensino e aprendizagem.

1.1 Justificativa

A criação de um jogo sério voltado para o ensino de geolocalização pode atender à demanda por novas abordagens educacionais, capazes de superar as dificuldades que muitos estudantes apresentam na compreensão desse conteúdo.

O uso da **RV** possibilita que os alunos vivenciem, de forma prática, conceitos que são apresentados apenas na teoria em sala de aula. Além disso, jogos sérios apresentam potencial para aumentar a motivação e o engajamento dos estudantes durante o processo de aprendizagem, uma vez que incorporam elementos interativos. Por fim, o desenvolvimento deste projeto também contribui para investigar e difundir o uso dessa ferramenta como meio de ensino, reforçando seu potencial como recurso de aprendizagem.

1.2 Objetivos

Este trabalho tem como objetivo principal o desenvolvimento de um jogo sério em **RV** voltado ao ensino de estudantes sobre conceitos introdutórios no tema de geolocalização. Para alcançar esse propósito, foram definidos os seguintes objetivos específicos:

- Projetar um ambiente virtual tridimensional que estimule a aprendizagem de conteúdos educacionais por meio da exploração e interação;
- Implementar mecânicas de jogo acessíveis e intuitivas, compatíveis com as capacidades cognitivas do público-alvo e com os conteúdos abordados;
- Incorporar elementos de personalização com *Moodle*, onde o tutor possa disponibilizar material dentro do jogo;
- Investigar a aplicação do Canvas de Design de Jogos Educativos (CDJE) como ferramenta de apoio ao processo de desenvolvimento de jogos sérios em realidade virtual.

1.3 Organização do Trabalho

Este documento está estruturado de forma a proporcionar uma visão clara e progressiva do desenvolvimento da pesquisa e de seus resultados. Sua organização é descrita a seguir:

Capítulo 1: Introdução. Apresenta o contexto geral do estudo, sua relevância acadêmica e social, a justificativa da proposta, os objetivos estabelecidos e a estrutura do trabalho.

Capítulo 2: Revisão Bibliográfica. Aborda os fundamentos teóricos que sustentam a pesquisa, e trabalhos relacionados e utilizados como embasamento.

Capítulo 3: Desenvolvimento. Detalha a metodologia aplicada, e a implementação prática do projeto.

Capítulo 4: Estudo de Caso. Discute o caso de uso obtido neste trabalho, assim como as dificuldades encontradas.

Capítulo 5: Considerações Finais. Apresenta a conclusão do trabalho e sugestões para futuros trabalhos.

2 Revisão Bibliográfica

Este capítulo tem como objetivo contextualizar o trabalho desenvolvido, apresentando conceitos principais e estudos anteriores que dão suporte à este projeto. Inicialmente, são abordados os principais conceitos e ferramentas que embasam o trabalho na Seção 2.1, seguidos pela análise de trabalhos relacionados que contribuíram para idealização e implementação deste projeto na Seção 2.2.

2.1 Fundamentação Teórica

Nesta seção são apresentados ao leitor conceitos fundamentais que cercam o desenvolvimento do projeto. Portanto, é abordando brevemente o conceito de **RV**, na Seção 2.1.1. Seguido de uma breve definição sobre *Jogos Sérios*, na Seção 2.1.2, destacando sua relevância no contexto educacional. Em seguida, é abordado conceitos técnicos e ferramentas utilizadas na fase de implementação, como a linguagem C#, na Seção 2.1.3, *Application Programming Interfaces (APIs)*, na Seção 2.1.4 e a ferramenta de criação de jogos Unity, na Seção 2.1.5, assim como seus principais componentes.

2.1.1 Realidade Virtual (RV)

A **RV** pode ser compreendida como uma interface avançada que permite ao usuário navegar e interagir em tempo real com ambientes tridimensionais gerados por computador, frequentemente utilizando dispositivos multissensoriais (TORI; KIRNER; SISCOUTTO, 2006). Nos jogos digitais, essa tecnologia tem se destacado por proporcionar experiências imersivas e interativas, ampliando significativamente o engajamento e a percepção sensorial do jogador (FREITAS; GODINHO-PAIVA, 2021).

A imersão em **RV** normalmente ocorre por meio de dispositivos específicos, como os *Head Mounted Displays (HMDs)*, que exibem imagens que acompanham os movimentos da cabeça, enquanto controles manuais permitem a interação com objetos no ambiente virtual. A **RV** possui aplicações em diversas áreas, como treinamento profissional, cirurgias na medicina, simulações na área da aviação e automobilismo, educação e entretenimento.

2.1.2 Jogos Sérios

Os jogos sérios, ou *Serious Games*, podem ser definidos como jogos que não tem como objetivo principal lazer ou diversão. Sendo assim, seu objetivo é educar, treinar ou informar o usuário (MICHAEL; CHEN, 2006). Esse tipo de aplicação utiliza elementos característicos dos

jogos digitais, como desafios, regras e interatividade, para apoiar processos de ensino, treinamento ou conscientização.

Jogos sérios podem ser encontrados em diferentes áreas. Na educação, por exemplo, são utilizados para apoiar o ensino de conteúdos escolares por meio de atividades interativas. Já em contextos profissionais, jogos sérios também são utilizados para treinamento e simulação de situações que exigem tomada de decisão ou prática de procedimentos específicos.

2.1.3 C#

O C# é uma linguagem de programação orientada a objetos criada pela *Microsoft*, relativamente moderna, tendo seu lançamento em julho dos anos 2000 ([Microsoft, 2025](#)). Seus códigos são projetados pela plataforma .NET, sendo extremamente utilizados ao criar jogos na *Unity Engine*.

2.1.4 Application Programming Interface

Segundo [Ofoeda, Boateng e Effah \(2019\)](#), uma API é um conjunto de definições e protocolos que permite a interação entre diferentes softwares e sistemas, facilitando a comunicação e o compartilhamento de funcionalidades.

2.1.5 Unity

Unity é uma ferramenta para desenvolvimento de jogos digitais amplamente utilizada, destacando-se por suporte a diversas plataformas e infraestrutura robusta. Trata-se de um ambiente de desenvolvimento tanto para jogos 2D quanto 3D, com suporte tanto à codificação com a linguagem C#, quanto à lógica *drag-and-drop*.

A ferramenta é particularmente popular por possuir licenças gratuitas, e de ampla comunidade que contribui com discussões e até criação de material para ser reutilizado entre os desenvolvedores, desde assets básicos como texturas, modelos 3D, *sprites* até ferramentas mais complexas como pacotes e *plugins*.

2.1.5.1 Pacotes

Os pacotes (*packages*) no *Unity* são módulos distribuídos, que oferecem funcionalidades específicas como bibliotecas de código, ferramentas de editor, APIs gráficas e coleções de recursos, além de permitir a construção de pacotes personalizados ([UNITY, 2025](#)).

2.1.5.2 Objetos de Jogo

Objetos de Jogo ou *GameObjects* é qualquer objeto de um jogo em ambiente Unity, desde personagens, iluminação, câmeras e efeitos especiais. Por padrão um *GameObjet* não tem função

alguma, porém o desenvolvedor molda o objeto desejado adicionando a ele propriedades, como colisões, malhas de texturas, componentes de áudio e infinitos outros (UNITY, 2025).

2.1.5.3 Transform

Um *transform*, armazena a posição e rotação de um *GameObject*. Um *GameObject* sempre contém um *transform* não sendo possível sua remoção (UNITY, 2025).

2.1.5.4 Prefabs

Um *Prefab* é um recurso que age como uma predefinição de um *GameObject*, permitindo que você crie e configure *GameObjects* assim como seus respectivos componentes e valores (UNITY, 2025). Consequentemente, os *Prefabs* são especificamente úteis para reutilizar *GameObjects* já criados, por si próprio ou por terceiros.

2.1.5.5 Cenas

Uma Cena, ou *Scene*, corresponde ao espaço onde o desenvolvedor organiza e gerencia todo o conteúdo do jogo ou aplicação, armazenando personagens, obstáculos, decorações e *Interface de Usuário (UI)* (UNITY, 2025).

2.1.5.6 Objeto Programável

Um Objeto Programável ou *ScriptableObject* permitem a definição de classes personalizadas utilizadas como contêineres de dados configuráveis no Editor. Uma das principais vantagens do uso de *ScriptableObjects* é o compartilhamento eficiente de dados entre múltiplos objetos em tempo de execução (UNITY, 2025). Evitando duplicação de objetos e persistência de dados, podendo ser utilizadas para definir configurações globais.

2.1.5.7 OpenXR

Antes do surgimento do *OpenXR*, os desenvolvedores tinham de enfrentar o problema de fragmentação do ecossistema de realidade virtual, onde existem diversas plataformas de *hardware*, cada uma com seu próprio caminho de código. Portanto, o *OpenXR* visa resolver este problema provendo uma API comum, que funciona como uma camada de abstração, para uma ampla variedade de hardwares de RV (KHRONOS, 2023). Portanto, o *plugin OpenXr* disponível na plataforma da *Unity*, é usado para habilitar o uso desta API.

2.1.5.8 XR Interaction Toolkit

O pacote *XR Interaction Toolkit* é um sistema de interação de alto nível, para criar experiências em RV. Ele fornece uma estrutura que permite interações entre 3D e UI a partir de eventos de entrada do *Unity*. O principal do sistema é oferecer os componentes *Interactor* e

Interactable, e um *Interaction Manager* que liga estes dois componentes, além de oferecer também componentes para locomoção (UNITY, 2025).

2.1.6 Perlin Noise

Perlin Noise é um padrão pseudoaleatório de valores flutuantes gerados em um plano 2D. Em vez de conter valores completamente aleatórios, consiste em ondas que aumentam ou diminuem gradualmente ao longo do padrão. O ruído pode ser usado como base para efeitos em texturas, assim como para animações, geração de *heightmaps* e outros afins (UNITY, 2025).



Figura 2.1 – Amostra de *Perlin Noise*, a escala de cinza representa valores entre 0 e 1. Fonte: Unity (2025).

2.2 Trabalhos Relacionados

Esta seção é dedicada a abordar em detalhes os trabalhos que foram usados como inspiração e diretamente relacionados com a idealização e desenvolvimento do projeto.

Bursztyn, D’Antonio e Masters (2025) realizou um estudo intitulado “*Using Immersive Virtual Reality to Measure Strike and Dip and Reach Geological Mapping Concepts*”, sendo uma análise do uso da RV como ferramenta de ensino para conceitos geológicos de mapeamento para estudantes de grau universitário em um ambiente virtual. O termo “*Strike and dip*” ou “rumo e mergulho” se refere à forma padrão de descrever a orientação de formações rochosas.

O público-alvo do estudo consistiu em estudantes de graduação em Geociências na Universidade de Montana, em Oregon, Estados Unidos, inscritos em uma disciplina prática sobre mapeamento geológico. A ideia surgiu como alternativa ao trabalho de campo durante o período de quarentena da COVID-19, quando os geocientistas tentavam ensinar técnicas de mapeamento geológico, anteriormente dadas em campo, agora através do ensino remoto.

Antes da implementação em RV, os autores já haviam conduzido testes com uma versão *desktop* acessada via navegador. A versão fora da RV já havia recebido boas avaliações quanto à proposta pedagógica. Porém, apresentou limitações técnicas, como a dificuldade e a familiaridade dos alunos com o controle via teclado e mouse, baixa qualidade gráfica devido a limitações do dispositivo e da internet dos usuários, e dificuldade de interação com as ferramentas, como a

bússola, resultando em uma diferença significativa em relação à experiência física da prática. Contudo, a evolução do projeto para a RV mostrou avanços expressivos em termos de fidelidade à prática de campo, engajamento e eficácia de aprendizado.

Os estudantes relataram maior sensação de presença, controle mais natural dos instrumentos e facilidade para compreender o mapeamento das formações geológicas. Além disso, os resultados obtidos indicaram que a interação direta com o ambiente tridimensional aumentou consideravelmente o nível de motivação, autoconfiança e compreensão conceitual dos participantes, além de favorecer a inclusão de alunos com restrições de mobilidade física.

Selzer et al. (2025) também realizou avaliação de uma aplicação em RV para o ensino de conceitos geológicos, intitulada “*Immersive Learning: Evaluating Virtual Reality for Geological Compass Education*”. O trabalho, porém, tem intuito mais específico, na reprodução realista e no ensino técnico do uso da *bússola de Brunton*, ferramenta utilizada para medições de rumo e mergulho de rochas.

A principal motivação deste estudo também está na limitação dos métodos tradicionais de ensino geológico, que dependem de poucas saídas de campo e atividades em sala mais simplificadas e com pouca variedade de situações. Para contornar essas limitações, foi desenvolvido um ambiente virtual que simula a região da *Sierra de la Ventana*, na Argentina, abrangendo cerca de 10.000 m² virtuais que permitem exploração livre, medições em múltiplos planos e visualização tridimensional de formações rochosas. O público-alvo incluiu estudantes de graduação do curso de Geologia da Universidade Nacional do Sul, *Bahía Blanca*, Argentina.

Foram realizados testes com grupos de alunos dentro e fora da realidade virtual. Os resultados indicaram que a interação em RV foi positiva, tendo os participantes destacando a semelhança entre RV e o real, o conforto de navegação e a utilidade pedagógica do ambiente. Durante os experimentos comparativos, os alunos que haviam recebido treinamento em RV apresentou desempenho superior em atividades de campo reais, demonstrando maior precisão nas medições, menor tempo de execução e mais autoconfiança no uso da bússola física, em relação ao grupo de controle que não utilizou o sistema. Logo, reafirma-se o uso da RV como ferramenta eficaz para o desenvolvimento de habilidades práticas e para a transferência de conhecimento entre o virtual e o real.

Portanto com os resultados dos estudos de (BURSZTYN; D'ANTONIO; MASTERS, 2025) e (SELZER et al., 2025) pode-se concluir que RV é uma alternativa válida e eficaz para o ensino de geolocalização. Dessa forma, o presente trabalho se inspira nesta abordagem porém se difere nos seguinte pontos:

Trabalho	Este Trabalho	(BURSZTYN; D'ANTONIO; MASTERS, 2025)	(SELZER et al., 2025)
Público-Alvo	Estudantes do Ensino Fundamental I.	Estudantes universitário.	Estudantes universitários.
Objetivo Principal	Ensino de conceitos introdutórios de geolocalização por meio de um jogo sério	Ensino de mapeamento geológico com RV.	Treinamento técnico com a bússola em RV.
Ferramentas Virtuais	Bússola e Mapa, <i>Slider</i> , <i>Audio Player</i>	Bússola.	Bússola.
Ambiente	Ambiente externo procedural.	Ambientes virtuais abstratos (plataformas com rochas).	Ambiente externo baseado um local geológico real.
Integração com Sistema de Gestão de Aprendizagem (LMS)	Integração com <i>Moodle</i> .	Sem integração.	Sem integração.

Tabela 2.1 – Comparativo entre os trabalhos relacionados e a proposta atual. Fonte: O autor (2025).

Os seguintes trabalhos foram diretamente utilizados na idealização e na produção do projeto respectivamente, dando continuidade e aplicação pratica de suas propostas.

Marinho (2024) propôs e avaliou uma metodologia para qualificar professores na idealização de interfaces imersivas educativas em RV. O trabalho delineou um curso de oito módulos, abordando conceitos como Jogos Digitais, Gamificação, Pensamento Computacional, Metodologias Ativas e Realidade Estendida, utilizando como ferramenta para tal, um *Canvas de Design de Jogos Educativos (CDJE)* (Figura 2.2), desenvolvido especificamente para o contexto de RV. Na ocasião, a metodologia foi aplicada junto a professores do ensino básico, que ao final elaboraram quatro propostas de jogos educativos imersivos. Por fim, o *feedback* dos participantes e a análise dos projetos demonstraram que a abordagem é eficiente para auxiliar os professores no processo de design de experiências em RV aplicada à educação.

Canvas de Design de Jogos Educativos Imersivos			Nome do projeto	Versão
1. Problema Quais problemas o jogo visa resolver, como dificuldades de aprendizado em uma determinada matéria e/ou falta de interesse no assunto por parte dos alunos	2. Jogador/Aluno Público alvo do jogo: qual a faixa etária dos jogadores, nível escolar e onde irão jogar (na escola, em casa, etc)	3. Resumo Breve descrição do jogo: início, meio e fim; gênero(s); um ou mais jogadores; dentre outras informações básicas		
4. Missão final O que o aluno deve fazer para ganhar/terminar o jogo. Se quiser, pode definir também outras missões extras, como, por exemplo, ganhar moedas para obter mais pontos	6. Conteúdo Pedagógico Qual disciplina e quais tópicos dela o jogo pretende abordar	7. Objetivos de aprendizagem Seguindo a Taxonomia de Bloom, definir os objetivos pensando em completar a frase: Por meio do jogo, o aluno deverá [VERBO] [CONTEÚDO PEDAGÓGICO]	8. Tarefas específicas Principais tarefas que o jogador deverá concluir durante o jogo, visando alcançar a missão final	9. Feedback e dicas Feedback: informações sobre o jogo para o professor, principalmente sobre o desempenho do aluno Dicas: como o jogo irá informar ao aluno o progresso e desempenho dele, além de ajudar quando ele tiver dificuldades ou errar algo
5. Inspirações O que serviu de inspiração para o jogo? Outros jogos, discussões, estudos? Se forem outros jogos, pense no que eles fizeram de certo e/ou de errado que pode ajudar na idealização do seu jogo			10. Interações com o Espaço Como o jogador poderá interagir com o mundo do jogo: ele pode correr, voar, pegar objetos, guardá-los para depois? Quais são os principais objetos que ele pode interagir? Quais são as regras que regem o mundo do jogo e as possibilidades do jogador?	11. Mundo do jogo Ambientação do jogo: sua história, onde ele se passa (floresta, laboratório, etc) e personagens, como inimigos, aliados, animais, etc. É possível ter mais de um local no jogo


 Desenvolvido para o projeto EducaMinasVR
 Créditos dos ícones: Freepik, Abdul-Aziz e Andrian Prabowo (Flaticon)








Figura 2.2 – Canvas de Design de Jogos Educativos. Fonte: (MARINHO, 2024).

Um **CDJE** é uma ferramenta que reúne, em um único painel, as principais informações sobre um jogo, evitando conceitos mais técnicos que podem gerar confusão. Para tal, a autora sugere o preenchimento destes blocos em ordem numérica. Inicialmente, define-se o *Bloco Base do Jogo*, representado na cor bege, no qual são definidos os elementos centrais do jogo. Em seguida, é preenchido o *Bloco Pedagógico* em roxo, responsável por definir o conteúdo educacional que será abordado. Posteriormente, é definido o *Bloco Aprendizado* em vermelho, que descreve as estratégias utilizadas para transmitir o conteúdo pedagógico ao aluno. Por fim, o *Bloco Imersão* em verde, que define como as interações com o ambiente virtual irão ocorrer.

Em seu trabalho, Marinho (2024), relata os projetos criados por professores do ensino básico seguindo sua metodologia. Definido como *Projeto 3*, um dos professores participantes propôs a ideia de um jogo de geografia no qual os estudantes deveriam localizar tesouros utilizando instrumentos de orientação, como mapas e bússolas. Portanto, esta proposta evidenciava uma dificuldade real no ensino de conceitos básicos de geolocalização.

A partir deste projeto inicial, este trabalho buscou desenvolver e expandir o conceito proposto pelo idealizador do *Projeto 3*, assim como implementá-lo. Para tal, o referido *canvas* foi utilizado como instrumento de apoio à concepção do jogo sério desenvolvido, sendo analisado também sob a perspectiva do desenvolvedor de jogos, observando-se sua aplicabilidade na transição entre a etapa de idealização e a implementação prática do protótipo.

Pimenta (2022) criou uma arquitetura de modularização de componentes reutilizáveis para a criação de jogos sérios usando *Unity*. O trabalho teve como foco a criação de um pacote *Unity*, o *Serious Game Components*, que contém componentes independentes como slides, questionários e

reprodutores de áudio integrados a um sistema [LMS](#). O desenvolvedor consegue arrastar *Prefabs* para dentro da cena e, via API, consegue se comunicar com alguma plataforma web para buscar os conteúdos e utilizá-los dentro do jogo para populá-los.

Neste projeto, esses componentes foram inicialmente utilizados na fase de prototipação do jogo, como mostra a Figura 2.3, servindo de base para a implementação das funcionalidades pedagógicas integradas ao ambiente virtual.

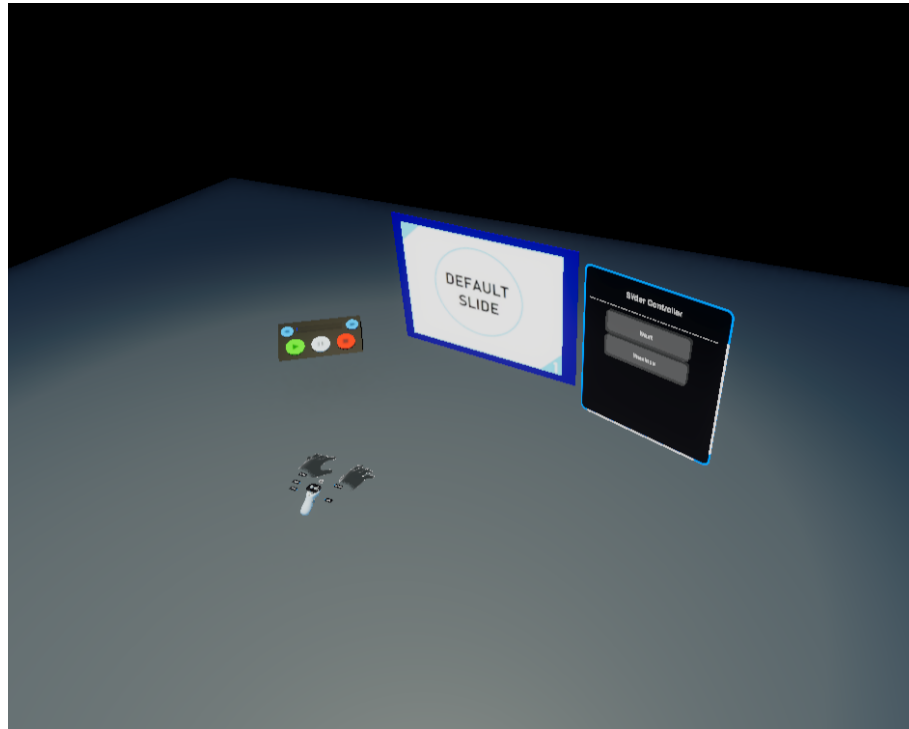


Figura 2.3 – Utilização dos componentes de slides e do reprodutor de áudio do pacote proposto por (PIMENTA, 2022) durante a fase de prototipação. Fonte: Autor (2026).

3 Desenvolvimento

Este capítulo aborda de forma detalhada as etapas de concepção e implementação do projeto. Para tanto, na Seção 3.1, é abordado o uso da metodologia de *design* de interfaces imersivas educativas em realidade virtual, com uso do CDJE como ferramenta proposta por Marinho (2024). E, na Seção 3.2, é relatada em detalhes a implementação do projeto e decisões de *design* feitas antes e durante o desenvolvimento.

3.1 Idealização e Metodologia

Esta seção apresenta o processo de idealização do jogo sério proposto, utilizando como base a metodologia CDJE desenvolvida por Marinho (2024), apresentada na Seção 2.2. Para isso, seguiu-se a recomendação de uso do *canvas* da metodologia, tomando como ponto de partida o conceito inicial concebido no *Projeto 3*, porém realizando uma nova idealização da proposta a partir de seus elementos fundamentais.

A Figura 3.1 apresenta a versão final do CDJE preenchido para este projeto, sintetizando, em blocos, os principais componentes definidos durante a fase de planejamento. Tais fases de planejamento serão abordadas a seguir.

Canvas de Design de Jogos Educativos Imersivos			Nome do projeto	Can You Find It?	Versão
1. Problema Dificuldade no aprendizado de geolocalização por parte dos estudantes e como operar ferramentas como mapas e bússolas.	2. Jogador/Aluno Público alvo, estudantes do ensino fundamental I, de 9 a 10 anos de idade.	3. Resumo Estudante explora um ambiente virtual em busca de um tesouro, e por meio de mapas e bússola se identifica no terreno e ir atrás do objetivo. O estudante poderá também ter uma breve aula dos conceitos básicos de geolocalização antes de iniciar a tarefa.	8. Tarefas específicas - Assistir material disponibilizado pelo professor. - Interagir com o mapa e bússola para localizar os objetivos e a si mesmo. - Achar o tesouro via exploração e aplicação dos conceitos de geolocalização.		9. Feedback e dicas - Um contador de tempo apresentará ao jogador o progresso de tempo. - O jogador receberá seu desempenho, assim como a notificação de recorde se alcançado. - Um radar que informe a aproximação do tesouro
4. Missão final Coletar todos o tesouro no ambiente virtual em menor tempo possível.	6. Conteúdo Pedagógico Interpretação e uso de mapas e bússolas. Ensino de conceitos como pontos cardeais.	7. Objetivos de aprendizagem Entender conceito de pontos cardeais, uso de mapas e bússola. Percepção de referências geográficas como relevo, objetos, e posição do sol para orientação. Analisar as informações dos objetos disponíveis. Aplicar os conhecimentos de geolocalização e informações dos objetos para se locomover.	10. Interações com o Espaço Jogador: Andar, olhar ao redor, segurar itens, se locomover. Mapa e bússola: Segurá-los e rotacioná-los livremente. Slides: voltar e passar para o próximo. Player de Áudio: Play, pause, voltar e passar.		11. Mundo do jogo Ambiente interno para aprendizado, e ambientes externos como florestas para caça dos itens de interesse.
5. Inspirações O jogo Sea of Thieves, e Green Hell. Os dois apresentando mecânicas de exploração via mapa e bússolas.					

Desenvolvido para o projeto EducaMinasVR
 Créditos dos ícones: Freepik, Abdul-Aziz e Andrian Prabowo (Flaticon)

opCoders JOURNAL XR4GOOD UFPA FAPEMIG

Figura 3.1 – Versão final do CDJE preenchido para este projeto. Fonte: do autor (2026).

No **Bloco base do jogo** foi registrado inicialmente o problema educacional identificado, relacionado à dificuldade de estudantes em compreender conceitos de geolocalização e utilizar

instrumentos de orientação. Também foram definidos o público-alvo e o conceito geral da experiência, estabelecendo-se como foco estudantes do ensino fundamental I, com faixa etária entre 9 e 10 anos. A partir dessas definições, estruturou-se a proposta de um jogo no qual o estudante explora um ambiente virtual tridimensional em busca de tesouros utilizando mapa e bússola como ferramentas de navegação.

Ainda foram definidos os elementos que descrevem o funcionamento geral da experiência. O resumo do jogo estabelece que o estudante deve explorar o ambiente virtual, utilizando instrumentos de orientação, para localizar os tesouros distribuídos no cenário. A qualquer momento da atividade principal, o jogador tem acesso a um conteúdo introdutório e personalizado pelo professor sobre os conceitos fundamentais de geolocalização. A missão final consiste em coletar todos os tesouros no menor tempo possível, incentivando a aplicação prática dos conhecimentos apresentados. Nesse mesmo bloco, também foram registradas as principais inspirações utilizadas na concepção do jogo, incluindo títulos como *Sea of Thieves* e *Green Hell*, que apresentam mecânicas de exploração baseadas na interpretação de mapas e na orientação espacial.

No **Bloco Pedagógico** foram definidos os conceitos que seriam abordados no jogo, incluindo a interpretação de mapas, o uso de bússolas e a compreensão dos pontos cardeais. A proposta pedagógica busca desenvolver nos estudantes habilidades de orientação espacial, incentivando a identificação de referências geográficas presentes no ambiente virtual, como relevo e objetos de referência, que auxiliam no processo de navegação.

Neste mesmo bloco também foram definidos os objetivos de aprendizagem do jogo. Entre eles estão a compreensão dos conceitos fundamentais de orientação geográfica, a capacidade de analisar informações presentes nos instrumentos de navegação e a aplicação desses conhecimentos durante a exploração do ambiente virtual. Dessa forma, o estudante é incentivado a interpretar os elementos disponíveis no jogo e utilizá-los para determinar sua posição e deslocar-se corretamente no espaço.

Na sequência, foi definido o **Bloco Aprendizado**, no qual foram estabelecidas as tarefas que o estudante deveria realizar durante a atividade. Estas tarefas consistem em interpretar informações presentes no mapa e na bússola para localizar os objetivos e orientar sua movimentação no cenário. Além disso, o jogo permite que o estudante acesse materiais explicativos disponibilizados e selecionados previamente pelo professor, oferecendo suporte adicional para a compreensão dos conceitos abordados.

O Bloco Aprendizado também define os mecanismos de *feedback* oferecidos ao jogador durante a experiência. Entre eles, um contador de tempo permite acompanhar o progresso da partida, além de o registro do tempo final gasto ao localizar o tesouro, possibilitando o armazenamento de recordes e incentivando novas tentativas por parte dos alunos. Além disso, foi incluído um radar como configuração opcional, capaz de indicar a aproximação do jogador em relação ao tesouro, auxiliando no processo de orientação dentro do ambiente virtual.

Por fim, foi estruturado o **Bloco Imersão**, onde define que o jogador pode se locomover pelo cenário, observar o ambiente ao seu redor e manipular objetos relevantes para a atividade, como o mapa e a bússola. O sistema também inclui elementos de interface voltados à apresentação de conteúdos educacionais, como slides e reprodutores de áudio utilizados durante a etapa de aprendizagem.

O ambiente do jogo foi concebido para integrar espaços de aprendizagem e exploração. Inicialmente, o estudante tem acesso a um ambiente interno focado à apresentação do conteúdo educacional e configurações da partida. Em seguida, o jogador é direcionado para os ambientes externos, como áreas de floresta, onde ocorre a atividade de exploração e busca pelos tesouros. Essa estrutura permite estabelecer uma transição entre o momento de aprendizagem conceitual e a aplicação prática dos conhecimentos dentro da experiência imersiva.

3.2 Implementação e Decisões de Design

Esta seção é dedicada a detalhar o processo de desenvolvimento prático do projeto, seguindo o planejamento descrito na Seção 3.1, que definiu as mecânicas centrais e interações previstas. Além disto, esta seção também apresenta registro histórico das reformulações realizadas a partir do desenvolvimento do projeto desde um protótipo inicial até sua versão final, com base em testes de usabilidade e críticas do próprio autor durante o desenvolvimento. Isto se deve com finalidade para registro de caso de uso do [CDJE](#) como ferramenta viável pela perspectiva de desenvolvedor. Tais resultados serão abordados no Capítulo 4, a seguir.

De forma geral, o jogo implementado permite ao jogador explorar uma ilha gerada de forma procedural, com o objetivo de localizar um tesouro, utilizando conceitos de orientação espacial. Durante a atividade, o jogador utiliza instrumentos virtuais de navegação, como mapa, bússola e radar, além de poder acessar conteúdos educacionais integrados ao *Moodle*, como *slides* e áudios. Esses elementos compõem a base das mecânicas de exploração e aprendizagem que orientaram as decisões de implementação descritas nas subseções seguintes.

Todos os *scripts* e o código-fonte completo do projeto estão disponíveis em um repositório público no *GitHub* ([RIBEIRO, 2026](#)).

3.2.1 Configuração do Ambiente de Desenvolvimento

Para o desenvolvimento do jogo, foi utilizado o motor *Unity*, versões 6 (6000.0.30f1) e (6000.2.7f2), configurado com uma predefinição inicial para realidade virtual, que inclui pacotes como *OpenXR* e *XR Interaction Toolkit*, essenciais para garantir a compatibilidade com dispositivos de realidade virtual.

Além dos pacotes iniciais da predefinição VR, o principal pacote adicionado ao projeto foi o *Serious Game Components* que adiciona a integração de uma *LMS* e para inserir conteúdo

aos seus *Prefabs*, como *Slider eplayer* de áudio.

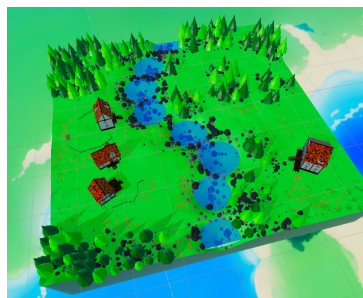
Para o desenvolvimento e teste de usabilidade final do jogo, foram utilizados os equipamentos *Meta Quest 2*, *Meta Quest 3* e *Meta Quest Pro*.

3.2.2 Ambiente Virtual

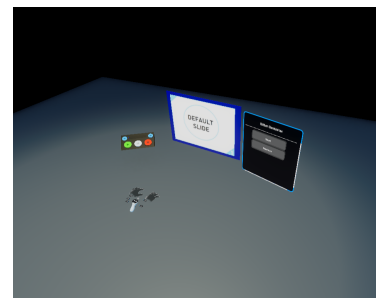
Inicialmente, foram criadas três principais cenas para o jogo: o ambiente externo de floresta (Figura 3.2b) e o ambiente interno para interação com os *Slides* e áudio (Figura 3.2c), conforme definido no **Bloco Imersão** (Seção 3.1), o qual define como e quais serão os ambientes virtuais do jogo, e, adicionalmente, um ambiente exclusivo para o menu inicial (Figura 3.2a).



(a) Cena inicial do menu do jogo.



(b) Ambiente externo de floresta explorável pelo jogador.



(c) Ambiente interno voltado ao aprendizado e interação.

Figura 3.2 – Cenas do ambiente do jogo. Fonte: do autor (2025).

A cena de ambiente externo na floresta (Figura 3.2b) utilizava como base do terreno um *prefab* obtido no site *Sketchfab*. Já localização dos itens de interesse, que no caso até então: uma chave e um baú, eram aleatoriamente instanciados em uma das áreas predeterminadas neste terreno, como mostra a Figura 3.3. Em testes de usabilidade, foi constatado um fator de previsibilidade, minimizando o esforço cognitivo em aplicar os conceitos de geolocalização através da tentativa de decorar o cenário.

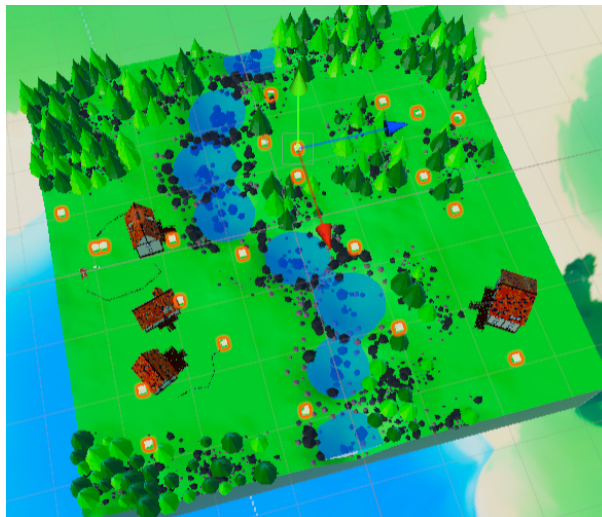


Figura 3.3 – Áreas predeterminadas para instanciar objetos de interesse. Fonte: do autor (2025).

Portanto, houve uma reformulação do terreno, passando a ser gerado proceduralmente usando *Perlin Noise*, descrito na Seção 2.1.6, para gerar uma ilha (Figura 3.4), com relevos, vegetações e estruturas geradas através de uma semente aleatória, garantindo assim maior fator de rejogabilidade e reduzindo a possibilidade do usuário decorar o terreno, assim, forçando-o a aplicar os conceitos aprendidos.

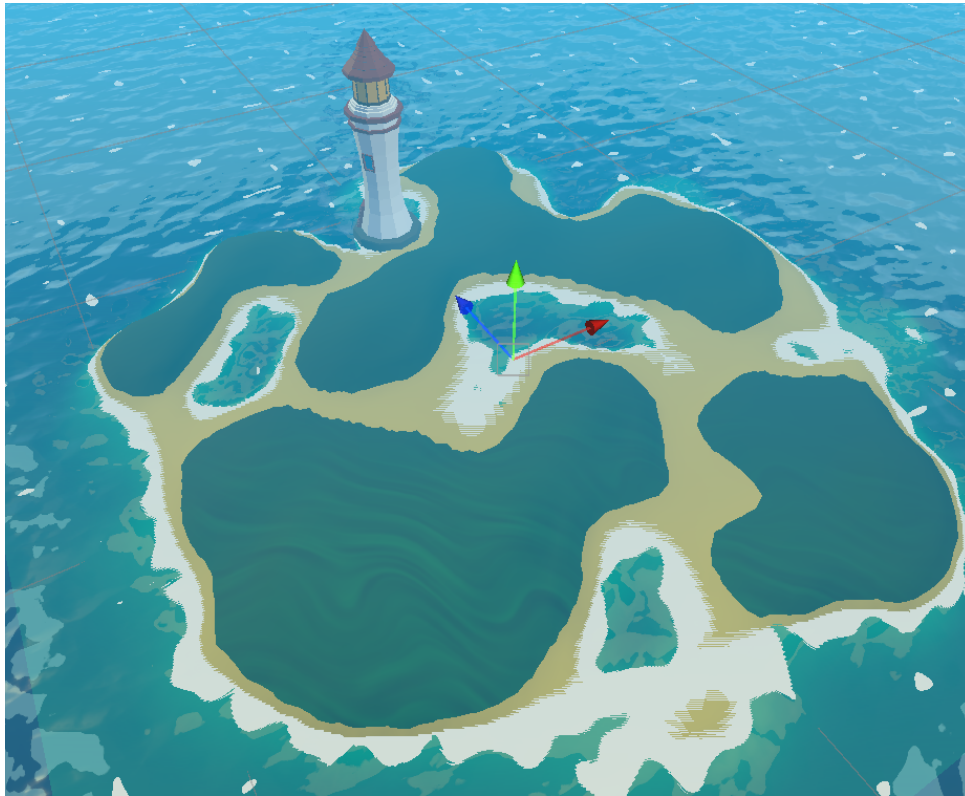


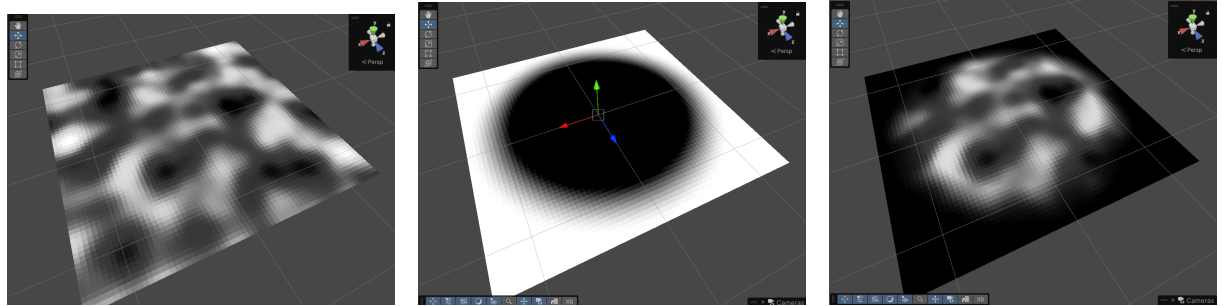
Figura 3.4 – Terreno gerado proceduralmente. Fonte: do autor (2026).

O processo de geração deste terreno consiste em 5 etapas. Criação de um mapa de ruído (Figura 2.1), geração do terreno, da vegetação, de objetos de referência e do Tesouro.

A primeira etapa, que consiste na criação do mapa de ruído, serve de base para todo o restante da geração do terreno. Para isso, foi implementada a função *GenerateNoiseMap*, responsável por produzir um mapa bidimensional de valores contínuos a partir do algoritmo de *Perlin Noise* (Seção 2.1.6) somado a várias camadas que refinam o resultado, gerando uma matriz de pontos flutuantes de 0 a 1 (Figura 3.5a).

Uma vez pronta a matriz, ela é utilizada em combinação com uma máscara de gradiente radial, na qual os valores decaem de 1 para 0 no sentido do centro (Figura 3.5b), formando assim um formato de “ilha” (Figura 3.5c).

Na segunda etapa, após a geração do mapa de ruído, são definidas as regiões do relevo. Para cada região, é atribuída uma altura mínima para cada porção de terreno baseada em valores de 0 a 1, um nome e uma cor, que serão usados para determinar instâncias de vegetação, objetos e texturas do terreno (Figura 3.6).

(a) Mapa de ruído *Perlin*.

(b) Mapa de gradiente radial.

(c) Combinação do mapa de ruído com o gradiente radial, resultando em um mapa de alturas com formato de radial.

Figura 3.5 – Etapas da geração do mapa de alturas, dispostos como textura em um *GameObject*, onde tons escuros representam valores próximos a 0 e tons claros valores próximos a 1. Fonte: do autor (2026).

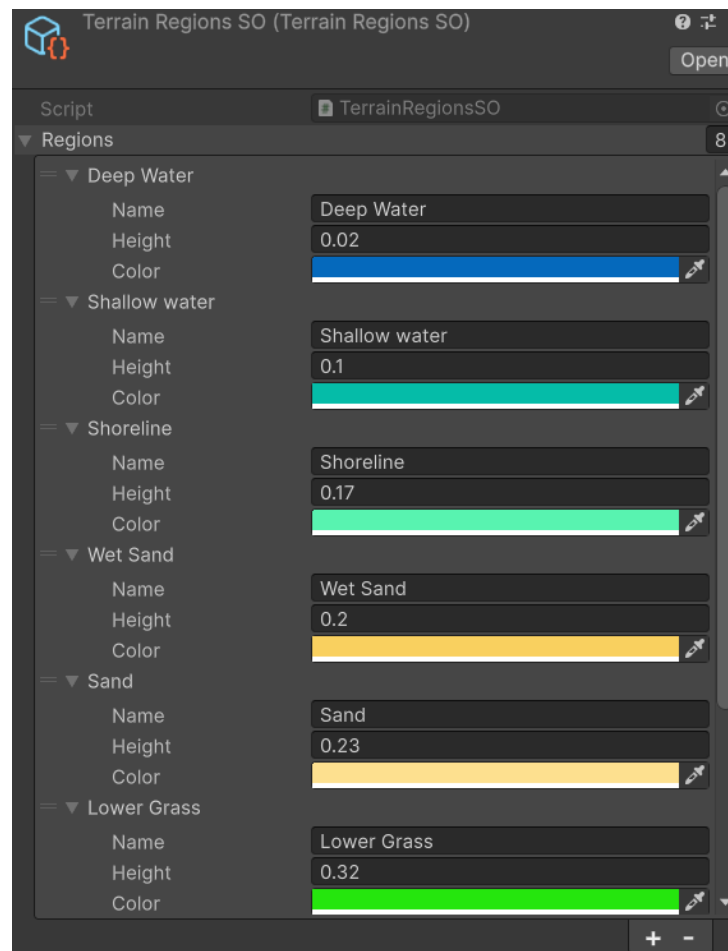


Figura 3.6 – Objeto Programável responsável por armazenar informações sobre as regiões. Fonte: do autor (2026).

Portanto, inicia-se a geração da malha tridimensional do terreno. O mapa de alturas é interpretado como uma grade regular de pontos, em que cada valor do mapa de ruído define a coordenada vertical de um vértice no espaço tridimensional. Esses vértices são distribuídos

uniformemente no plano horizontal, formando uma estrutura retangular, composta por triângulos que representam a base do terreno (Figura 3.7).

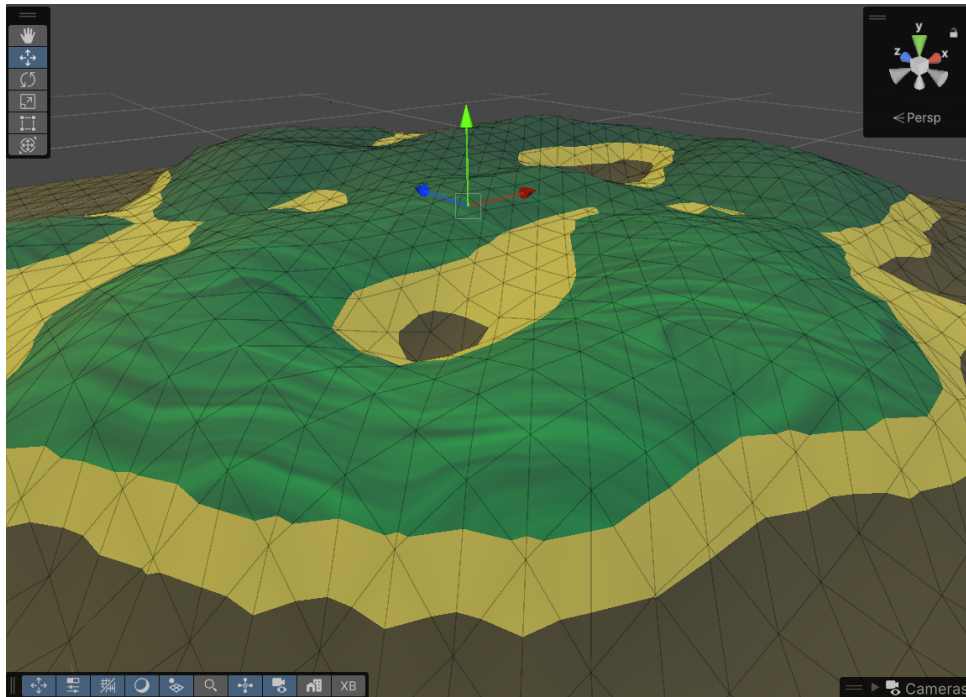


Figura 3.7 – Malha do terreno com altura dos vértices baseada no mapa de ruído. Fonte: do autor (2026).

E por fim, todas as demais informações do terreno são calculadas e armazenadas em um objeto da classe *MapData* como mostrado na Figura 3.8, responsável por centralizar os dados estruturais e procedurais do mapa, incluindo o mapa de ruído, dimensões, curva e multiplicador de altura, mapa de cores do mapa, além das posições e rotações de vegetações, ponto de inicialização do jogador e farol, fornecendo ainda métodos para interpolação de altura e conversão entre coordenadas do mapa de ruído e coordenadas no espaço global. Esta classe também calcula e armazena a posição do tesouro, que agora é o único item de interesse.

3.2.3 Objetos Interativos

Os objetos interativos definidos no **Bloco Imersão**, Seção 3.1, sendo eles mapa, bússola, pá, slides e *player* de áudio, foram implementados de forma a cumprir as tarefas e fornecer os *feedbacks* estabelecidos durante a fase de idealização no **Bloco Aprendizado**.

3.2.3.1 Decisões de *Design* e Reformulação Arquitetural

Na primeira versão do projeto, os objetos interativos previstos no CDJE estavam distribuídos fisicamente no ambiente virtual como elementos independentes entre si, de livre manipulação e de física própria aplicada. Entretanto, durante testes de usabilidade realizados ao longo do desenvolvimento do projeto, foram identificadas limitações associadas a essas mecânicas, que serão discutidas neste capítulo, o que motivou a reformulação desses objetos.

```

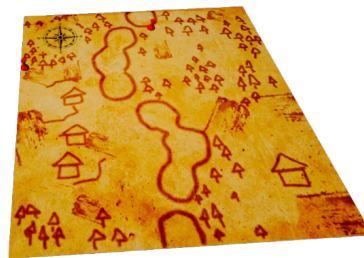
public class MapData{
    public float[,] NoiseMap;
    public int Height;
    public int Width;
    public List<Matrix4x4> GrassPostions;
    public List<Vector2Int> TreasurePositions;
    public List<Vector3> TreesPositions;
    public Vector3 PlayerSpawn = new Vector3(0, 100, 0);
    public Texture2D MiniMapTexture;
    public TerrainRegionsSO RegionsSO;
    public Color[] ColorMap;
    public float heightMultiplier;
    public AnimationCurve heightCurve;
    public int terrainSize;
    public float terrainScale = 10f;
    public Vector3 LightHousePosition;
    public float[,] FalloutNoiseMap;
}

```

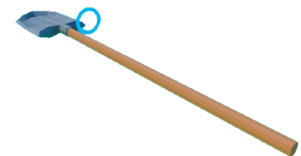
Figura 3.8 – Classe *MapData* e seus atributos. Fonte: do autor (2026).



(a) Modelo 3D da bússola.



(b) Modelo 3D do mapa.



(c) Modelo 3D da pá.

Figura 3.9 – Objetos interativos do protótipo inicial. Fonte: do autor (2026).

A bússola, o mapa e a pá (Figura 3.9) eram utilizados como instrumentos físicos de exploração, podendo ser agarrados e atracados aos controles do personagem, assim como em posições fixas ao corpo do jogador, para carregar os itens sem a necessidade de permanecer pressionando o botão de agarre. A pá, em específico, tinha a função de revelar o tesouro ao tocar nas áreas de interesse predeterminadas (Figura 3.3).

Esse modelo visava privilegiar a imersão por meio da interação direta com as múltiplas ferramentas, trazendo maior engajamento na atividade. Porém, durante as avaliações exploratórias, observou-se que a necessidade constante de manipular, física e individualmente, cada objeto introduzia uma carga cognitiva adicional não prevista, criando barreiras ao aprendizado do objetivo central do jogo sério: o aprendizado de conceitos de geolocalização. Isso fazia com que os jogadores se esforçassem mais para aprender as mecânicas de jogo, como agarrar, manipular objetos e procurar objetos constantemente perdidos no espaço virtual, do que para aplicar os princípios de orientação espacial propostos pela atividade.

Os objetos interativos com recursos pedagógicos, o *slider* e *audio player* foram inicialmente implementados a partir de *prefabs* disponibilizados pelo pacote *SeriousGameComponents*. Esses componentes eram instanciados diretamente em uma cena separada (Figura 3.2c) e mantinham sua estrutura funcional original do pacote proveniente.

Embora essa abordagem tenha facilitado a integração inicial das funcionalidades pedagógicas, a utilização direta dos *prefabs* resultava em uma experiência fragmentada e de difícil inserção no *loop* de *gameplay*. Como consequência, o jogador precisava alternar constantemente entre diferentes pontos do ambiente para acessar conteúdos complementares.

A partir desses diagnósticos, tornou-se evidente a necessidade de reorganizar as mecânicas envolvendo os objetos interativos, com o objetivo de minimizar barreiras cognitivas adicionais que desviassem o foco do entendimento do conteúdo pedagógico.

Dessa forma, optou-se por centralizar as funcionalidades, inicialmente idealizadas no CDJE, em dois principais objetos: um *tablet* e um relógio inteligente, ambos fixos ao modelo de mãos do jogador. Assim, as mecânicas deixaram de ser gerenciadas por múltiplos objetos livres e passaram a ser gerenciadas por telas integradas a estes objetos fixos nas mãos do jogador, como é exemplificado pela Figura 3.10.



Figura 3.10 – Modelos 3D do *tablet* e do relógio inteligente. Fonte: do autor (2026).

O *tablet* opera como núcleo de acesso às funcionalidades pedagógicas de integração com o *Moodle* e às configurações da partida, estruturado em diferentes telas navegáveis por meio da interação direta ao toque, com os ícones exibidos na interface. Em determinadas situações, como o menu de início de partida ou o acesso a conteúdos instrucionais, o ambiente virtual é temporariamente obscurecido por uma esfera escura ao redor do jogador, reduzindo estímulos

externos e favorecendo a concentração no conteúdo exibido.

De maneira geral, o *tablet* contém as telas:

- Tela inicial com opções de início de partida e configurações;
- Menu acessível durante o *gameplay*;
- Visualização do mapa da ilha;
- Exibição de slides didáticos;
- Reprodução de áudios explicativos;
- Tela de reinicialização da partida;
- Painel de configurações (tempo, teleporte e ativação do radar).

O relógio inteligente, por sua vez, foi concebido como um dispositivo de acesso rápido durante a atividade de exploração. Suas telas podem ser alternadas por meio do gatilho frontal do controle de realidade virtual e incluem:

- Contador regressivo de tempo da partida;
- Bússola minimalista com indicação do Norte;
- Radar opcional para auxílio de localização.

A pá, por sua vez, foi completamente removida do jogo, substituída pela simples ação de se aproximar do tesouro.

Desta forma, o foco deixou de estar na manipulação física constante de múltiplos objetos e passou a privilegiar a clareza do fluxo de aprendizagem. A centralização destas funcionalidades em dispositivos fixos visa reduzir as barreiras ao objetivo pedagógico, uma vez que não há mais interações físicas, como a perda ou queda de objetos no ambiente virtual. Além disso, os objetos e modelos de mãos visam manter a imersão, porém com grau de familiaridade maior por parte dos jogadores.

3.2.3.2 Bússola

A bússola tem como objetivo permitir que o jogador consiga identificar no ambiente virtual o “Norte” e, por conseguinte, se localizar e direcionar-se pelo mapa para alcançar os itens de interesse.



Figura 3.11 – Modelo 3D da tela de bússola. Fonte: do autor (2026).

Inserida como parte do fluxo de telas do relógio, a interface de bússola apresentada na Figura 3.11 é composta por dois *gameObjects*: um elemento gráfico que representa a agulha direcional e um objeto textual contendo a letra “N”, responsável por indicar o Norte geográfico. A agulha é dividida nas cores vermelha e preta, seguindo a convenção tradicional em que a extremidade vermelha aponta para o Norte, enquanto o marcador “N”, também destacado em vermelho, atua como referência fixa.

```
public Transform north;

Mensagem do Unity | 0 referências
void Update()
{
    if (north != null && transform.parent != null)
    {
        Vector3 relativeTarget = transform.parent.InverseTransformPoint(north.position);

        float needleRotation = Mathf.Atan2(relativeTarget.y, relativeTarget.z) * Mathf.Rad2Deg;

        transform.localRotation = Quaternion.Euler(-needleRotation, 0, 0);
    }
}
```

Figura 3.12 – Script *NeedleController*. Fonte: do autor (2026).

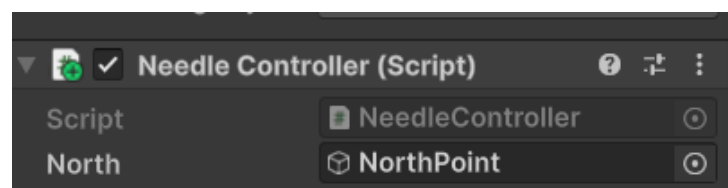


Figura 3.13 – Script *NeedleController* no *inspector*. Fonte: do autor (2025).

O *GameObject* correspondente à agulha da bússola (Figura 3.11) é controlado pelo *script*

NeedleController.cs (Figura 3.12). Um *GameObject* qualquer deve ser atribuído pelo *inspector* ao campo *North* (Figura 3.13) para fazer a referência geográfica.

Para calcular a orientação da agulha, a posição do objeto *north* é convertida das coordenadas globais da cena para as coordenadas relativas à agulha por meio da função *InverseTransformPoint()*. Essa conversão é necessária para garantir que a rotação aplicada à agulha seja consistente, mesmo que o objeto da bússola esteja rotacionando no ambiente virtual.

Em seguida, o ângulo do objeto *north* e o eixo de posição da agulha é obtido com a função *Mathf.Atan2()*, resultando em um valor radiano que é convertido para graus através da constante *Mathf.Rad2Deg*, armazenando na variável *needleRotation*, representando assim a rotação necessária para alinhar a agulha ao objeto *north*. Por fim, o valor de *needleRotation* é aplicado ao *transform* do objeto agulha.

3.2.3.3 Mapa

A funcionalidade do mapa é representar o terreno atual do jogo, e marcar o tesouro a cada de forma aleatória a cada nova partida.



Figura 3.14 – Modelo 3D da tela do mapa. Fonte: do autor (2026).

O mapa possui uma imagem que consiste em uma representação *pixelizada* do terreno atual. As letras N, E, S e W indicam, respectivamente, Norte, Leste, Sul e Oeste, funcionando como referências cardinais fixas na interface. A variação de cores está diretamente associada à altitude e ao tipo de superfície do terreno: tons de azul representam áreas com presença de água, sendo os mais claros correspondentes a regiões rasas e os mais escuros a regiões mais profundas; tons de verde indicam áreas de vegetação, nas quais cores mais claras representam regiões de

menor elevação e tons progressivamente mais escuros indicam áreas mais altas; o verde escuro intenso é utilizado para representar árvores, enquanto o bege claro identifica faixas de areia.

Além disso, dois elementos visuais desempenham papel central na orientação do jogador: o farol, com seu modelo 3D visível na Figura 3.4, que atua como marco fixo de grande escala de referência espacial, e o símbolo vermelho “X”, que indica a localização do tesouro.

Para a geração da textura do mapa são utilizadas as regiões previamente definidas no objeto programável (Figura 3.6), armazenadas na instância da classe *MapData* (Figura 3.8) além do mesmo mapa de ruído gerado proceduralmente na criação do terreno. A função *UpdateColorMap* percorre todo o mapa de ruído, analisando cada valor de altura individualmente e comparando-o com os limites estabelecidos para cada região; a partir dessa verificação, é atribuída ao vetor de cores a tonalidade correspondente à faixa de altura identificada, construindo assim um vetor de cores correspondente ao terreno, que será aplicado como textura do *GameObject* responsável pelo mapa.

```
for (int i = -size; i <= size; i++)
{
    int x1 = pos.x + i;
    int z1 = pos.y + i;

    int x2 = pos.x + i;
    int z2 = pos.y - i;

    if (x1 >= 0 && x1 < Width && z1 >= 0 && z1 < Height)
        ColorMap[z1 * Width + x1] = Color.red;

    if (x2 >= 0 && x2 < Width && z2 >= 0 && z2 < Height)
        ColorMap[z2 * Width + x2] = Color.red;
}
```

Figura 3.15 – Correção de cores para marcação do X. Fonte: do autor (2026)

Para a representação em específico do farol e do símbolo “X” marcando o tesouro, o mapa de cores modificado durante os cálculos de posições realizado na classe *MapData* (Figura 3.8). No caso do marcador “X” como exemplo na Figura 3.15 são alterados todos os *pixels* e suas diagonais, enquanto para o farol é aplicada uma variação de cores em uma região circular ao redor do ponto escolhido. Dessa forma, elementos especiais são incorporados diretamente na textura por meio da manipulação direta do vetor de cores.

3.2.3.4 Temporizador

Como definido no **Bloco Aprendizado 3.1**, um dos *feedbacks* definidos é de apresentar ao jogador o progresso de tempo da partida, isto por sua vez é feito pela tela de temporizador.



Figura 3.16 – Modelo 3D da tela do Temporizador. Fonte: do autor (2026).

O tempo da partida pode ser ajustado antes de iniciá-la através do menu de opções, como mostra a Figura 3.17.

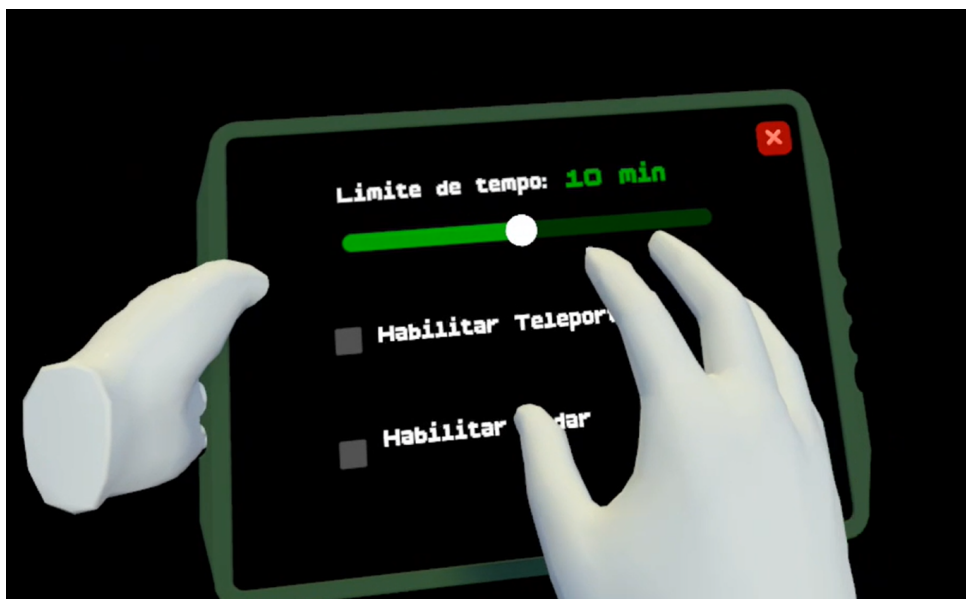


Figura 3.17 – Tela de configurações. Fonte: do autor (2026).

Uma vez definido o tempo da partida, ela é iniciada e o temporizador inicia a contagem regressiva, controlado pelo *script MatchController*.

3.2.3.5 Radar

O radar é responsável por auxiliar o jogador a identificar o local do tesouro, sendo uma tela opcional do relógio.



Figura 3.18 – Modelo 3D da tela do Radar. Fonte: do autor (2026).

O radar pode ser ativado na tela de configuração antes do início da partida (Figura 3.18). Seu funcionamento é baseado em um ciclo de animação no qual a pulsação e a cor variam de acordo com a distância entre o jogador e o tesouro. Esse comportamento é controlado pela variável *Distance*, adicionada ao *Animator* dessa interface. O valor dessa variável é atualizado a cada quadro na função *Update* do script *WatchController*, conforme apresentado na Figura 3.19.

```
public void CalculaDistanceFromChest()
{
    Vector3 aPos = treasureTransform.position;
    Vector3 bPos = playerTransform.position;

    aPos.y = 0f;
    bPos.y = 0f;

    playerDistance = Vector3.Distance(aPos, bPos);
}

Mensagem do Unity | 0 referências
void Update()
{
    if( mc.GetGameState() == MatchController.GameState.Play )
    {
        CalculaDistanceFromChest();
        radarAnim.SetFloat("Distance", playerDistance);
    }
}
```

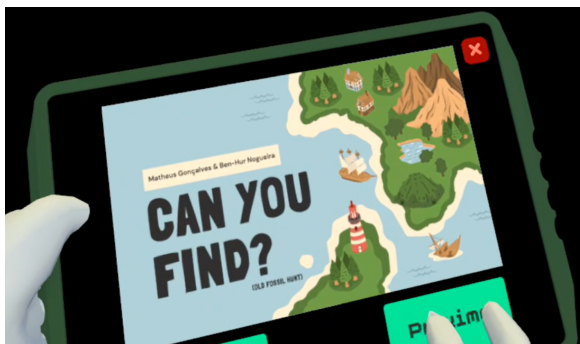
Figura 3.19 – Cálculo de distância realizado no script *WatchController*. Fonte: do autor (2026).

O *script* oferece ainda funções auxiliares para pausar, resumir e começar o temporizador, e, em caso de o tempo acabar antes de achar o tesouro, a partida é automaticamente reiniciada.

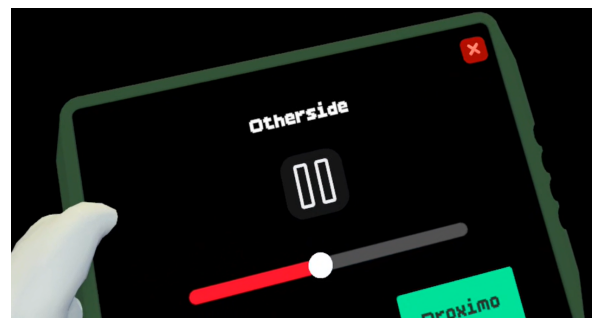
3.2.3.6 Slides e Player de Áudio

Além do mapa e das configurações da partida, o *tablet* também concentra os recursos educacionais utilizados durante a experiência do jogo. Sendo eles os *slides* e o *player* de áudio, que permitem ao professor disponibilizar conteúdos complementares e personalizados diretamente no ambiente virtual, via plataforma *Moodle*.

Conforme abordado na subseção anterior, na versão inicial do projeto, esses recursos eram acessados por meio de *prefabs* independentes do pacote *SeriousGameComponents*, posicionados em uma cena específica do ambiente virtual. Entretanto, essa abordagem, apesar de prática para prototipagem do jogo, dificultava a integração com o fluxo da atividade, exigindo que o jogador trocasse de cena para acesso aos materiais, cada um instanciado de forma independente. A partir da reorganização do projeto, essas funcionalidades foram incorporadas diretamente à interface do *tablet* como mostrado na Figura 3.20.



(a) Modelo 3D do *slider*. Fonte: do autor (2026).



(b) Modelo 3D do *player* de áudio. Fonte: do autor (2026).

Figura 3.20 – Modelos dos componentes de recursos educacionais mostrando conteúdos baixados do *Moodle*. Fonte: do autor (2026).

A funcionalidade de *slides* (Figura 3.20a) consiste em uma sequência de imagens exibidas na interface do *tablet*, permitindo ao jogador avançar ou retroceder entre as imagens por meio dos botões na tela do dispositivo. Já o *player* de áudio (Figura 3.20b) possibilita a reprodução de faixas sonoras. A interação também ocorre por meio de botões na tela do *tablet*, permitindo navegação entre faixas, além das funções de *play* e *pause*. Além disso, é permitido que o áudio continue sendo reproduzido em segundo plano mesmo quando o jogador sai da tela do *player* de áudio, possibilitando que o jogador escute o conteúdo enquanto explore o ambiente virtual.

Para o gerenciamento e armazenamento dos slides e áudios, ainda foi mantido o uso do pacote *SeriousGameComponents*, mais especificamente a classe *LmsLoader*. No entanto, o processo de download dos arquivos não é realizado diretamente por esse componente. Para essa finalidade, foi desenvolvido um *script* próprio denominado *MyLoader*, cuja implementação foi

inspirada na lógica de funcionamento do *LmsLoader*, mas adaptada às necessidades específicas deste trabalho, uma vez que utiliza uma API com interações ligeiramente diferentes da padronizada por este pacote¹.

Dessa forma, o *MyLoader* realiza as requisições à API, identifica os módulos corretos dentro da seção do curso e efetua o download dos arquivos correspondentes aos *slides* e aos áudios. Após essa etapa, os conteúdos são inseridos nas estruturas internas do *LmsLoader*, preservando a organização proposta pelo pacote e evitando a criação de um sistema paralelo de armazenamento e gerenciamento, permitindo possível integração com seus componentes futuramente.

Após o download dos arquivos, os dados são associados a um identificador padrão (*DEFAULT_IDENTIFIER*) e inseridos nos dicionários internos do *LmsLoader*. A Figura 3.21 apresenta um exemplo desse processo para o caso dos *slides*, onde o método responsável atualiza o sistema com os conteúdos obtidos.

```
1 referência
private void InjetarSlidesNoLms(List<Texture2D> slides)
{
    LmsLoader.title = slideTitle;
    LmsLoader.description = slideDescription;

    string id = LmsLoader.DEFAULT_IDENTIFIER;

    if (LmsLoader.slides.ContainsKey(id))
        LmsLoader.slides[id] = slides;
    else
        LmsLoader.slides.Add(id, slides);

    SetLmsLoaderFlag("slidesFetched", true);
    Debug.Log($"Slides Injetados: {slides.Count}");
}
```

Figura 3.21 – Injeção dos slides no *LmsLoader* após o download do conteúdo. Fonte: do autor (2026)

Em seguida, no componente responsável pela exibição, os slides são recuperados diretamente do *LmsLoader* e armazenados em uma lista local para controle de navegação, como ilustrado na Figura 3.22. A partir dessa lista, o sistema controla o índice atual e atualiza a textura exibida no elemento de interface.

¹ Para acessar os conteúdos do ambiente *Moodle*, foi utilizada uma API REST desenvolvida em um projeto de mestrado em andamento. Essa API é responsável por intermediar a comunicação entre a aplicação desenvolvida neste trabalho e a plataforma Moodle, permitindo autenticação, listagem de conteúdos e download de arquivos utilizados nos slides e áudios. Como o projeto ainda se encontra em fase de desenvolvimento e não possui publicação formal até o momento, sua descrição detalhada não está disponível publicamente.

```

private void InicializarSlides()
{
    if (LmsLoader.slides != null && LmsLoader.slides.ContainsKey(slideIdentifier))
    {
        slidesAtuais = LmsLoader.slides[slideIdentifier];

        if (slidesAtuais.Count > 0)
        {
            slidesCarregados = true;
            indiceAtual = 0;
            AtualizarSlide();

            Debug.Log($"[MySlider] {slidesAtuais.Count} slides carregados.");
        }
        else
        {
            Debug.LogWarning("[MySlider] Lista de slides vazia.");
        }
    }
    else
    {
        Debug.LogError($"[MySlider] Slides '{slideIdentifier}' não encontrados.");
    }
}

3 referências
private void AtualizarSlide()
{
    if (slidesAtuais == null || slidesAtuais.Count == 0)
        return;

    Texture2D textura = slidesAtuais[indiceAtual];

    rawImage.texture = textura;
}

```

Figura 3.22 – Inicialização e recuperação dos slides a partir do LmsLoader para exibição no sistema. Fonte: do autor (2026)

O fluxo desta integração com [LMS](#) esta representado no diagrama da Figura 3.23. Nele, observa-se que o professor insere os conteúdos na plataforma *Moodle*, que por sua vez os disponibiliza através da [API](#). Essa comunicação é intermediada pelo *script MyLoader* que injeta os dados na classe *LmsLoader* do pacote *SeriousGameComponent*, responsável por armazenar os dados. Por fim, os conteúdos são acessados pelos scripts *MySlider* e *MyAudioPlayer* para serem apresentados ao jogador como tela do componente *tablet*, além de permitir que sejam acessados pelos próprios componentes do pacote *SeriousGameComponent* caso necessário futuramente.

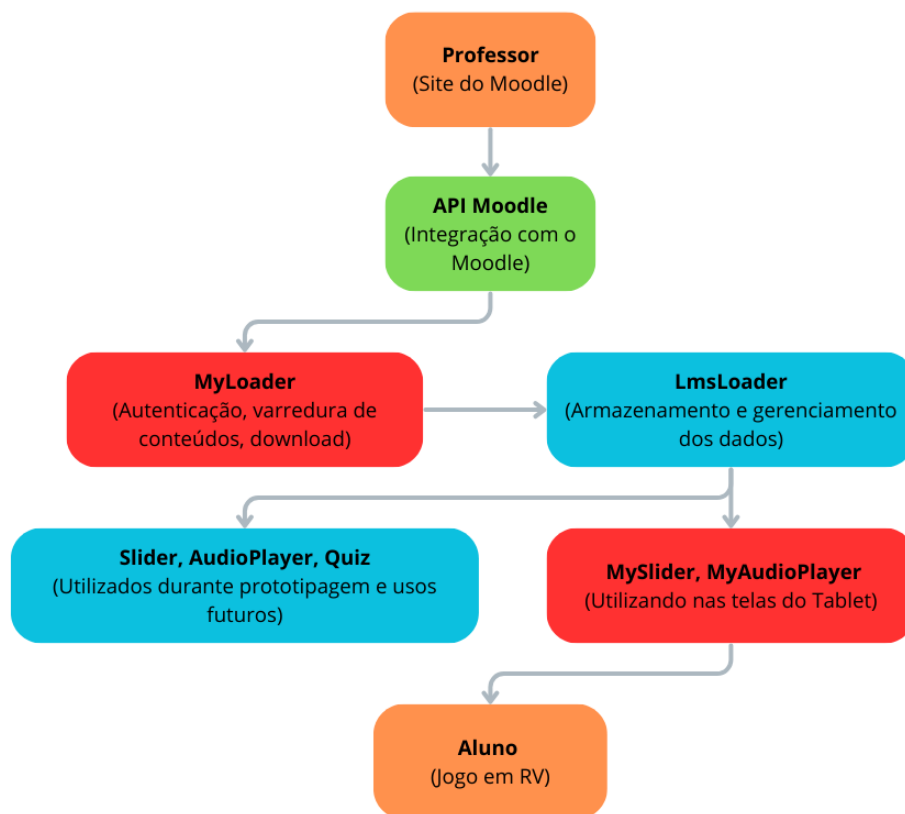


Figura 3.23 – Diagrama de fluxo do uso do pacote *SeriousGameComponent* neste projeto. Fonte: do autor (2026).

3.2.4 Observações Durante o Desenvolvimento

Durante o desenvolvimento do protótipo inicial do jogo, foram realizados testes de usabilidade para observar o funcionamento das mecânicas implementadas e identificar possíveis dificuldades de interação. Esses testes ocorreram de maneira informal, envolvendo o próprio desenvolvedor e utilizando os equipamentos: *Meta Quest 2*, *Meta Quest 3* e *Meta Quest Pro*.

O objetivo dessas sessões foi compreender como os jogadores interagem com o ambiente virtual e com os objetos disponíveis, bem como identificar limitações do design inicial que pudessem impactar o foco pedagógico da experiência.

A partir dessas observações, foram identificados alguns problemas relacionados principalmente à manipulação dos objetos, à organização do ambiente e ao fluxo de interação do jogo. Essas limitações, discutidas anteriormente na subseção de Implementação e Decisões de Design 3.2, indicaram a necessidade de reformular parte das mecânicas implementadas.

Assim, os testes de usabilidade desempenharam um papel importante no direcionamento das decisões de design adotadas posteriormente, contribuindo para a reorganização da interação do jogador e para a centralização das funcionalidades nos dispositivos apresentados na subseção sobre Objetos Interativos 3.2.3.

4 Estudo de Caso

Este capítulo apresenta o caso de uso obtido a partir do desenvolvimento do jogo sério em RV proposto neste trabalho. Inicialmente, a Seção 4.1 apresenta uma visão geral do jogo sério resultante, bem como de seu funcionamento. Em seguida, a Seção 4.2 discute as principais dificuldades encontradas durante o desenvolvimento e as estratégias adotadas para contorná-las. Por fim, a Seção 4.3 detalha a utilização do CDJE sob a perspectiva do desenvolvedor.

4.1 Visão Geral do Funcionamento do Jogo

O resultado final deste projeto é um jogo sério em RV no qual o jogador explora um ambiente virtual com o objetivo de localizar um tesouro utilizando conceitos de orientação espacial.

A atividade ocorre em uma ilha, a qual é gerada proceduralmente a cada nova partida, e ao iniciar a aplicação, o jogador tem acesso ao *tablet*, dispositivo responsável por concentrar as funcionalidades principais da interface do sistema, em um ambiente obscurecido, reduzindo qualquer estímulo externo. Nessa etapa é possível configurar pelo próprio *tablet* parâmetros da partida, como o tempo limite e habilitar ou não o teleporte, para ajudar na movimentação do personagem, assim como o radar que auxilia na detecção do tesouro. Além disto, é possível acessar conteúdos educacionais disponibilizados pelo professor, como *slides* e áudios integrados à plataforma *Moodle*. E, por fim, nesta tela ainda é possível iniciar a partida.

Uma vez iniciada a atividade, o jogador passa a explorar livremente o ambiente da ilha em busca do tesouro. Para auxiliá-lo nesta missão, ele conta com dois objetos essenciais: o mapa acessível pelo *tablet* e a bússola, que é uma das telas do relógio inteligente acoplado ao modelo de mãos do jogador. O mapa apresenta uma representação simplificada do terreno da ilha, indicando com um “X” o local do tesouro, assim como outras informações importantes do cenário, como relevo, vegetação (indicada por variações de cores) e estruturas como o farol, uma estrutura visível por grande parte da ilha.

O relógio inteligente funciona como um instrumento de consulta rápida durante a exploração. Entre suas telas disponíveis estão o temporizador da partida, que informa o tempo restante para conclusão da atividade, a bússola, responsável por indicar a direção do norte no ambiente virtual, e o radar, caso habilitado. Esse último recurso fornece pistas visuais sobre a proximidade do tesouro, alterando sua animação e sua cor conforme o jogador se aproxima da localização correta.

Durante a exploração, o jogador pode alternar entre as funcionalidades do *tablet* e do relógio de forma integrada ao fluxo de *gameplay*, sem a necessidade de trocar de cena ou acessar

objetos independentes no ambiente, podendo, caso queira, acessar os conteúdos de slides e áudio, pausando o tempo da partida.

Quando o jogador se aproxima suficientemente da área onde o tesouro está localizado, o sistema identifica essa proximidade e revela o objeto no ambiente virtual, concluindo a atividade proposta mostrando o tempo em que o jogador a conclui, e registrando como recorde caso alcançado um tempo menor ao recorde já registrado.

Dessa forma, o ciclo de jogo combina exploração, orientação espacial e consulta a instrumentos de navegação, estruturando uma experiência que busca reforçar, de forma prática, os conceitos da geolocalização.



Figura 4.1 – Jogador dentro da experiência imersiva. Fonte: do autor (2026).

4.2 Evolução do Protótipo ao Sistema Final

A jornada de desenvolvimento deste projeto não foi estritamente linear. Embora a etapa de idealização utilizando o [CDJE](#) tenha contribuído significativamente para o planejamento e a prototipagem inicial do jogo sério, ao longo do processo foram necessárias reformulações e revisões no design da experiência.

Estas reformulações ocorreram conforme os testes exploratórios descritos na Seção [3.2.4](#). Os quais detectaram problemas de jogabilidade que interferiam diretamente no objeto de aprendizagem do jogo sério, criando uma barreira a mais de dificuldade ao obrigar o jogador a dominar mecânicas prévias como a manipulação livre de cada objeto em cena, além da troca de cenários para acesso aos recursos educativos, interrompendo o fluxo de *gameplay*.

Diante dessas observações, as mecânicas foram repensadas e reorganizadas, concentrando as principais funcionalidades em dois elementos centrais: o *tablet* e o relógio inteligente. Esses

dispositivos passaram a reunir as funções anteriormente distribuídas em diferentes objetos do cenário, além de permitir um acesso mais rápido e organizado aos conteúdos do jogo. Também foram incorporados alguns recursos de qualidade de vida, como o registro de recordes de partida, que incentivam os jogadores a se superarem a cada tentativa, e um mecanismo que reposiciona o jogador em terra firme caso ele entre na água, auxiliando na continuidade da exploração.

4.3 Uso do Canvas no Processo de Desenvolvimento

Conforme discutido na Seção 4.2, o **CDJE** desempenhou um papel importante ao longo do desenvolvimento deste trabalho. Sua principal contribuição ocorreu na fase inicial de idealização do projeto, na qual o canvas serviu como ferramenta central para a organização das ideias e a definição dos principais elementos que compõem o jogo sério proposto.

A utilização do **CDJE** possibilitou estruturar, de forma rápida, visual e sistemática, os principais aspectos, como os objetivos pedagógicos, as mecânicas de interação, os recursos educacionais e o fluxo geral da experiência do jogador. Esse processo facilitou a construção de uma visão inicial do sistema e contribuiu para que a etapa de prototipagem fosse conduzida com maior clareza e rapidez. Nesse contexto, a implementação inicial do jogo ocorreu de maneira mais direta, uma vez que diversas decisões de design já estavam previamente definidas e documentadas no *canvas*.

Além de auxiliar na organização das ideias do projeto, o **CDJE** também se mostrou útil como instrumento de apoio durante a reformulação do jogo. Conforme novos desafios foram identificados ao longo dos testes exploratórios e do desenvolvimento prático, o *canvas* serviu de referência para reavaliar decisões previamente tomadas e planejar ajustes nas mecânicas e na estrutura da experiência. Dessa forma, o **CDJE** não foi utilizado apenas como um guia inicial, mas também como um elemento guia para o processo de tomada de decisões de design.

Sob a perspectiva do desenvolvimento, essa experiência permitiu observar que o uso do *canvas* torna mais explícita e fácil a relação entre os objetivos educacionais e as soluções implementadas no jogo. Ao organizar esses elementos em um único artefato visual, o **CDJE** contribui para a manutenção do alinhamento entre a proposta pedagógica e as decisões técnicas adotadas durante a implementação.

5 Considerações Finais

Este capítulo apresenta uma síntese dos resultados obtidos ao longo do desenvolvimento do trabalho, destacando as contribuições do projeto e as conclusões alcançadas com a implementação do jogo sério em realidade virtual na Seção 5.1. Além disso, são discutidos possíveis trabalhos futuros na Seção 5.2.

Ao longo deste trabalho, foi desenvolvido um jogo sério em realidade virtual com foco no ensino de conceitos básicos de geolocalização, o qual permite ao jogador explorar um ambiente virtual tridimensional em RV, utilizando instrumentos de orientação e acessando conteúdos educacionais integrados ao ambiente de aprendizagem *Moodle*.

Outro aspecto relevante do trabalho foi a aplicação do CDJE ao longo do processo de desenvolvimento, auxiliando tanto na fase inicial de idealização quanto na reorganização das mecânicas do sistema.

5.1 Conclusão

Com base nos objetivos específicos definidos na Seção 1.2, é possível analisar como o desenvolvimento do sistema atendeu às metas propostas neste trabalho:

- **Projetar um ambiente virtual tridimensional que estimule a aprendizagem de conteúdos educacionais por meio da exploração e interação.** O jogo desenvolvido apresenta um ambiente imersivo em realidade virtual composto por uma ilha gerada proceduralmente, permitindo que o jogador explore o espaço com ajuda de um mapa e uma bússola, e aplique conceitos de orientação espacial durante a atividade.
- **Implementar mecânicas de jogo acessíveis e intuitivas, compatíveis com as capacidades cognitivas do público-alvo e com os conteúdos abordados.** Durante o desenvolvimento, as mecânicas foram reformuladas a partir dos testes exploratórios, resultando na centralização e simplificação das interações em dois dispositivos principais, o *tablet* e o relógio, reduzindo a complexidade de manipulação de objetos e tornando a experiência mais intuitiva e mantendo a necessidade de entendimento das ferramentas para atingir o objetivo central.
- **Incorporar elementos de personalização com Moodle, onde o tutor possa disponibilizar material dentro do jogo.** Foi implementado um sistema de comunicação com o ambiente *Moodle*, permitindo a utilização de slides e áudios diretamente no jogo por meio do *tablet*, integrando o conteúdo pedagógico à experiência em realidade virtual.
- **Investigar a aplicação do Canvas de Design de Jogos Educativos (CDJE) como ferramenta de apoio ao processo de desenvolvimento de jogos sérios em realidade virtual.**

O [CDJE](#) foi utilizado desde a fase de idealização do projeto, auxiliando na organização das ideias, na definição das mecânicas e no replanejamento das interações do sistema, demonstrando sua utilidade no processo de desenvolvimento de um jogo sério.

Portanto, de forma geral, os resultados apresentados indicam que os objetivos do trabalho foram alcançados. Demonstrando a viabilidade da criação de jogos sérios em realidade virtual como ferramenta de apoio ao ensino de geolocalização.

5.2 Trabalhos Futuros

Como trabalhos futuros, diversas possibilidades podem ser exploradas a partir da base desenvolvida.

Uma das principais extensões previstas é a ampliação das formas de integração com a plataforma *Moodle*. Durante o desenvolvimento, implementou-se o carregamento de conteúdos educacionais, como slides e áudios. Entretanto, o pacote *SeriousGameComponents* já prevê outros recursos pedagógicos que não foram totalmente explorados neste projeto. Entre eles, destaca-se a implementação de *quizzes* integrados ao ambiente virtual, permitindo a aplicação de atividades avaliativas diretamente na experiência em realidade virtual.

Outra possibilidade de evolução é a integração com sistemas de [Sistema de Registro de Aprendizagem \(LRS\)](#), possibilitando o registro e acompanhamento das interações dos jogadores durante a atividade educacional, também previstos no pacote *SeriousGameComponents*. Essa abordagem permitiria coletar dados sobre o progresso do estudante, o tempo de interação, as decisões tomadas e o desempenho em atividades, contribuindo para análises pedagógicas mais aprofundadas.

Além disso, a metodologia utilizada neste trabalho, baseada no uso do [CDJE](#) aliado ao desenvolvimento em realidade virtual e à integração com [LMS](#), pode ser aplicada na criação de novos jogos educativos em diferentes áreas do conhecimento para diferentes públicos-alvo. Portanto, o modelo proposto pode servir como base para o desenvolvimento de inúmeras experiências educacionais imersivas, ampliando o potencial de aplicação da abordagem apresentada.

Dessa forma, espera-se que os trabalhos futuros propostos contribuam para ampliar o potencial educacional do sistema desenvolvido, além de incentivar novas pesquisas na área de jogos sérios em realidade virtual voltados à educação.

Referências

- BURSZTYN, N.; D'ANTONIO, A.; MASTERS, B. Using immersive virtual reality to measure strike and dip and teach geological mapping concepts. *Earth Science, Systems and Society*, v. 5, n. 1, p. e2024-008, 2025. Disponível em: <<https://www.lyellcollection.org/doi/abs/10.1144/esss2024-008>>.
- FREITAS, F.; GODINHO-PAIVA, R. Jogos em realidade virtual: medição da qualidade de imersão. *Revista Poliedro*, v. 5, p. 444–467, 12 2021.
- KHRONOS. *OpenXR Tutorial*. [S.l.], 2023. Acesso em 23 jul. 2025. Disponível em: <<https://openxr-tutorial.com/android/vulkan/1-introduction.html#introduction>>.
- MARINHO, L. M. d. C. C. *EducaMinasVR: qualificação de professores em design de interfaces imersivas educativas em realidade virtual*. Dissertação (Dissertação (Mestrado em Ciência da Computação)) — Universidade Federal de Ouro Preto, Ouro Preto, MG, 2024. Orientador: Saul Emanuel Delabrida Silva. Coorientador: Reinaldo Silva Fortes. Disponível em: <<https://www.repositorio.ufop.br/items/9bd95420-3075-4be0-89f3-00f47ffca9cc>>. Acesso em: 14 ago. 2025.
- MICHAEL, D.; CHEN, S. *Serious Games: Games that Educate, Train and Inform*. Thomson Course Technology, 2006. ISBN 9781592006229. Disponível em: <<https://books.google.com.br/books?id=49kTAQAAIAAJ>>.
- Microsoft. *C# Guide – .NET managed language*. [S.l.], 2025. Acesso em 24 jul. 2025. Disponível em: <<https://learn.microsoft.com/dotnet/csharp/>>.
- OFOEDA, J.; BOATENG, R.; EFFAH, J. Application programming interface (api) research: A review of the past to inform the future. *International Journal of Enterprise Information Systems*, v. 15, p. 76–95, 07 2019.
- PIMENTA, A. S. G. M. *Uma arquitetura de modularização de componentes para desenvolvimento de jogos sérios na Unity Engine*. Dissertação (Mestrado) — Universidade Federal de Ouro Preto, Ouro Preto, 2022. Disponível em: <<https://www.monografias.ufop.br/handle/35400000/4710>>.
- RIBEIRO, M. G. *CanYouFindIt: Repositório oficial do projeto*. 2026. <<https://github.com/xr4good/CanYounFindIt>>.
- SELZER, M. N.; ASIAIN, L.; FERRACUTTI, G.; URRIBARRI, D. K.; LARREA, M. L. Immersive learning: Evaluating virtual reality for geological compass education. *IEEE Pervasive Computing*, v. 24, n. 2, p. 17–27, 2025.
- SILVA, R.; CIROLINI, A.; BRUCH, A.; FONSECA, E.; CAMPOS, S. Gonçalves de; FREITAS, F. A percepção dos níveis de aprendizagem de cartografia dos alunos do ensino fundamental. *Expressa Extensão*, v. 24, p. 85, 12 2018.
- SILVA, S. E. D. *Smart Wearable: let's think wearable from a new perspective*. 104 p. Tese (Doutorado em Ciência da Computação) — Universidade Federal de Ouro Preto, Ouro Preto, 2018. Instituto de Ciências Exatas e Biológicas. Disponível em: <<https://www.repositorio.ufop.br/handle/123456789/9838>>.

SOUZA, L. P.; SOUSA, K. F. D.; JUCá, S. C. S. Gamificação e realidade aumentada como ferramentas pedagógicas para o ensino: uma agenda de pesquisa. *Educação Temática Digital*, Universidade Estadual de Campinas, v. 27, p. e025020, mar. 2025. Acesso em: 24 ago. 2025. Disponível em: <<https://www.periodicos.sbu.unicamp.br/ojs/index.php/etd/article/view/8674840>>.

TORI, R.; KIRNER, C.; SISCOOTTO, R. (Ed.). *Fundamentos e Tecnologia de Realidade Virtual e Aumentada*. Porto Alegre: SBC – Sociedade Brasileira de Computação, 2006. Livro do Pré-Simpósio do VIII Symposium on Virtual Reality, Belém – PA. ISBN 85-7669-068-3.

UNITY. *Unity Manual*. [S.l.], 2025. Acesso em 23 jul. 2025. Disponível em: <<https://docs.unity3d.com/6000.0/Documentation/Manual/index.html>>.