Consequências – um jogo de exploração utilizando realidade virtual com ilusão de ótica sobre conscientização ambiental

Vitor Hugo Helmbrecht, Dalton Solano dos Reis – Orientador

Curso de Bacharel em Ciência da Computação

Departamento de Sistemas e Computação

Universidade Regional de Blumenau (FURB) – Blumenau, SC – Brasil

vhelmbrecht@furb.br, dalton@furb.br

**Resumo:** Este artigo apresenta o processo de desenvolvimento e avaliação de um jogo para plataformas de realidade virtual, utilizando-se de ilusão de ótica e com foco na conscientização ambiental do jogador. O jogo foi desenvolvido utilizando o motor gráfico Unity3D, além da linguagem C# e as ferramentas XR Interaction Toolkit, Cinemachine, Timeline e NavMesh. Tanto para o desenvolvimento quanto para a realização dos testes foi utilizado o dispositivo Meta Quest 2. Os resultados obtidos foram satisfatórios, de forma que o jogo cumpriu seus objetivos.

**Palavras-chave**: Ciência da computação. Realidade virtual. Jogo. Exploração. Ilusão de ótica. Unity. Conscientização ambiental.

# Introdução

O aquecimento global é um aumento da temperatura média superficial global (SILVA; PAULA, 2009). Esse fenômeno pode acontecer por algumas razões diferentes, sejam elas internas ou externas. De acordo com Silva e Paula (2009), fatores internos do aquecimento global são complexos e inconstantes, além de dependerem de diversas variáveis diferentes, como a atividade solar e o vulcanismo. Já os fatores externos são causados por ações como a queima de combustíveis fósseis ou queimadas. Os autores apresentam fatores tanto externos quanto internos que vêm realizando essas alterações climáticas ao longo de séculos, mas uma coisa é afirmada: o ser humano possui um impacto sobre a mudança climática do planeta.

Conscientização ambiental está relacionado às ações que o indivíduo exerce que estão vinculadas ao meio ambiente (SÁNCHEZ; LAFUENTE, 2010). Ao mesmo tempo, educação ambiental já é no Brasil um dos temas contemporâneos transversais da Base Nacional Comum Curricular (BNCC), ou seja, ele deve ser abordado durante o ensino, mas sem ser tratado como uma imposição de conteúdo (BRASIL, 2022). Isso nos mostra como a conscientização ambiental já está sendo abordada no Brasil, além de ser um assunto importante para que possamos diminuir o nosso impacto sobre as mudanças negativas que ocorrem no nosso planeta.

O mercado de videogames mundial gerou 196.8 bilhões de dólares em 2022 (NEWZOO, 2023), enquanto o mercado de cinema gerou 77 bilhões no mesmo período (IBISWORLD, 2022), o que demonstra como as pessoas estão cada vez mais interessadas em videogames. Junte a isso o fato de que o uso de equipamentos de realidade virtual teve um grande aumento recentemente (TAO *et al*., 2021), e temos a fundação de um mercado promissor.

Assim como TAO *et al*. (2021) falaram, não é apenas para uso pessoal que as pessoas começaram a adotar mais os videogames. Existe também o uso voltado para, por exemplo, o tratamento de condições clínicas. Visto que esse mercado continua crescendo e que a realidade virtual possui múltiplas funcionalidades, esse trabalho tem como objetivo construir um jogo para plataformas de realidade virtual que faça o uso de ilusão de ótica como mecânica principal do jogo, além de ter como foco a conscientização ambiental do jogador.

# FUNDAMENTAÇÃO TEÓRICA

Nessa seção são apresentados os assuntos que fundamentam o desenvolvimento do jogo. A primeira subseção trata sobre realidade virtual, a segunda subseção trata sobre a ilusão de ótica, a terceira subseção trata sobre conscientização ambiental e a quarta subseção trata sobre os trabalhos correlatos.

## REALIDADE VIRTUAL

Segundo Azuma (1997, p.2), Realidade Virtual (RV) faz com que, enquanto imerso, o usuário não possa ver o mundo que está ao seu redor. Essa tecnologia coloca o usuário em um mundo à parte, completamente novo. De acordo com Di, Li e Cheng (2020), a RV é uma tecnologia que combina uma variedade de outras tecnologias, como computação, multimídia, processamento de imagem, entre outros, e por fim cria uma tecnologia que, por sua vez, está dentro do campo de computação e simulação. Os autores ainda afirmam que a tecnologia de RV inclui a simulação de ambientes, percepções, habilidades naturais (como a rotação da cabeça, olhos, gestos etc.) e equipamentos sensoriais.

Tao *et al*. (2021) comenta que recentemente ocorreu um grande aumento no uso de RV não apenas para uso pessoal, como também no tratamento de diferentes condições clínicas. Os autores afirmam que isso se deve ao fato de os *Head-Mounted Displays* (HMD) de alta qualidade, que são os equipamentos através dos quais é possível ter uma experiência RV mais imersiva, estão se tornando cada vez mais acessíveis. A Figura 1 exemplifica essa normalização de HMDs de alta qualidade. A propaganda da empresa Meta é realizada com a utilização de pessoas aproveitando a experiência de RV com HMDs em suas casas.

Figura – Propagando do Meta Quest 2

Homem em pé com chapéu

Descrição gerada automaticamente

Fonte: Meta (2023).

## ilusão de ótica

É complicado encontrar uma definição satisfatória para o termo ilusão, mas se pode dizer que as melhores ilusões são aquelas as quais não é possível ver a olho nu, apenas com a ajuda de outros sentidos (como o toque), ou o uso outros instrumentos (como uma régua). De qualquer maneira, podemos buscar a definição da palavra “ilusão” no dicionário, que traz as seguintes definições para o termo: 1. algo que engana intelectualmente; 2. a percepção de algo que está existindo de uma maneira que cause a interpretação errônea da sua real natureza (BACH; POLOSCHEK, 2006).

Bach e Poloschek (2006) apresentam alguns exemplos de ilusões de ótica, como por exemplo a grade de Hermann, demonstrada na Figura 2, na qual ao olhar na parte esquerda da grade é possível ver pontos cinzas aparecendo, mas ao olhar na parte direita, os pontos não aparecem devido à mudando no formato da grade.

Figura - Grade de Hermann à esquerda e variação à direita

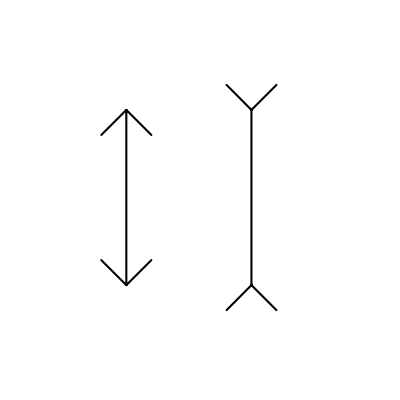
Padrão do plano de fundo

Descrição gerada automaticamente

Fonte: Bach; Poloschek (2006, p. 1).

Outra ilusão de ótica bem conhecida é a ilusão de Müller-Lyer, apresentada na Figura 3. Nessa ilusão, apesar de os dois segmentos de retas possuírem o mesmo tamanho, o segmento que possui as alhetas para fora (direita) é percebido como maior que o que está com as alhetas para dentro (esquerdo) (SUGANUMA, 2006).

Figura - Ilusão de Müller-Lyer



Fonte: Suganuma (2006, p. 5).

## conscientização ambiental

Sánches e Lafuente (2010) definem conscientização ambiental como o conceito que refere a fatores especificamente psicológicos que estão relacionados à propensão do indivíduo de realizar comportamentos que favorecem o meio ambiente.

Homburg e Stolberg (2006) citam quatro exemplos de comportamentos que podem ser classificados como favorecedores do meio ambiente, sendo eles: ativismo ambiental (como envolvimento de organizações ambientais); comportamentos não-ativistas em esferas públicas (como petições); ambientalismo do setor privado (como poupar energia e comprar objetos feitos com materiais reciclados); e comportamentos em organizações (como o design de produtos).

## TRABALHOS CORRELATOS

São apresentados trabalhos com características semelhantes aos principais objetivos do estudo proposto. O primeiro está detalhado no Quadro 1**Erro! Fonte de referência não encontrada.**, e o trabalho se trata do jogo CidadaniaAR (NIENOW, 2019), que foi desenvolvido utilizando realidade aumentada e ilusão de ótica, além de abordar o tema reciclagem. O segundo trabalho, que está sendo apresentado no Quadro 2, explica um pouco mais sobre o jogo Vignettes (SKELETON BUSINESS, 2017), que tem o foco em exploração e na sua utilização de ilusão de ótica. Já o Quadro 3 traz informações sobre o jogo Monument Valley (USTWO GAMES, 2014), que é um jogo móvel que também se utiliza de ilusão de ótica como mecânica principal do jogo para avançar na história.

Quadro – CidadaniaAR

|  |  |
| --- | --- |
| Referência | Nienow (2019). |
| Objetivos | Criar um jogo que faça uso da ilusão de ótica como mecânica principal de jogabilidade. |
| Principais funcionalidades | Resolução de desafios utilizando ilusão de ótica. |
| Ferramentas de desenvolvimento | Unity 3D. |
| Resultados e conclusões | Após a aplicação de questionários, concluiu-se que o objetivo foi alcançado, apesar de pontos a melhorar terem sido apresentados. |

Fonte: elaborado pelo autor.

Nienow (2019) desenvolveu o CidadaniaAR, um jogo de puzzle e realidade aumentada focado em ilusão de ótica, incorporando o tema de reciclagem de forma interativa, conforme demonstrado na Figura 4. O jogo foi desenvolvido utilizando o motor gráfico Unity junto da linguagem de programação C#, além do *plugin* para realidade aumentada Vuforia e o software Blender para o desenvolvimento de customizações do jogo em relação a modelos 3D.

Figura – Puzzle de fragmentação e sua resolução

Diagrama

Descrição gerada automaticamente com confiança baixa

Fonte: Nienow (2019).

Quadro - Vignettes

|  |  |
| --- | --- |
| Referência | Skeleton Business (2017). |
| Objetivos | Produto comercial. |
| Principais funcionalidades | Utilização da ilusão de ótica para realizar a exploração da história. |
| Ferramentas de desenvolvimento | Unity 3D. |
| Resultados e conclusões | Vignettes recebeu e foi indicado a premiações desde 2015 até 2018, incluindo o prêmio IndieCade EU 2017. |

Fonte: elaborado pelo autor.

O estúdio Skeleton Business (2017) desenvolveu Vignettes, um jogo indie comercial desenvolvido utilizando o motor gráfico Unity. O jogo foi lançado oficialmente em 2017 para iOS e em 2019 para computador, e é focado em exploração com um toque incomum de ilusão de ótica.

Durante o jogo, o jogador deve rotacionar os objetos que aparecem em sua tela até que ele encontre um ângulo que transformará esse objeto em algum outro objeto ou mesmo um animal, conforme demonstrado na Figura 5. Além da rotação, o jogador também pode interagir com quase todos os objetos através de toques/cliques ou alteração do zoom. O que se torna não só um extra interessante para o jogador com interações como o barulho de um rádio ou afagar um gato, como também se torna algo necessário em determinados momentos para liberar objetos escondidos.

Figura – Exemplo de transformação de telefone para panela

Forma, Polígono

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

Quadro - Monument Valley

|  |  |
| --- | --- |
| Referência | Ustwo Games (2014). |
| Objetivos | Produto comercial. |
| Principais funcionalidades | Utilização da ilusão de ótica para resolver enigmas e avançar na história. |
| Ferramentas de desenvolvimento | Unity 3D. |
| Resultados e conclusões | Monument Valley recebeu diversas premiações, sendo o principal o prêmio de melhor jogo móvel de 2017 pelo The Game Awards. |

Fonte: elaborado pelo autor.

Monument Valley é um jogo indie comercial desenvolvido utilizando o motor gráfico Unity pela empresa USTWO Games (2014). O jogo foi lançado oficialmente em 2014 para iOS e Android, e é focado em puzzles com a utilização principal de ilusões de ótica para formar objetos impossíveis e assim conseguir avançar na história. Alguns trechos do jogo podem ser visualizados na Figura 6.

Figura - Alguns níveis do jogo Monument Valley



Fonte: Monument Valley.

# DESCRIÇÃO do jogo

Nessa seção são apresentados os aspectos fundamentais e mais importantes da implementação do jogo e está dividida em duas partes, das quais a primeira explica sobre a visão geral do jogo, enquanto a segunda descreve a implementação do jogo.

## Especificação do jogo

O jogo desenvolvido coloca o jogador como observador dos personagens principais e seus mundos. Existem um total de três histórias e dois personagens principais. O jogo também é dividido em dois tipos de interações com o jogador, a primeira sendo onde o jogador apenas acompanha os personagens principais e observa a história se desenrolando, e a segunda onde o jogador interage diretamente com determinados objetos.

Cada nível começa com a história do personagem principal se desenrolando, à qual o jogador apenas acompanha enquanto pode observar o cenário ao redor, já que ele está em um ambiente de realidade virtual. Em algumas partes predeterminadas da história é realizada uma transição para o cenário de interação com objetos, conforme demonstrado na Figura 7.

Figura - Transição da cena de história para a cena de interação

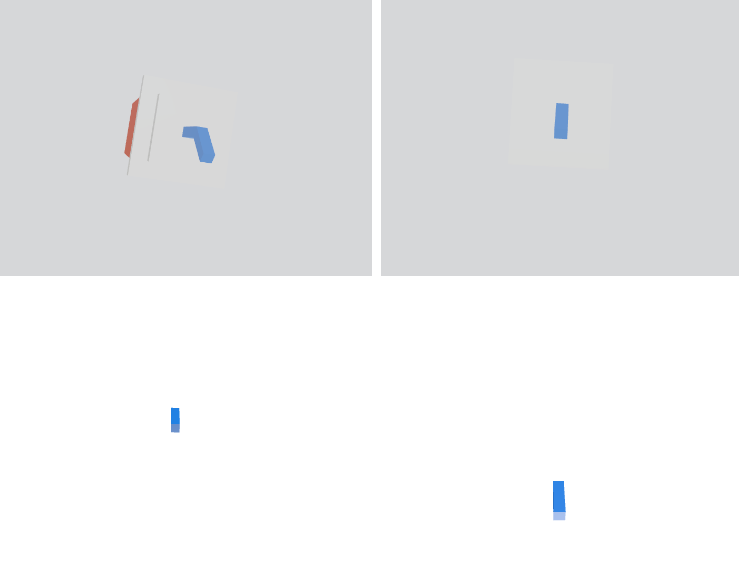
Interface gráfica do usuário, Site

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

A partir do momento que o jogador estiver na cena de interação com o objeto, ele deverá então utilizar do controle do seu aparelho de realidade virtual para rotacionar o objeto até um ângulo no qual ele consiga formar um novo objeto, assim como demonstrado na Figura 8, onde o jogador deve transformar o canudo do copo de plástico em uma gota de chuva, que iniciará assim a cena 2.

Figura - Transformação do copo em gota d'água e começando a cair



Fonte: elaborado pelo autor.

Após a transformação do objeto, a próxima cena de acompanhamento irá iniciar automaticamente, passando assim para a próxima parte da história. O jogo se baseia nessa mecânica para contar suas três histórias de como os atos cometidos pelos personagens principais possuem consequências, sejam elas positivas ou negativas, na vida delas mesmas e de outros seres vivos.

Para elucidar melhor as sequências de cenas que são seguidas em cada fase do jogo, foram criados diagramas de sequências para cada uma, conforme pode ser observado na Figura 9, Figura 10 e Figura 11.

Figura - Diagrama de sequências da fase 1

Uma imagem contendo Calendário

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

Figura - Diagrama de sequências da fase 2

Gráfico de barras

Descrição gerada automaticamente com confiança média

Fonte: elaborado pelo autor.

Figura - Diagrama de sequências da fase 3

Gráfico

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

## implementação

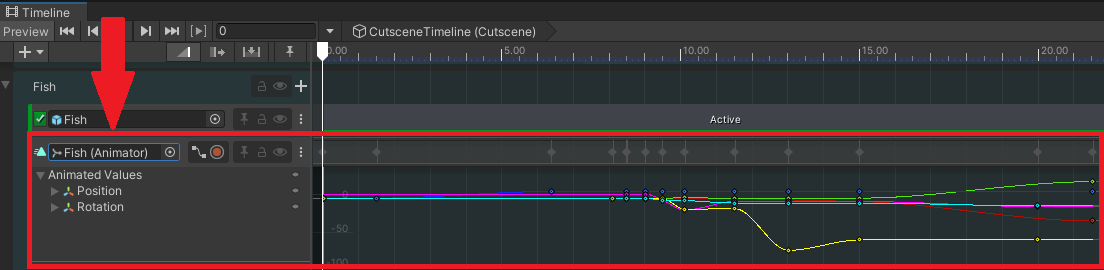
O jogo foi desenvolvido utilizando o motor gráfico Unity3D. Além disso, foi utilizado o ambiente de desenvolvimento Visual Studio Community, que é disponível de maneira gratuita.

As cenas do jogo que não possuem interação direta com o jogador são criadas utilizando dois componentes principais como base: o cenário da cena específica e a cinemática que será mostrada nessa cena. O cenário é montado utilizando modelos, sejam eles de construções, personagens, animais ou plantas, e todos os modelos utilizados no jogo foram adquiridos através da *asset store* que a própria Unity oferece. Quanto a parte da cinemática, é possível dividi-la em duas ferramentas essenciais disponibilizadas pelo próprio Unity: Cinemachine e Timeline.

Timeline é a ferramenta que o Unity disponibiliza para a criação de sequências de eventos (Unity Technologies, 2020a). Essa ferramenta foi utilizada para criar as sequências de posicionamento, rotação e demais possibilidades de comportamentos de câmeras e objetos nas cinemáticas do jogo. Quanto à Cinemachine, Unity   
Technologies (2020b) a classifica como um sistema de câmeras que permite a utilização de comportamentos mais complexos de câmeras sem a necessidade de escrever código. Essa ferramenta foi utilizada em todas as câmeras das cinemáticas do jogo.

No jogo existem três tipos diferentes de Non-Player Characters (NPC), com os quais o jogador não interage diretamente, apenas acompanha. Desses três tipos de NPC, dois deles são controlados via código e o terceiro se move conforme o que foi determinado na cinemáticaatravés da ferramenta Timeline. Na Figura 12 é possível observar um exemplo do terceiro tipo de NPC citado, no qual o objeto Fish (Animator) tem sua posição e rotação alteradas conforme o tempo da cena avança.

Figura - Exemplo de NPC controlado pela Timeline



Fonte: elaborado pelo autor.

Quanto aos outros dois tipos de NPCs, o mais simples deles é controlado por uma classe chamada NPC, a qual está demonstrada no Quadro 4. Essa classe determina os valores de animação desse NPC, sendo eles velocidade e se o NPC está encostado em uma parede ou não.

Quadro - Classe NPC

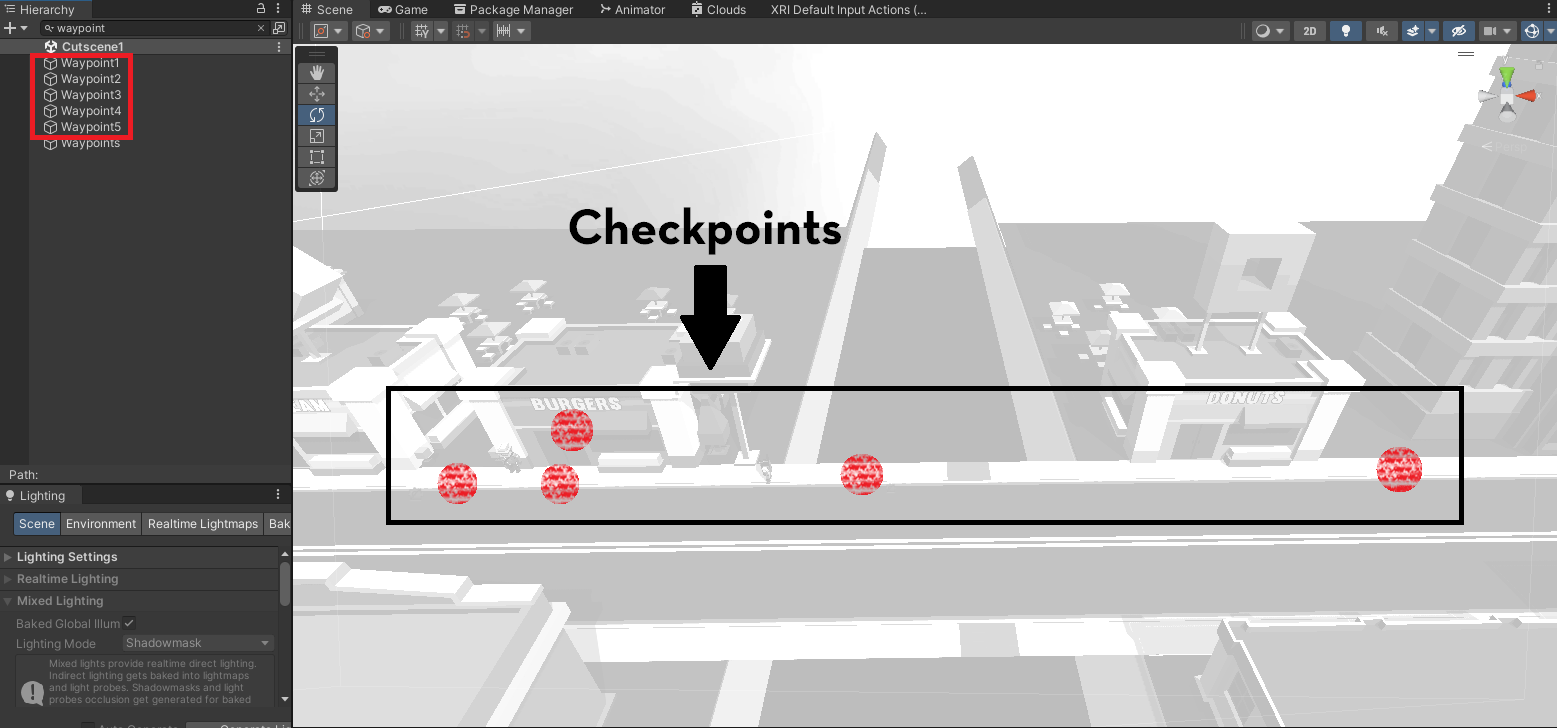
Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

O último tipo de NPC é um pouco mais detalhado em suas ações e é também o menos utilizado. Esse NPC possui um código que serve para avançar até algum local específico da cena e passar por *checkpoints* durante o caminho. Além disso, seu código também permite a manipulação de um objeto que o NPC possa estar segurando. Para que ele possa exercer sua função corretamente, é necessário determinar alguns *checkpoints* no cenário e passá-los para o código. Os *checkpoints* são posicionados no cenário através de GameObjects com o nome “Waypoint” seguido de seu índice, conforme apresentado na Figura 13. Já a classe que controla esse tipo de NPC se chama NPCMain, e ela pode ser observada por completo no Quadro 5.

Figura - Checkpoints da Cena 1, Fase 2



Fonte: elaborado pelo autor.

Quadro - Classe NPCMain

Texto

Descrição gerada automaticamente

Fonte: elaborado pelo autor

A classe NPCMain herda da classe MonoBehaviour, que é disponibilizada pelo Unity, permitindo assim que ela utilize alguns métodos da classe pai. Um desses métodos é o Update, que é executado a cada renderização realizada. O método Update está sendo utilizado nesse caso para poder determinar qual é o próximo destino para o qual o NPC deve seguir, bem como verificar se o objeto que está sendo segurado pelo NPC deve ser exibido ou desvinculado dele através dos métodos CheckShowObject e CheckDropObject.

Além dos *checkpoints*, que são recebidos através do atributo goals, a classe NPCMain também necessita de outros objetos para poder funcionar conforme o esperado. O atributo mais importante para isso é o agent que é atribuído no método Start, o qual também é herdado através da classe MonoBehaviour.

O atributo agent é do tipo NavMeshAgent, o qual faz parte da ferramenta NavMesh que o Unity disponibiliza. Essa ferramenta serve para determinar quais são os objetos sobre os quais os agentes podem andar. Sua aplicação pode ser visualizada na Figura 14, aonde todos os objetos que estão em azul são aqueles sobre os quais os personagens com a classe NPCMain podem andar. Para fins de demonstrar todos os locais, a figura não retrata a hamburgueria que existe ao lado da sorveteria.

Figura - Locais sobre os quais o NPC pode andar

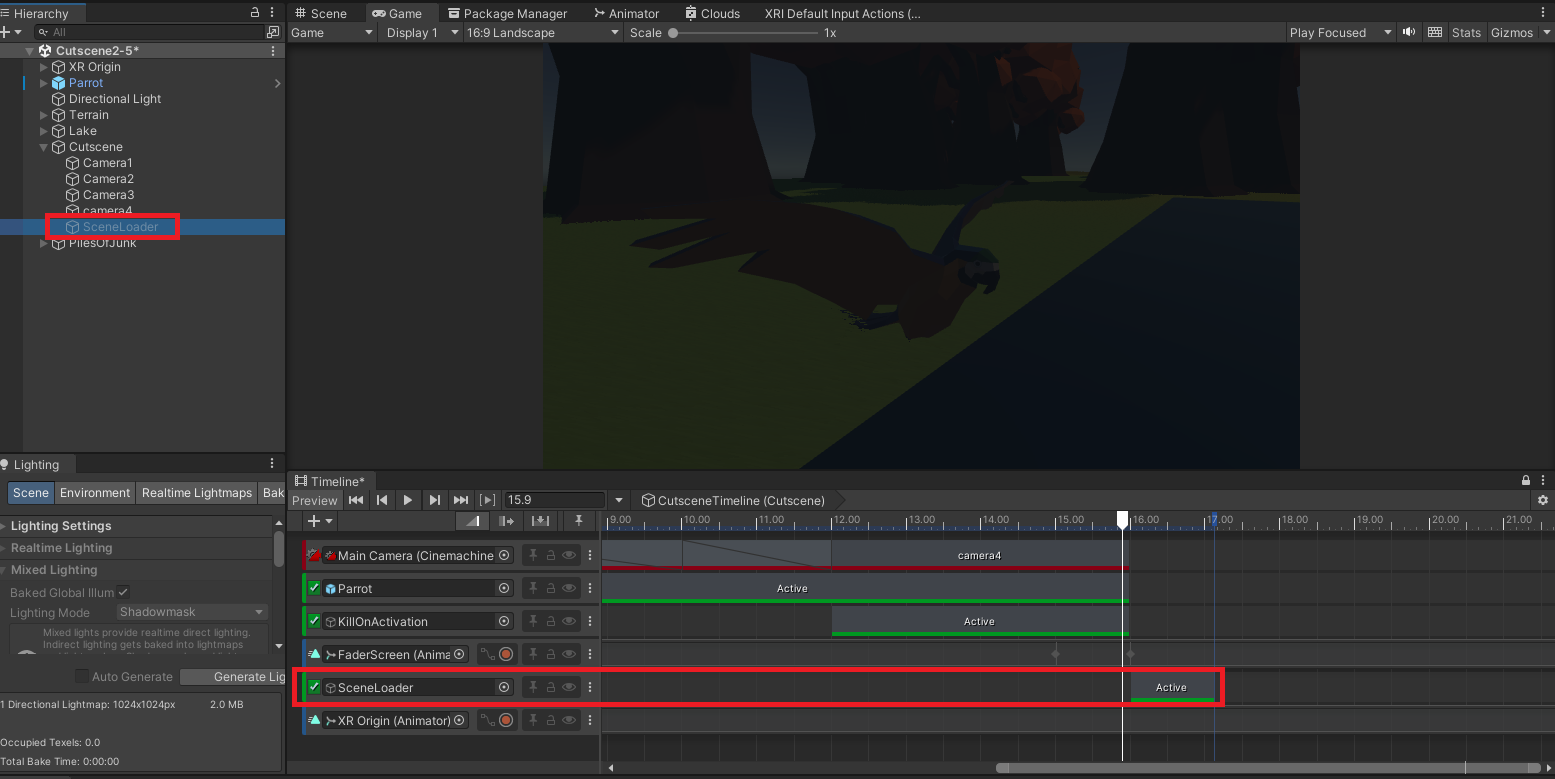
Interface gráfica do usuário, Site

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

Ao final de uma cinemática que leva a outra cinemática, a tela escurece e o objeto SceneLoader é ativado através da Timeline, conforme demonstrado na Figura 15. Esse objeto por sua vez possui um código vinculado a ele que utiliza o SceneManager do próprio Unity para carregar a próxima cena.

Figura - Ativação do objeto SceneLoader



Fonte: elaborado pelo autor.

Nos casos em que a cinemática deve carregar não outra cinemática, mas sim a cena de interação, o código vinculado ao objeto SceneLoader é o LoadInteractibleLevelTemplate, que pode ser visualizado no Quadro 6. Esse código recebe todos os parâmetros necessários para carregar a cena de interação, sendo eles o item com o qual o jogador irá interagir, o ângulo final que o item deve estar para poder avançar de cena, qual é a cena para a qual o jogador será direcionado ao concluir a interação e a cor de fundo que deverá ser utilizada na interação.

Quadro - Classe LoadInteractibleLevelTemplate

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

Conforme também é possível observar no Quadro 6, a classe LoadInteractibleLevelTemplate utiliza uma classe intermediária chamada Utils para determinar os parâmetros que serão passados para a cena de interação. Essa classe pode ser vista no Quadro 7, e é uma classe estática que tem duas utilidades principais. Sua primeira utilidade é justamente a passagem de parâmetros de maneira única entre as cenas cinemáticas e as de interação. A sua segunda funcionalidade é fornecer o método IsInRange para outras classes, que é utilizado para verificar se um vetor de três posições está dentro de um outro vetor de três posições, permitindo ter uma certa faixa de diferença aceitável para dizer que ambos possuem valores equivalentes.

Quadro - Classe Utils

Texto

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

Por fim, a outra classe que utiliza a Utils é a InteractibleObject, que é a principal classe responsável pela interação do jogador com os objetos, e está vinculada a todos os objetos com os quais é possível interagir. O cenário de interação consiste em um cubo que pode ter suas paredes mudadas de cor. O jogador estará dentro desse cubo e visualizando um dos múltiplos objetos possíveis para realizar sua interação. A princípio, todos os objetos com os quais é possível interagir estarão ativos na cena, mas aqueles que não corresponderem ao objeto que está armazenado na classe Utils serão desativados durante a interação, conforme é demonstrado no Quadro 8.

Quadro - Código que desativa o item caso ele não seja o objeto de interação

Tela de celular com texto preto sobre fundo branco

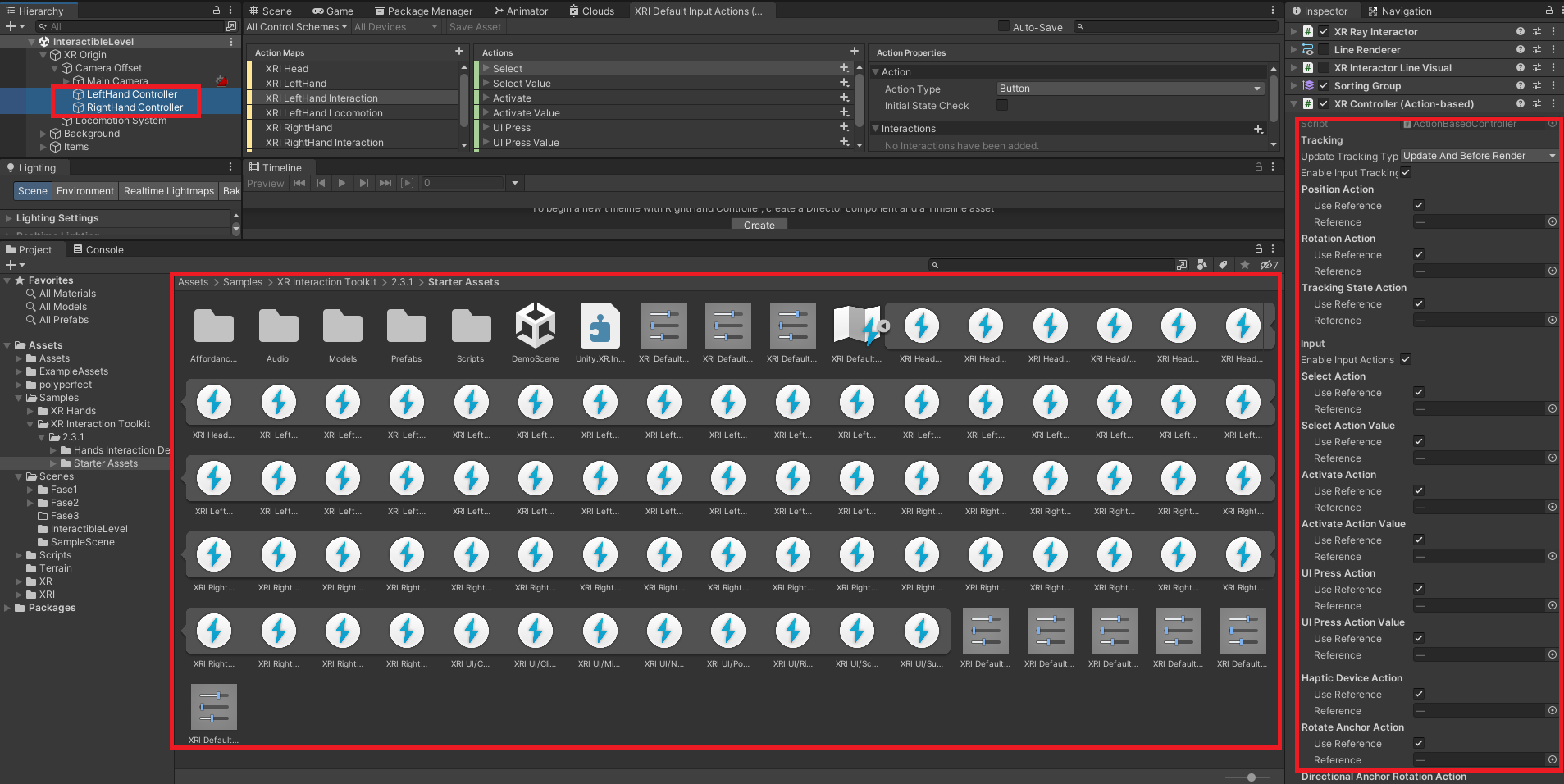
Descrição gerada automaticamente com confiança média

Fonte: elaborado pelo autor.

Para permitir a interação com o jogador através dos óculos de realidade virtual, é necessário existir um jeito de receber as entradas que o jogador realizará, sejam elas apertos de botões ou movimentações dos controles. Para isso, o jogo utiliza o XR Interaction Toolkit, que de acordo com a Unity Technologies (2023), é um sistema de interação para criação de experiências tanto em realidade virtual quanto realidade aumentada.

Esse Toolkit oferece, entre vários outros, um componente chamado XR Origin, que é utilizado para colocar o jogador no mundo do jogo efetivamente, além de armazenar as diferentes interações possíveis com o jogador através dos óculos de realidade virtual. Consequentemente, existe a possibilidade de configurar manualmente cada possibilidade de interação, mas a ferramenta também disponibiliza um conjunto padrão de configurações que pode ser baixado a parte e utilizado automaticamente. Todas as configurações podem ser visualizadas na Figura 16. Estão destacados à esquerda superior os componentes aos quais as configurações devem ser anexadas, à esquerda inferior todos os arquivos que vem com o pacote de Starter Assets que a ferramenta provê, e à direita os locais onde esses arquivos devem ser referenciados.

Figura - Arquivos que a ferramenta XR Interaction Toolkit provê



Fonte: elaborado pelo autor.

Após as configurações terem sido finalizadas, estará quase tudo pronto para receber as entradas que o jogador fornece através dos seus óculos VR. O único ponto que falta é gerar um código que ficará responsável por, em cada cena em que a interação ocorre, rastrear se existem controles válidos para serem utilizados e armazená-los de uma maneira que fique de fácil acesso para as classes que quiserem utilizá-los. Esse processo pode ser visualizado no Quadro 9, através da classe InputData. Nessa classe, a cada renderização será realizada uma validação, para verificar se os objetos correspondentes aos controles ainda estão válidos. Caso eles não estiverem, os métodos InitializeInputDevices e InitializeInputDevice serão chamados para tentar instanciar corretamente os objetos.

Quadro - Classe InputData

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Fonte: elaborado pelo autor.

Uma vez que a configuração é concluída e é possível acessar as entradas que o jogador fornece, e assim entra em cena mais uma vez a classe InteractibleObject. Conforme pode ser visualizado no Quadro 10, enquanto o jogador não interagir com o objeto, o objeto estará rodando sobre o próprio eixo. No momento em que o jogador começar sua interação, o método RotateItem faz com que o objeto receba a mesma rotação do controle do jogador. Por sua vez, o método CheckIfTheAngleIsRight realiza a validação da rotação do objeto, para ver se o jogador conseguiu chegar na rotação desejada e pode avançar para a próxima cena.

Quadro - métodos de interação da classe InteractableObject

Texto

Descrição gerada automaticamente com confiança média

Fonte: elaborado pelo autor.

# RESULTADOS

Nessa seção são apresentados os dois tipos de testes realizados com o jogo. Na primeira sunseção são discutidos os testes de funcionalidades do jogo, realizados durante o processo de desenvolvimento. Já na segunda subseção são discutidos os resultados oriundos da pesquisa realizada com os usuários que testaram o jogo após o desenvolvimento.

## testes de funcionalidade

Os testes de funcionalidade foram realizados constantemente durante o desenvolvimento do jogo, para garantir que as funcionalidades estivessem de acordo com o esperado. Esses testes foram realizados principalmente com a utilização do HMD Meta Quest 2. Parte dos testes também foram realizados diretamente no Unity, mas esses testes dificilmente representavam o que aconteceria ao rodar o jogo com o HMD, já que diferente do Unity, o HMD permite que o jogador olhe ao redor do cenário livremente.

Um dos pontos principais que faz com que testes sem o uso do HMD não representem o resultado corretamente é o fato de que a rotação da câmera utilizada durante a cinemática não representa necessariamente a rotação na qual o componente XR Origin utilizará. Isso pode fazer com que enquanto a câmera esteja focada no personagem principal da cena, o componente esteja virado, por exemplo, 180º no eixo x, perdendo assim todo o foco da cena, ou então obrigando o jogador a movimentar sua cabeça no mundo real. Devido a isso, era necessário continuamente realizar pequenos ajustes nas rotações desse componente junto das câmeras, para garantir que o jogador não perdesse o foco da história conforme ela avançasse.

Inicialmente, o objetivo era que o jogo fosse desenvolvido para celulares, de uma maneira que o jogador poderia utilizar um Google Cardboard, o que aumentaria muito a quantidade de pessoas que teriam acesso ao jogo, já que esses dispositivos são muito mais baratos do que um HMD imersivo. De qualquer maneira, se provou um grande desafio facilitar a interação do usuário com o objeto, e devido a isso, a plataforma de desenvolvimento foi alterada para o HMD, no caso o Meta Quest 2.

## testes com usuários

Após o desenvolvimento, foram realizados testes com quatro usuários. Não houve um local específico para a aplicação dos testes. Todos os testes foram realizados com a presença do autor, utilizando o HMD dele. Durantes a aplicação dos testes, foi primeiramente apresentada a ideia do jogo, e em seguida os jogadores começavam a jogar. Após todos terem concluído seus testes, foi solicitado o preenchimento do formulário.

Quadro - Respostas dos usuários

|  |  |  |
| --- | --- | --- |
| **Pergunta** | **Respostas** | **Quantidade** |
| Qual a sua faixa etária? | De 10 a 14 anos  De 19 a 25 anos  De 36 a 50 anos | 25%  25%  50% |
| Você já havia utilizado um dispositivo de realidade virtual imersiva antes? (Oculus/Meta Quest ou parecidos) | Sim | 100% |
| Numa escala de 0 a 10, quanto você gostou da sua experiência com o jogo Consequências? | 7  9  10 | 25%  25%  50% |
| Quão difícil você achou o jogo (escala 1[muito fácil] a 5 [muito difícil])? | 2  3  4 | 50%  25%  25% |
| Você sentiu algum tipo de tontura ou desconforto devido a movimentações abruptas da câmera durante o jogo? | Não | 100% |
| Foi possível entender a história de todas as fases? | Sim | 100% |
| Caso não, por gentileza, descreva qual(is) história(s) não foi possível compreender e por quê. | - |  |
| Na sua opinião, seria possível utilizar o jogo Consequências em escolas para ajudar na conscientização ambiental dos alunos? | Sim | 100% |
| Caso não, por gentileza, descreva o motivo. | - |  |

Fonte: elaborado pelo autor.

A última pergunta foi aberta, para que os jogadores pudessem deixar seus comentários sobre o que acharam do jogo e o que poderia ser melhorado. Foram obtidas três respostas para essa questão. Duas delas foram positivas e comentaram como gostaram da história do jogo e que ele é bem educativo. Houve um terceiro e último comentário que pontuou algumas coisas a melhorar no jogo, sendo elas: adicionar sons ao jogo; facilitar a interação do jogador com os objetos; realizar uma melhor ambientação dos cenários subaquáticos; adicionar animações em interações dos NPCs, como por exemplo ao passar por portas.

Os resultados são majoritariamente satisfatórios. O objetivo de fazer com que os jogadores tivessem facilidade de compreender a história sendo narrada foi concluído com êxito, e o fato de todos terem concordado que o jogo seria viável em uma sala de aula também é positivo para o objetivo. Todavia, alguns jogadores tiveram uma grande dificuldade com a interação com os objetos no jogo.

# CONCLUSÕES

Com base nos resultados obtidos, o jogo alcançou seu objetivo de utilizar a RV para promover a conscientização ambiental utilizando ilusão de ótica como mecânica principal de interação. A incorporação do tema “conscientização ambiental” foi facilmente implementada e agregou muito valor ao resultado do jogo. Outro ponto relevante é que um especialista que ajudou no desenvolvimento dos roteiros comentou sobre como o jogo pode ser utilizado para ajudar na compreensão do tema por crianças. Por fim, houve alguns pontos que não foram desenvolvidos conforme o esperado, sendo o principal deles a dificuldade que alguns jogadores apresentaram durante a interação do jogo.

O Unity se mostrou uma ferramenta mais do que suficiente para o desenvolvimento em RV e atendeu todas as necessidades com excelência. Algumas das ferramentas utilizadas possuíam uma curva de aprendizado mais atenuada, mas a utilização da linguagem C# balanceou essa curva. De qualquer maneira, a comunidade que existe ao redor do Unity facilitou o aprendizado e a correção dos problemas que apresentaram maior dificuldade. A integração do Unity com o Meta Quest 2 também foi simples e direta, sem dificuldades e rápida.

Quanto às ferramentas utilizadas: O XR Interaction Toolkit foi um dos processos mais complicados de realizar as configurações, e isso acabou incorrendo em problemas durante a implementação, mas uma vez configurado corretamente, funcionou conforme o esperado. No começo, a Timeline e a Cinemachine foram duas ferramentas complicadas de utilizar de maneira eficaz. De qualquer forma, uma vez compreendidas, elas aceleraram o processo de desenvolvimento das cinemáticas das cenas, fazendo com que as últimas cenas criadas tomassem cerca de um vigésimo do tempo que a primeira cena levou.

Apesar do sucesso e alcance dos objetivos propostos, foram identificadas algumas possibilidades para continuação e extensão da pesquisa desse projeto:

1. aumentar a quantidade de fases;
2. aumentar a quantidade de objetos com os quais é possível interagir;
3. disponibilizar o jogo também para Google Cardboard, facilitando assim o acesso a ele;
4. aplicar o jogo em turmas do ensino fundamental e avaliar o impacto realizado;
5. adicionar música e efeitos sonoros no jogo;
6. facilitar a interação do jogador com os objetos.

Referências

AZUMA, Ronald T. A survey of augmented reality. **Presence: Teleoperators & Virtual Environments**, Cambridge, v. 6, n. 4, p. 355-385, ago. 1997.

BACH, Michael; POLOSCHEK, Charlotte M. Optical illusions. **ACNR**, Mere, v. 6, n. 2, p.20-21, mar./jun.. 2006.

BRASIL. Ministério da Educação. Secretaria de Educação Básica. **Caderno meio ambiente**. Brasília, DF, 2022. Disponível em: http://basenacionalcomum.mec.gov.br/images/implementacao/cadernos\_tematicos/caderno\_meio\_ambiente\_consolidado\_v\_final\_27092022.pdf. Acesso em: 24 jun. 2023.

DING, Y.; LI, Y.; CHENG, L.. Application of Internet of Things and Virtual Reality Technology in College Physical Education. **IEEE Access**, New York, v. 8 n. 1, p.96065-96074, maio 2020.

Homburg A., Stolberg A. Explaining pro-environmental behavior with a cognitive theory of stress. **Journal of Environmental Psychology**, Amsterdan, v. 26, n. 1, p.1-14, mar. 2006.

IBISWORLD, **Global Movie Production & Distribution Industry** – Market Research Report. 2022. Disponível em: https://www.ibisworld.com/global/market-research-reports/global-movie-production-distribution-industry/. Acesso em: 25 jun. 2023.

META. **Get Meta** **Quest 2 for less**, 2023. Disponível em: https://www.meta.com/quest/products/quest-2. Acesso em: 25 jun. 2023.

NIENOW, Matheus N. **CidadaniaAR – Jogo de puzzle utilizando realidade aumentada com ilusão de ótica**. 2019. 24f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

NEWZOO, **Newzoo Global Games Market Report 2022 | Free Version**. 2023. Disponível em: https://newzoo.com/resources/trend-reports/newzoo-global-games-market-report-2022-free-version. Acesso em: 24 jun. 2023.

SÁNCHEZ, M.J.; LAFUENTE, Regina. Defining and measuring environmental consciousness. **Revista Internacional de Sociologia**, Madrid, v. 68, n. 3, p.731-735, set./dez. 2010.

SILVA R. W. C., PAULA B. L. Causa do aquecimento global: antropogênica versus natural. **Terræ Didatica**, São Paulo, v. 5 n. 1, p.42-49, jul. 2009.

SKELETON BUSINESS. **Vignettes**. 2017. Disponível em: https://vignettesga.me/. Acesso em: 18 set. 2022.

SUGANUMA, Elisa. **Percepção da ilusão de Müller-Lyer em macaco-prego (Cebus spp.)**. 2006. 111 f. Dissertação (Mestrado em Ciências da Saúde) - Universidade de Brasília, Brasília.

TAO, G. *et al*. Immersive virtual reality health games: a narrative review of game design. **Journal of NeuroEngineering and Rehabilitation**, v. 18, n. 31, p.1-21, fev. 2021.

UNITY TECHNOLOGIES, **Introduction to Timeline**, 2020a. Disponível em: https://learn.unity.com/tutorial/introduction-to-timeline-2019. Acesso em: 24 jun. 2023.

UNITY TECHNOLOGIES. **Overview of Cinemachine**, 2020b. Disponível em: https://learn.unity.com/tutorial/overview-of-cinemachine. Acesso em: 24 jun. 2023.

UNITY TECHNOLOGIES, **Working with NavMesh Agents**, 2021. Disponível em: https://learn.unity.com/tutorial/working-with-navmesh-agents. Acesso em: 24 jun. 2023.

UNITY TECHNOLOGIES, **XR Interaction** **Toolkit**, 2023. Disponível em: https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.3/manual/index.html. Acesso em: 24 jun. 2023.

USTWO GAMES. **Monument Valley**. 2014. Disponível em: https://www.monumentvalleygame.com/. Acesso em: 28 set. 2022.

APÊNDICE A – formulário de avaliação

Nesse apêndice é apresentado o formulário de avaliação utilizado durante os testes com usuários realizados após o desenvolvimento do jogo. O formulário é apenas um, mas devido ao seu tamanho, foi dividido na Figura 17 e na Figura 18.

Figura - Perguntas iniciais do formulário

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Fonte: elaborado pelo autor

Figura - Perguntas finais do formulário

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Fonte: elaborado pelo autor