

UNIOESTE - UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ
CENTRO DE ENGENHARIAS E CIÊNCIAS EXATAS
CURSO DE CIÊNCIA DA COMPUTAÇÃO

Plataforma Metaverso para Capacitação e Treinamento Imersivo

Luis Guilherme Vosgerau Baroni

FOZ DO IGUAÇU

2024

Luis Guilherme Vosgerau Baroni

Plataforma Metaverso para Capacitação e Treinamento Imersivo

Monografia submetida à Universidade Estadual do Oeste do Paraná, Curso de Ciência da Computação - Campus de Foz do Iguaçu, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Dr. Claudio Roberto Marquetto Mauricio

FOZ DO IGUAÇU

2024

Assinado eletronicamente por:
Claudio Roberto Marquetto Mauricio
CPF: ***.616.369-**
Data: 01/04/2025 16:08:08 -03:00



**Dr. Claudio Roberto Marquetto
Mauricio**
Orientador(a)

Assinado eletronicamente por:
Fabiana Frata Furlan Peres
CPF: ***.361.919-**
Data: 01/04/2025 13:58:26 -03:00



Drª. Fabiana Frata Furlan Peres
Membro

Assinado eletronicamente por:
Adriano Peretti Da Silva Santos
CPF: ***.868.149-**
Data: 01/04/2025 13:54:49 -03:00



Adriano Peretti da Silva Santos
Membro

Dedico este trabalho a todos que contribuíram para o meu crescimento e sucesso.

Agradecimentos

Agradeço, em primeiro lugar, a Deus por ter me dado forças para enfrentar os desafios e seguir em frente com determinação e fé. Agradeço à minha mãe, à minha avó e à minha namorada, por todo o amor, apoio e incentivo em cada etapa desta jornada. Sem o suporte de vocês, esta conquista não teria sido possível. Agradeço também ao restante da minha família, que sempre esteve ao meu lado, oferecendo apoio, palavras de encorajamento e força nos momentos mais difíceis. Sou grato aos meus professores da Unioeste e LabCG, que compartilharam não apenas conhecimento técnico, mas também lições de vida e inspiração para seguir em frente nos momentos difíceis. Um agradecimento especial ao meu orientador, pelo suporte, paciência e conselhos valiosos durante o desenvolvimento deste trabalho. Agradeço à equipe do CEASB, que contribuiu de maneira significativa para o meu desenvolvimento, tanto como pessoa quanto como profissional. O aprendizado e as experiências que tive com vocês foram essenciais para minha formação. Também agradeço aos meus amigos e colegas de curso, que tornaram essa caminhada mais leve e cheia de aprendizado. Os momentos de troca, apoio e parceria foram fundamentais para chegar até aqui.

*“Go placidly amid the noise and haste, and remember what peace there may be in
silence.”*
(Max Ehrman)

Resumo

A realidade virtual tem se consolidado como uma ferramenta eficiente para treinamento e capacitação em diversos setores, proporcionando experiências imersivas que simulam cenários reais com alta precisão. O uso de ambientes virtuais compartilhados e interativos permite que múltiplos usuários colaborem em tempo real, aumentando orealismo e a eficácia dos treinamentos. Este trabalho tem como objetivo o desenvolvimento de uma plataforma de metaverso, com um treinamento de procedimentos de manutenção em disjuntores de 500 kV, instalados na subestação blindada da usina de Itaipu. O sistema foi implementado utilizando o motor gráfico Unity, a biblioteca XR Interaction Toolkit para controle e manipulação de objetos em realidade virtual, e o Photon PUN 2 para a sincronização das ações em ambiente multiplayer. O treinamento foi estruturado com base em uma máquina de estados, permitindo a execução de tarefas complexas de forma sequencial e organizada. O sistema foi avaliado por meio de experimentos com usuários que realizaram as atividades de treinamento e responderam a questionários para análise de cinetose e usabilidade que resultou em uma média de 68.21 para a aplicação desenvolvida. A plataforma desenvolvida oferece ainda a possibilidade de expansão para novos treinamentos, permitindo a reutilização dos componentes e facilitando a adaptação a diferentes cenários operacionais.

Palavras-chaves: Simulação de Treinamento, Realidade Virtual, Aprendizagem ativa, Espaço Virtual

Lista de ilustrações

Figura 1 – Fotografia de um dos primeiros simuladores de voo, o <i>Link Trainer</i>	5
Figura 2 – Fluxo da prospecção e do desenvolvimento de um simulador de treinamento	7
Figura 3 – Fotografias de um dos primeiros dispositivos de realidade virtual, o <i>Sword of Damocles</i>	8
Figura 4 – <i>HoloLens</i> para um treinamento em realidade aumentada no setor de engenharia	10
Figura 5 – Representação do sistema integrador de realidade virtual com exoesqueleto	12
Figura 6 – Níveis de adoção da realidade virtual para os seis casos de uso definidos no setor AEC (Arquitetura, Engenharia e Construção)	15
Figura 7 – Modelo de <i>Gameful Design</i> e Gamificação	16
Figura 8 – Representação do <i>continuum</i> de realidade-virtualidade	20
Figura 9 – Linha do tempo da evolução dos dispositivos de RA e RV	21
Figura 10 – Comparação entre características da RV e RA, similaridades e diferenças	24
Figura 11 – Diagrama representativo solução proposta	29
Figura 12 – Funcionamento da ação de agarrar da biblioteca XR Interaction Toolkit	33
Figura 13 – Meta Quest 2 e notebook utilizado, com seus periféricos	36
Figura 14 – Modelo da GIS	42
Figura 15 – Modelo da sala de treinamento	43
Figura 16 – Realização da manutenção <i>overhaul</i>	44
Figura 17 – Capacete vermelho do tutor do treinamento	45
Figura 18 – Modelo do controle representativo da mão do jogador	47
Figura 19 – Teletransporte e seleção da direção final	48
Figura 20 – Diagrama representativo do fluxo da configuração de cada cena	50
Figura 21 – <i>Canvas</i> assíncrono no <i>Lobby</i>	52
Figura 22 – <i>Canvas</i> síncrono na sala de treinamento	53
Figura 23 – Ação de segurar objeto	56
Figura 24 – Máquina de estados do treinamento implementado	57
Figura 25 – Transpaleteira movida após primeira ação	58
Figura 26 – Gancho movido para baixo após pressionar o botão	59
Figura 27 – Gancho movido para cima após acoplar acumulador hidráulico	60
Figura 28 – Guindaste movido para o lado após movimento para cima	60

Figura 29 – Peças sujas removidas do acumulador	61
Figura 30 – <i>Canvas</i> final do treinamento, com as duas opções possíveis	62
Figura 31 – Funções chamadas pelo botão de carregar sala como <i>trainee</i>	67
Figura 32 – Fotografia da realização do experimento.	71

Lista de tabelas

Tabela 1 – Métodos de sincronização utilizados em cada etapa do treinamento	65
Tabela 2 – Resultados individuais do SSQ	74
Tabela 3 – Estatísticas descritivas da análise do SSQ	74
Tabela 4 – Estatísticas descritivas da análise do SUS	76

Lista de Abreviaturas e Siglas

3D Tridimensional

6DOF Six Degrees of Freedom

ABB Asea Brown Boveri

AEC Arquitetura, Engenharia e Construção

API Application Programming Interface

AVC Acidente Vascular Cerebral

CEASB Centro Avançado em Soluções para Barragens

FSM Finite State Machine

GIS Gas Insulated Substation

HMD Head-Mounted Display

HUD Heads-Up Display

MT Mirror Therapy

NFT Non-Fungible Token

PC Computador Pessoal

PUN Photon Unity Netwo

RA Realidade Aumentada

RPC Remote Procedure Call

RV Realidade Virtual

SSQ Simulator Sickness Questionnaire

SUS System Usability Scale

USAF United States Air Force

XR Extended Reality

Sumário

1	Introdução	1
1.1	Objetivo	3
1.1.1	Objetivo Geral	3
1.1.2	Objetivos Específicos	3
2	Revisão da Literatura	4
2.1	Simuladores de Treinamentos	4
2.1.1	Conceito e Terminologias	4
2.1.2	Histórico	4
2.1.3	Ascensão de Dispositivos	7
2.1.4	Áreas de Aplicação	11
2.2	Gamificação em Simuladores de Treinamento	16
2.2.1	Origem e Primeiros Aparições de Gamificação em Simuladores	16
2.2.2	Aplicações em Diversas Áreas	17
2.2.3	Benefícios da Gamificação em Simuladores de Treinamento	18
2.2.4	Consolidação dos <i>Serious Games</i>	18
2.3	Realidade Virtual e Realidade Aumentada	19
2.3.1	Definições e Histórico	19
2.3.2	Evolução dos Dispositivos de Realidade Aumentada e Realidade Virtual	20
2.3.3	Realidade Virtual	22
2.3.3.1	Aplicações e Benefícios	23
2.3.4	Realidade Aumentada	23
2.3.4.1	Aplicações e Benefícios	23
2.3.5	Comparação e Oportunidades	24
2.4	Metaverso	25
2.4.1	Definição e Estrutura do Metaverso	25
2.4.2	Histórico do Metaverso	26
2.4.3	Relação com Realidade Virtual e Realidade Aumentada	26
2.4.4	Aplicações do Metaverso	26
2.4.4.1	Educação e Treinamento	26
2.4.4.2	Colaboração e Trabalho Remoto	27
2.4.4.3	Entretenimento e Socialização	27
2.4.4.4	Economia Digital	27
2.4.5	O Futuro do Metaverso	27

3 Proposta da Solução	29
4 Materiais e Métodos	31
4.1 <i>Software</i>	31
4.1.1 Unity	31
4.1.2 C#	32
4.1.3 XR Interaction Toolkit	32
4.1.4 Photon PUN 2	33
4.1.5 Blender	34
4.1.6 GitHub	34
4.1.7 Parallel Sync	34
4.2 <i>Hardware</i>	35
4.2.1 Meta Quest 2 e 3	35
4.2.2 PCs e Notebooks	35
5 Requisitos e Funcionalidades	37
5.1 Funcionalidades	37
5.2 Requisitos	39
5.2.1 Requisitos Funcionais	39
5.2.2 Requisitos Não Funcionais	40
6 Cenários e Modelos Utilizados	41
6.1 <i>Lobby</i>	41
6.2 Sala de Treinamento	42
6.3 Modelagem do Capacete e da Lente	43
6.4 Contextualização sobre o Treinamento	44
7 Desenvolvimento	45
7.1 Personagem do Usuário	45
7.1.1 Controles e Modos de Interação	46
7.1.2 Locomoção	47
7.2 Gerenciador de Salas	48
7.3 Gerenciador da Sala de Treinamento	49
7.4 Configuração do Jogador	49
7.5 Canvas e Interface de Usuário	50
7.6 Troca de Sala e Sincronização de Estado	51
7.6.1 Desenvolvimento e Configuração	51
7.7 <i>Canvas</i> Síncrono e Sincronização de Estado	52
7.8 Sincronização de Estado com ‘SyncActivation’	53
7.9 Sincronização de Animações com ‘SyncAnimation’	54

7.10	Sincronização de Comportamento com ‘GrabSync’	55
7.11	Considerações sobre Sincronização	56
7.12	Gerenciamento de Estados no Treinamento	56
7.12.1	Estado 1: Movimentação da Transpaleteira	58
7.12.2	Estado 2: Movimento do Gancho para Baixo	58
7.12.3	Estado 3: Movimento do Gancho com o Acumulador Hidráulico para Cima	59
7.12.4	Estado 4: Movimento do Guindaste com o Acumulador para o Lado	60
7.12.5	Estado 5: Remoção das Peças Sujas do Acumulador	61
7.12.6	Estado 6: Limpeza das Peças	61
7.13	Tela de Retorno ou Repetição do Treinamento	61
7.14	Sincronização dos Estados no <i>Multiplayer</i>	62
7.14.1	Componentes de Sincronização	63
7.14.2	Sincronização via RPC para Interface	63
7.14.3	Sincronização Contínua de Posição com RPC	64
7.14.4	Sincronização de Animações com RPC	64
7.15	Gerenciadores de Salas Adaptados	65
7.16	Integração de Novos Treinamentos	66
8	Resultados	69
8.1	Avaliação do Sistema	69
8.1.1	Objetivo da Avaliação	69
8.1.2	Metodologia de Avaliação	70
8.1.3	Execução das Atividades	71
8.1.3.1	Atividade 1 (AT1): Movimentação e operação de uma transpaleteira	71
8.1.3.2	Atividade 2 (AT2): Manutenção de um cilindro hidráulico	71
8.1.4	Coleta de Dados	72
8.2	Análise dos Resultados do <i>Simulator Sickness Questionnaire (SSQ)</i>	72
8.2.1	Estatísticas Gerais	74
8.2.2	Discussão dos Resultados	74
8.3	Análise dos Resultados de Usabilidade (<i>System Usability Scale - SUS</i>)	75
8.3.1	Discussão dos Resultados	75
9	Conclusões e Trabalhos Futuros	77
9.1	Trabalhos Futuros	78
Referências	81	
A System Usability Scale (SUS)	85	

1 Introdução

A prática de simulação em treinamento tem raízes profundas, remontando ao início do século XX. Durante a Primeira Guerra Mundial, essa abordagem ganhou destaque, quando surgiu a necessidade urgente de métodos eficazes para capacitar pilotos de aeronaves. Um dos primeiros simuladores de voo, o *Link Trainer*, foi criado por Edwin Link na década de 1920, permitindo que pilotos praticassem navegação e procedimentos de voo em um ambiente seguro e controlado, sem os riscos inerentes ao voo real [RADIANTI et al., 2020; LI; CHEN, 2022].

Com o avanço da tecnologia digital na década de 1960, os simuladores começaram a incorporar sistemas de computação, proporcionando simulações mais precisas e realistas. A indústria de aviação civil e militar liderou a adoção dessas tecnologias, desenvolvendo simuladores que podiam replicar condições de voo e cenários de emergência complexos. Esse período também marcou o início da aplicação de simuladores em outras áreas, como treinamento de condução e operações marítimas [RADIANTI et al., 2020].

Na década de 1980, com o advento dos computadores pessoais, a gamificação começou a ser integrada aos simuladores de treinamento. A gamificação introduziu elementos de jogos, como pontuações, níveis e desafios, tornando o treinamento mais envolvente e motivador. Esta abordagem mostrou-se eficaz em aumentar a retenção de conhecimento e o engajamento dos trainees, sendo amplamente adotada em setores como a saúde, para treinar cirurgiões e outros profissionais médicos em procedimentos complexos e raros [JARAMILLO-MEDIAVILLA et al., 2024; DETERDING et al., 2011].

Com a evolução das tecnologias, a transição para plataformas de realidade aumentada (RA) e realidade virtual (RV) trouxe benefícios significativos. Enquanto a simulação tradicionalmente se refere à replicação de condições reais em um ambiente controlado, a evolução para RA e RV permite uma integração mais sofisticada entre o mundo físico e o digital. De acordo com Milgram e Kishino [1994], a simulação em RV pode ser vista como parte de um *continuum* de realidades mistas, onde a realidade virtual representa a imersão total em um ambiente gerado por computador, desconectado do mundo físico. Nesse cenário, o usuário é completamente imerso em um ambiente virtual 3D (tridimensional), onde todas as percepções são manipuladas digitalmente. Por outro lado, a RA está na outra extremidade desse *continuum*, onde elementos digitais são sobrepostos ao ambiente físico, enriquecendo a percepção do mundo real com informações virtuais. Azuma [1997] define a RA como a tecnologia que combina elementos virtuais e reais de maneira interativa e em tempo real, mantendo a percepção do ambiente físico.

Segundo Radianti et al. [2020], a imersão e a interatividade são elementos cruciais

que diferenciam a RV de outros tipos de multimídia educacional. Estudos indicam que *setups* mais imersivos, como os proporcionados pela RV, resultam em um maior sentimento de presença, definido como a sensação subjetiva de “estar lá” dentro da simulação. Esse maior sentimento de presença tem sido associado a níveis mais altos de engajamento em tarefas de aprendizado do que métodos tradicionais [MAKRANSKY; PETERSEN, 2021].

A realidade virtual tem se mostrado eficaz em várias áreas, como gestão do conhecimento, testes de usabilidade industrial e engenharia. Gurian, Bolisani e Kirchner [2023] destacam que a RV melhora o engajamento e a retenção de conhecimento dos usuários, criando ambientes imersivos e interativos que facilitam o aprendizado e a colaboração. Esses benefícios são cruciais em organizações intensivas em conhecimento, onde a disseminação eficiente de informações é essencial.

Além disso, Freitas, Gomes e Winkler [2022] identificam que a aplicação de RV em testes de usabilidade industrial proporciona economia de custos e tempo, além de aumentar a segurança dos processos. A RV permite a visualização detalhada de protótipos e a realização de testes em ambientes virtuais, reduzindo a necessidade de protótipos físicos e minimizando riscos associados a testes em ambientes reais.

Em especial, a RV também é amplamente utilizada em treinamentos de engenharia civil, mecânica e elétrica, onde os profissionais podem interagir com modelos tridimensionais de infraestrutura, operar equipamentos virtuais e realizar diagnósticos em sistemas complexos sem os riscos associados ao mundo físico [SAMPAIO et al., 2010]. Além disso, o uso de simuladores em RV tem sido eficaz na formação de engenheiros, promovendo um aprendizado mais interativo e imersivo, conforme demonstrado por diversas pesquisas na área [ABULRUB; ATTRIDGE; WILLIAMS, 2011].

Górski [2017] aborda a personalização de soluções de RV para engenharia, demonstrando como essas aplicações podem ser adaptadas para resolver problemas específicos em diferentes áreas da engenharia. Soluções customizadas permitem que engenheiros visualizem, simulem e otimizem projetos complexos de maneira mais eficiente e precisa.

Com o desenvolvimento das tecnologias, o conceito de metaverso tem ganhado destaque como uma ferramenta potencial para melhorar a gestão do conhecimento e a eficácia dos treinamentos em setores industriais críticos. O metaverso é definido como um ambiente virtual imersivo e persistente onde usuários podem interagir em tempo real, facilitado por tecnologias como RV e RA [WEINBERGER, 2022].

Diante desse panorama de avanços, o uso de ambientes imersivos e do metaverso para treinamento apresenta um potencial transformador. A aplicação de realidade virtual, combinada à interatividade e à capacidade de personalização dos conteúdos, oferece uma experiência de aprendizado adaptada às demandas de setores críticos, como a engenharia e a manutenção industrial. Este trabalho explora essa convergência de tecnologias,

propondo uma plataforma de treinamento que utiliza RV para simular cenários de alto risco em um ambiente seguro, colaborativo e interativo. Ao investigar as possibilidades do metaverso como ferramenta de capacitação, esta pesquisa busca contribuir para a construção de metodologias mais eficazes e seguras no treinamento profissional, especialmente em áreas onde a precisão e o conhecimento prático são essenciais.

1.1 Objetivo

1.1.1 Objetivo Geral

Desenvolver uma plataforma de metaverso para treinamento em realidade virtual que permita a interação em tempo real entre tutores e usuários *trainees*.

1.1.2 Objetivos Específicos

- Desenvolver a plataforma metaverso com as funcionalidades definidas
- Implementar um protótipo de treinamento para incluir no metaverso, seguindo conceitos de gamificação
- Realizar testes de usabilidade do *software*

2 Revisão da Literatura

2.1 Simuladores de Treinamentos

2.1.1 Conceito e Terminologias

O conceito de simuladores de treinamento se baseia na ideia de replicar, em ambiente controlado, condições e cenários do mundo real para que os usuários possam praticar e desenvolver habilidades específicas. Conforme definido por Rolfe e Staples [1986], um simulador de treinamento é um sistema que imita os aspectos críticos de uma tarefa ou ambiente operacional, permitindo ao usuário adquirir experiência prática e responder a cenários simulados de forma segura e eficaz. Esse tipo de treinamento é especialmente útil em situações onde o erro pode ter consequências graves, como na aviação, operações militares e procedimentos médicos.

Ao recriar as condições reais, simuladores oferecem uma experiência de aprendizado imersiva e controlada. A eficácia dos simuladores se baseia na capacidade de reproduzir estímulos sensoriais e reações físicas semelhantes às que o usuário encontraria no ambiente real. Isso permite que o *trainee*, ou seja, o indivíduo em processo de formação prática para desenvolver habilidades específicas, adquira experiência em um ambiente seguro e controlado, que simula as exigências do contexto real de sua futura atuação.

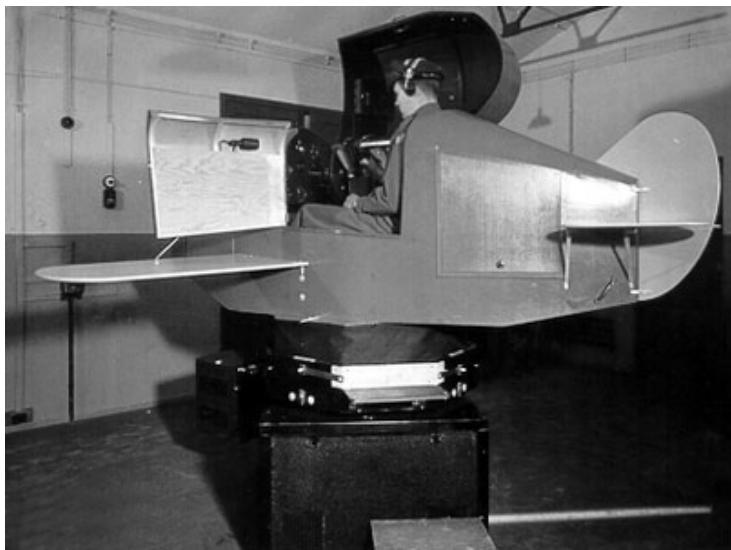
O papel do *trainee* é fundamental no processo de treinamento, pois ele transita do conhecimento teórico para a prática, construindo não apenas uma memória muscular, mas também uma memória cognitiva que poderá ser aplicada diretamente em situações reais. Essa prática é essencial em áreas que exigem respostas rápidas e precisas, como aviação, saúde e operações militares, onde o *trainee* pode desenvolver reflexos e habilidades práticas que promovem a transição para a competência real. Desde os primeiros simuladores de voo, que permitiam aos pilotos praticar manobras básicas sem deixar o solo, até os dispositivos modernos que integram *feedback* visual e tátil, a simulação no treinamento visa proporcionar um aprendizado intensivo e imersivo, mitigando os riscos e preparando o *trainee* para enfrentar desafios complexos do mundo real.

2.1.2 Histórico

O conceito de simuladores de treinamento remonta ao início do século XX, período em que a simulação foi empregada pela primeira vez para auxiliar no desenvolvimento de habilidades práticas sem expor os aprendizes a riscos diretos. Essa abordagem tornou-se especialmente proeminente durante a Primeira Guerra Mundial, quando a necessidade

urgente de capacitar pilotos de aeronaves motivou a criação de dispositivos específicos para simulação de voo. Um dos marcos desse período foi o desenvolvimento do *Link Trainer*, projetado por Edwin Link na década de 1920, conforme ilustrado na Figura 1. Este simulador mecânico permitia que pilotos praticassem navegação e procedimentos de voo em um ambiente controlado, reproduzindo condições básicas de operação e controle sem os perigos do voo real. Sua inovação e eficácia tornaram-no uma ferramenta essencial na aviação militar e civil, sendo amplamente utilizado até meados do século XX [HAYS; SINGER, 1989].

Figura 1 – Fotografia de um dos primeiros simuladores de voo, o *Link Trainer*



Fonte: [TONG; BIL, 2007]

A partir da década de 1960, o desenvolvimento tecnológico possibilitou a introdução de sistemas de computação nos simuladores, proporcionando simulações mais realistas e precisas. Esse avanço foi liderado pela indústria de aviação, que rapidamente adotou simuladores digitais capazes de replicar condições de voo com alto nível de detalhe, incluindo cenários de emergência complexos. Essa evolução não apenas aprimorou a eficácia do treinamento de pilotos, mas também demonstrou o potencial dos simuladores para outras áreas, como condução de veículos e operações marítimas. O aprimoramento desses dispositivos com sistemas de controle e sensores sofisticados, que elevavam a fiabilidade da simulação, marcou o início de uma nova era para os simuladores [ROLFE; STAPLES, 1986].

Na década de 1980, com o surgimento dos computadores pessoais (PCs), os simuladores tornaram-se mais acessíveis e começaram a ser utilizados em setores adicionais. Além de estarem presentes na aviação, os simuladores foram adaptados para áreas como saúde e segurança, onde a prática em ambiente simulado permite o desenvolvimento de habilidades críticas sem expor os *trainees* a riscos reais. Nesse período, houve também um aumento no interesse por aplicações em treinamento militar e operações industriais,

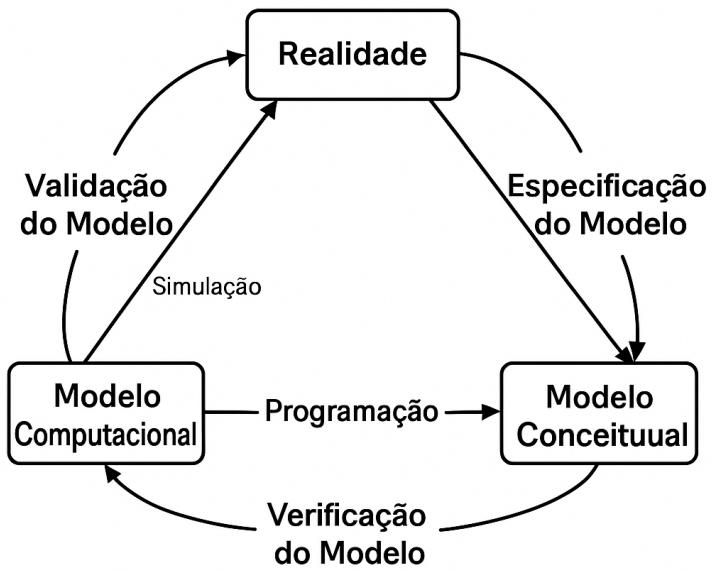
nos quais os simuladores eram utilizados para preparar trabalhadores para operações de alta complexidade e risco. Esses primeiros simuladores computacionais estabeleciam as bases para sistemas cada vez mais precisos e abrangentes, tornando-se indispensáveis no treinamento técnico e prático de diversos setores profissionais.

Nos anos seguintes, os simuladores continuaram a evoluir, beneficiando-se de avanços em fidelidade visual e sensorial. A introdução de simuladores com sistemas de realidade aumentada e realidade virtual nos anos 1990 e 2000 ampliou ainda mais suas capacidades, permitindo níveis superiores de imersão e interatividade. Nessa fase, áreas como medicina e segurança ganharam destaque, pois os simuladores permitiam práticas que imitavam condições de trabalho intensas e complexas, muitas vezes impossíveis de serem simuladas com segurança em ambientes reais. Além disso, esses novos simuladores ampliaram a utilização de dados em tempo real para criar cenários variáveis e dinâmicos, adaptando as condições da simulação ao desempenho dos usuários e facilitando o treinamento progressivo e contínuo.

Atualmente, os simuladores de treinamento são aplicados em uma vasta gama de setores, indo além da aviação e da condução para alcançar a medicina, a engenharia civil, a segurança industrial e o controle de processos em indústrias de risco. Em cada uma dessas áreas, os simuladores permitem que os profissionais pratiquem, em ambiente seguro, operações que exigem precisão, resposta rápida e atenção constante, reduzindo o risco de erros críticos no exercício de suas funções. As abordagens modernas, que integram elementos sensoriais e visuais realistas, têm permitido treinar profissionais para reagir em situações de alto estresse, onde a preparação e a experiência prática podem ser fatores determinantes na prevenção de acidentes e no aprimoramento da performance [ROLFE; STAPLES, 1986; HAYS; SINGER, 1989].

O desenvolvimento de simuladores de treinamento segue um fluxo contínuo, onde as etapas de especificação do modelo, programação, validação e verificação do modelo são realizadas de forma interligada, conforme demonstrado na Figura 2. Este ciclo contínuo de validação e análise do modelo computacional e conceitual é essencial para garantir que o simulador seja capaz de refletir com precisão a realidade, adaptando-se às necessidades do treinamento e oferecendo uma experiência de aprendizado eficaz e segura.

Figura 2 – Fluxo da prospecção e do desenvolvimento de um simulador de treinamento



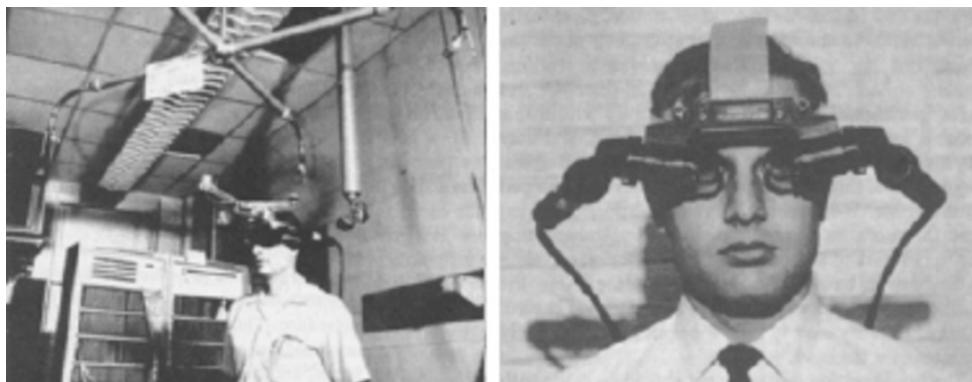
Fonte: [ATASHGAH; BAHRAMI; NASIRI, 2005]

Em suma, o desenvolvimento dos simuladores de treinamento reflete uma trajetória contínua de aprimoramento e especialização. Desde o início com dispositivos mecânicos básicos, passando por sistemas computacionais até os atuais simuladores imersivos de realidade virtual, os simuladores têm evoluído para proporcionar um aprendizado mais seguro e efetivo, permitindo que profissionais de diversas áreas pratiquem e aprimorem suas habilidades em ambientes seguros e altamente controlados.

2.1.3 Ascensão de Dispositivos

A trajetória de desenvolvimento dos dispositivos de suporte a simuladores de treinamento teve início com a criação de tecnologias que possibilitavam a simulação de ambientes controlados. A partir da década de 1960, essas tecnologias começaram a incluir dispositivos visuais e tátteis que permitiam aos usuários experimentar a sobreposição de dados informativos em ambientes físicos. O primeiro dispositivo de grande importância foi o *Sword of Damocles* de Ivan Sutherland, criado em 1968 [SUTHERLAND, 1968], conforme visto na Figura 3. Esse dispositivo era um dos primeiros exemplos de um *display* montado na cabeça (HMD), e apesar de ser rudimentar, ele representava uma abordagem inovadora ao sobrepor gráficos tridimensionais ao ambiente do usuário, permitindo um treinamento imersivo para tarefas específicas [MILGRAM; KISHINO, 1994]. Esse sistema foi um marco, pois permitiu que operadores praticassem interações controladas em um ambiente simulado, criando uma experiência segura e educativa.

Figura 3 – Fotografias de um dos primeiros dispositivos de realidade virtual, o *Sword of Damocles*



Fonte: [KREVELEN, 2007]

Nos anos 1970, a tecnologia começou a expandir seu alcance para campos práticos, com dispositivos focados em HUDs (*head-up displays*) que permitiam aos pilotos de aeronaves militares visualizar informações de voo diretamente no campo de visão. Essa tecnologia possibilitou que informações críticas, como altitude, velocidade e orientação, fossem integradas ao campo de visão sem desviar a atenção do piloto, otimizando o treinamento em condições controladas. O uso do HUD foi pioneiro na Força Aérea e se consolidou como um dispositivo essencial para o treinamento de pilotos, pois permitia que eles respondessem a cenários simulados com informações em tempo real, sem a necessidade de olhar para os instrumentos no painel [ROLFE; STAPLES, 1986].

A década de 1980 foi marcada pelo uso crescente de sistemas de realidade aumentada em ambientes industriais. Empresas como a Boeing começaram a utilizar HMDs para melhorar o treinamento e a eficiência na montagem de aeronaves. Um exemplo notável foi o uso de sobreposições visuais para guiar técnicos em tarefas complexas de montagem, sobrepondo instruções e diagramas diretamente sobre o que o técnico visualizava. Isso não só aprimorou a precisão das operações como também permitiu o treinamento prático e contínuo em tempo real, sem a presença constante de supervisores. Esses sistemas começaram a aplicar *feedback* visual e auditivo, tornando-se componentes cruciais em simuladores de treinamento que exigiam precisão e repetição constante.

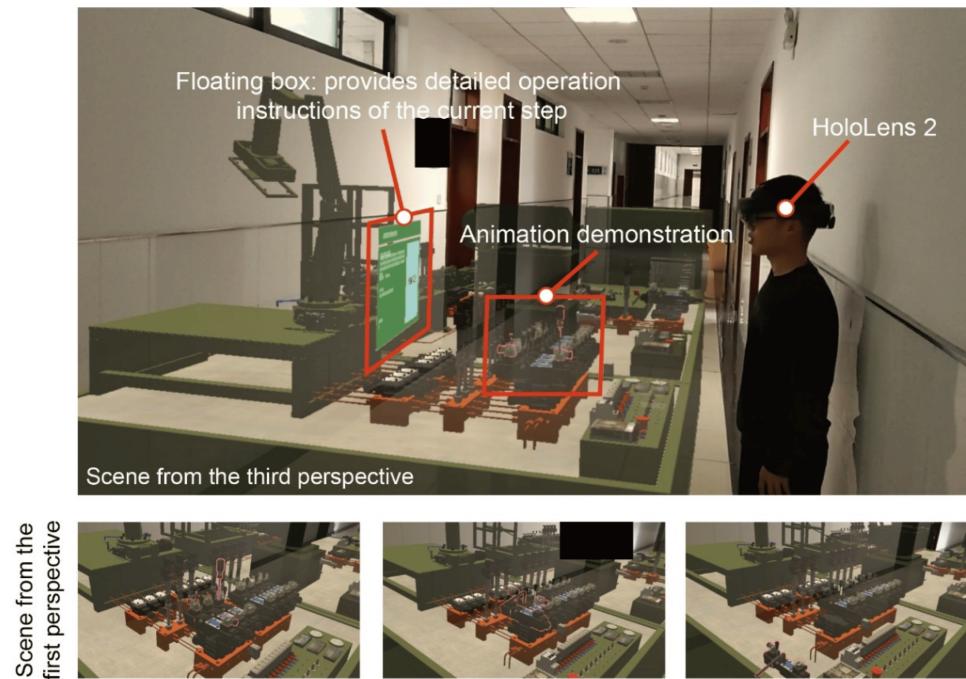
Na década de 1990, o desenvolvimento de dispositivos como o *Virtual Fixtures*, criado pelo Laboratório de Pesquisa da Força Aérea dos Estados Unidos (USAF), revolucionou o treinamento para tarefas que requerem precisão tátil. Esse sistema combinava a RA com tecnologia háptica, criando um ambiente onde os usuários recebiam *feedback* físico ao manipular ferramentas virtuais sobrepostas no campo visual. Dessa forma, os *trainees* podiam sentir resistência e texturas simuladas, o que era essencial para o treinamento em operações cirúrgicas e manutenção técnica de equipamentos delicados. O *Virtual Fixtures* se tornou um dos primeiros dispositivos a unir a realidade aumentada com simulação tátil, elevando orealismo e a aplicabilidade dos simuladores de treinamento [AZUMA, 1997].

A transição para os anos 2000 trouxe dispositivos de RA mais compactos e especializados. *Headsets* da Vuzix, por exemplo, foram introduzidos para oferecer um sistema leve e portátil de sobreposição de dados, especialmente em setores industriais e militares. Esse tipo de tecnologia permitia que os operadores visualizassem informações críticas enquanto realizavam operações de campo, como a manutenção de veículos e a operação de máquinas pesadas. Esses dispositivos se destacaram por sua capacidade de transformar o treinamento de tarefas arriscadas em um ambiente controlado, ao projetar procedimentos e alertas diretamente no campo visual dos *trainees*. Essa inovação foi particularmente valiosa para áreas como o treinamento em plataformas de petróleo e a operação de maquinário pesado, onde cada erro pode representar um risco significativo.

Em 2013, o Google Glass trouxe uma abordagem mais acessível e compacta de dispositivos de realidade aumentada para o público geral, mas seu uso em treinamento foi limitado. Somente em 2016, com o lançamento do Microsoft HoloLens, as possibilidades de RA em simuladores de treinamento foram elevadas a um novo patamar. O HoloLens combinou mapeamento espacial e interação por gestos, possibilitando o treinamento de profissionais em ambientes virtuais sobrepostos ao mundo físico. Este dispositivo permitiu que os usuários visualizassem e interagissem com hologramas em seus ambientes reais, aplicando essa tecnologia ao treinamento em setores como a medicina, engenharia e manufatura. Esse nível de realismo imersivo e interativo marcou o início da RA como uma tecnologia integrada e funcional nos simuladores de treinamento moderno [FREITAS; GOMES; WINKLER, 2022; WEINBERGER, 2022].

A Figura 4 ilustra o uso do *HoloLens 2* em um treinamento em realidade aumentada no setor de engenharia, onde um usuário interage com a simulação em um ambiente físico, recebendo instruções detalhadas sobre a tarefa atual por meio de uma interface flutuante e visualizando a demonstração da animação, o que contribui para a aprendizagem e melhora a eficácia do treinamento.

Figura 4 – *HoloLens* para um treinamento em realidade aumentada no setor de engenharia



Fonte: [LIU et al., 2022]

Os conceitos vistos na Figura 4 acima podem ser traduzir livremente como:

- **Floating box:** Caixa flutuante.
 - **Animation demonstration:** Demonstração de animação.
 - **HoloLens 2:** HoloLens 2.
 - **Scene from the first perspective:** Cena da primeira perspectiva.
 - **Scene from the third perspective:** Cena da terceira perspectiva.

A Figura 4 ilustra uma aplicação do *HoloLens 2*, tecnologia de realidade aumentada, em um ambiente de operação industrial. A caixa flutuante na interface fornece instruções detalhadas sobre a operação do passo atual, auxiliando na execução de tarefas. Além disso, a demonstração de animação apresenta uma visualização dos processos, promovendo uma compreensão mais clara do procedimento. As cenas mostram diferentes perspectivas do ambiente, permitindo uma interação mais intuitiva e imersiva com o espaço e os processos apresentados.

Cada um desses avanços, dos HUDs militares aos *headsets* modernos de realidade aumentada, representou um salto significativo no campo dos simuladores de treinamento. Hoje, dispositivos de RA modernos oferecem um treinamento integrado e em tempo real, permitindo que trabalhadores e técnicos treinem em situações complexas e de alto risco.

em um ambiente seguro e controlado, reforçando habilidades e conhecimentos práticos que podem ser diretamente aplicados no mundo real.

2.1.4 Áreas de Aplicação

Os simuladores de treinamento têm sido amplamente aplicados em diversas áreas, proporcionando um ambiente controlado para o desenvolvimento de habilidades complexas e a prática de tarefas em situações de alto risco. Uma das primeiras áreas a adotar simuladores foi a aviação, que identificou a necessidade de treinamento prático para capacitar pilotos sem expô-los aos perigos de voos reais. Dispositivos como o *Link Trainer*, desenvolvido na década de 1920, permitiram que pilotos de guerra e civis praticassem navegação e manobras, evitando riscos e preparando-os para enfrentar uma variedade de situações de voo [RADIANTI et al., 2020; LI; CHEN, 2022]. Atualmente, simuladores de voo evoluíram para equipamentos altamente sofisticados, que replicam condições de voo reais, incluindo turbulência, falhas de sistema e aterrissagens de emergência, permitindo que os pilotos adquiram experiência prática em uma gama completa de cenários possíveis.

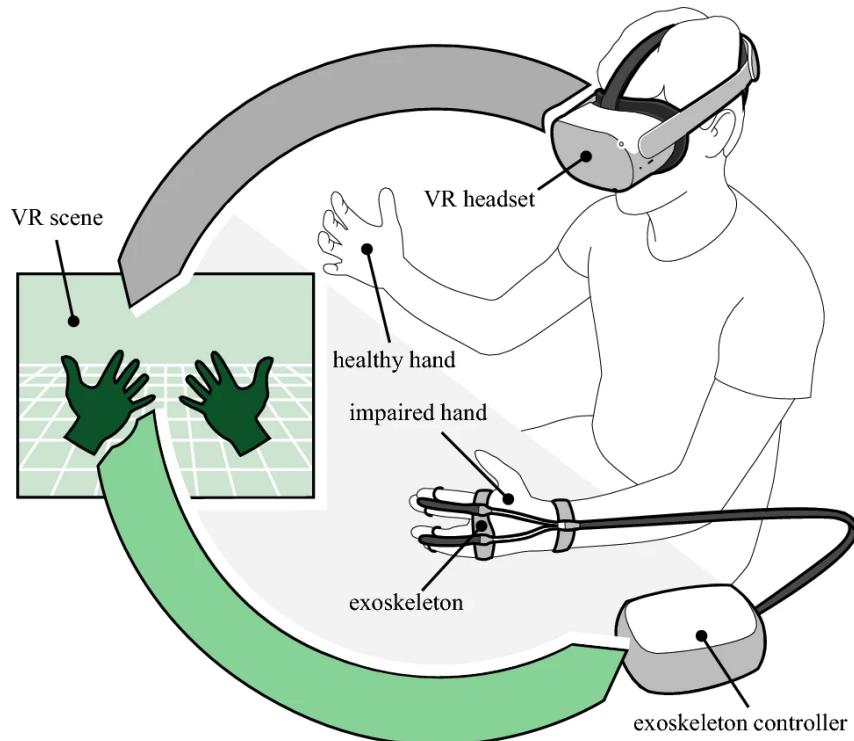
No setor militar, o uso de simuladores se expandiu significativamente a partir da Segunda Guerra Mundial. Inicialmente aplicados para treinamento de pilotos e motoristas de tanques, esses simuladores evoluíram para sistemas que replicam situações de combate em terra, ar e mar. Através de simulações imersivas que combinam efeitos visuais, sonoros e até táteis, os soldados podem se preparar para missões críticas, como infiltrações, operações em áreas de conflito e resgates. O uso de simuladores é particularmente valioso para forças militares modernas, pois permite ensaiar coordenação de equipes, estratégias de combate e respostas a eventos inesperados sem o perigo de campo. Sistemas como o *Virtual Battlespace*, um simulador militar de combate, são amplamente utilizados por forças armadas em todo o mundo para treinar soldados e comandantes em operações coordenadas [FREITAS; GOMES; WINKLER, 2022].

Na área da saúde, simuladores de treinamento são indispensáveis para capacitar profissionais em procedimentos de alta complexidade e precisão, especialmente em especialidades médicas como a cirurgia. Dispositivos de realidade aumentada e virtual, como o *SimMan 3G* e simuladores de laparoscopia, permitem que médicos e enfermeiros pratiquem procedimentos sem colocar pacientes reais em risco. Esses simuladores incorporam tecnologias que fornecem *feedback* tátil e visual detalhado, replicando a resistência e a textura dos tecidos humanos. De acordo com Makransky e Petersen [2021], o uso de simuladores na educação médica aumenta a proficiência e a confiança dos profissionais, reduzindo a ocorrência de erros e melhorando a segurança do paciente. Hospitais e centros de treinamento de saúde em todo o mundo investem em simuladores como o *da Vinci Surgical System*, um robô cirúrgico que permite praticar procedimentos minimamente invasivos em um ambiente seguro e controlado, antes de realizar as operações em

pacientes.

Além disso, um estudo de Abbate et al. [2023], que propõe um sistema de terapia com espelho, ou MT (Mirror Therapy), combina RV com um exoesqueleto para tratar deficiências motoras nas mãos, pode ser visto na Figura 5. A terapia tradicional com espelho usa o reflexo da mão saudável para enganar o cérebro, estimulando a recuperação motora. No sistema proposto, o movimento da mão saudável é espelhado na mão afetada em um ambiente virtual, enquanto um exoesqueleto físico reproduz esse movimento na mão real, promovendo neuroplasticidade e senso de incorporação, ou *embodiment*.

Figura 5 – Representação do sistema integrador de realidade virtual com exoesqueleto



Fonte: [ABBATE et al., 2023]

A tradução livre da Figura 5 é vista na seguinte legenda:

- **VR scene:** Cena de RV.
- **VR headset:** Headset de RV.
- **Healthy hand:** Mão saudável.
- **Impaired hand:** Mão comprometida.
- **Exoskeleton:** Exoesqueleto.
- **Exoskeleton controller:** Controlador do exoesqueleto.

Os resultados mostraram que o sistema é eficaz e bem aceito pelos usuários, oferecendo potencial para melhorar a terapia de pacientes com limitações motoras após um acidente vascular cerebral (AVC). O estudo destaca que o uso combinado de RV e dispositivos robóticos pode reduzir o custo e aumentar a acessibilidade da reabilitação, com possibilidade de aplicação em telemedicina.

Esse avanço reforça o potencial da RV para criar terapias personalizadas e imersivas, melhorando a recuperação funcional de pacientes com deficiências motoras. O sistema desenvolvido mostra como a integração entre RV e exoesqueleto oferece uma abordagem mais eficaz para reabilitação neuro-motora.

Além do setor de saúde, a engenharia civil e industrial também se beneficia dos simuladores de treinamento. Em particular, engenheiros civis utilizam simuladores para planejar e visualizar projetos complexos, como construção de pontes, edifícios e estradas. Simuladores permitem que engenheiros pratiquem o uso de equipamentos pesados e realizem operações de manutenção sem os riscos e custos associados ao uso em campo. No setor de construção, modelos 3D e simuladores de realidade aumentada são aplicados para treinar trabalhadores em montagem de estruturas e manuseio de máquinas pesadas [SAMPAIO et al., 2010]. Isso permite que os *trainees* desenvolvam habilidades essenciais e aprendam a operar de maneira segura antes de entrar no ambiente real de trabalho, onde erros podem ter consequências graves.

A indústria automotiva também adota simuladores, especialmente para a capacitação em segurança e operações de manutenção. Simuladores de condução, como o *SimuRide*, são utilizados para o treinamento de motoristas de veículos de carga pesada, ônibus e veículos de emergência, permitindo que esses profissionais pratiquem condução em condições adversas, como chuva, neve e neblina. Esses simuladores replicam situações de risco, como frenagens bruscas e manobras de desvio, capacitando motoristas para reagir de maneira eficaz e segura em situações críticas. Além disso, fabricantes de veículos, como Toyota e Ford, utilizam simulações para treinar técnicos em diagnósticos e reparos, proporcionando uma experiência prática e eficiente antes mesmo de lidar com os veículos reais.

A área de energia, especialmente em operações de petróleo e gás e em usinas nucleares, emprega simuladores para treinar técnicos e engenheiros em procedimentos de segurança e manutenções complexas. Em plataformas de petróleo *offshore*, os profissionais são expostos a riscos elevados, e os simuladores permitem que técnicos pratiquem respostas a vazamentos, incêndios e explosões sem se exporem a perigos reais. Freitas, Gomes e Winkler [2022] destaca que esse tipo de treinamento reduz custos e riscos associados ao treinamento em campo, oferecendo aos trabalhadores um ambiente controlado onde podem desenvolver suas habilidades sem as pressões e os perigos das operações reais. A indústria nuclear também utiliza simuladores para treinar operadores de reatores,

replicando situações de emergência e procedimentos de contenção de incidentes nucleares.

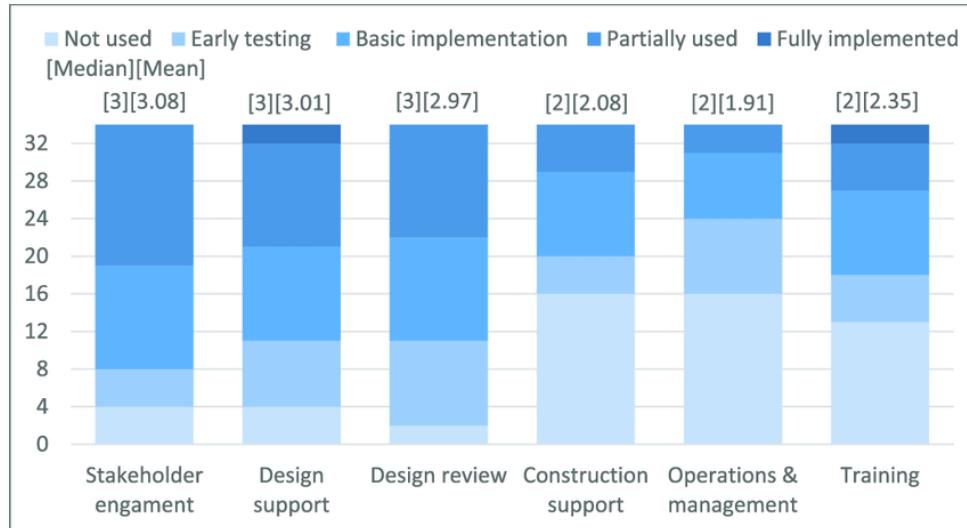
No setor educacional, especialmente no ensino técnico e profissionalizante, simuladores estão sendo cada vez mais utilizados para ensinar habilidades práticas de forma interativa. Simuladores de soldagem, por exemplo, permitem que estudantes de cursos técnicos pratiquem operações de soldagem em um ambiente virtual antes de utilizar equipamentos reais, ajudando-os a adquirir familiaridade com as ferramentas e técnicas em um ambiente seguro. Esse tipo de tecnologia tem sido implementado em escolas técnicas e centros de formação profissional para facilitar o aprendizado e melhorar a eficiência dos cursos práticos.

Simuladores também desempenham um papel essencial no setor de transportes ferroviários, onde são utilizados para treinar operadores de trens em situações de alta pressão e resposta rápida. Simuladores de condução ferroviária replicam as condições de operação dos trens, permitindo que os operadores aprendam a lidar com falhas de sinal, frenagem de emergência e manobras em condições de baixa visibilidade. Esse treinamento é crucial para garantir a segurança dos passageiros e a eficiência do sistema de transporte.

Essas áreas e cases exemplificam o amplo potencial dos simuladores de treinamento, que continuam a evoluir para atender às necessidades de setores específicos e aumentar a segurança, eficiência e eficácia dos processos de treinamento. Com a evolução contínua da tecnologia, o uso de simuladores se torna cada vez mais crucial para capacitar profissionais em habilidades complexas, promovendo uma aprendizagem prática e segura que prepara os usuários para enfrentar os desafios do mundo real.

A Figura 6 ilustra os níveis de adoção da realidade virtual em diferentes setores dentro do AEC (Arquitetura, Engenharia e Construção), destacando como a tecnologia tem sido progressivamente utilizada para melhorar a qualidade e a eficiência dos treinamentos.

Figura 6 – Níveis de adoção da realidade virtual para os seis casos de uso definidos no setor AEC (Arquitetura, Engenharia e Construção)



Fonte: [DELGADO et al., 2020]

Os termos do gráfico visto na Figura 6 podem ser traduzidos livremente como:

- **Stakeholder engagement:** Engajamento das partes interessadas.
- **Design support:** Suporte ao design.
- **Design review:** Revisão do design.
- **Construction support:** Suporte à construção.
- **Operations & management:** Operações e gestão.
- **Training:** Treinamento.
- **Not used:** Não utilizado.
- **Early testing:** Teste inicial.
- **Basic implementation:** Implementação básica.
- **Partially used:** Parcialmente utilizado.
- **Fully implemented:** Totalmente implementado.
- **Median:** Mediana.
- **Mean:** Média.

2.2 Gamificação em Simuladores de Treinamento

A gamificação refere-se ao uso de elementos de design de jogos em contextos não lúdicos, com o intuito de aumentar o engajamento, a motivação e a eficácia dos participantes em atividades que tradicionalmente não são vistas como jogos. Segundo Deterding et al. [2011], a gamificação não envolve a criação de jogos completos, mas sim a aplicação de componentes e mecânicas de jogos, como pontuações, níveis e recompensas, em tarefas e sistemas do mundo real. Esses elementos têm como objetivo promover um comportamento específico dos usuários, incentivando-os a participar de maneira mais envolvente e comprometida.

No contexto dos simuladores de treinamento, a gamificação começou a ser integrada durante a década de 1980, influenciada pelo sucesso dos jogos eletrônicos e pelo seu potencial para ambientes de aprendizado prático. Deterding et al. [2011] organizam o campo da gamificação e do design lúdico em um diagrama bidimensional, onde a gamificação, ou *gameful design*, é vista como o uso de partes dos jogos, ou elementos, aplicados a contextos distintos, enquanto os *serious games* englobam o uso de jogos inteiros para finalidades não lúdicas, como educação e treinamento. Conforme ilustrado na Figura 7, a gamificação situa-se no quadrante superior direito, onde as mecânicas de jogos são utilizadas em fragmentos, em comparação ao *design lúdico* e aos brinquedos, que oferecem experiências completas sem o foco de “jogo” tradicional.

Figura 7 – Modelo de *Gameful Design* e Gamificação



Fonte: [DETERDING et al., 2011]

2.2.1 Origem e Primeiros Aparições de Gamificação em Simuladores

Os primeiros conceitos de gamificação em ambientes de simuladores surgiram com a popularização dos jogos eletrônicos, que demonstraram ser uma ferramenta eficaz para

o desenvolvimento de habilidades em um contexto divertido e interativo. Na década de 1980, o advento dos PCs e consoles de *videogames* trouxe uma nova abordagem ao treinamento, possibilitando que elementos de jogos, como pontuações, níveis e recompensas, fossem aplicados em contextos de aprendizado [DETERDING et al., 2011]. Esse movimento começou a ser explorado em treinamentos militares e de aviação, onde simuladores começaram a incorporar elementos de competição e progressão, incentivando os *trainees* a alcançar melhores desempenhos e aumentar a retenção de conhecimento.

Um dos primeiros casos documentados de gamificação em simuladores foi no treinamento de combate das forças armadas dos Estados Unidos, que integraram elementos de jogos para treinar soldados em ambientes simulados de maneira mais engajante. Essa abordagem foi eficaz em promover a cooperação e o espírito de equipe, e logo foi adotada por outras áreas, como a aviação civil e a medicina [JARAMILLO-MEDIAVILLA et al., 2024]. A incorporação desses conceitos foi impulsionada pelo sucesso dos jogos eletrônicos no entretenimento, que demonstraram o potencial da gamificação para atrair a atenção dos participantes e mantê-los engajados em tarefas desafiadoras.

2.2.2 Aplicações em Diversas Áreas

Com o passar do tempo, a gamificação expandiu-se para diversas áreas além do setor militar, incluindo a medicina, a educação, a engenharia e o setor corporativo. Na medicina, simuladores gamificados passaram a ser utilizados para treinar profissionais de saúde em procedimentos complexos e de alto risco. Esses simuladores combinam elementos de pontuação, níveis de dificuldade e *feedback* em tempo real para proporcionar uma experiência de aprendizado mais interativa e realista [MAKRANSKY; PETERSEN, 2021]. Em centros de formação médica, os *trainees* utilizam esses simuladores para praticar procedimentos, como cirurgias laparoscópicas e intervenções cardíacas, em ambientes virtuais onde erros não acarretam riscos à vida dos pacientes.

Na educação, a gamificação encontrou terreno fértil, especialmente em disciplinas que exigem prática intensiva, como ciências exatas e cursos técnicos. Estudantes de engenharia, por exemplo, utilizam simuladores gamificados para aprender sobre princípios de física e mecânica, com desafios progressivos que os ajudam a dominar o conteúdo. Projetos de simulação em engenharia civil, descritos por Sampaio et al. [2010], incluem modelos tridimensionais e ambientes de construção gamificados, onde os alunos podem praticar tarefas de construção e manutenção com *feedback* imediato sobre seus desempenhos.

O setor corporativo também adota a gamificação em programas de treinamento, utilizando simuladores para ensinar práticas de segurança, liderança e habilidades de comunicação. Empresas implementam simuladores gamificados para treinar funcionários em cenários de atendimento ao cliente, onde eles podem responder a reclamações e resolver

problemas de maneira prática e lúdica. Essas simulações incentivam a melhoria contínua e a competição saudável entre os *trainees*, promovendo um aprendizado eficaz e duradouro.

2.2.3 Benefícios da Gamificação em Simuladores de Treinamento

A gamificação em simuladores de treinamento oferece uma série de benefícios, sendo um dos principais a motivação aumentada. A introdução de elementos de jogos, como recompensas e conquistas, estimula os *trainees* a se dedicarem ao aprendizado, promovendo o engajamento e reduzindo a monotonia. Estudos indicam que ambientes gamificados aumentam a retenção de conhecimento e a participação ativa, o que é essencial para a internalização de habilidades práticas [DETERDING et al., 2011].

Outro benefício é a possibilidade de personalização do aprendizado, onde os *trainees* podem progredir em níveis de dificuldade e focar nas áreas onde precisam de mais prática. Em simuladores de treinamento médico, por exemplo, a gamificação permite que os *trainees* revisitem procedimentos específicos e enfrentem novos desafios a cada tentativa, ajustando a simulação de acordo com o nível de habilidade do usuário. A possibilidade de ajustar o ambiente de aprendizado a partir de *feedbacks* obtidos durante as atividades gamificadas também contribui para uma curva de aprendizado mais eficiente.

Além disso, a gamificação proporciona um ambiente seguro para o desenvolvimento de habilidades em situações de alto risco. Profissionais de segurança, por exemplo, podem praticar procedimentos de emergência em simuladores que recompensam a tomada de decisões rápidas e corretas, ajudando a preparar o *trainee* para eventos reais. Esse aspecto é particularmente valioso em áreas como combate a incêndios e operações de resgate, onde a experiência prática em situações reais é limitada e arriscada.

2.2.4 Consolidação dos *Serious Games*

A evolução da gamificação levou à consolidação dos chamados *serious games*, jogos que, além do entretenimento, têm como objetivo principal o aprendizado e o desenvolvimento de habilidades específicas. Esses jogos são amplamente utilizados em treinamentos corporativos, educação e desenvolvimento pessoal, sendo projetados para criar um ambiente envolvente e imersivo onde o usuário pode aprender enquanto se diverte. Jaramillo-Mediavilla et al. [2024] identificam que *serious games* aplicados à educação têm um impacto positivo no engajamento e na motivação dos estudantes, e são uma ferramenta eficaz para o aprendizado autodirigido e interativo.

Os *serious games* são aplicados a uma ampla gama de áreas, incluindo simulações de combate para treinamento militar, simulações médicas para estudantes de medicina e programas de liderança e desenvolvimento de habilidades interpessoais para o setor corporativo. Exemplos incluem jogos de simulação de gestão de recursos para formar

líderes e gerentes em organizações e jogos educacionais para o ensino de matemática e ciências. Esses jogos vão além da simulação, incentivando o usuário a explorar, resolver problemas e tomar decisões em um ambiente interativo e seguro.

Um exemplo de *serious game* amplamente utilizado é o jogo de simulação médica *Pulse!!*, desenvolvido para ensinar profissionais de saúde sobre procedimentos de emergência em ambientes de alto estresse. Em *Pulse!!*, os jogadores assumem o papel de médicos e enfermeiros em um pronto-socorro, onde enfrentam situações de vida ou morte que requerem diagnósticos rápidos e precisos. Outro exemplo é o *Virtual Battlespace*, um *serious game* usado por forças militares para simular cenários de combate e treinar soldados em táticas de combate e tomada de decisão em campo.

A consolidação dos *serious games* como uma ferramenta de treinamento se deve ao seu potencial de criar ambientes onde o usuário pode experimentar, aprender e desenvolver habilidades em um contexto que simula a realidade, mas sem os riscos associados a ela. Essa abordagem tem se mostrado eficaz na formação de profissionais em áreas diversas, garantindo que o conhecimento adquirido seja não apenas teórico, mas também prático e aplicável em situações reais.

A gamificação e os *serious games* têm transformado a maneira como os simuladores de treinamento são aplicados e percebidos em diferentes setores. Ao integrar elementos de jogos e desafios, esses simuladores aumentam o engajamento e a retenção de conhecimento, oferecendo uma experiência prática valiosa para áreas que exigem alta precisão e habilidades específicas. A implementação contínua de elementos de gamificação e a adoção de *serious games* sugerem que essas ferramentas continuarão a evoluir e a se consolidar como métodos fundamentais para o desenvolvimento de competências, adaptando-se às necessidades de um mercado em constante transformação.

2.3 Realidade Virtual e Realidade Aumentada

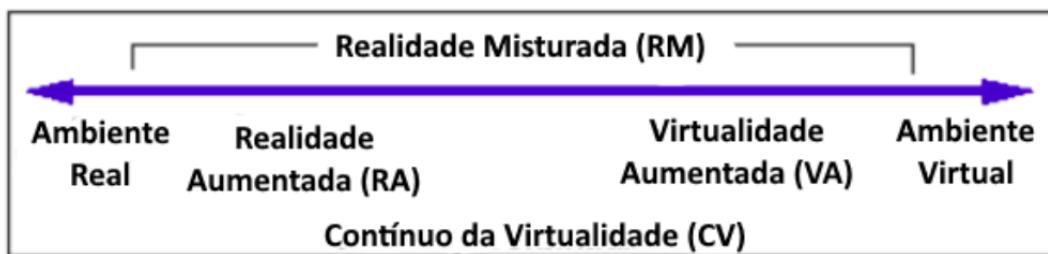
2.3.1 Definições e Histórico

A RV e a RA são tecnologias imersivas que transformam a interação dos usuários com o ambiente, permitindo vivências além das limitações físicas. A RV proporciona uma experiência completamente digital, na qual o usuário é imerso em um ambiente tridimensional gerado por computador, sendo um conceito introduzido nos anos 1960 por Ivan Sutherland, com o famoso dispositivo *Sword of Damocles*, um dos primeiros HMDs [SUTHERLAND, 1968]. Esse dispositivo pioneiro, mostrado na Figura 3, representou uma inovação significativa ao proporcionar uma interface visual controlada digitalmente, permitindo a percepção de objetos tridimensionais gerados por computador sobrepostos à visão do usuário. Desde então, a RV evoluiu para incluir tecnologias que proporcionam

experiências sensoriais mais amplas e realistas. Em contrapartida, a RA projeta elementos digitais sobre o mundo físico, oferecendo uma interface que complementa o ambiente real com informações adicionais, sem ocultá-lo totalmente, conforme conceituado por Milgram e Kishino [1994] no *continuum* de realidade-virtualidade.

O conceito de *continuum* de realidade-virtualidade, desenvolvido por Milgram e Kishino [1994], descreve uma escala que vai desde o mundo real, passando por ambientes com sobreposição de elementos digitais (realidade aumentada), até a imersão total em ambientes virtuais (realidade virtual). Neste *continuum*, a realidade aumentada ocupa uma posição intermediária, onde elementos digitais são sobrepostos ao ambiente físico para enriquecer a percepção do usuário, mantendo-se uma conexão direta com o mundo real. Já a realidade virtual situa-se no extremo oposto, oferecendo um ambiente completamente gerado por computador, onde o usuário é imerso em um espaço digital que substitui totalmente a realidade física. Esse *continuum* é fundamental para entender as diversas aplicações e interações possíveis em cada tecnologia, pois permite uma combinação flexível e adaptativa de elementos reais e virtuais, atendendo às necessidades específicas de áreas como treinamento, educação e entretenimento. A Figura 8 ilustra essa escala entre o ambiente real e o ambiente virtual, destacando as posições intermediárias da realidade aumentada e da virtualidade aumentada dentro do *continuum*.

Figura 8 – Representação do *continuum* de realidade-virtualidade

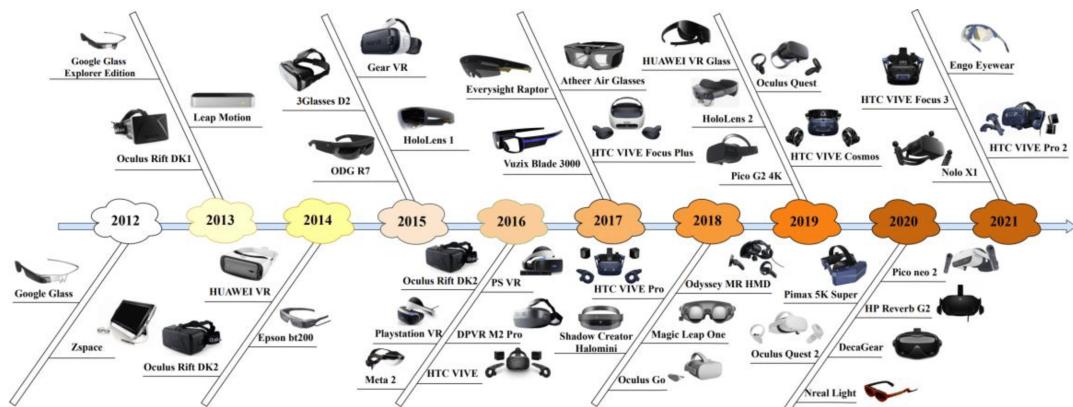


Fonte: [MILGRAM; KISHINO, 1994]

2.3.2 Evolução dos Dispositivos de Realidade Aumentada e Realidade Virtual

A evolução dos dispositivos de RA e RV tem sido marcada por inovações tecnológicas que ampliaram as possibilidades de imersão e interação dos usuários com ambientes virtuais. Conforme ilustrado na Figura 9, dispositivos ao longo dos anos evoluíram significativamente, desde os primeiros protótipos até sistemas avançados com alta resolução e recursos de rastreamento precisos. A seguir, detalhamos essa linha do tempo, destacando os principais dispositivos e suas contribuições ao campo da RA e RV [LI; CHEN, 2022].

Figura 9 – Linha do tempo da evolução dos dispositivos de RA e RV



Fonte: [LI; CHEN, 2022]

Em 2012, o lançamento do *Google Glass Explorer Edition* marcou uma das primeiras tentativas de um dispositivo de RA para o mercado consumidor. O *Google Glass* oferecia uma tela montada na cabeça, capaz de sobrepor informações visuais ao campo de visão do usuário, proporcionando um primeiro contato com a integração entre o digital e o real. Embora tivesse funcionalidades limitadas, abriu caminho para o desenvolvimento de dispositivos de RA focados em usabilidade diária.

Com o surgimento do *Oculus Rift DK1* em 2013, o mercado de RV começou a ganhar força. O *Oculus Rift* foi um dos primeiros dispositivos de RV acessíveis ao consumidor, oferecendo uma experiência de imersão em ambientes virtuais com visualização estereoscópica e rastreamento de movimento. Em 2014, dispositivos como o *Samsung Gear VR* e o *3Glasses D1* começaram a se popularizar, com diversas empresas explorando o potencial da RV para entretenimento.

Em 2015, a segunda geração do *Oculus Rift DK2* trouxe melhorias em termos de resolução e taxa de atualização, oferecendo uma experiência de RV mais fluida. O ano também foi marcado pelo lançamento do *HTC Vive*, que introduziu o rastreamento espacial, permitindo aos usuários se movimentarem em um espaço físico enquanto exploravam o ambiente virtual. Essa funcionalidade foi fundamental para consolidar a RV como uma tecnologia de imersão avançada.

O ano de 2016 foi um marco com o lançamento de dispositivos como o *PlayStation VR*, que trouxe a experiência de RV para o mercado de consoles de jogos. Dispositivos como o *Microsoft HoloLens* também foram introduzidos, oferecendo uma experiência de RA sofisticada com mapeamento espacial e interação por gestos. Essa expansão demonstrou a viabilidade da RA e RV para uma gama maior de aplicações, incluindo entretenimento, educação e treinamento industrial.

Com o lançamento do *HTC Vive Pro* e o *Oculus Go* em 2017, o mercado de

dispositivos de RV continuou a evoluir, aprimorando a qualidade visual e a ergonomia. O *Oculus Go* foi um dos primeiros dispositivos autônomos, eliminando a necessidade de conexão a um computador, o que ampliou o alcance da tecnologia para um público mais amplo. A inclusão de controles de movimento mais precisos também contribuiu para experiências mais interativas.

Em 2018, dispositivos como o *Magic Leap One* e o *Oculus Quest* trouxeram inovações em portabilidade e qualidade de imagem. O *Oculus Quest* foi o primeiro dispositivo de RV totalmente autônomo com rastreamento de seis graus de liberdade (6DoF), permitindo uma experiência de RV completa sem a necessidade de sensores externos. Essas melhorias aumentaram a acessibilidade dos dispositivos de RV e RA para consumidores e empresas.

Nos anos de 2019 e 2020, o mercado de RA e RV começou a se consolidar com dispositivos como o *HTC Vive Cosmos* e o *Oculus Quest 2*. O *Quest 2* destacou-se pela sua acessibilidade e desempenho aprimorado, tornando-se um dos dispositivos de RV mais populares. Nesse período, a RA também avançou, com aplicações focadas em áreas profissionais, incluindo treinamento e visualização industrial.

Em 2021, dispositivos como o *Vuzix Shield* e o *Microsoft HoloLens 2* consolidaram a RA no mercado corporativo, oferecendo recursos avançados de sobreposição de dados e colaboração remota. O *HoloLens 2* destacou-se por sua capacidade de rastreamento ocular e interação natural com o ambiente, permitindo novas aplicações em manutenção industrial, saúde e ensino.

Essa linha do tempo evidencia como os dispositivos de RA e RV evoluíram em resposta às demandas de diferentes setores, resultando em tecnologias mais acessíveis, sofisticadas e funcionais, capazes de transformar a forma como interagimos com o ambiente digital e o mundo físico [LI; CHEN, 2022].

2.3.3 Realidade Virtual

A RV se caracteriza por gerar um ambiente simulado onde o usuário pode explorar e interagir. O desenvolvimento dessa tecnologia foi fortemente impulsionado pela indústria de entretenimento, mas rapidamente expandiu para outras áreas, como educação, treinamento e saúde. Entre os primeiros dispositivos de RV destaca-se o *Sensorama* de Morton Heilig, um simulador que oferecia estímulos visuais, auditivos e táteis, marcando um avanço importante no desenvolvimento de dispositivos imersivos [HEILIG, 1962].

Dispositivos como os modernos HMDs *Oculus Rift*, *HTC Vive* e *PlayStation VR* aprimoraram a qualidade gráfica e a capacidade de rastreamento, proporcionando uma experiência de alta imersão. Esses dispositivos utilizam sistemas de rastreamento de posição, incluindo sensores iniciais, ópticos e de câmeras externas, que permitem captar

o movimento e a orientação da cabeça, criando uma sensação de presença [HEILIG, 1962; SUTHERLAND, 1968]. A RV tem sido amplamente utilizada em aplicações de treinamento industrial, permitindo simulações realistas de ambientes de trabalho complexos, como subestações elétricas e ambientes de operação de risco, conforme demonstrado em estudos de caso no setor energético [FREITAS; GOMES; WINKLER, 2022].

2.3.3.1 Aplicações e Benefícios

O uso da RV em treinamento é consolidado pela capacidade de criar ambientes seguros e controlados, onde usuários podem praticar habilidades em situações simuladas. Essa tecnologia tem sido essencial em áreas de alto risco, como aviação, onde os simuladores de voo permitem que pilotos desenvolvam habilidades sem os perigos e custos associados aos voos reais. Outras áreas, como saúde, também utilizam RV para treinamento de cirurgias complexas, oferecendo uma alternativa para a prática de procedimentos sem riscos ao paciente.

Na engenharia e na arquitetura, a RV é amplamente utilizada para visualização de projetos e para realizar simulações de operações de construção. Os engenheiros podem explorar modelos tridimensionais de projetos complexos, o que facilita a detecção de problemas antes mesmo da execução. Além disso, as equipes de segurança podem ensaiar protocolos em áreas de risco, treinando as melhores práticas sem comprometer a segurança dos trabalhadores [FREITAS; GOMES; WINKLER, 2022]. Na educação, essa tecnologia permite que estudantes explorem ambientes históricos ou simulem experimentos científicos complexos, proporcionando uma aprendizagem prática e interativa.

2.3.4 Realidade Aumentada

A RA, em contrapartida, busca complementar o ambiente físico com informações digitais em tempo real, sendo aplicada com frequência em dispositivos móveis e óculos de RA. Ela se popularizou com o avanço dos *smartphones* e dispositivos como o *Microsoft HoloLens*, que sobrepõem informações digitais diretamente sobre o campo de visão do usuário. A RA tem um impacto significativo em áreas como manutenção industrial e medicina, onde informações visuais sobrepostas auxiliam na execução de tarefas complexas, orientando o usuário em tempo real [AZUMA, 1997].

2.3.4.1 Aplicações e Benefícios

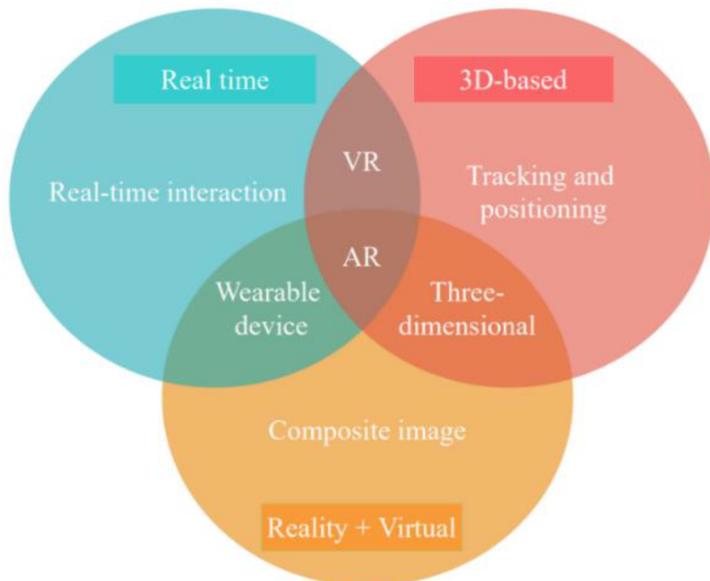
Aplicações de RA são vastas e incluem áreas como a educação, onde a tecnologia é utilizada para enriquecer materiais de estudo com elementos interativos e animados, aumentando o engajamento e a compreensão dos alunos. Além disso, na medicina, a RA tem sido utilizada em cirurgias, sobrepondo imagens de exames diretamente sobre o

paciente, permitindo maior precisão [AZUMA, 1997]. A RA também tem sido aplicada em setores como varejo e indústria automotiva, permitindo que os trabalhadores visualizem componentes internos de máquinas ou recebam orientações de montagem em tempo real, minimizando erros e aumentando a eficiência.

2.3.5 Comparação e Oportunidades

A principal diferença entre RV e RA reside no grau de imersão e no modo como cada tecnologia modifica a percepção do usuário. A RV proporciona uma imersão completa em um ambiente digital, desconectando o usuário do mundo físico, enquanto a RA mantém o usuário conectado ao ambiente real, adicionando camadas de informações úteis sobre ele. Como ilustrado na Figura 10, a RA e a RV compartilham algumas características, como a interação em tempo real e o uso de dispositivos vestíveis, mas diferem em sua abordagem de integração com o mundo físico e virtual. A Figura destaca como a RA combina imagens compostas do mundo real com elementos virtuais, enquanto a RV é predominantemente baseada em ambientes tridimensionais gerados por computador [LI; CHEN, 2022]. O desenvolvimento de ambas as tecnologias oferece novas oportunidades para criar experiências de aprendizado, treinamento e entretenimento mais interativas e eficazes [MILGRAM; KISHINO, 1994; AZUMA, 1997].

Figura 10 – Comparação entre características da RV e RA, similaridades e diferenças



Fonte: [LI; CHEN, 2022]

A lista abaixo representa a tradução livre dos termos vistos na Figura 10:

- **Real time:** Tempo real.
- **Real-time interaction:** Interação em tempo real.

- **VR:** RV.
 - **AR:** RA.
 - **Wearable device:** Dispositivo vestível.
 - **Tracking and positioning:** Rastreamento e posicionamento.
 - **Three-dimensional:** Tridimensional.
 - **Composite image:** Imagem composta.
 - **3D-based:** Baseado em 3D.
 - **Reality + Virtual:** Realidade + Virtual.

2.4 Metaverso

O conceito de metaverso refere-se a um espaço virtual compartilhado e persistente, criado pela convergência de ambientes de realidade virtual e realidade aumentada, onde os usuários podem interagir entre si e com elementos digitais de maneira imersiva e em tempo real. O termo foi popularizado pela obra *Snow Crash* de Neal Stephenson, em 1992, e, desde então, evoluiu de uma visão futurista para um campo ativo de pesquisa e desenvolvimento com diversas aplicações práticas [WEINBERGER, 2022; MYSTAKIDIS, 2022]. Mystakidis [2022] define o metaverso como um ambiente multiusuário e persistente que integra o mundo físico e o digital, utilizando tecnologias de interação multissensorial para permitir experiências digitais dinâmicas e imersivas.

2.4.1 Definição e Estrutura do Metaverso

O metaverso pode ser compreendido como um ecossistema digital tridimensional que conecta múltiplos mundos virtuais, proporcionando uma experiência contínua e interativa. Essa experiência é facilitada pela tecnologia de RV, que imerge o usuário em ambientes digitais, e pela RA, que adiciona camadas informativas ao mundo físico. Em um cenário de metaverso, esses elementos combinam-se para permitir que os usuários experimentem um “mundo” persistente, onde as interações e os dados mantêm-se além de uma sessão isolada, integrando de maneira coesa aspectos do mundo digital e físico [WEINBERGER, 2022; MYSTAKIDIS, 2022]. De acordo com Mystakidis [2022], a estrutura do metaverso apoia-se em interações dinâmicas e comunicação em tempo real, promovendo uma imersão completa através de tecnologias como RV e RA.

2.4.2 Histórico do Metaverso

Embora o termo metaverso tenha sido popularizado por Stephenson na década de 1990, as ideias centrais do metaverso, como a imersão e a interatividade, podem ser rastreadas até o desenvolvimento inicial da RV e da RA, tecnologias que começaram a evoluir nos anos 1960 e 1970. Em 2003, a criação do *Second Life* trouxe o conceito para o público em geral, apresentando um mundo virtual onde os usuários podiam criar avatares, interagir com outros usuários, construir ambientes e participar de uma economia digital baseada em ativos virtuais.

Com o avanço das tecnologias de RV e RA, além do desenvolvimento de redes de quinta geração e de tecnologias de computação em nuvem, tornou-se possível criar experiências mais imersivas e interativas. Esse progresso permitiu que o metaverso se expandisse para áreas além do entretenimento, como educação e trabalho colaborativo. Empresas como a Meta, anteriormente *Facebook*, têm investido no desenvolvimento do metaverso, promovendo o conceito e incentivando o desenvolvimento de tecnologias que suportam mundos virtuais persistentes e interativos [RADIANTI et al., 2020; MYSTAKIDIS, 2022].

2.4.3 Relação com Realidade Virtual e Realidade Aumentada

O metaverso é essencialmente uma extensão das potencialidades oferecidas pela RV e RA, ampliando a capacidade de interação e imersão dos usuários. Enquanto a RV oferece uma imersão completa em um ambiente virtual gerado por computador, a RA possibilita a inserção de elementos digitais sobre o mundo físico, criando um continuum de interações que permite a sobreposição e a combinação desses dois ambientes. Esse continuum, conforme descrito por [MILGRAM; KISHINO, 1994], posiciona o metaverso como uma fusão desses extremos, permitindo tanto experiências totalmente imersivas quanto híbridas, adaptadas às necessidades e preferências dos usuários.

2.4.4 Aplicações do Metaverso

As aplicações do metaverso são amplas e diversificadas, abrangendo desde o entretenimento até áreas técnicas e educacionais. Abaixo, destacamos algumas das principais áreas de aplicação:

2.4.4.1 Educação e Treinamento

O metaverso proporciona um ambiente seguro e imersivo para o treinamento em diversas áreas, permitindo simulações de cenários complexos e de alto risco. Em setores como medicina, engenharia e forças armadas, o uso do metaverso facilita o treinamento prático sem os riscos e custos associados ao treinamento no mundo físico. Estudos indicam

que a imersão oferecida pelo metaverso contribui para uma maior retenção de conhecimento e um engajamento mais ativo dos usuários [GURIAN; BOLISANI; KIRCHNER, 2023; FREITAS; GOMES; WINKLER, 2022; MYSTAKIDIS, 2022]. Mystakidis [2022] destaca ainda que, na educação, o metaverso transforma o ensino à distância, criando experiências de aprendizado interativas e práticas, fundamentais para o ensino de habilidades complexas.

2.4.4.2 Colaboração e Trabalho Remoto

Com o crescimento do trabalho remoto, o metaverso emerge como uma solução para conectar equipes em ambientes de colaboração virtual. Empresas estão explorando o metaverso para criar escritórios virtuais onde os funcionários podem se encontrar, colaborar em projetos e realizar reuniões em um ambiente mais interativo e pessoal do que as plataformas de vídeo convencionais. Essa aplicação tem sido especialmente útil em setores como desenvolvimento de software e design, onde o trabalho em equipe e a visualização de produtos em 3D são essenciais [RADIANTI et al., 2020; MYSTAKIDIS, 2022].

2.4.4.3 Entretenimento e Socialização

O entretenimento é uma das áreas onde o metaverso ganhou maior popularidade, com jogos como *Fortnite* e *Roblox* adotando elementos de metaverso que permitem que os jogadores participem de eventos sociais e econômicos dentro de seus mundos virtuais. Essas plataformas oferecem experiências de socialização, concertos virtuais e até transações econômicas, criando um ecossistema dinâmico e social que atrai milhões de usuários.

2.4.4.4 Economia Digital

O metaverso possibilita a criação de economias virtuais completas, onde os usuários podem comprar, vender e trocar ativos digitais. Isso inclui bens como imóveis virtuais, itens colecionáveis e arte digital. A tecnologia de *blockchain* tem sido fundamental para garantir a propriedade e a autenticidade desses ativos, por meio dos *tokens* não fungíveis (NFTs), que são utilizados para representar bens digitais únicos no metaverso. Esses ativos criam uma economia robusta, permitindo transações seguras e sustentáveis dentro do metaverso [WEINBERGER, 2022].

2.4.5 O Futuro do Metaverso

Com o contínuo desenvolvimento das tecnologias de RA e RV e o avanço da infraestrutura digital, o metaverso tem o potencial de transformar a maneira como interagimos, aprendemos e trabalhamos. À medida que novas aplicações e dispositivos imersivos surgem, espera-se que o metaverso se expanda ainda mais, proporcionando um ambiente

virtual interconectado e acessível a um número crescente de pessoas. A consolidação do metaverso como uma plataforma multifuncional e acessível exigirá um desenvolvimento constante de tecnologias de segurança, interoperabilidade e acessibilidade, permitindo que pessoas de todo o mundo possam interagir e criar nesse novo espaço virtual [WEINBERGER, 2022; MYSTAKIDIS, 2022].

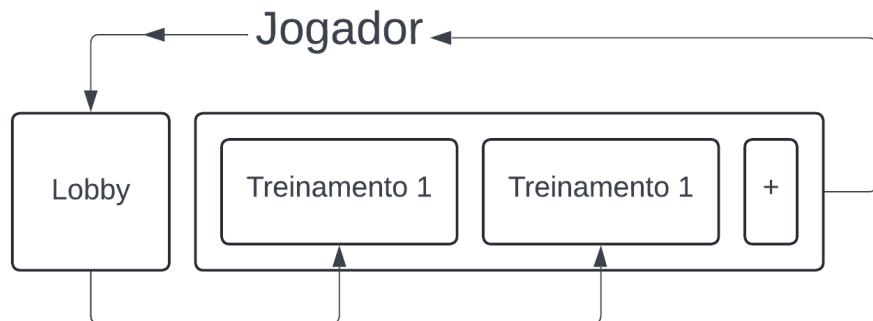
3 Proposta da Solução

O avanço das tecnologias imersivas tem possibilitado a criação de ambientes interativos para diferentes aplicações, desde o entretenimento até o treinamento profissional. A utilização de RV e metaverso no contexto de capacitação profissional tem se mostrado uma abordagem promissora, pois proporciona um ambiente de simulação realista e seguro para a realização de treinamentos, reduzindo custos e aumentando a eficácia do aprendizado [RADIANTI et al., 2020].

Dentre as principais ferramentas disponíveis para o desenvolvimento de ambientes imersivos, os motores de jogos desempenham um papel essencial, pois oferecem suporte para renderização gráfica, simulação física, interação com dispositivos de entrada e otimização da experiência do usuário. Essas tecnologias permitem que desenvolvedores criem plataformas altamente interativas, focando na experiência e na usabilidade dos participantes do treinamento [FREITAS; GOMES; WINKLER, 2022].

A solução proposta neste trabalho consiste na implementação de uma plataforma de metaverso para treinamento e capacitação, podendo ser vista na Figura 11, desenvolvida utilizando um motor gráfico moderno, permitindo que *trainees* realizem atividades práticas em um ambiente virtual. A plataforma conta com um sistema estruturado de treinamento, no qual os usuários podem interagir com objetos simulados, executar tarefas específicas com a presença de tutores em cena. Segundo Li e Chen [2022], a integração da realidade virtual e aumentada no treinamento ainda carece de metodologias bem estruturadas para garantir sua efetividade, sendo este um dos maiores desafios presentes nas soluções da área.

Figura 11 – Diagrama representativo solução proposta



Fonte: Autoria própria

Conforme visto na Figura 3, um dos principais diferenciais da solução proposta

é a possibilidade de expansão e agregação de novos treinamentos na mesma plataforma. A arquitetura desenvolvida foi projetada para permitir a inclusão de novos cenários e módulos de treinamento sem a necessidade de modificar a estrutura principal da aplicação. A implementação de novos treinamentos pode ser realizada por meio da criação de novas cenas e carregamento delas a partir do *Lobby*, um cenário inicial de espera.

Diferentemente também das abordagens tradicionais de ensino, que dependem de demonstrações presenciais e materiais didáticos estáticos, a solução proposta permite que os participantes vivenciem o treinamento de maneira ativa, interagindo com cenários dinâmicos. Isso possibilita um acompanhamento mais preciso do desempenho individual, permitindo análises personalizadas para cada *trainee*.

Para garantir uma experiência otimizada, a plataforma incorpora diferentes métodos de locomoção no ambiente virtual, incluindo locomoção física no espaço real, com *joystick* do controle e por teleporte [BALLESTRIN; CHESSA; SOLARI, 2021].

Outro diferencial da solução é a possibilidade de gestão dos treinamentos realizados, promovendo um sistema de gestão do conhecimento que permite um aprimoramento contínuo de soluções de ensino. Estudos indicam que sistemas imersivos bem projetados podem contribuir para a retenção do conhecimento e a efetividade do treinamento, tornando a experiência mais realista e engajante [WEINBERGER, 2022].

Portanto, a solução proposta visa não apenas fornecer um ambiente de simulação realista para treinamento profissional, mas também estabelecer um modelo estruturado para a implementação de sistemas de capacitação baseados em metaverso. Ao possibilitar uma experiência de aprendizado interativa, controlada e personalizável, a plataforma contribui para a evolução das metodologias de ensino imersivo, buscando promover engajamento nos processos de capacitação profissional [MAKRANSKY; PETERSEN, 2021].

4 Materiais e Métodos

A implementação e execução de uma plataforma de treinamento em ambiente de metaverso envolveu o uso de diversos softwares e dispositivos de hardware para garantir a criação de um ambiente virtual imersivo, interativo e acessível em tempo real. O sistema foi projetado para oferecer uma experiência de treinamento colaborativa, permitindo a interação de múltiplos usuários em um ambiente virtual compartilhado, com sincronização de estado e comportamento em tempo real.

Para garantir a implementação eficaz deste sistema, foram utilizadas ferramentas de *software* especializadas e dispositivos de *hardware* que permitiram a criação de um ambiente virtual robusto e responsivo. A combinação entre ferramentas de desenvolvimento, plataformas de modelagem 3D e motores de rede possibilitou a construção de um sistema dinâmico e funcional para execução das atividades de treinamento.

4.1 Software

O desenvolvimento do sistema foi baseado em uma combinação de ferramentas especializadas para modelagem 3D, interações em RV e integração de redes *multiplayer*. A escolha dos *softwares* utilizados foi baseada em sua robustez, confiabilidade, desempenho e suporte para ferramentas de realidade virtual.

4.1.1 Unity

A plataforma foi desenvolvida utilizando o motor gráfico Unity, amplamente reconhecido por sua capacidade de criar ambientes virtuais interativos e multiplataforma. A Unity oferece um conjunto robusto de ferramentas para desenvolvimento de jogos e simulações, incluindo sistemas de física, iluminação, renderização e *scripts*. A integração com dispositivos de realidade virtual via plugins proprietários foi um fator determinante para a escolha da Unity como plataforma de desenvolvimento [UNITY, 2023a].

No contexto deste projeto, a Unity foi utilizada para a criação dos cenários 3D, implementação das interações dos usuários com o ambiente e sincronização das ações dos usuários em tempo real. A Unity permitiu a configuração de elementos gráficos complexos e a execução de simulações realistas dos componentes com ferramentas manipuladas durante o treinamento.

4.1.2 C#

A linguagem de programação C# foi utilizada para a implementação da lógica do sistema, incluindo a configuração das interações e o controle de estados dos objetos em tempo real. A escolha de C# foi motivada por sua integração nativa com a Unity e por suas características de orientação a objetos, que facilitaram a implementação de comportamentos modulares e reutilizáveis [TROELSEN; JAPIKSE, 2021].

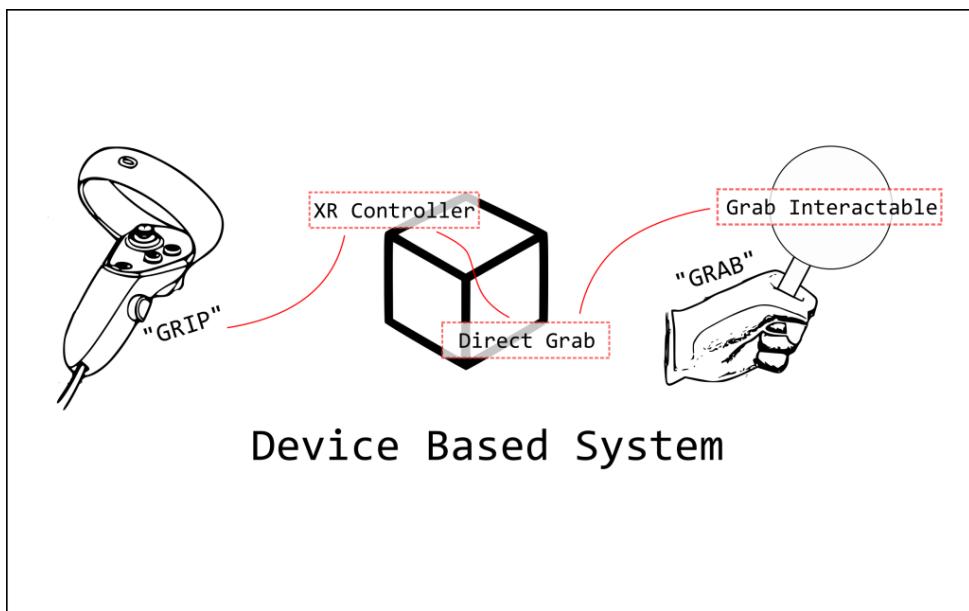
Os *scripts* em C# foram responsáveis pela lógica das interações dos usuários com os objetos virtuais, incluindo a manipulação de ferramentas, deslocamento dos objetos e execução das etapas de treinamento. A linguagem também foi utilizada para o controle de sincronização dos estados dos objetos entre os diferentes dispositivos e usuários conectados à mesma sala.

4.1.3 XR Interaction Toolkit

Além da implementação lógica com C#, o XR Interaction Toolkit, uma ferramenta de realidade extendida (XR), foi utilizado para permitir que os usuários manipulassem os objetos virtuais de forma intuitiva, facilitando o controle e a imersão. Esse pacote da Unity fornece componentes prontos para manipulação de objetos e controle de movimentação, permitindo que os usuários interajam com os elementos do cenário de forma natural e intuitiva [UNITY, 2023b].

O XR Interaction Toolkit possibilitou a configuração de diferentes métodos de locomoção, incluindo teletransporte, movimentação física e com o uso de *joystick*. A biblioteca também foi responsável pelo reconhecimento das interações com os objetos manipuláveis, demonstrado na Figura 12. Isso permite a execução de ações como agarrar, soltar, mover e rotacionar os componentes durante o treinamento.

Figura 12 – Funcionamento da ação de agarrar da biblioteca XR Interaction Toolkit



Fonte: [JUDD, 2020]

Os termos vistos na Figura 12 podem ser traduzidos livremente como:

- **Device Based System:** Sistema Baseado em Dispositivo.
- **GRIP:** Apertar.
- **XR Controller:** Controlador XR.
- **Direct Grab:** Pegar Direto.
- **Grab Interactable:** Pegar Interativo.
- **GRAB:** Pegar.

4.1.4 Photon PUN 2

A sincronização das ações dos usuários em tempo real foi implementada utilizando a biblioteca Photon PUN 2 (Photon Unity Networking). O Photon PUN 2 oferece suporte para criação de jogos *multiplayer*, com baixa latência e alta escalabilidade [PHOTON, 2023].

O Photon PUN 2 foi utilizado para criar as salas virtuais e gerenciar a sincronização entre objetos e jogadores conectados ao sistema. A biblioteca permitiu que múltiplos usuários compartilhassem o mesmo ambiente virtual, com atualização em tempo real das posições dos objetos e ações dos jogadores.

Até a data de hoje, a plataforma Photon Unity Networking (PUN) oferece diversas opções de planos para atender diferentes necessidades de desenvolvimento de jogos

multijogador. Dentre essas opções, o PUN Plus disponibiliza suporte a até 100 usuários conectados simultaneamente por 12 meses, ao custo de 95 dólares americanos. Para projetos que exigem maior capacidade de usuários simultâneos, a plataforma também oferece planos mensais, como o PUN 500 CCU, que permite a conexão de até 500 usuários simultâneos, com custo de 95 dólares americanos mensais. Essas alternativas possibilitam aos desenvolvedores escalabilidade e flexibilidade conforme as necessidades do jogo e o número de usuários a serem atendidos.

4.1.5 Blender

O software de modelagem 3D Blender foi utilizado para a criação de objetos para o ambiente de treinamento. O Blender é um software de código aberto amplamente utilizado para modelagem, animação e renderização de objetos tridimensionais [BLENDER, 2022].

Os modelos dos cenários e dos componentes utilizados no treinamento foram criados no Blender, antes da execução deste trabalho, pela equipe do CEASB (Centro Avançado em Soluções para Barragens), com base em dados técnicos. Apenas uma pequena parte dos objetos que não foram disponibilizados pelo CEASB precisou ser modelada do zero, incluindo as peças que compõem o avatar do jogador.

Estes modelos foram essenciais para criar uma experiência de treinamento realista, permitindo que os usuários manipulassem objetos complexos e interagissem com componentes detalhados em um ambiente virtual.

4.1.6 GitHub

O controle de versão e a colaboração entre os desenvolvedores foram gerenciados utilizando a plataforma GitHub. A plataforma permitiu o gerenciamento eficiente do código-fonte e dos arquivos do projeto, garantindo a integridade e consistência do sistema [GITHUB, 2021].

O GitHub foi utilizado para armazenar o código-fonte, gerenciar alterações e permitir a implementação em diferentes dispositivos. A plataforma também foi essencial para o rastreamento de erros e o controle de versões durante as fases de desenvolvimento e testes.

4.1.7 Parallel Sync

Após garantir a consistência do código-fonte e o controle de versões com o GitHub, o Parallel Sync foi utilizado para validar o comportamento do sistema em cenários multiusuário, assegurando a sincronização em tempo real das interações.

A biblioteca permitiu a execução simultânea de múltiplas instâncias em um único dispositivo. Isso possibilitou a simulação de cenários reais de treinamento, com múltiplos usuários interagindo em um ambiente compartilhado.

O uso do Parallel Sync foi essencial para verificar a estabilidade em rede e a responsividade do sistema em condições de uso real. A ferramenta permitiu testar a sincronização dos movimentos, interações e atualizações de estado entre as instâncias conectadas na mesma sala.

4.2 *Hardware*

Para garantir a execução eficiente da plataforma em ambiente de realidade virtual, foram utilizados dispositivos específicos de hardware para renderização gráfica, controle de interação e execução dos modelos de treinamento.

4.2.1 Meta Quest 2 e 3

O uso de dispositivos Meta Quest 2 e 3 foi fundamental para garantir que o sistema proporcionasse uma experiência imersiva e responsiva, facilitando a execução das tarefas de treinamento com precisão e sem latência significativa.

O Meta Quest 2 possui uma resolução de 1832 x 1920 pixels por olho e uma taxa de atualização de até 90 Hz, enquanto o Meta Quest 3 oferece resolução superior e uma taxa de atualização de até 120 Hz. Ambos os dispositivos possuem sensores para rastreamento de movimento e controles manuais para interação com o ambiente virtual.

O Meta Quest foi configurado para suportar diferentes métodos de locomoção, incluindo movimentação física, uso de *joystick* e teletransporte. A capacidade de rastreamento de posição e orientação dos dispositivos foi considerada para garantir uma interação precisa e sem atrasos com os componentes do treinamento.

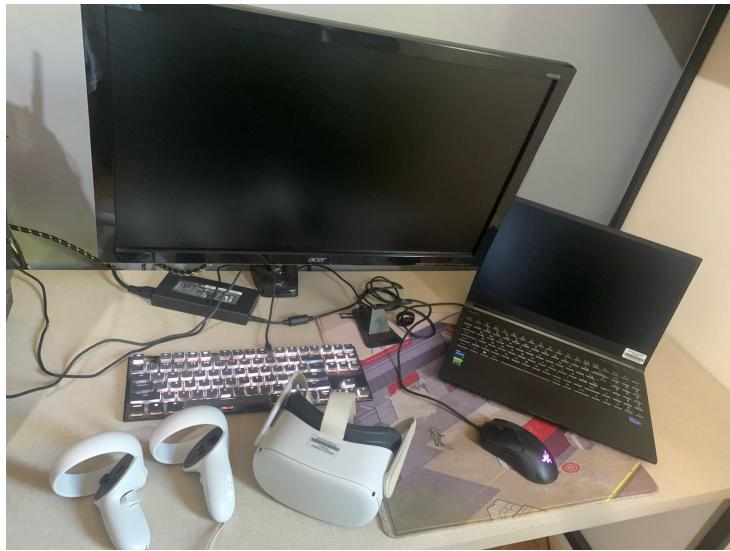
4.2.2 PCs e Notebooks

O desenvolvimento inicial foi realizado em PCs e notebooks com capacidade gráfica avançada para garantir que o sistema fosse capaz de renderizar cenários complexos e processar a física dos objetos em tempo real.

Foram utilizados PCs equipados com processador Intel Xeon E-2286G, 64 GB de memória RAM e placa gráfica NVIDIA Quadro RTX 4000 para garantir o desempenho necessário para renderização em tempo real e execução de física complexa. Além disso, foram utilizados notebooks com processador Intel Core i7-11800H, 16 GB de memória RAM e placa gráfica NVIDIA GeForce RTX 3060, proporcionando mobilidade e capacidade de

processamento para testes e ajustes em diferentes cenários. O *setup* para desenvolvimento utilizando um notebook pode ser visto na Figura 13.

Figura 13 – Meta Quest 2 e notebook utilizado, com seus periféricos



Fonte: Autoria Própria

O uso de *hardware* de alto desempenho foi essencial para garantir a estabilidade da aplicação e a responsividade das interações em tempo real. O desenvolvimento em ambiente de *desktop* também permitiu validar as funcionalidades em diferentes configurações antes da execução em dispositivos móveis.

A combinação de tecnologias de *software* e *hardware* de alta performance possibilitou a construção de um sistema de treinamento robusto e eficiente, com capacidade de resposta em tempo real e interação imersiva entre os usuários conectados ao ambiente virtual. A sincronização entre os dispositivos e o comportamento realista dos objetos foram fatores determinantes para a criação de uma plataforma de treinamento de alta fidelidade.

5 Requisitos e Funcionalidades

A definição clara e precisa dos requisitos e funcionalidades de um sistema é uma etapa essencial no processo de desenvolvimento de *software*, garantindo que o sistema atenda às necessidades dos usuários e aos objetivos do projeto. Na engenharia de *software*, os requisitos são definidos como as condições ou capacidades que um sistema deve possuir para satisfazer uma necessidade ou objetivo, enquanto as funcionalidades descrevem o comportamento específico que o sistema deve apresentar para atender a esses requisitos.

No contexto de sistemas baseados em metaverso, os requisitos e funcionalidades envolvem aspectos específicos relacionados à interação em ambientes virtuais, sincronização em tempo real e comportamento de múltiplos usuários. A natureza imersiva e interativa desses sistemas demanda uma abordagem robusta na definição das funcionalidades, especialmente em relação à movimentação, manipulação de objetos e interação entre os usuários.

Serão descritas as funcionalidades e os requisitos que compõem o sistema de treinamento e capacitação baseado em metaverso, com suporte para múltiplos usuários utilizando dispositivos Quest 2 e Quest 3. As funcionalidades foram organizadas de forma lógica, cobrindo desde o acesso inicial ao sistema até a execução e finalização das atividades de treinamento. Os requisitos foram classificados em funcionais e não funcionais para melhor organização e clareza.

As funcionalidades descritas foram definidas com base em uma análise detalhada de sistemas similares disponíveis no mercado, combinada com uma prospecção direta junto a profissionais da área, a fim de identificar as melhores práticas e necessidades específicas para um sistema de treinamento em ambiente de metaverso. Além disso, a fundamentação teórica foi estruturada a partir de uma imersão na literatura especializada, permitindo consolidar as decisões com base em evidências empíricas e estudos prévios sobre RV e interação em ambientes virtuais multiusuário. Essa abordagem metodológica assegura que as funcionalidades atendam aos objetivos do sistema e às expectativas dos usuários em relação à experiência de seu uso.

5.1 Funcionalidades

As funcionalidades do sistema de treinamento no metaverso foram organizadas e descritas conforme o comportamento esperado durante a execução do sistema.

- #### • Gerenciamento de Usuários e Acesso

- Permitir que os usuários acessem o sistema utilizando dispositivos Quest 2 ou Quest 3.
 - Criar e gerenciar uma *room* inicial para o *lobby*.
 - Permitir que vários usuários acessem o *lobby* simultaneamente.
 - Permitir que o usuário selecione o tipo de usuário.
 - Garantir que apenas o tutor tenha permissão para iniciar o treinamento.
 - Diferenciar os usuários com capacetes de cores distintas.
 - **Locomoção e Navegação no Ambiente**
 - Permitir movimentação física no ambiente com rastreamento posicional.
 - Permitir locomoção por teletransporte, com seleção da direção de orientação após o teletransporte.
 - Permitir movimentação usando *joystick*.
 - Permitir que o usuário olhe para os lados usando o *joystick* ou movendo fisicamente a cabeça.
 - **Sincronização Multiplayer usando Photon PUN 2**
 - Sincronizar a posição e rotação dos jogadores em tempo real.
 - Sincronizar movimento de objetos em todas as instâncias conectadas.
 - Sincronizar o estado de ativação e desativação de botões, animações e painéis.
 - Sincronizar a posição e estado dos jogadores ao entrar na cena de treinamento.
 - Garantir que os capacetes de cores diferentes sejam visíveis para todos os jogadores.
 - **Interface de Seleção de Treinamento**
 - Disponibilizar uma interface no *lobby* para seleção do treinamento desejado.
 - Permitir a seleção do tipo de usuário, *trainee* ou tutor.
 - Visualizar quais treinamentos estão disponíveis.
 - **Início e Gerenciamento do Treinamento**
 - Transferir todos os usuários do *lobby* para a cena de treinamento após o início.
 - Atribuir capacete vermelho para o tutor e capacete amarelo para os *trainees*.
 - Ativar um sistema de fases para o treinamento, permitindo avanço apenas após conclusão da fase anterior.

- **Execução das Etapas do Treinamento**
 - Permitir que todos os jogadores interajam com determinados objetos.
 - Sincronizar em tempo real o movimento dos objetos e ações dos jogadores.
 - Permitir que o treinamento seja executado em paralelo por múltiplos jogadores.
- **Finalização e Repetição do Treinamento**
 - Apresentar uma tela de conclusão ao final do treinamento.
 - Permitir que o tutor e *trainees* escolham entre retornar ao *lobby* ou repetir o treinamento.
 - Garantir que a repetição individual do *trainee* não afete os outros jogadores conectados.

5.2 Requisitos

De acordo com Sommerville [2011], os requisitos de *software* podem ser classificados em funcionais e não funcionais, sendo que os requisitos funcionais descrevem o comportamento e as funcionalidades específicas que o sistema deve apresentar, enquanto os requisitos não funcionais definem as restrições e propriedades de qualidade que o sistema deve obedecer, como desempenho, usabilidade e segurança. Neste trabalho, os requisitos foram estruturados com base nessa classificação proposta por Sommerville [2011], permitindo uma organização clara e detalhada das funcionalidades e restrições do sistema de treinamento em ambiente de metaverso. Essa abordagem proporciona uma definição precisa dos objetivos e comportamentos esperados, além de facilitar o planejamento e implementação das funcionalidades, garantindo que o sistema atenda aos critérios de qualidade e eficiência estabelecidos.

Os requisitos foram divididos em funcionais e não funcionais, visando garantir que o sistema atenda às necessidades específicas de interação e desempenho esperadas em um ambiente de metaverso.

5.2.1 Requisitos Funcionais

- O sistema deve permitir que o usuário selecione o tipo de usuário, *trainee* ou tutor.
- A movimentação deve ser possível por teletransporte, *joystick* e movimentação física.
- A sincronização *multiplayer* deve garantir que todos os usuários visualizem as mesmas interações.

- O sistema deve permitir execução de treinamento em paralelo por múltiplos jogadores.
- A ativação e desativação de botões e objetos interativos devem ser sincronizadas.

5.2.2 Requisitos Não Funcionais

- O sistema deve ser compatível com dispositivos Meta Quest 2 e Quest 3.
- A latência da sincronização *multiplayer* deve ser suficientemente baixa para garantir fluidez.
- A interface deve ser intuitiva e de fácil compreensão.
- O sistema deve suportar pelo menos quatro jogadores simultâneos sem perda significativa de desempenho.

6 Cenários e Modelos Utilizados

A construção de ambientes virtuais imersivos requer a utilização de cenários e modelos tridimensionais detalhados, que sejam representações fiéis dos ambientes reais onde as atividades são realizadas. No desenvolvimento deste projeto, foram empregados dois cenários principais: o *Lobby* e a Sala de Treinamento. Ambos foram previamente modelados e texturizados pela equipe do CEASB, tendo como base o projeto de desenvolvimento do simulador de revisão do acionamento hidráulico dos disjuntores de 500 kV da GIS (Gas Insulated Substation), localizado na Usina de Itaipu [CEASB, 2023].

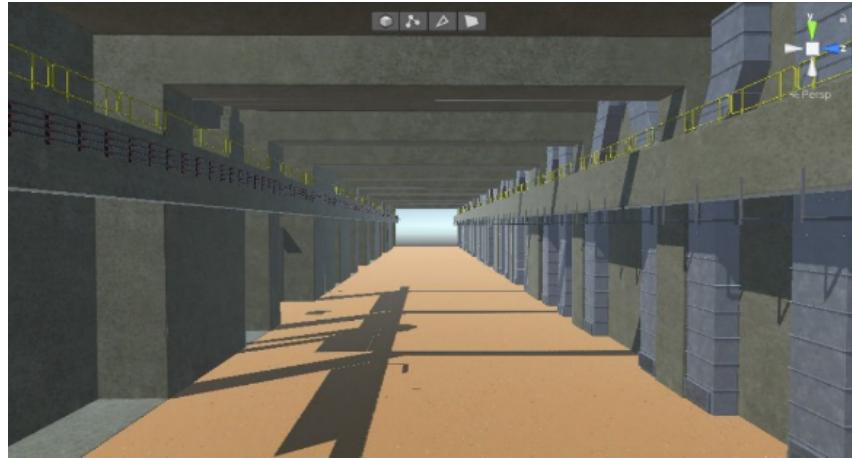
No entanto, o processo de adaptação desses modelos para o ambiente de realidade virtual e a implementação da lógica de treinamento foram realizados neste projeto. Esse trabalho envolveu a integração dos modelos tridimensionais com a Unity, o desenvolvimento de *scripts* em C# para a execução das etapas de treinamento, e a sincronização dos estados dos objetos em ambiente *multiplayer* utilizando Photon PUN 2.

O uso desses modelos não só garante maior realismo e fidelidade ao contexto de treinamento, como também proporciona um ambiente virtual familiar aos operadores, facilitando o processo de aprendizado e transferência de conhecimento. A modelagem desses cenários foi realizada em *softwares* como SolidWorks e Blender, com aplicação de texturização detalhada, permitindo que o ambiente virtual se aproxime ao máximo das condições reais de operação.

6.1 *Lobby*

O *Lobby* é o ambiente inicial do sistema, funcionando como ponto de encontro para os usuários. Esse cenário foi obtido a partir de um modelo detalhado da GIS de 500 kV, localizado na usina de Itaipu. Este modelo foi previamente desenvolvido para a plataforma desktop utilizada no treinamento de manutenção em disjuntores e posteriormente adaptado para o ambiente de realidade virtual, representado na Figura 14.

Figura 14 – Modelo da GIS



Fonte: [CEASB, 2023]

No *lobby*, os usuários podem se locomover utilizando diferentes métodos:

- **Locomoção física:** o usuário pode caminhar fisicamente dentro da área delimitada pelo headset de realidade virtual.
 - **Teletransporte:** é possível selecionar a posição de destino no ambiente e a orientação desejada após o teletransporte.
 - **Joystick:** o usuário pode navegar livremente pelo espaço utilizando os controles manuais.

A interação inicial ocorre no *lobby*, onde os participantes podem selecionar o treinamento desejado e o tipo de usuário (*trainee* ou *tutor*). Apenas o tutor tem permissão para iniciar o treinamento, momento em que todos os usuários são movidos para a cena de treinamento correspondente.

6.2 Sala de Treinamento

A Sala de Treinamento representa o espaço de manutenção dos disjuntores da GIS, visto na Figura 15. Este cenário também foi obtido a partir de modelos desenvolvidos em projetos anteriores pelo CEASB, adaptados para o ambiente de realidade virtual. A modelagem inclui não apenas a estrutura física da sala, mas também as ferramentas e os componentes utilizados nos procedimentos de manutenção, garantindo que o treinamento ocorra em um ambiente virtual realista e detalhado.

Figura 15 – Modelo da sala de treinamento



Fonte: [CEASB, 2023]

No treinamento, os usuários (*trainees* e tutores) podem interagir diretamente com os equipamentos, executar tarefas específicas e observar os resultados em tempo real. Durante o treinamento:

- Os usuários podem manipular ferramentas e componentes utilizando o XR Interaction Toolkit.
 - O tutor tem o papel de acompanhar o desempenho dos trainees e fornecer orientações em tempo real.
 - A troca de estado dos componentes, como desmontagem, limpeza e remontagem é realizada por meio de interações manuais e eventos programados no sistema.

6.3 Modelagem do Capacete e da Lente

O capacete e a lente utilizados pelos personagens foram modelados no Blender para proporcionar maior realismo e personalização dos personagens no ambiente virtual.

O capacete é utilizado como um identificador visual para diferenciar *trainees* e tutores no ambiente virtual. Os tutores são representados por capacetes vermelhos, enquanto os *trainees* usam capacetes amarelos. Essa diferenciação facilita a comunicação visual e a organização das atividades no treinamento.

As lentes foram modeladas para replicar o efeito de profundidade e campo de visão no ambiente virtual. A textura aplicada nas lentes foi ajustada para simular reflexos e distorções ópticas, garantindo uma experiência de imersão mais fiel à realidade.

6.4 Contextualização sobre o Treinamento

O treinamento realizado na plataforma é baseado no procedimento de revisão do acionamento hidráulico dos disjuntores ELK SH3141 – 500 kV, fabricados pela Brown Boveri & Cie, atualmente ABB (Asea Brown Boveri). Os disjuntores estão em operação há mais de 30 anos na subestação blindada de 500 kV da usina de Itaipu, sendo submetidos a inspeções periódicas a cada 2 e 4 anos, e a uma revisão completa *overhaul* a cada 15 anos, conforme visto na Figura 16 abaixo.

Figura 16 – Realização da manutenção *overhaul*



Fonte: [CEASB, 2023]

A motivação para o desenvolvimento do simulador desktop desenvolvido pelo CEASB surgiu da necessidade de capacitar novos operadores para a realização dos procedimentos de manutenção, devido à iminente aposentadoria de profissionais experientes. O treinamento aborda a desmontagem, verificação, troca de componentes e remontagem do acionamento hidráulico, seguindo as diretrizes estabelecidas pelo fabricante e documentadas durante o processo de revisão realizado em 2015.

O treinamento é dividido em etapas sequenciais, cobrindo desde a movimentação inicial dos componentes até a montagem final e testes de funcionamento. Os usuários recebem feedback em tempo real sobre o desempenho e podem repetir o treinamento para consolidar o aprendizado.

Esse formato de treinamento em ambiente virtual não apenas melhora a retenção de conhecimento, como também proporciona uma plataforma segura para simulação de falhas e correções sem risco de danos aos equipamentos reais.

7 Desenvolvimento

7.1 Personagem do Usuário

O personagem no ambiente de treinamento em metaverso foi cuidadosamente projetado para proporcionar uma experiência realista e intuitiva ao usuário. A estrutura do jogador é composta por um capacete, uma lente de óculos e controles que permitem a interação e locomoção no cenário virtual. Essa configuração foi pensada para maximizar a sensação de imersão e proporcionar um controle preciso sobre o ambiente virtual. Para reforçar a diferenciação dos papéis dentro do treinamento, foi estabelecido um sistema de codificação por cores no capacete: os tutores utilizam um capacete vermelho, conforme a Figura 17, enquanto os *trainees* são identificados com um capacete amarelo. Esse esquema de cores permite uma identificação rápida e eficiente dos papéis, o que facilita a dinâmica de interação entre os participantes.

O capacete e a lente de óculos não têm apenas um papel estético, eles também são elementos fundamentais para a configuração da experiência visual e da interação com o ambiente virtual. A definição da cor do capacete é feita automaticamente no momento da inicialização do jogador no treinamento, com base no papel que o usuário escolheu ao ingressar nele. Esse processo é realizado por meio do ‘Game Manager’ um *script* que lê as propriedades definidas no Photon Network e ajusta o material do capacete de acordo com a função atribuída. O uso de diferentes cores para o capacete garante que os usuários consigam rapidamente identificar quem são os tutores e quem são os *trainees*, o que reforça a organização e o alinhamento dos papéis durante o treinamento.

Figura 17 – Capacete vermelho do tutor do treinamento



Fonte: Autoria própria

7.1.1 Controles e Modos de Interação

O jogador interage com o ambiente virtual por meio de controles específicos que utilizam o XR Interaction Toolkit, um pacote da Unity desenvolvido para facilitar a implementação de interações em realidade virtual. Os modelos desse controle podem ser vistos na Figura 18. No ambiente de treinamento, foram configurados três principais tipos de interação:

- **Poke Interactor:** Esse modo de interação permite que o jogador realize ações de toque direto em objetos do ambiente virtual. Ele é utilizado para ações de precisão, como pressionar botões ou manipular objetos pequenos.
- **Near-Far Interactor:** Este modo de interação permite que o jogador manipule objetos tanto a curta quanto a longa distância. Ele é especialmente útil para situações em que o jogador precisa interagir com elementos que estão fora de seu alcance físico, sem que haja a necessidade de se locomover até o objeto.
- **Teleport Interactor:** Esse modo de interação é usado para permitir que o jogador se desloque rapidamente dentro do ambiente virtual por meio de teletransporte. O sistema permite que o jogador selecione o local de destino e a direção em que ficará virado após o teletransporte, proporcionando uma sensação mais natural e confortável de locomoção.

Figura 18 – Modelo do controle representativo da mão do jogador



Fonte: Autoria própria

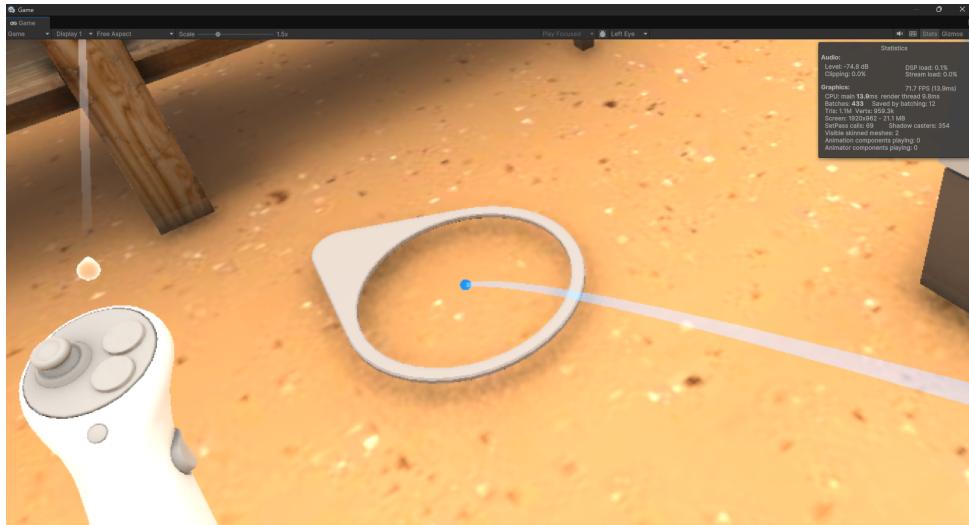
A configuração desses sistemas de interação foi pensada para proporcionar uma experiência fluida e responsiva, garantindo que o jogador tenha controle total sobre suas ações no ambiente virtual.

7.1.2 Locomoção

A locomoção dentro do ambiente virtual foi configurada para permitir diferentes formas de movimentação, com o objetivo de acomodar diferentes níveis de experiência e preferências dos jogadores. O sistema de locomoção inclui:

- **Turn:** Permite que o jogador gire a visão utilizando o *joystick* do controle. Esse modo é especialmente útil para realizar ajustes rápidos de ângulo de visão sem precisar movimentar fisicamente a cabeça.
- **Move:** Permite que o jogador caminhe virtualmente pelo ambiente usando o *joystick* para definir a direção e a velocidade do movimento.
- **Teleportation:** O teletransporte é uma forma rápida e eficiente de locomoção. O jogador seleciona o ponto de destino e é imediatamente transportado até lá. Esse modo de locomoção é útil para reduzir o risco de desconforto e enjojo em realidade virtual, conforme visto na Figura 19 abaixo.

Figura 19 – Teletransporte e seleção da direção final



Fonte: Autoria própria

- **Movimento físico:** O jogador também pode se movimentar fisicamente dentro do ambiente real, o que será refletido em tempo real no ambiente virtual. Esse método explora os seis graus de liberdade do sistema, permitindo que o jogador se mova e olhe livremente em qualquer direção.

A implementação de múltiplos métodos de locomoção foi essencial para acomodar diferentes estilos de jogo e níveis de conforto entre os participantes.

7.2 Gerenciador de Salas

O ‘Room Manager’ é o componente responsável pela configuração inicial do ambiente virtual e pela conexão dos jogadores em uma sala compartilhada. Esse sistema é baseado na arquitetura de rede oferecida pelo Photon PUN 2, que facilita a criação de ambientes *multiplayer* em realidade virtual.

O ‘Room Manager’ é configurado para conectar automaticamente o jogador ao servidor Photon assim que o aplicativo é iniciado. Caso não exista uma sala disponível, o sistema cria uma sala chamada “test” e aloca o jogador nessa sala. Caso uma sala já exista, o jogador é automaticamente inserido na sala em andamento. Essa abordagem garante que todos os jogadores estejam sempre conectados ao mesmo ambiente do servidor.

O processo de seleção de papel é realizado por meio do método `SelectRole()`, que permite que o jogador defina se atuará como tutor ou *trainee*. Essa escolha é armazenada nas propriedades customizadas do Photon Network, permitindo que o ‘Game Manager’ configure os elementos visuais e de interação de acordo com o papel selecionado.

Quando o treinamento é iniciado, o ‘Room Manager’ é responsável por instanciar os jogadores no ambiente e garantir que todas as configurações de papel e posição sejam replicadas para todos os participantes conectados.

7.3 Gerenciador da Sala de Treinamento

O ‘Game Manager’ é o componente que gerencia o comportamento do jogador e o fluxo de execução do treinamento. Ele é responsável por instanciar o jogador no ambiente virtual, configurando os controles e o capacete de acordo com o papel selecionado no ‘Room Manager’.

Após a definição do papel, o ‘Game Manager’ verifica as propriedades armazenadas na rede Photon e configura os elementos gráficos e interativos do personagem. Os tutores recebem um capacete vermelho e os *trainees* recebem um capacete amarelo. Além disso, o ‘Game Manager’ também é responsável por iniciar o treinamento e controlar o progresso das etapas de execução.

O ‘Game Manager’ garante que todas as ações dos jogadores, incluindo movimentação e interações com objetos, sejam sincronizadas entre todos os participantes. Essa sincronização é essencial para manter a consistência e a integridade do ambiente virtual durante o treinamento.

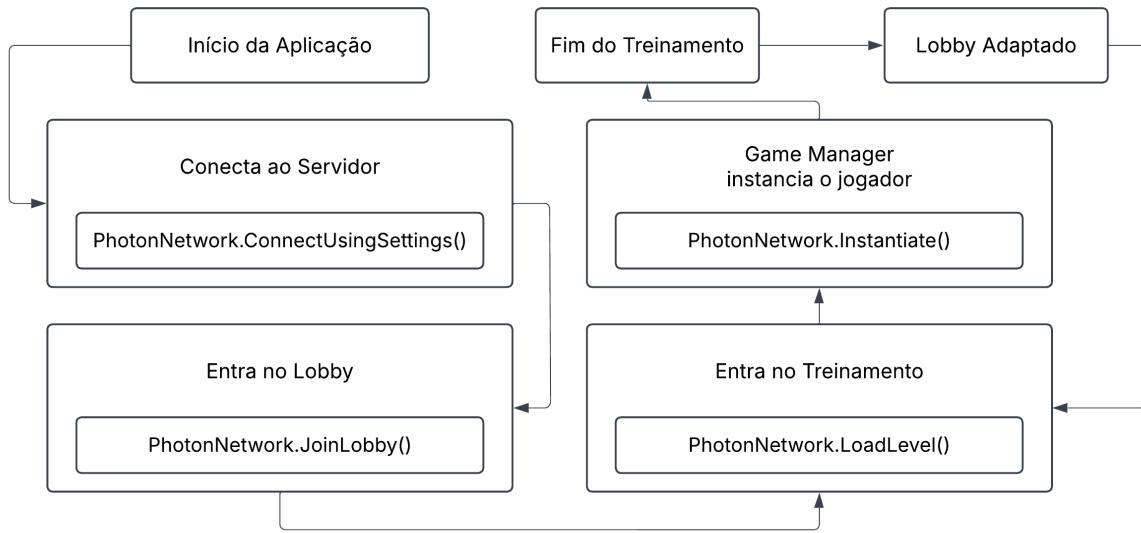
7.4 Configuração do Jogador

O ‘Player Setup’ é o componente que configura os controles e o comportamento local do jogador. Ele é ativado assim que o jogador é instanciado no ambiente virtual, garantindo que o personagem seja totalmente configurado antes do início do treinamento.

O ‘Player Setup’ ativa os sistemas de movimentação, os interatores e os mecanismos de *feedback* tátil. Ele também habilita a câmera e os dispositivos de entrada, garantindo que o jogador tenha controle total sobre o personagem.

O *script* foi desenvolvido para garantir que apenas o jogador local tenha controle sobre o personagem. Isso significa que o jogador tem controle exclusivo sobre seus próprios dispositivos e mecanismos de movimentação, enquanto as ações dos demais jogadores são sincronizadas em tempo real pela rede Photon. A representação de todo esse fluxo pode ser vista na Figura 20.

Figura 20 – Diagrama representativo do fluxo da configuração de cada cena



Fonte: Autoria própria

7.5 Canvas e Interface de Usuário

A interface gráfica de seleção de sala e papel do jogador foi desenvolvida utilizando um componente *Canvas* dentro da Unity, que representa o ponto de interação inicial do usuário no ambiente de *lobby*. O *Canvas* é um elemento fundamental no sistema, pois permite que o jogador configure sua experiência de treinamento antes de ingressar na sala de execução. Essa configuração inclui a seleção da sala onde o treinamento será realizado e a definição do tipo de usuário, que pode ser *tutor* ou *trainee*, a representação de uma tela deste *Canvas* pode ser visto na Figura 21.

O *Canvas* foi configurado para funcionar de maneira independente para cada jogador, sem sincronização via Photon PUN. Essa decisão foi tomada para garantir que cada usuário pudesse personalizar sua experiência de maneira isolada, sem interferência ou sobreposição das escolhas de outros jogadores. Quando um jogador seleciona o tipo de papel ou a sala desejada, essa escolha é processada localmente, sem impacto nas configurações dos demais jogadores conectados. Esse comportamento é possível porque o *Canvas* e seus componentes, como botões, *dropdowns* e textos, são elementos locais da Unity, não sendo replicados pela rede do Photon. Assim, cada jogador visualiza apenas o próprio *Canvas* e suas alterações, o que permite que diferentes jogadores realizem seleções distintas sem conflito.

7.6 Troca de Sala e Sincronização de Estado

A troca de sala é gerenciada pelo método `LoadLevel()` do Photon PUN, que é utilizado para carregar dinamicamente diferentes cenas dentro do ambiente virtual. Esse método é uma funcionalidade integrada do Photon que permite que o jogo altere o ambiente em execução sem que o jogador precise sair e reconectar à plataforma. Quando um jogador seleciona a sala desejada e confirma a entrada no treinamento, o método `LoadLevel()` é invocado para carregar a cena correspondente ao treinamento escolhido.

Contudo, o método `LoadLevel()` só pode ser executado pelo *Master Client*. No Photon PUN, o *Master Client* é o jogador que foi o primeiro a se conectar à sala. Ele atua como o servidor principal e tem permissão para controlar a execução de eventos globais na sessão *multiplayer*, como a troca de cena, o início do treinamento e a sincronização de elementos críticos dentro do ambiente virtual.

Esse comportamento implica que, se o tutor for responsável por iniciar o treinamento, ele deve ser o primeiro jogador a se conectar ao ambiente de *lobby*. Caso o tutor não seja o primeiro a ingressar na sala, ele não será automaticamente designado como *Master Client* e, consequentemente, não terá permissão para executar o comando de troca de cena. Para garantir que o tutor tenha controle sobre o início do treinamento, ele deve ser o primeiro a conectar-se ao ambiente.

Para que o comando de troca de cena seja executado, o sistema verifica se o jogador que está executando o comando é o *Master Client*. Essa verificação é feita por meio da função `IsMasterClient()`, que retorna um valor booleano indicando se o jogador local é o *Master Client*. Se a função retornar verdadeiro, o comando de troca de cena é permitido e o método `LoadLevel()` é executado, transferindo todos os jogadores para a cena de treinamento.

7.6.1 Desenvolvimento e Configuração

Durante o desenvolvimento, o *Canvas* foi estruturado de forma modular, permitindo fácil manutenção e expansão futura. A interface foi organizada em um sistema de hierarquia que permite a fácil localização e modificação de cada componente. A seleção de papel e sala foi implementada por meio de *dropdowns* e botões da Unity, que foram configurados para armazenar as escolhas do jogador em variáveis locais. Essas variáveis são posteriormente utilizadas pelo sistema para configurar o comportamento do jogador na sala de treinamento.

O *script* que controla a seleção de papel e sala foi desenvolvido utilizando o método `SelectRole()`, que é executado quando o jogador confirma sua escolha na interface. O método armazena o papel selecionado em uma propriedade customizada do Photon

Network, o que permite que o sistema configure o comportamento do jogador após a transição de cena. Além disso, o método também invoca o comando de troca de cena utilizando `LoadLevel()`, caso o jogador local seja o *Master Client*. Esse processo garante que o comportamento de inicialização do treinamento seja consistente e coordenado entre todos os jogadores conectados.

A implementação desse sistema foi essencial para garantir uma experiência organizada e sem conflitos para os jogadores. A separação das configurações locais e das ações globais permite que cada jogador tenha controle sobre sua configuração individual, enquanto as ações que afetam o estado global do ambiente, como a troca de cena, permanecem restritas ao controle do *Master Client*. Essa abordagem reforça a estabilidade e a integridade do ambiente de treinamento, proporcionando uma experiência coesa e sincronizada para todos os participantes.

Figura 21 – *Canvas* assíncrono no *Lobby*



Fonte: Autoria própria

7.7 *Canvas* Síncrono e Sincronização de Estado

Além do *Canvas* localizado no *lobby*, que funciona de maneira local e isolada para cada jogador, há também um *Canvas* síncrono, demonstrado pela Figura 22, que é atualizado em tempo real para todos os jogadores conectados à sala. Esse *Canvas* é utilizado para exibir informações globais ou configurar elementos que precisam estar visíveis de maneira uniforme para todos os usuários.

A sincronização desse *Canvas* é feita por meio de chamadas remotas de procedimento, chamadas RPCs (Remote Procedure Calls), utilizando o Photon PUN. Quando um jogador realiza uma ação que altera o estado do *Canvas*, essa ação é transmitida para todos os jogadores por meio de um método RPC, garantindo que o estado do *Canvas* permaneça consistente entre todos os usuários conectados. Essa funcionalidade é essen-

cial para que a experiência de treinamento seja compartilhada e coesa, sem discrepâncias entre os jogadores.

A implementação dessa sincronização foi realizada por meio do uso de um componente ‘PhotonView’, que é um *script* da API (Application Programming Interface) Photon PUN associado a cada objeto que deve ser sincronizado. O ‘PhotonView’ permite que um objeto seja identificado de maneira única na rede e que seus métodos possam ser chamados remotamente por qualquer jogador. No caso do *Canvas*, o objeto é sincronizado utilizando o *script* ‘SyncActivation’, desenvolvido para permitir a transmissão de comandos RPC que alteram o estado do *Canvas* e de outros elementos da interface.

Figura 22 – *Canvas* síncrono na sala de treinamento



Fonte: Autoria própria

7.8 Sincronização de Estado com ‘SyncActivation’

O *script* ‘SyncActivation’ é responsável por controlar a ativação e desativação de objetos no *Canvas* e em outras partes do cenário. Esse controle é realizado utilizando o método `ToggleObject()`, que é executado localmente e, em seguida, transmitido para os demais jogadores utilizando o comando RPC. Quando um jogador ativa ou desativa um objeto, o comando é enviado para todos os jogadores conectados por meio do atributo `AllBuffered` da classe `RpcTarget`, o que garante que todos os jogadores, incluindo novos jogadores que ingressarem na sala, receberão o estado atualizado.

O *script* ‘SyncActivation’ inclui diversas funcionalidades que foram desenvolvidas para permitir a manipulação de objetos de maneira eficiente e sincronizada. As principais funcionalidades implementadas incluem:

- **ToggleObject:** Alterna o estado de um objeto (ativado ou desativado) e transmite essa alteração para todos os jogadores conectados.

- **SetActiveState**: Ativa ou desativa um objeto específico com base em um valor booleano fornecido como parâmetro.
- **ActivateObject** e **DeactivateObject**: Ativam ou desativam um objeto específico, identificado pelo nome, e sincronizam essa mudança com todos os jogadores.
- **ToggleChildrenObjects**: Ativa ou desativa todos os objetos filhos de um objeto pai e sincroniza esse comportamento na rede.

Por exemplo, ao ativar o Canvas de treinamento após a transição de cena, o método `ActivateObject()` é chamado para ativar o *Canvas* e garantir que ele seja visível para todos os jogadores. Como a ativação é transmitida via RPC, qualquer jogador que se conecte posteriormente verá o *Canvas* no estado correto, mesmo que a ativação tenha ocorrido antes de sua conexão.

7.9 Sincronização de Animações com ‘SyncAnimation’

O *script* ‘SyncAnimation’ foi desenvolvido para controlar a execução de animações de maneira sincronizada entre todos os jogadores conectados à sala. Para garantir essa sincronização, o *script* utiliza o método `PlayAnimation()`, que recebe o nome de um gatilho da animação e transmite esse comando para todos os jogadores utilizando o comando RPC.

Quando o comando RPC é recebido por outros jogadores, é chamado localmente o método `SyncAnimationRPC()`, ativando o gatilho da animação no componente *Animator* associado ao objeto. Essa abordagem garante que todos os jogadores vejam a mesma sequência de animação em tempo real, mantendo a consistência visual do treinamento.

As principais funcionalidades do *script* ‘SyncAnimation’ incluem:

- **PlayAnimation**: Recebe o nome do gatilho da animação e transmite o comando para todos os jogadores conectados.
- **SyncAnimationRPC**: Ativa o gatilho da animação no componente *Animator* localmente, garantindo que a animação seja executada de maneira sincronizada para todos os jogadores.

Essa abordagem é especialmente útil em situações onde a execução de eventos visuais deve ser simultânea para todos os jogadores. Por exemplo, ao iniciar uma nova etapa do treinamento, uma animação de transição pode ser exibida de maneira sincronizada para garantir que todos os jogadores tenham a mesma percepção do estado do ambiente.

7.10 Sincronização de Comportamento com ‘GrabSync’

O *script* ‘GrabSync’ é responsável por sincronizar o comportamento de objetos que podem ser manipulados diretamente pelos jogadores, ação demonstrada pela Figura 23. Esse script é fundamental para garantir que as interações físicas com o ambiente sejam consistentes para todos os jogadores conectados à sala.

O *script* utiliza o componente ‘PhotonView’ para gerenciar a propriedade do objeto manipulado. Quando um jogador pega um objeto utilizando o componente ‘XR Grab Interactable’ do XR Interaction Toolkit, o *script* solicita a posse do objeto por meio do método `RequestOwnership()`. Esse método transfere a posse do objeto para o jogador local, permitindo que ele manipule o objeto fisicamente. Essa manipulação é transmitida para os demais jogadores utilizando o comando RPC, garantindo que a posição e o estado do objeto sejam replicados corretamente na rede.

As principais funcionalidades do *script* ‘GrabSync’ incluem:

- **OnSelectEnter:** Chamado quando o jogador pega um objeto. Solicita a posse do objeto por meio de `RequestOwnership()` e permite que o jogador manipule o objeto.
- **Update:** Garante que a física do objeto seja aplicada corretamente apenas para o jogador que detém a posse do objeto. Se o objeto não estiver sob posse do jogador local, o comportamento físico é desativado para evitar inconsistências.

Essa abordagem é essencial para garantir que objetos manipuláveis respondam de maneira consistente às interações dos jogadores, sem atrasos ou divergências de comportamento entre diferentes jogadores. A combinação de transferência de posse e replicação de estado em tempo real permite que a experiência de treinamento em realidade virtual seja colaborativa.

Figura 23 – Ação de segurar objeto



Fonte: Autoria própria

7.11 Considerações sobre Sincronização

A implementação dos *scripts* ‘SyncActivation’, ‘SyncAnimation’ e ‘GrabSync’ foi essencial para garantir a consistência e a coerência das interações no ambiente de treinamento. A abordagem baseada em RPC permitiu que eventos locais fossem transmitidos para todos os jogadores conectados, enquanto o uso de transferência de posse e replicação de estado garantiu que as interações físicas e visuais fossem sincronizadas. Essa combinação de técnicas proporcionou uma experiência de treinamento imersiva e colaborativa, permitindo que jogadores localizados em diferentes regiões geográficas interagissem com o ambiente e entre si de maneira natural e sem discrepâncias de estado.

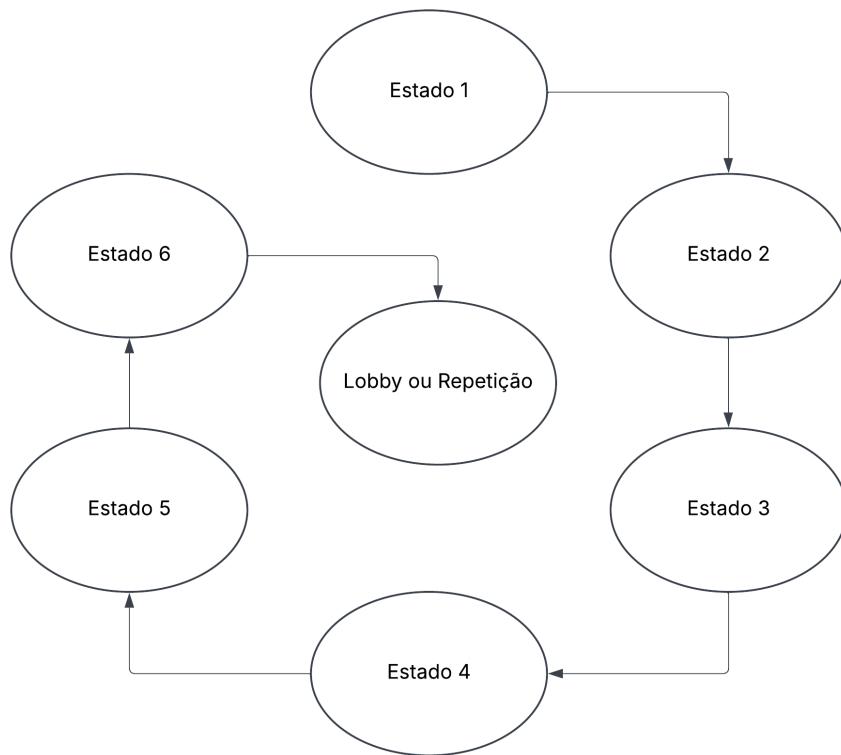
7.12 Gerenciamento de Estados no Treinamento

O gerenciamento dos estados do treinamento foi implementado utilizando uma máquina de estados finitos (FSM). As máquinas de estados são um modelo computacional que permite a definição de um conjunto finito de estados, transições entre esses estados e ações associadas a cada transição ou estado específico. Esse modelo é amplamente utilizado em jogos e sistemas interativos, pois permite estruturar o comportamento de sistemas complexos de maneira organizada e previsível, facilitando o controle do fluxo de execução e a implementação de respostas a eventos.

Segundo Millington [2019], as máquinas de estados são um dos principais métodos para implementação de lógica de jogo, pois permitem estruturar o comportamento de personagens, sistemas de combate, movimentação e interações com o ambiente em estados bem definidos e interligados por transições claras. Em jogos *multiplayer*, o uso de FSM

é ainda mais importante, pois o estado de um jogador ou objeto precisa ser atualizado e refletido em todos os clientes conectados em tempo real, o que requer sincronização eficiente através de redes. A máquina de estados referente ao treinamento implementado é representada pela Figura 24, onde o estado inicial é o “Estado 1”.

Figura 24 – Máquina de estados do treinamento implementado



Fonte: Autoria própria

No sistema desenvolvido, o *script* ‘StageManager’ é responsável por controlar o comportamento do treinamento com base nesse modelo de máquina de estados. O sistema é projetado para funcionar em um ambiente *multiplayer*, o que significa que o estado atual e as transições entre estados devem ser sincronizados entre todos os jogadores conectados à sessão, independentemente de sua função (tutor ou *trainee*). Para isso, o Photon PUN 2 é utilizado para gerenciar a sincronização em tempo real por meio de chamadas remotas de procedimento.

No contexto da aplicação, a variável `stage`, definida como um valor inteiro no *script* ‘Stage Manager’, representa o estado atual do treinamento. A lógica de transição entre estados é implementada utilizando blocos condicionais e métodos específicos que monitoram eventos de interação e mudanças no ambiente. O sistema é dividido em seis estados principais, que são detalhados a seguir:

7.12.1 Estado 1: Movimentação da Transpaleteira

No primeiro estado, o jogador deve movimentar a transpaleteira até uma posição específica dentro do ambiente virtual. Esse movimento é feito com um *click* no botão de “Começar”, acima da transpaleteira. Esse estado é crítico para o sucesso das próximas etapas, pois o posicionamento correto da transpaleteira define o ponto inicial para o uso do gancho e o manuseio do acumulador hidráulico.

Quando a transpaleteira atinge a posição correta, conforme visto na Figura 25, o *script* ‘Stage Manager’ detecta essa condição e aciona a transição para o próximo estado. Essa transição é sincronizada com todos os jogadores através de uma chamada RPC, garantindo que o novo estado seja atualizado em tempo real para todos os usuários conectados na sessão.

Figura 25 – Transpaleteira movida após primeira ação



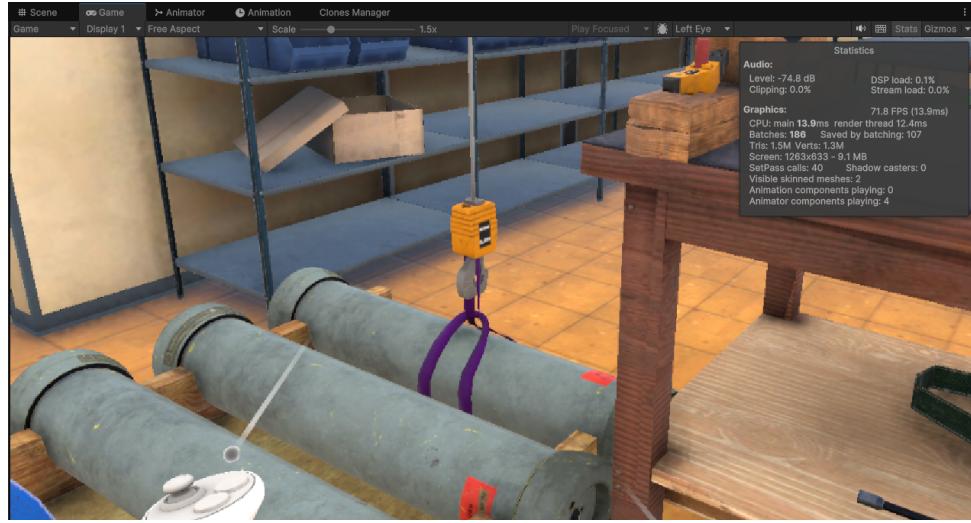
Fonte: Autoria própria

7.12.2 Estado 2: Movimento do Gancho para Baixo

No segundo estado, o jogador deve ativar o movimento do gancho para baixo para que ele se acople ao acumulador hidráulico. Esse movimento é implementado com um *Poke-Interactor*, ao pressionar um botão em cima da mesa de manutenção, que muda gradualmente a posição do gancho. Quando o jogador aciona o botão, o *script* ‘Stage Manager’ envia uma chamada RPC para que o movimento seja iniciado simultaneamente em todos os dispositivos conectados.

O *script* monitora continuamente a posição do gancho, onde `MoveTowards()` é utilizado para ajustar o valor da posição ao longo do eixo Y. Quando o gancho atinge a posição desejada, visto na Figura 26, o sistema atualiza automaticamente o valor de `stage` e aciona a próxima transição de estado.

Figura 26 – Gancho movido para baixo após pressionar o botão



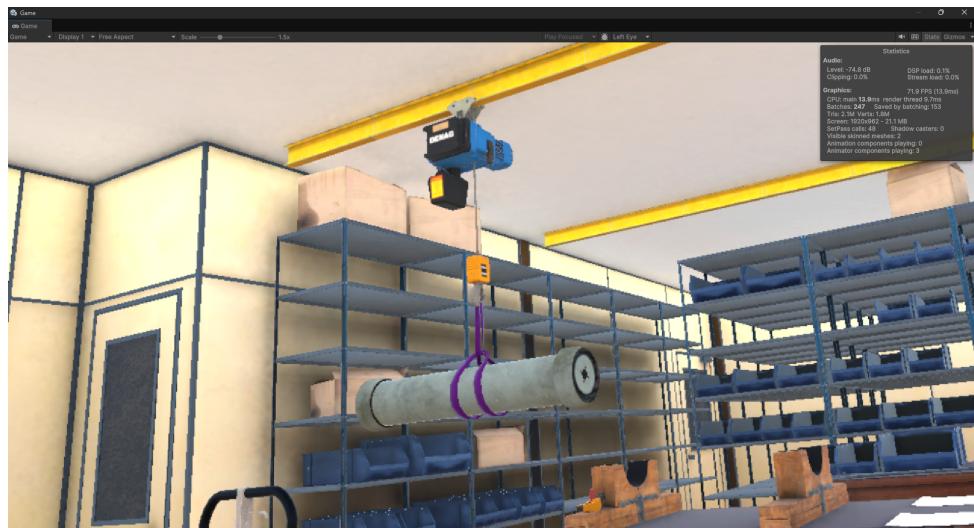
Fonte: Autoria própria

7.12.3 Estado 3: Movimento do Gancho com o Acumulador Hidráulico para Cima

Após o gancho ter se acoplado ao acumulador, o jogador deve levantar o acumulador até uma posição específica. Esse movimento é controlado por uma sincronização de posição associada ao acumulador. A transição para esse estado é ativada automaticamente pelo sistema quando o gancho atinge a posição final na etapa anterior.

A movimentação é sincronizada por meio de chamadas RPC, garantindo que a posição do acumulador seja refletida corretamente para todos os jogadores conectados. Além disso, o *script* monitora o progresso do movimento e, ao atingir a posição desejada, atualiza o valor de `stage` para o próximo estado. A posição desejada desse estado pode ser vista na Figura 27.

Figura 27 – Gancho movido para cima após acoplar acumulador hidráulico



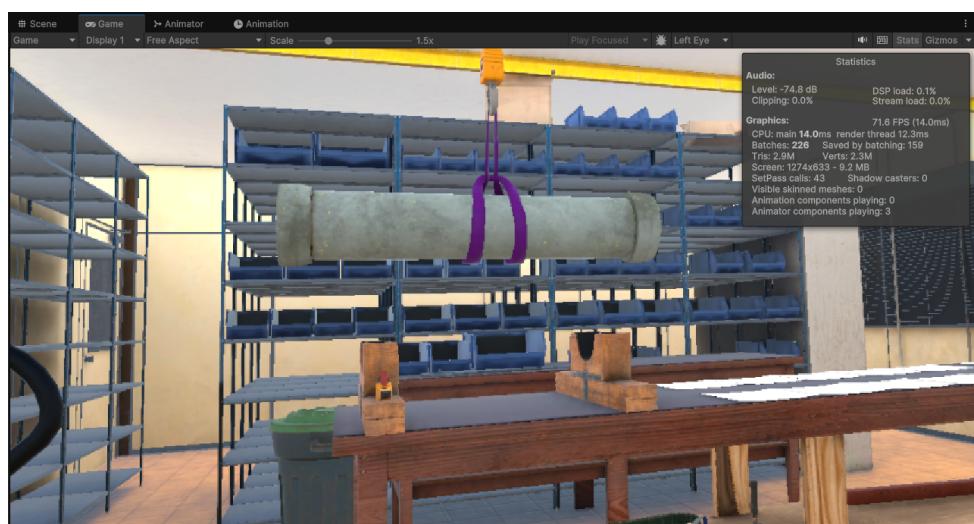
Fonte: Autoria própria

7.12.4 Estado 4: Movimento do Guindaste com o Acumulador para o Lado

No quarto estado, o jogador deve realizar o movimento lateral do guindaste com o acumulador preso. Esse movimento também é implementado atualizando a posição do guindaste. A ativação da animação é feita através de uma chamada RPC para garantir que todos os jogadores conectados vejam o movimento lateral em tempo real.

Durante esse estado, o *script* monitora o progresso da movimentação e, quando o acumulador atinge a posição final, conforme a Figura 28, o valor de **stage** é atualizado automaticamente para o próximo estágio. Esse controle baseado em FSM assegura que o jogador não consiga avançar para a próxima fase sem completar essa etapa com sucesso.

Figura 28 – Guindaste movido para o lado após movimento para cima



Fonte: Autoria própria

7.12.5 Estado 5: Remoção das Peças Sujas do Acumulador

Com o acumulador posicionado corretamente, o jogador deve remover as peças sujas. Esse processo é realizado utilizando uma animação que é sincronizada para todos os jogadores quando realizada.

A remoção de cada peça é monitorada pelo *script* ‘Stage Manager’. Quando uma peça é removida com sucesso, o sistema registra essa ação e atualiza o estado. O progresso da remoção das peças é refletido para todos os jogadores conectados através de RPCs, garantindo consistência na execução da tarefa. As peças removidas e posicionadas na mesa podem ser vistas na Figura 29.

Figura 29 – Peças sujas removidas do acumulador



Fonte: Autoria própria

7.12.6 Estado 6: Limpeza das Peças

O sexto e último estado corresponde à limpeza das peças removidas. Esse estado é tratado internamente no código como duas etapas (estados 6 e 7), mas do ponto de vista da experiência do usuário, ele é apresentado como um único estágio de limpeza. Cada peça removida precisa ser manipulada de forma a simular o processo de limpeza, alterando a textura da peça para representar um estado “limpo”, conforme visto na Figura 30.

A conclusão dessa etapa é monitorada pelo *script* ‘Stage Manager’, que contabiliza o número de peças limpas e, ao atingir o valor esperado, registra a conclusão do treinamento.

7.13 Tela de Retorno ou Repetição do Treinamento

Após a conclusão de todas as etapas do treinamento, o sistema exibe uma tela de finalização no *Canvas*. Nessa tela, o usuário tem a opção de retornar ao *lobby* ou

repetir o treinamento. Essa funcionalidade é essencial para permitir que o usuário reforce o aprendizado por meio da repetição ou que finalize a sessão e retorne ao ambiente inicial para potencialmente realizar novos treinamentos ou interações.

Caso o usuário opte por retornar ao *lobby*, ele será direcionado para uma cena adaptada do *lobby* inicial. Essa cena adaptada é gerenciada por dois *scripts* principais: o ‘Adapted Game Manager’ e o ‘Adapted Room Manager’, os quais foram desenvolvidos para manter a consistência da experiência do usuário e garantir a sincronização no ambiente *multiplayer*.

Figura 30 – *Canvas* final do treinamento, com as duas opções possíveis



Fonte: Autoria própria

7.14 Sincronização dos Estados no *Multiplayer*

A sincronização dos estados entre os jogadores é um aspecto crítico da implementação da FSM no sistema. O *script* ‘Stage Manager’ utiliza RPCs para acionar as mudanças de estado e para ativar ou desativar animações e interações. Essa abordagem assegura que o estado atual do treinamento seja consistente para todos os jogadores conectados, eliminando discrepâncias entre o que é exibido para cada participante.

A estrutura baseada em máquina de estados permitiu estruturar o treinamento de maneira clara e organizada, facilitando a manutenção do código e permitindo a expansão futura das funcionalidades do sistema. Essa implementação também assegura que o progresso do treinamento e as interações com os objetos virtuais sejam consistentes entre todos os jogadores conectados na sessão *multiplayer*.

A sincronização de estados e comportamentos entre múltiplos usuários em tempo real é um aspecto essencial para garantir a consistência e orealismo em um ambiente de treinamento em realidade virtual. A plataforma desenvolvida implementa diferentes

métodos para garantir que as interações e movimentos dos objetos no treinamento sejam refletidos para todos os usuários conectados à mesma sala. Para isso, foram utilizados principalmente componentes da biblioteca Photon PUN 2, que oferece suporte para redes *multiplayer* com baixa latência e alta escalabilidade.

A sincronização foi implementada de diferentes formas, dependendo da natureza e do comportamento do objeto. A seguir, são detalhados os métodos utilizados:

7.14.1 Componentes de Sincronização

O componente ‘Photon View’ é a base para a sincronização de objetos em ambientes *multiplayer* com Photon. Ele permite que objetos na cena sejam reconhecidos como entidades de rede, permitindo que suas propriedades sejam atualizadas e replicadas automaticamente para todos os jogadores conectados à mesma sala.

Já o ‘Photon Transform View’ é responsável pela sincronização de propriedades físicas, como posição, rotação e escala. Este componente é essencial para garantir que objetos manipuláveis, como a transpaleteira e o guindaste, apareçam com a posição e orientação corretas para todos os jogadores.

No sistema desenvolvido, o ‘Photon View’ e o ‘Photon Transform View’ foram aplicados para sincronizar:

- Posição e rotação da transpaleteira durante o movimento inicial.
- Posição e rotação do guindaste durante o movimento vertical e horizontal.
- Posição e rotação do acumulador hidráulico durante o movimento vertical e horizontal.
- Movimento dos objetos seguráveis pelos usuários com XR Interaction Toolkit.

7.14.2 Sincronização via RPC para Interface

A interface de seleção de treinamento e tipo de usuário foi implementada com um *Canvas* que é ativado e desativado dinamicamente usando chamadas RPCs. A sincronização da interface permite que:

- O *canvas* de seleção de tipo de usuário e treinamento seja ativado ou desativado dinamicamente com uma única chamada de rede.
- A visibilidade da interface é controlada por eventos específicos, sendo possível exemplificar com o início do treinamento ou a finalização de uma etapa.

- A decisão de trocar de cena ou retornar ao *lobby* após o treinamento seja sincronizada para todos os usuários conectados.

A sincronização do *canvas* foi feita utilizando RPC, que transmite o comando para todos os jogadores conectados com o atributo **AllBuffered**, garantindo que qualquer usuário que entre posteriormente na sala veja o mesmo estado da interface.

7.14.3 Sincronização Contínua de Posição com RPC

Em algumas situações, a sincronização automática de posição e rotação pelo ‘Photon Transform View’ não foi suficiente para garantir uma atualização precisa do estado dos objetos, especialmente em casos de movimentos rápidos ou manipulação direta por interações. Para resolver esse problema, foi implementado um método adicional de sincronização utilizando RPC para enviar diretamente a posição e rotação dos objetos em cada atualização de frame.

Este método foi utilizado para:

- Sincronização do gancho durante o movimento vertical e horizontal nos estados 2, 3 e 4.
- Sincronização da posição e rotação do acumulador durante o movimento de deslocamento e montagem.
- Sincronização da posição dos objetos durante o processo de limpeza e reinstalação.

Esse método foi implementado com o RPC e transmitido para todos os jogadores conectados com **AllBuffered**, para garantir consistência no estado de cada objeto.

7.14.4 Sincronização de Animações com RPC

Para garantir que animações complexas, como o movimento da transpaleteira e a remoção das peças sujas sejam exibidas de forma sincronizada para todos os usuários, foi implementada uma RPC específica para o controle das animações.

Essa abordagem permitiu que:

- As animações de movimentação de peças fossem iniciadas de forma sincronizada.
- Os estados das animações fossem propagados para todos os jogadores.

A sincronização das animações foi feita usando RPC para acionar os gatilhos das animações dos componentes ‘Animator’ de cada objeto.

A Tabela 1 a seguir resume quais métodos de sincronização foram utilizados em cada etapa do treinamento:

Tabela 1 – Métodos de sincronização utilizados em cada etapa do treinamento

Estado	Descrição	Método de Sincronização
1	Movimentação inicial da transpaleteira	RPC para ativação e desativação + RPC para animações
2	Movimento vertical do gancho	RPC para atualização de posição
3	Movimento do acumulador para cima	RPC para atualização de posição
4	Movimento lateral do guindaste com acumulador	RPC para atualização de posição
5	Desmontagem e limpeza das peças	RPC para animações
6	Limpeza das peças e montagem final	Photon View + Photon View Transform + RPC para ativação e desativação

Fonte: Autoria própria

7.15 Gerenciadores de Salas Adaptados

O *script* ‘Adapted Game Manager’ é responsável por instanciar o jogador na posição inicial dentro da cena adaptada. Esse processo ocorre automaticamente após o jogador ser transportado para a nova cena. Para garantir que o jogador seja configurado corretamente, o *script* utiliza o método `Instantiate()`, que permite que o jogador e suas propriedades sejam sincronizados em tempo real com os demais participantes na sala.

No método `Start()`, o *script* verifica se o jogador já está em uma sala com o método `InRoom()`. Caso essa verificação retorne verdadeira, o jogador é instanciado na posição definida por um ponto definido como *spawn*, indicado pela variável `spawnPoint`. Após ser instanciado, o *script* configura o jogador utilizando um método que atribui ao jogador as funcionalidades e interações adequadas para o ambiente.

Essa configuração permite que o jogador, após retornar ao *lobby*, mantenha o mesmo comportamento de movimentação, interação e uso dos dispositivos de controle, como o *joystick* e o teleporte, definidos anteriormente na fase inicial do treinamento. Dessa forma, o processo de transição entre o treinamento e o *lobby* ocorre de maneira fluida e integrada ao restante do sistema.

O *script* ‘Adapted Room Manager’ é responsável por gerenciar a sala e definir o

papel do jogador dentro da nova cena. O método `SelectRole()` permite que o jogador escolha entre os papéis de tutor e *trainee*, armazenando essa informação como uma propriedade personalizada no sistema do Photon. Essa informação é armazenada no objeto `playerProperties`, que é uma instância da classe `Hashtable`. A propriedade `Role` recebe o valor correspondente à função selecionada pelo jogador e, em seguida, é configurada para o jogador local com o método `SetCustomProperties()`.

Após a seleção do papel, o jogador tem a opção de iniciar o treinamento novamente. Essa funcionalidade é implementada por meio do método `StartGame()`, que carrega a cena correspondente ao treinamento. Esse método é protegido para garantir que apenas o *Master Client*, o primeiro jogador a entrar na sala, tenha permissão para iniciar a troca de cena. Caso o jogador que tentar iniciar o treinamento não seja o *Master Client*, uma mensagem de aviso é exibida no console, informando que apenas o jogador com essa posição pode iniciar o processo de troca de cena.

A sincronização das ações entre os jogadores é garantida pelo Photon, que assegura que todos os jogadores na sala compartilhem o mesmo estado da cena. Esse comportamento é essencial para que o ambiente *multiplayer* funcione corretamente, permitindo que todos os jogadores tenham a mesma percepção do estado do ambiente e das interações realizadas pelos demais jogadores. Essa funcionalidade reforça o caráter colaborativo do treinamento e garante a consistência da experiência do usuário.

A tela de retorno ou repetição foi desenvolvida para proporcionar ao usuário o controle sobre a continuidade do treinamento. Ao permitir que o usuário repita o treinamento de maneira individual ou retorne ao ambiente inicial para explorar outras opções de treinamento, o sistema promove um ciclo de aprendizado contínuo e adaptativo.

7.16 Integração de Novos Treinamentos

A plataforma foi desenvolvida de maneira modular, permitindo a adição de novos treinamentos sem a necessidade de modificações estruturais no sistema base. Esse modelo de implementação oferece flexibilidade e escalabilidade, garantindo que novos cenários e atividades possam ser facilmente incorporados ao sistema, ampliando sua aplicabilidade para diferentes contextos de treinamento profissional.

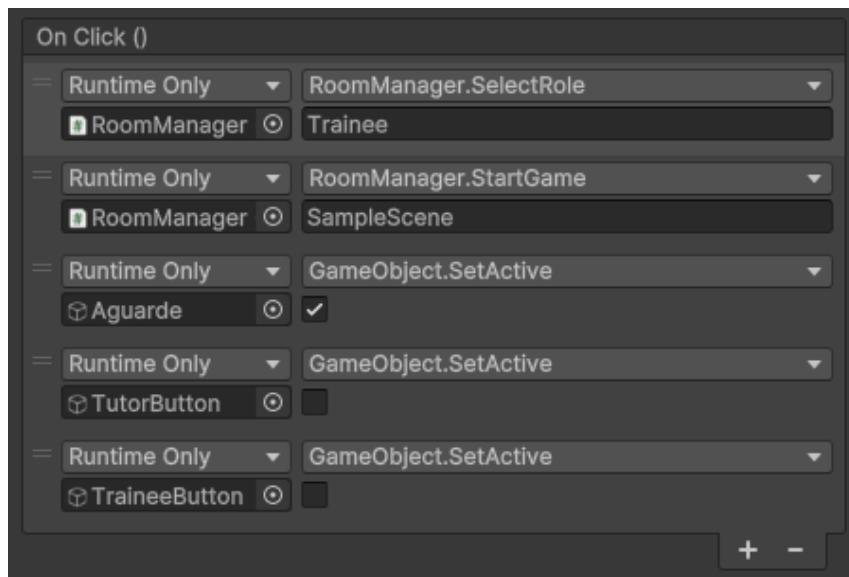
A integração de um novo treinamento à plataforma pode ser realizada por meio da adição de um novo botão na interface de seleção de treinamentos disponível no *Lobby*. O sistema foi estruturado para que cada treinamento seja representado por uma cena independente, que pode ser carregada dinamicamente utilizando o sistema de gerenciamento de cenas do Photon PUN 2. Essa abordagem permite que o processo de carregamento e troca entre os ambientes de treinamento ocorra de forma rápida e sincronizada entre

todos os participantes conectados.

O processo de integração de um novo treinamento pode ser descrito em quatro etapas principais:

- 1. Criação da cena do novo treinamento:** A criação de um novo treinamento na plataforma envolve o desenvolvimento de uma nova cena no Unity. Essa cena deve conter, no mínimo, um ponto de *spawn* para a posição inicial do jogador e um ‘GameManager’ para gerenciar a lógica do treinamento e a sincronização de eventos entre os jogadores conectados. Essa estrutura garante que todos os jogadores sejam posicionados corretamente ao ingressarem no novo treinamento.
 - 2. Adição de um novo botão na interface do Lobby:** Para que o novo treinamento fique disponível para seleção, é necessário adicionar um novo botão na interface do ‘Canvas do Lobby’. Esse botão será responsável por acionar o sistema de troca de cenas, permitindo que o treinamento seja carregado dinamicamente.
 - 3. Configuração do evento de clique do botão:** O evento `onClick()` do botão deve ser configurado para chamar a função de carregamento de cena por meio do comando `LoadLevel()` do Photon PUN. Esse comando é utilizado para carregar a nova cena de maneira sincronizada entre todos os jogadores conectados. Assim, ao selecionar o treinamento, todos os participantes serão transferidos para a nova cena de treinamento simultaneamente. Nota-se o exemplo da Figura 31, onde para se carregar outro treinamento, o nome do parâmetro *SampleScene* deve ser mudado para o nome referente ao novo treinamento.

Figura 31 – Funções chamadas pelo botão de carregar sala como *trainee*



Fonte: Autoria própria

4. **Configuração do ‘GameManager’ na nova cena:** Após o carregamento da nova cena, o ‘GameManager’ será responsável por inicializar o treinamento, definir o comportamento dos objetos e gerenciar as interações dos usuários. O ‘GameManager’ também será o responsável por registrar e sincronizar o progresso do treinamento entre os participantes conectados.

8 Resultados

O processo de desenvolvimento resultou em um sistema funcional, que foi submetido a testes em ambiente controlado para validar sua usabilidade e impacto na experiência do usuário. A plataforma foi projetada para oferecer uma experiência imersiva e interativa, permitindo que *trainees* e tutores realizassem treinamentos de forma colaborativa em um ambiente virtual simulado.

8.1 Avaliação do Sistema

8.1.1 Objetivo da Avaliação

A avaliação foi conduzida com o objetivo de verificar a usabilidade da plataforma de treinamento em ambiente de metaverso, considerando o impacto na experiência dos usuários. O experimento foi previamente aprovado pelo Comitê de Ética em Pesquisa sob o número de CAAE 86355425.4.0000.0107, assegurando que todos os procedimentos estivessem em conformidade com os princípios éticos e regulamentares. O estudo também buscou identificar pontos de melhoria na interface e na interação com o ambiente virtual, coletados por meio das sugestões dos usuários, além de avaliar a suscetibilidade a sintomas de cinetose (*motion sickness*), com uso de um questionário chamado *Simulator Sickness Questionnaire* (SSQ) e a percepção geral dos participantes em relação à navegação e à utilidade da plataforma com um questionário chamado *System Usability Scale* (SUS).

O SSQ é utilizado para medir os sintomas de cinetose (enjoo, tontura e desconforto ocular) durante a interação com ambientes virtuais ou simuladores. Ele ajuda a avaliar o impacto de sistemas de realidade virtual na saúde do usuário, identificando a intensidade desses sintomas em três áreas: náusea, desorientação e mal-estar ocular. A pontuação total do SSQ é calculada com base na soma dessas subescalas, ajudando a identificar se a experiência virtual está causando desconforto excessivo ao usuário. Enquanto o SUS avalia a usabilidade de sistemas interativos, como plataformas de treinamento. Composto por 10 afirmações, o SUS mede a percepção dos usuários sobre a facilidade de uso, eficiência e satisfação geral com a plataforma. As respostas são ajustadas e convertidas em uma pontuação final que varia de 0 a 100. O SUS é amplamente utilizado para identificar problemas na interface e aprimorar a experiência do usuário.

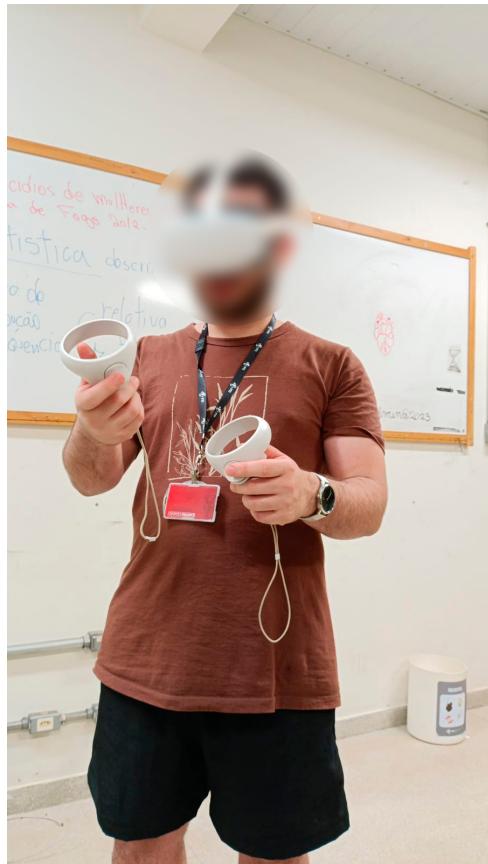
8.1.2 Metodologia de Avaliação

O experimento foi realizado com um total de 20 participantes (13 homens e 7 mulheres), com idades entre 20 e 40 anos. O estudo foi conduzido em dois dias distintos, sendo que 15 participantes participaram no primeiro dia e 5 participantes no segundo dia. O ambiente de teste foi uma sala espaçosa, fechada e bem iluminada, para garantir conforto e consistência durante a execução das tarefas de treinamento. Os critérios de inclusão foram:

- Pessoas com mais de 18 anos;
 - Ausência de limitações motoras ou visuais;
 - Ausência de doenças que pudessem causar desconforto ou inviabilizar o uso dos dispositivos HMD.

Os participantes foram recrutados em ambiente universitário e receberam uma explicação detalhada sobre o objetivo do estudo e o funcionamento da plataforma de metaverso, conforme visto na Figura 32. Após a explicação, tiveram 5 minutos para explorar livremente o ambiente virtual, identificando funcionalidades e se familiarizando com os controles de locomoção e interação. A fase de exploração inicial teve como objetivo reduzir o desconforto inicial e permitir que os participantes se adaptassem ao ambiente virtual. Essa familiarização foi realizada no cenário da GIS modelado, onde para o início das atividades, é necessário o usuário selecionar um tipo (*trainee*, ou *tutor*).

Figura 32 – Fotografia da realização do experimento.



Fonte: Autoria própria

8.1.3 Execução das Atividades

Após a fase de familiarização, os participantes foram instruídos a realizar duas atividades dentro da plataforma:

8.1.3.1 Atividade 1 (AT1): Movimentação e operação de uma transpaleteira

O participante deveria mover a transpaleteira até a posição correta utilizando um clique. Após o movimento, o participante pressionava um botão para que o gancho descesse e pegasse o cilindro.

8.1.3.2 Atividade 2 (AT2): Manutenção de um cilindro hidráulico

O participante deveria desmontar o cilindro, realizar a limpeza dos componentes internos e, em seguida, remontar o cilindro corretamente.

As atividades foram executadas uma única vez por cada participante.

8.1.4 Coleta de Dados

Para avaliar o sistema, foi conduzido um processo de análise baseado em coleta de dados quantitativos e qualitativos, por meio da aplicação de questionários específicos e da observação direta das sessões de treinamento. O processo de coleta de dados envolveu três etapas principais:

- Aplicação de um questionário de caracterização, com o objetivo de coletar informações gerais sobre os participantes, como idade, experiência prévia com tecnologias imersivas e estado de saúde geral;
 - Aplicação de um *Simulator Sickness Questionnaire* (SSQ) adaptado, para avaliar a suscetibilidade dos participantes a efeitos colaterais como enjoo, tontura e desconforto visual durante o uso da plataforma;
 - Aplicação de um *System Usability Scale* (SUS), para avaliar a percepção subjetiva dos participantes sobre a usabilidade do sistema;
 - Aplicação de um questionário pós-experimento, desenvolvido para obter informações qualitativas sobre a experiência dos usuários com o treinamento e identificar possíveis pontos de melhoria na plataforma.

Os dados obtidos foram tabulados e analisados de forma sistemática, permitindo identificar padrões de comportamento e percepção dos participantes em relação ao sistema. A análise dos resultados foi realizada com o apoio de ferramentas estatísticas, permitindo uma avaliação detalhada dos aspectos técnicos e subjetivos do sistema.

8.2 Análise dos Resultados do *Simulator Sickness Questionnaire* (SSQ)

O *Simulator Sickness Questionnaire* (SSQ) foi utilizado para avaliar os sintomas de desconforto experimentados pelos participantes após a execução das atividades de treinamento em realidade virtual. O SSQ é um método amplamente utilizado para medir a suscetibilidade à cinetose e os sintomas associados à exposição a ambientes virtuais, simuladores e dispositivos de realidade aumentada e virtual.

A análise dos dados foi realizada utilizando um código em Python, que calculou as pontuações de cada subescala (Náusea, Desorientação e Mal-estar Ocular) e a pontuação total do SSQ para cada participante. Os valores de cada sintoma foram convertidos em uma escala numérica, onde:

- 0: Nenhum sintoma (Nada)

- **1:** Sintoma leve (Fraco)
 - **2:** Sintoma moderado (Moderado)
 - **3:** Sintoma severo (Forte)

Os sintomas foram agrupados de acordo com as categorias definidas pelo modelo de Kennedy et al. [1993]: Náusea, Desorientação e Mal-estar Ocular. Os valores brutos foram multiplicados por fatores de normalização específicos para cada subescala, conforme as seguintes fórmulas 8.1, 8.2, 8.3 e 8.4:

$$\text{Náusea} = \sum_{n=1}^{\infty} (\text{Sintomas de Náusea}) \times 9.54 \quad (8.1)$$

$$\text{Desorientação} = \sum_{n=1}^{\infty} (\text{Sintomas de Desorientação}) \times 13.92 \quad (8.2)$$

$$\text{Mal-estar Ocular} = \sum_{n=1}^{\infty} (\text{Sintomas de Mal-estar Ocular}) \times 7.58 \quad (8.3)$$

$$\text{Total SSQ} = \sum_{n=1}^{\infty} (\text{Todos os Sintomas}) \times 3.74 \quad (8.4)$$

A análise foi conduzida em um total de 20 participantes, e os resultados foram compilados e organizados em uma tabela com a pontuação individual de cada participante para as três subescalas e o valor total do SSQ. A Tabela 2 apresenta os resultados obtidos.

Tabela 2 – Resultados individuais do SSQ

Participante	Náusea	Desorientação	Mal-estar Ocular	Total SSQ
1	9.54	69.60	7.58	26.18
2	19.08	97.44	7.58	37.40
3	0.00	0.00	0.00	0.00
4	19.08	41.76	30.32	33.66
5	0.00	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00
7	0.00	0.00	22.74	11.22
8	0.00	0.00	0.00	0.00
9	0.00	0.00	7.58	3.74
10	0.00	27.84	0.00	7.48
11	0.00	0.00	0.00	0.00
12	9.54	0.00	0.00	3.74
13	0.00	0.00	0.00	0.00
14	0.00	13.92	0.00	3.74
15	0.00	27.84	0.00	7.48
16	19.08	111.36	45.48	59.84
17	0.00	55.68	37.90	33.66
18	0.00	0.00	0.00	0.00
19	0.00	0.00	0.00	0.00

Fonte: Autoria própria

8.2.1 Estatísticas Gerais

A Tabela 3 apresenta as estatísticas descritivas (média, desvio padrão, mínimo e máximo) das pontuações obtidas para cada subescala e para o valor total do SSQ.

Tabela 3 – Estatísticas descritivas da análise do SSQ

Métrica	Náusea	Desorientação	Mal-estar Ocular	Total SSQ
Média	3.82	21.31	7.63	10.86
Desvio Padrão (DP)	6.74	35.05	11.63	15.31
Mínimo	0.00	0.00	0.00	0.00
Máximo	19.08	111.36	45.48	59.84

Fonte: Autoria própria

8.2.2 Discussão dos Resultados

Os resultados obtidos mostraram que a média para a pontuação de Náusea foi de 3.82, o que indica um nível de desconforto leve para a maioria dos participantes. A pontuação média para Desorientação foi de 21.31, o que sugere um desconforto moderado em termos de percepção de movimento e orientação espacial. Já a pontuação média para Mal-estar Ocular foi de 7.63, o que indica um desconforto leve relacionado à fadiga ocular e dificuldade de foco.

A pontuação total média foi de 10.86, o que sugere um desconforto leve durante a execução das tarefas de treinamento. A pontuação máxima de 59.84 foi observada em um único participante, o que indica um nível severo de desconforto para esse caso específico.

Os resultados indicam que, de forma geral, o sistema proporcionou uma experiência estável para a maioria dos participantes, com sintomas leves ou moderados. Os casos isolados de desconforto mais intenso podem estar relacionados a fatores individuais, como sensibilidade pessoal à realidade virtual ou duração da exposição.

8.3 Análise dos Resultados de Usabilidade (*System Usability Scale* - SUS)

A usabilidade da plataforma de treinamento em metaverso foi avaliada por meio do questionário *System Usability Scale*, uma ferramenta amplamente utilizada para medir a facilidade de uso e a eficiência de sistemas interativos. O SUS é composto por 10 afirmações, alternando entre afirmações positivas e negativas, que devem ser respondidas utilizando uma escala Likert de 1 a 5, onde:

- **1:** Discordo completamente;
 - **2:** Discordo;
 - **3:** Neutro;
 - **4:** Concordo;
 - **5:** Concordo completamente.

As afirmações positivas (itens 1, 3, 5, 7 e 9) têm sua pontuação ajustada subtraindo-se 1 ponto do valor informado pelo participante. As afirmações negativas (itens 2, 4, 6, 8 e 10) são invertidas, subtraindo-se o valor informado de 5. Essa conversão ajusta os valores para uma escala que vai de 0 a 4.

A pontuação final do SUS foi obtida somando todas as respostas ajustadas e multiplicando esse valor por 2.5, resultando em uma escala de 0 a 100, conforme a metodologia proposta por Brooke [1996]. Essa escala é comumente utilizada para avaliar a experiência geral do usuário e identificar aspectos críticos relacionados à usabilidade e eficiência do sistema.

8.3.1 Discussão dos Resultados

Os resultados obtidos estão apresentados na Tabela 4, contendo as estatísticas descritivas das pontuações do SUS.

Tabela 4 – Estatísticas descritivas da análise do SUS

Estatística	Valor
Média	68.21
Desvio Padrão (DP)	12.82
Mínimo	40.00
Mediana (50º percentil)	75.00
Máximo	87.50
1º Quartil (25%)	55.00
3º Quartil (75%)	77.50

Fonte: Autoria própria

A pontuação média do SUS foi de 68.21, o que sugere que a plataforma apresenta uma usabilidade considerada boa, segundo a escala de referência proposta por Brooke [1996]. O valor médio de 68 é amplamente reconhecido como a média global de sistemas de software, indicando que a plataforma oferece uma experiência de uso que atende aos padrões esperados.

A pontuação máxima foi de 87.5, indicando que alguns participantes consideraram a plataforma altamente eficiente e de fácil utilização. Essa faixa indica uma experiência de usabilidade excelente, segundo a escala de avaliação do SUS. Por outro lado, a pontuação mínima foi de 40, revelando que alguns participantes enfrentaram dificuldades durante a execução das atividades.

Observou-se que 50% dos participantes avaliaram a plataforma com uma pontuação superior a 75, reforçando que a experiência geral foi considerada positiva. No entanto, 25% dos participantes deram pontuações inferiores a 55, o que indica que parte do público teve dificuldades que podem estar relacionadas à navegação na interface, adaptação ao ambiente virtual ou execução das tarefas.

9 Conclusões e Trabalhos Futuros

Este trabalho foi desenvolvido em um contexto de avanço tecnológico e crescente interesse na utilização de plataformas de realidade virtual para treinamento e capacitação. A aplicação de tecnologias imersivas em treinamentos profissionais, apesar de já ter sido explorada em algumas áreas, ganhou relevância nos últimos anos devido ao desenvolvimento de dispositivos de alta qualidade e ao aprimoramento de ferramentas de desenvolvimento, tornando possível criar ambientes virtuais cada vez mais realistas e acessíveis.

Entretanto, a implementação de plataformas de treinamento em metaverso apresenta desafios técnicos significativos, especialmente no que diz respeito à sincronização de ações e interações entre múltiplos usuários em um ambiente compartilhado. A complexidade na execução de tarefas de forma simultânea e a necessidade de manter uma experiência imersiva e estável são barreiras que dificultam a criação de sistemas robustos e eficazes para treinamento em ambientes virtuais colaborativos.

Diante desse cenário, o presente trabalho propôs o desenvolvimento de uma plataforma de treinamento em metaverso, estruturada de maneira a permitir a interação entre *trainees* e tutores em tempo real, com sincronização de ações e estados dos objetos virtuais. O processo de implementação foi detalhado, incluindo a criação de funcionalidades específicas para manipulação de objetos, locomoção no ambiente virtual, sincronização de eventos e *feedback* visual em tempo real. A plataforma foi desenvolvida utilizando o motor gráfico Unity, com o suporte da biblioteca Photon PUN 2 para gerenciamento das interações *multiplayer* e sincronização de estado entre clientes.

Durante o desenvolvimento da plataforma, foram enfrentados desafios relacionados à sincronização de estados e ao comportamento dos objetos no ambiente virtual. Embora não tenha sido realizado uma análise de estabilidade sistemática, a observação durante o desenvolvimento revelou que métodos diferentes de sincronização apresentam tempos de resposta e consistência distintos, por isso foi optado por não executar testes sistemáticos de estabilidade. Dito isso, os resultados das observações indicaram que a manipulação direta de transformações de posição foi menos rápida e consistente em comparação com a execução por meio de animações, o que sugere que, para ações que exigem resposta rápida e onde há também um grande volume de usuários em cena, a sincronização por transformação não é adequada.

A avaliação da plataforma foi realizada por meio de um experimento com 20 participantes, que executaram duas tarefas distintas na plataforma: a movimentação de uma transpaleteira (por meio de animação) e a movimentação de um gancho (por meio de manipulação direta de transformações). Os dados coletados foram analisados com base em

três instrumentos principais: o *Simulator Sickness Questionnaire* (SSQ), o *System Usability Scale* (SUS) e um questionário pós-experimento (QPE). Os resultados mostraram que a plataforma apresentou uma pontuação média de 68.21 no SUS, indicando uma usabilidade considerada boa. A análise do SSQ revelou que os sintomas de desconforto foram geralmente baixos, com valores médios de 3.82 para náusea, 21.31 para desorientação e 7.63 para mal-estar ocular, indicando que a plataforma ofereceu uma experiência relativamente confortável para a maioria dos participantes.

As experiências adquiridas durante o desenvolvimento da plataforma evidenciaram que decisões relacionadas à implementação de técnicas de interação e manipulação de objetos devem considerar não apenas a precisão técnica, mas também a experiência subjetiva dos usuários.

O conhecimento adquirido é de grande valor para o desenvolvimento de futuras plataformas de treinamento em metaverso. A descrição detalhada dos processos de sincronização e manipulação de estados pode servir como referência para a implementação de novos sistemas em contextos industriais, educacionais e profissionais. O uso da Unity como plataforma de desenvolvimento proporciona flexibilidade para adaptações e ampliações, permitindo que as funcionalidades implementadas neste trabalho sejam expandidas para diferentes tipos de treinamento e contextos de aplicação.

9.1 Trabalhos Futuros

Embora os resultados obtidos tenham demonstrado boa aceitação da plataforma, algumas limitações foram identificadas e podem ser abordadas em trabalhos futuros para ampliar o potencial do sistema. Como visto na Seção 8.3, a pontuação mínima de 40 evidencia que alguns usuários encontraram obstáculos durante o uso da plataforma. A análise dos resultados sugere que esses problemas estão diretamente relacionados à experiência do usuário com a interface, à resposta do sistema durante a execução das tarefas e à clareza das instruções fornecidas no início do treinamento. Esses pontos de melhoria podem ser abordados de forma estruturada em futuros desenvolvimentos da plataforma. Entre as principais causas para a avaliação mais baixa na usabilidade estão:

- **Complexidade na navegação:** Alguns participantes relataram dificuldades em se locomover e executar tarefas específicas, o que pode estar relacionado à falta de clareza na interface ou ao posicionamento de elementos no ambiente virtual. Para resolver esse problema, trabalhos futuros podem focar em uma reformulação da interface, otimizando o posicionamento dos elementos e facilitando a navegação por meio de ajustes na configuração dos controles e nas opções de locomoção. A introdução de um sistema de teleporte mais intuitivo ou a utilização de guias visuais

para indicar a posição dos objetos pode simplificar a experiência de navegação para usuários menos experientes.

- **Ausência de *feedback* visual em tempo real:** A inclusão de indicadores visuais mais claros para confirmar ações pode melhorar significativamente a experiência do usuário. Em futuros desenvolvimentos, um sistema de *feedback* em tempo real pode ser implementado utilizando mudanças de cor, animações visuais ou efeitos sonoros para reforçar as ações bem-sucedidas. Além disso, o uso de *haptics* (*feedback* tátil) nos controles pode tornar a resposta das ações mais realista e satisfatória para o usuário.
- **Aprimoramento do tutorial inicial:** Embora tenha sido oferecida uma etapa de familiarização de 5 minutos, uma explicação mais detalhada com instruções visuais pode facilitar a adaptação de usuários com pouca ou nenhuma experiência prévia com realidade virtual. Trabalhos futuros podem explorar a criação de um tutorial interativo que permita que o usuário execute ações simples antes de iniciar o treinamento real, garantindo que todos os participantes estejam preparados para as etapas principais do treinamento. Esse tutorial poderia incluir uma simulação de ambiente controlado para que os usuários testem comandos básicos e se familiarizem com o sistema de controle antes de iniciar o treinamento propriamente dito.

Além da resolução dessas limitações, outras direções para trabalhos futuros incluem:

- **Ampliação das funcionalidades de interação:** A implementação de novos métodos de interação, como o uso de controle por gestos pode aumentar a naturalidade e a imersão durante o treinamento. Isso pode incluir a integração de dispositivos de rastreamento de mão (*hand tracking*) para permitir a manipulação direta de objetos sem o uso de controles físicos.
- **Avaliação sistemática de estabilidade e latência:** Embora a plataforma tenha apresentado um desempenho satisfatório em termos de resposta e sincronização, não foram realizados testes sistemáticos para avaliar a estabilidade e a latência do sistema de forma controlada. Trabalhos futuros podem explorar a realização de testes estruturados para medir com precisão o tempo de resposta entre as ações dos usuários e a sincronização de eventos em ambiente *multiplayer*. A análise detalhada da latência entre diferentes métodos de sincronização (como a atualização por transformações diretas e a execução por meio de animações) pode fornecer *insights* para otimizar o comportamento do sistema, especialmente em cenários com múltiplos jogadores e interações complexas. Essa abordagem permitirá identificar gargalos na comunicação de rede e melhorar a consistência das respostas do sistema.

- **Expansão para múltiplos cenários:** A criação de novos cenários de treinamento, incluindo ambientes externos e com maior complexidade, permitirá avaliar o desempenho da plataforma em contextos variados e com diferentes níveis de exigência. A simulação de cenários industriais, por exemplo, pode proporcionar uma validação mais abrangente das funcionalidades do sistema, além de permitir a adaptação para setores como saúde, construção civil e indústria energética.
 - **Aprimoramento na avaliação de desempenho:** A inclusão de métricas automáticas para avaliação de desempenho, como análise de tempo de execução, precisão nas tarefas e taxa de sucesso, permitirá uma análise mais detalhada sobre a efetividade do treinamento.

O desenvolvimento desta plataforma constituiu um avanço relevante na utilização de tecnologias de metaverso para treinamentos profissionais. A experiência adquirida com a implementação das técnicas de sincronização, controle de interação e manipulação de estado será de grande utilidade para futuras iniciativas nesse campo. Os resultados obtidos indicam que é viável criar ambientes de treinamento imersivos que proporcionam uma experiência de aprendizado realista e consistente.

Em um nível pessoal, a execução deste trabalho representou um desafio técnico e metodológico, demandando a integração de conhecimentos em programação, redes e realidade virtual. A superação dessas dificuldades resultou em um sistema funcional e aprovado pelos usuários, evidenciando o potencial das plataformas de metaverso para capacitação e treinamento em ambientes virtuais complexos. Espera-se que este trabalho sirva como base para a evolução de práticas de treinamento e inspire o desenvolvimento de novas soluções fundamentadas em tecnologias de realidade virtual e metaverso.

Referências

- ABBATE, G. et al. A mirror therapy system using virtual reality and an actuated exoskeleton for the recovery of hand motor impairments: a study of acceptability, usability, and embodiment. *Scientific Reports*, v. 13, n. 1, p. 22881, Dec 2023. ISSN 2045-2322. Disponível em: [⟨https://doi.org/10.1038/s41598-023-49571-7⟩](https://doi.org/10.1038/s41598-023-49571-7). Citado na página 12.
- ABULRUB, A. G.; ATTRIDGE, A. N.; WILLIAMS, M. A. Virtual reality in engineering education: The future of creative learning. *International Journal of Emerging Technologies in Learning*, v. 6, n. 4, p. 4–11, 2011. Disponível em: [⟨https://online-journals.org/index.php/i-jet/article/view/1766⟩](https://online-journals.org/index.php/i-jet/article/view/1766). Citado na página 2.
- ATASHGAH, M. A.; BAHRAMI, M.; NASIRI, N. Developing a flexible 6dof modeling and simulation environment for a sounding rocket. In: . [s.n.], 2005. Disponível em: [⟨https://www.researchgate.net/publication/279524437_Developing_a_Flexible_6DOF_Modeling_and_Simulation_Environment_for_a_Sounding_Rocket⟩](https://www.researchgate.net/publication/279524437_Developing_a_Flexible_6DOF_Modeling_and_Simulation_Environment_for_a_Sounding_Rocket). Citado na página 7.
- AZUMA, R. T. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments*, v. 6, n. 4, p. 355–385, 1997. Disponível em: [⟨https://direct.mit.edu/pvar/article/6/4/355/18336⟩](https://direct.mit.edu/pvar/article/6/4/355/18336). Citado 4 vezes nas páginas 1, 8, 23 e 24.
- BALLESTRIN, G.; CHESSA, M.; SOLARI, F. A registration framework for the comparison of video and optical see-through devices in interactive augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, v. 27, n. 11, p. 4273–4283, 2021. Disponível em: [⟨https://ieeexplore.ieee.org/document/9416430⟩](https://ieeexplore.ieee.org/document/9416430). Citado na página 30.
- BLENDER, F. *Blender Documentation*. [S.l.], 2022. Disponível em: [⟨https://docs.blender.org/manual/en/latest/⟩](https://docs.blender.org/manual/en/latest/). Citado na página 34.
- BROOKE, J. Sus: A quick and dirty usability scale. In: JORDAN, P. W. et al. (Ed.). *Usability Evaluation in Industry*. Taylor and Francis, 1996. p. 189–194. ISBN 9780748404605. Disponível em: [⟨https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale⟩](https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale). Citado 2 vezes nas páginas 75 e 76.
- CEASB. *Reunião de encerramento do projeto: Simulador da Revisão do Acionamento Hidráulico dos Disjuntores 500kV da GIS*. 2023. Não publicado. Citado 4 vezes nas páginas 41, 42, 43 e 44.
- DELGADO, M. D. et al. A research agenda for augmented and virtual reality in architecture, engineering and construction. *Advanced Engineering Informatics*, v. 45, p. 101122, 08 2020. Citado na página 15.
- DETERDING, S. et al. From game design elements to gamefulness: Defining gamification. In: *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. Tampere, Finland: Association for Computing Machinery, 2011. p. 9–15. ISBN 978-1-4503-0799-5. Disponível em: [⟨https://dl.acm.org/doi/10.1145/2181037.2181040⟩](https://dl.acm.org/doi/10.1145/2181037.2181040). Citado 4 vezes nas páginas 1, 16, 17 e 18.

- FREITAS, F. V. d.; GOMES, M. V. M.; WINKLER, I. Benefits and challenges of virtual-reality-based industrial usability testing and design reviews: A patents landscape and literature review. *Applied Sciences*, v. 12, n. 3, p. 1755, 2022. Disponível em: <https://www.mdpi.com/2076-3417/12/3/1755>. Citado 7 vezes nas páginas 2, 9, 11, 13, 23, 27 e 29.
- GITHUB, I. *GitHub Documentation*. [S.l.], 2021. Disponível em: <https://docs.github.com/>. Citado na página 34.
- GÓRSKI, F. Building virtual reality applications for engineering with knowledge-based approach. *Management and Production Engineering Review*, v. 8, n. 3, p. 37–45, 2017. Disponível em: <https://www.degruyter.com/document/doi/10.1515/mper-2017-0037/html>. Citado na página 2.
- GURIAN, G.; BOLISANI, E.; KIRCHNER, K. Virtual reality for supporting knowledge sharing: An exercise of technology assessment. In: *Proceedings of the 24th European Conference on Knowledge Management (ECKM 2023)*. Reading, UK: Academic Conferences International Limited, 2023. p. 399–407. Disponível em: <https://papers.academic-conferences.org/index.php/eckm/article/view/1507>. Citado 2 vezes nas páginas 2 e 27.
- HAYS, R. T.; SINGER, M. J. *Simulation Fidelity in Training System Design: Bridging the Gap between Reality and Training*. New York: Springer, 1989. ISBN 978-1-4757-6376-6. Disponível em: <https://link.springer.com/book/10.1007/978-1-4757-6374-2>. Citado 2 vezes nas páginas 5 e 6.
- Morton L. Heilig. *Sensorama Simulator*. 1962. US3050870A. Disponível em: <https://patents.google.com/patent/US3050870A>. Citado 2 vezes nas páginas 22 e 23.
- JARAMILLO-MEDIAVILLA, L. et al. Impact of gamification on motivation and academic performance: A systematic review. *Education Sciences*, v. 14, n. 6, p. 639, 2024. Disponível em: <https://www.mdpi.com/2227-7102/14/6/639>. Citado 3 vezes nas páginas 1, 17 e 18.
- JUDD, A. *What you need to know about Unity's XR Toolkit and its new input system*. 2020. Disponível em: <https://mavricresearch.com/2020/11/24/what-you-need-to-know-about-unitys-xr-toolkit-and-its-new-input-system/>. Citado na página 33.
- KENNEDY, R. S. et al. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The International Journal of Aviation Psychology*, Taylor & Francis, v. 3, n. 3, p. 203–220, 1993. Disponível em: https://www.tandfonline.com/doi/abs/10.1207/s15327108ijap0303_3. Citado na página 73.
- KREVELEN, R. V. *Augmented Reality: Technologies, Applications, and Limitations*. [S.l.], 2007. Disponível em: https://www.researchgate.net/publication/292150312_Augmented_Reality_Technologies_Applications_and_Limitations. Citado na página 8.
- LI, S.; CHEN, K. Augmented and virtual reality (AR/VR) for education and training in the AEC industry: A systematic review of research and applications. *Buildings*, v. 12, n. 10, p. 1529, 2022. Disponível em: <https://www.mdpi.com/2075-5309/12/10/1529>. Citado 7 vezes nas páginas 1, 11, 20, 21, 22, 24 e 29.

- LIU, X.-W. et al. Research on training effectiveness of professional maintenance personnel based on virtual reality and augmented reality technology. *Sustainability*, v. 14, n. 21, 2022. ISSN 2071-1050. Disponível em: <https://www.mdpi.com/2071-1050/14/21/14351>. Citado na página 10.
- MAKRANSKY, G.; PETERSEN, G. B. The cognitive affective model of immersive learning (CAMIL): A theoretical research-based model of learning in immersive virtual reality. *Educational Psychology Review*, v. 33, n. 3, p. 937–958, 2021. Disponível em: <https://link.springer.com/article/10.1007/s10648-020-09586-2>. Citado 4 vezes nas páginas 2, 11, 17 e 30.
- MILGRAM, P.; KISHINO, F. A taxonomy of mixed reality visual displays. *IEICE Transactions on Information and Systems*, E77-D, n. 12, p. 1321–1329, 1994. ISSN 0916-8532. Disponível em: https://www.researchgate.net/publication/220613316_A_Taxonomy_of_Mixed_Reality_Visual_Displays. Citado 5 vezes nas páginas 1, 7, 20, 24 e 26.
- MILLINGTON, I. *Artificial Intelligence for Games*. 3rd. ed. Boca Raton, FL: CRC Press, 2019. ISBN 978-1138580553. Citado na página 56.
- MYSTAKIDIS, S. Metaverse. *Encyclopedia*, v. 2, p. 486–497, 2022. Disponível em: <https://www.mdpi.com/2673-8392/2/1/31>. Citado 4 vezes nas páginas 25, 26, 27 e 28.
- PHOTON, E. *Photon Unity Networking Documentation*. [S.l.], 2023. Disponível em: <https://doc.photonengine.com/en-us/pun/current/getting-started/pun-intro>. Citado na página 33.
- RADIANTI, J. et al. A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda. *Computers & Education*, v. 147, p. 103778, 2020. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0360131519303276>. Citado 5 vezes nas páginas 1, 11, 26, 27 e 29.
- ROLFE, J. M.; STAPLES, K. J. *Flight Simulation*. Cambridge, UK: Cambridge University Press, 1986. ISBN 9780521303510. Disponível em: <https://www.cambridge.org/core/books/flight-simulation/5B0C75F275B60B22747F61E8D527A527>. Citado 4 vezes nas páginas 4, 5, 6 e 8.
- SAMPAIO, A. Z. et al. 3d and virtual reality models in civil engineering education: Construction, rehabilitation and maintenance. *Automation in Construction*, v. 19, n. 7, p. 819–828, 2010. Disponível em: <https://www.sciencedirect.com/science/article/pii/S092658051000083X>. Citado 3 vezes nas páginas 2, 13 e 17.
- SOMMERVILLE, I. *Software Engineering*. 9. ed. Boston, MA: Addison-Wesley, 2011. ISBN 978-0-1370-3529-7. Citado na página 39.
- SUTHERLAND, I. E. A head-mounted three dimensional display. In: *Proceedings of the Fall Joint Computer Conference, AFIPS*. [s.n.], 1968. v. 33, p. 757–764. Disponível em: <https://dl.acm.org/doi/10.1145/1476589.1476686>. Citado 3 vezes nas páginas 7, 19 e 23.
- TONG, P.; BIL, C. Effects of errors on transfer of aircraft flight training. *National Aerospace Laboratory NLR - 32nd European Rotorcraft Forum, ERF 2006*, v. 3,

- p. 1390–1411, 01 2007. Disponível em: https://www.researchgate.net/publication/288205920_Effects_of_errors_on_transfer_of_aircraft_flight_training. Citado na página 5.

TROELSEN, A.; JAPIKSE, P. Pro c 9 with .net 5, foundational principles and practices in programming. 01 2021. Citado na página 32.

UNITY, T. *Unity 6 Documentation*. [S.l.], 2023. Disponível em: <https://docs.unity3d.com/Manual/>. Citado na página 31.

UNITY, T. *XR Interaction Toolkit Documentation*. [S.l.], 2023. Disponível em: <https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.1/manual/index.html>. Citado na página 32.

WEINBERGER, M. What is metaverse?—a definition based on qualitative meta-synthesis. *Future Internet*, v. 14, n. 11, p. 310, 2022. Disponível em: <https://www.mdpi.com/1999-5903/14/11/310>. Citado 6 vezes nas páginas 2, 9, 25, 27, 28 e 30.

A System Usability Scale (SUS)

*Eu acho que gostaria de usar esse sistema com frequência **

- Discordo completamente
- Discordo
- Neutro
- Concordo
- Concordo completamente

*Eu acho o sistema desnecessariamente complexo. **

- Discordo completamente
- Discordo
- Neutro
- Concordo
- Concordo completamente

*Eu achei o sistema fácil de usar. **

- Discordo completamente
- Discordo
- Neutro
- Concordo
- Concordo completamente

*Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema. **

- Discordo completamente
- Discordo
- Neutro
- Concordo
- Concordo completamente

*Eu acho que as várias funções do sistema estão muito bem integradas. **

Discordo completamente
 Discordo
 Neutro
 Concordo
 Concordo completamente

*Eu acho que o sistema apresenta muita inconsistência. **

Discordo completamente
 Discordo
 Neutro
 Concordo
 Concordo completamente

*Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente. **

Discordo completamente
 Discordo
 Neutro
 Concordo
 Concordo completamente

*Eu achei o sistema atrapalhado de usar. **

Discordo completamente
 Discordo
 Neutro
 Concordo
 Concordo completamente

*Eu me senti confiante ao usar o sistema. **

- Discordo completamente
- Discordo
- Neutro
- Concordo
- Concordo completamente

*Eu precisei aprender várias coisas novas antes de conseguir usar o sistema. **

- Discordo completamente
- Discordo
- Neutro
- Concordo
- Concordo completamente

B Simulator Sickness Questionnaire (SSQ)

Após executar as tarefas de **operação do Transpalet e manutenção do cilindro** dentro do ambiente virtual do Metaverso, marque a intensidade do que você sentiu durante ou logo após a experiência.

	nada	fraco	moderado	forte
Mal estar generalizado	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Cansaço	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dor de cabeça	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Vista cansada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dificuldade de manter o foco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Aumento da salivação	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sudorese	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Nausea	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dificuldade de concentração	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Cabeça pesada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Visão embaçada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tontura com olhos abertos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tontura com olhos fechados	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Vertigem	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Desconforto abdominal	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Arroto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



MANIFESTO DE ASSINATURAS



Código de validação: M4CEZ-7BCGX-S32CB-9HVWL

Esse documento foi assinado pelos seguintes signatários nas datas indicadas (Fuso horário de Brasília):

- ✓ Adriano Peretti Da Silva Santos (CPF ***.868.149-**) em 01/04/2025 13:54 -
Assinado eletronicamente

Endereço IP	Geolocalização
177.73.101.200	Lat: -25,479324 Long: -54,560704 Precisão: 8 (metros)
Autenticação	adriano.santos@itaipuparquetec.org.br
Email verificado	
TJsqS80OmvKBr+TiE/tmJpxCLzNcHlKgH9p9UvOISMg=	
SHA-256	

- ✓ Fabiana Frata Furlan Peres (CPF ***.361.919-**) em 01/04/2025 13:58 -
Assinado eletronicamente

Endereço IP	Geolocalização
177.73.98.61	Não disponível
Autenticação	fabiana.peres@unioeste.br
Email verificado	
Jq+6SiTJ+Awp9JNwPsY/RDURf6P1hM792GMqeTPEgJo=	
SHA-256	

- ✓ Claudio Roberto Marquetto Maurício (CPF ***.616.369-**) em 01/04/2025 16:08 -
Assinado eletronicamente

Endereço IP	Geolocalização
131.255.85.70	Não disponível
Autenticação	claudio.mauricio@unioeste.br
Email verificado	
jjElzJgmzn0crF+WFecyB6AfJ5P9KnzcJd7pNjXFCj8=	
SHA-256	

Para verificar as assinaturas, acesse o link direto de validação deste documento:

<https://assinaturas.itaipuparquetec.org.br/validate/M4CEZ-7BCGX-S32CB-9HVWL>

Ou acesse a consulta de documentos assinados disponível no link abaixo e informe o código de validação:

<https://assinaturas.itaipuparquetec.org.br/validate>