

ARTIGO ORIGINAL

Uso de realidade virtual para simular procedimentos da medicina nuclear

Use of virtual reality to simulate nuclear medicine procedures

Joaquim Eugênio Mattos dos Santos^{1,2}, Yuri Lemos Fernandes^{1,2}, André Cotelli do Espírito Santo ^{1,2}, Antonio Carlos de Abreu Mol ^{1,2}, Mônica Aquino de Assis², Paulo Francisco de Oliveira^{1,2}

¹Centro Universitário Carioca, ²Instituto De Engenharia Nuclear

*joaquim.eugenio.mattos@gmail.com; uri.lemos@gmail.com; cotelli.andre@gmail.com; mol@ien.gov.br; monica.assis@quintador.com.br; paulo_sisc@hotmail.com

Recebido: 09/12/2020. Revisado: 14/09/2021. Aceito: 04/05/2022.

Resumo

Uma vez que o acesso às tecnologias digitais vem aprimorando as formas como a sociedade realiza suas funções desde as mais simples até as mais complexas, é esperado também que esse avanço auxilie nas tarefas operacionais de áreas profissionais. Nesse sentido o uso da Realidade Virtual que também está em constante evolução no mercado mundial, pode se tornar uma ferramenta poderosa de simulação e treinamento para aplicações em áreas como medicina, engenharia e arquitetura e não apenas como entretenimento onde é comumente utilizada. Considerando essa perspectiva, a área de Medicina Nuclear por se tratar de uma parte integral e essencial no serviço de procedimentos radiodiagnósticos e radioterapêuticos, exige um treinamento muito sucinto por se tratar de uma atividade que inclui a manipulação de elementos que contém radiação ionizante, que pode oferecer sérios riscos à saúde. Portanto o presente trabalho tem como objetivo desenvolver uma ferramenta interativa digital baseada em Realidade Virtual para simular os procedimentos operacionais de um laboratório de radiofarmácia. Para este fim, será utilizado o software Unity 3D juntamente com um capacete de realidade virtual, para desenvolvimento do ambiente virtual e interações, contando com uma similaridade à laboratórios reais de radiofarmácia, incluindo todos os principais equipamentos. A proposta visa aprimorar e auxiliar o treinamento de profissionais da medicina nuclear reduzindo potencialmente os custos e periculosidade de uma prática em ambiente real.

Palavras-Chave: Realidade Virtual; Unity 3D; Radiofarmácia; Medicina Nuclear.

Abstract

Since access to digital technologies has been improving the ways in which society performs its functions from the simplest to the most complex, it is also expected that this advance will assist in the operational tasks of professional areas. In this sense, the use of Virtual Reality, which is also constantly evolving in the world market, can become a powerful simulation and training tool for applications in areas such as medicine, engineering and architecture and not just as entertainment where it is commonly used. Considering this perspective, the Nuclear Medicine area, as it is an integral and essential part in the service of radio diagnostic and radiotherapeutic procedures, requires very succinct training because it is an activity that includes the manipulation of elements that contain ionizing radiation, which can offer serious health risks. Therefore, the project aims to develop a digital interactive tool based on Virtual Reality to simulate the operational procedures of a radiopharmacy laboratory. For this purpose, Unity 3D software will be used to develop the virtual environment and interactions, with a similarity to real radio pharmacy laboratories, including all the main equipment contained in these sites. The proposal aims to improve and assist the training of nuclear medicine professionals, potentially reducing the costs and dangerousness of a practice in a real environment.

Keywords: Virtual Reality; Unity 3D; RadioPharmacy.

1 Introdução

O advento de tecnologias digitais nos mais diversos meios modificou a forma como o ser humano se comunica, se relaciona, trabalha e compreende a sociedade. Considerando esse cenário, os procedimentos de várias naturezas também foram gradualmente se modificando, cedendo espaço para as tecnologias e ferramentas que possam auxiliar diretamente no cumprimento de tarefas distintas. Como exemplo temos as ferramentas gráficas que são utilizadas mundialmente para simular ambientes, otimizar fotografias, criar conceitos de produto e também para entretenimento em filmes com efeitos especiais e jogos digitais.

A tecnologia de Realidade Virtual já é aplicada nas áreas de engenharia, arquitetura, medicina, entre outras. Conectando sobre essa tecnologia pode ser ainda mais útil, o estímulo à produção de simuladores mostra-se como uma solução, principalmente para áreas nas quais o treinamento em ambiente real seria considerado custoso e potencialmente perigoso.

Portanto a área de medicina nuclear que é de essencial importância para procedimentos radiodiagnósticos e radioterapêuticos exige uma profunda qualificação por se tratar de uma atividade considerada perigosa no que se refere à proteção física dos trabalhadores bem como do público em geral.

No contexto dos procedimentos operacionais da medicina nuclear, temos processos relacionados ao fluxo de elementos radioativos, atividades como manipulação, controle de qualidade, entre outras, exigem contato direto com tais substâncias. Além disso é descrito nas normas como resolução oficial que é dever do profissional farmacêutico a responsabilidade sobre todo o processo que envolve o radiofármaco. Considerando essas obrigações, em um desses processos pode ocorrer falhas ou erros que causariam ocasionais exposições à radiação ionizante, o que é considerado um acidente grave. Portanto é de suma importância que haja um treinamento por parte desses funcionários, além de simulações para que a capacitação e maior aprendizagem dos profissionais que atuarão na área de medicina nuclear a fim de evitar ao máximo que qualquer tipo de acidente possa ocorrer em um ambiente que exige tanto cuidado.

Visando essa problemática, e considerando todo esse cenário de inovação que a Realidade Virtual propõe, a ideia principal do trabalho é a construção e desenvolvimento de uma ferramenta de treinamento baseada em Realidade Virtual, em apoio à dissertação de mestrado da aluna Mônica Aquino de Assis, do IEN (Instituto de Engenharia Nuclear), com o objetivo de auxiliar os profissionais da área de medicina nuclear, de forma que se obtenha um ambiente virtual seguro e passível de uma simulação dos procedimentos operacionais considerando as normas e regulamentos referentes ao serviço de medicina nuclear publicados pela ANVISA, CNEN e recomendações internacionais do IAEA.

2 Trabalhos Relacionados

A tecnologia de RV na área nuclear já vem sendo utilizada amplamente em vários projetos e teses que exploram a possibilidade e eficácia de tal procedimento. A principal

característica que se faz presente nas vantagens da utilização da RV na área nuclear é a possibilidade de realizar treinamentos e simulações de ambientes e situações que são potencialmente perigosos e caros, sem que haja nenhum tipo de risco relacionado à exposição à radiação e com o custo expressivamente reduzido. Como exposto por Marins (2018) os modelos virtuais de instalações nucleares podem servir como boas ferramentas para medidas de proteção física relacionadas à manipulação de fontes radioativas entre outros aspectos de suma importância relacionados a esses procedimentos.

O uso de uma ferramenta simuladora com realidade virtual torna o processo de treinamento facilmente transportável e modificável, e resulta em profissionais que desempenham as funções ainda melhor do que os que foram treinados no sistema real e convencional, o que foi amplamente reconhecido durante o desenvolvimento do trabalho. (Winttenberg, 1995).

Considerando especificamente o aspecto do uso da RV na área nuclear o Instituto de Engenharia Nuclear (IEN) através do laboratório de realidade virtual (LABRV) já elaborou diversos projetos com o apoio de outras instituições de pesquisa e ensino (CDTN/CNEN, COPPE NUCLEAR/UFRJ, COPPE CIVIL/UFRJ, CRCN/UFPE/CNEN, UGF, UNICARIOCA), que mostraram o potencial e eficácia desses métodos. Dentro os projetos realizados podemos destacar:

2.1 Realidade Virtual Aplicada à Proteção Física de Instalações Nucleares

O sistema de Marins (2018) tem como objetivo auxiliar nas estratégias de proteção física em uma instalação nuclear de forma a evitar ataques criminosos de quaisquer naturezas envolvendo o uso de materiais radioativos presentes nesses ambientes. O projeto usa um sistema de RV, portanto um simulador computacional, no qual agentes de segurança podem traçar estratégias que evitem a invasão por parte de criminosos virtuais em um ambiente que recria de maneira realista toda a estrutura física do IEN (Fig. 1).

O sistema utiliza uma rede de forma que vários usuários possam interagir, separando um grupo de agentes de segurança e um grupo de criminosos, um com o objetivo de conter o outro no que tange às tentativas de invasão de ambiente controlado e furto de materiais radioativos.



Figura 1: Simulador de Proteção Física

2.2 Uso da Realidade Virtual no Planejamento de Repositórios de Rejeitos Radioativos

No projeto de Chelles (2017), houve o desenvolvimento de um ambiente usando RV como explorador de um modelo virtual de um repositório de rejeitos radioativos de baixo e médio níveis previamente projetado. Contando também com um vídeo informativo para seus usuários, virtualização do terreno que seria utilizado para a construção além da interação dos usuários com os ambientes propostos no repositório. Os usuários podem explorar livremente as instalações virtuais (Fig. 2), e averiguar a disposição dos principais objetos e edificações necessárias e previstas para construção real do ambiente proposto.



Figura 2: Simulador dos Repositórios de Rejeito

2.3 Realidade Virtual no ensino de procedimentos radiológicos

No estudo de O'Connor et al. (2020), alunos de primeiro ano de graduação em Radiologia foram submetidos ao uso de um simulador em Realidade Virtual para treinamento e aprendizado de procedimentos radiológicos em pacientes virtuais. O objetivo principal do estudo era a substituição de parte do treinamento em campo pelo treinamento virtual pela facilidade e baixo custo em comparação com métodos tradicionais. Os alunos tiveram a oportunidade de aplicar os conceitos teóricos na prática no simulador, com feedback em tempo real da completude ou não dos objetivos dos exames promovidos virtualmente. Além dos testes com alunos, todos foram submetidos a um questionário para avaliar a eficácia do uso de Realidade Virtual no ensino dos procedimentos radiológicos (Fig. 3). O resultado da pesquisa aponta uma porcentagem de 58% de aprovação do uso da tecnologia de acordo com o gráfico apresentado.

2.4 Uso de Realidade Virtual para simular níveis de radiação em diferentes cenários de isolamento

O trabalho de Hagita et al. (2020) utiliza uma aplicação de Realidade Virtual para simular o efeito de barreiras de

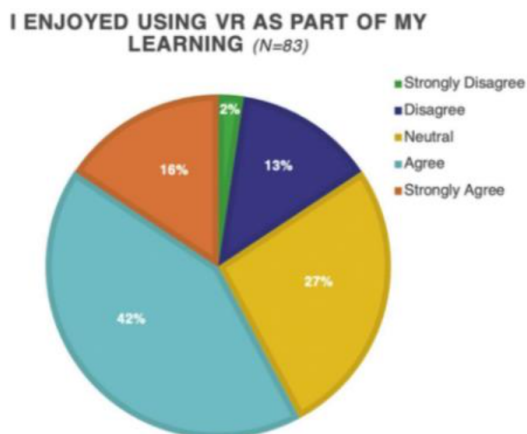


Figura 3: Gráfico de aprovação do uso de RV no ensino de procedimentos radiológicos

ferro e concreto para redução dos níveis de radioatividade emitidos por barris de material potencialmente radioativo (Fig. 4). O principal ponto da proposta é promover um treinamento inicial de proteção radiológica, permitindo a manipulação de elementos virtuais para produzir barreiras. O sistema conta com uma espécie de contador Geiger virtual para apresentar em tempo real o nível de radioatividade emitido pelas fontes e como a disposição das barreiras virtuais pode contribuir para diminuição desse valor. O artigo reflete a importância do uso desse tipo de ferramenta para treinamento mesmo que em situações mais simples e endereçadas a iniciantes da área de engenharia nuclear. Ao analisar todas essas ferramentas e as contribuições benéficas que as mesmas trouxeram ao âmbito da pesquisa na área nuclear, a utilização do RV continua sendo um bom método para treinamento e simulação no aspecto do objetivo do trabalho.

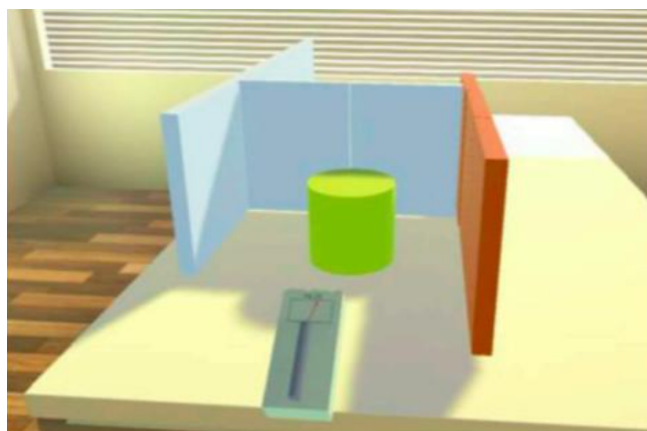


Figura 4: Simulador de barreiras para fontes radioativas

3 Fundamentação Teórica

Neste referido capítulo, aborda-se os principais conceitos utilizados no desenvolvimento da aplicação, a motivo de elucidar questões sobre estruturação teórica e embasamento do processo de construção das principais características do trabalho.

3.1 Serious Games

Serious Games é o termo utilizado para se referir à aplicações digitais interativas que usam as mesmas técnicas utilizadas em jogos para entretenimento, porém com objetivo técnico e importante em âmbito profissional. Segundo Zyda (2007), os serious games podem promover o entretenimento imersivo para treinamento corporativo ou governamental, estratégias de comunicação, saúde e educação. Jogadores podem ser motivados a jogar para adquirir conhecimento sobre conceitos, processos e estratégias específicas como mencionado por Melo and Adamatti (2020). O desenvolvimento de um serious game exige a participação de profissionais da área proposta, para que os objetivos do uso da aplicação sejam alcançados seguindo orientações técnicas.

3.2 Realidade Virtual

A Realidade Virtual, também conhecida como RV, é uma técnica de interface que tem como objetivo aprimorar a interação entre o homem e um sistema no sentido de conseguir uma maior sensação de imersão do usuário nos ambientes virtuais. Segundo Burdea (2003), uma das chaves principais para a RV é a resposta do mundo virtual às interações do usuário através de gestos, voz entre outros estímulos que enaltecem ainda mais a sensação de imersão que esse tipo de tecnologia provém. O uso de dispositivos como os capacetes de Realidade Virtual auxilia nessas interações e principalmente na estereoscopia, que é o uso de imagens 360° e tridimensionais, além de transmitir a localização do usuário para o ambiente tridimensional virtual de forma que o mesmo se sinta inserido naquele contexto e possa interagir diretamente com os elementos expostos virtualmente. Para o uso de RV nos serious games é necessário o aparato desses equipamentos específicos, que reforcem o objetivo proposto, com tecnologias de estereoscopia, sensores de movimento e localização virtual baseada na real, de forma que o usuário consiga imediatamente fazer a correlação entre a experiência e suas funções no dia a dia como profissional.

3.2.1 Capacetes de Realidade Virtual

Para utilização da tecnologia de RV com todo seu potencial faz-se necessário o uso de equipamentos específicos conhecidos como capacetes de RV. Essas ferramentas foram sendo aprimoradas com o avanço da tecnologia, e finalmente em 2016 os modelos de capacetes para consumidor desenvolvidos por diferentes empresas tornam a experiência acessível ao público geral (Tsai, 2016). Porém antes da popularização dos capacetes houveram várias experiências sensoriais que já podem ser consideradas como precursoras desta tecnologia de imersão. Como destaques podemos

citar:

- **Sensorama:** Criado nos anos 50 e patenteado nos anos 60, por Morton Heiling, era uma experiência visual que exibia um filme, contando com visão 3D, som binaural, além de ventos e vibrações para tornar a sensação de imersão ainda maior. A ferramenta se assemelhava a uma cabine, onde o espectador posicionava apenas a cabeça em um espaço específico (Fig. 5). Apesar de todas essas funções, não havia nenhum tipo de interação (Mazuryk, 1996).



Figura 5: Sensorama Fonte: Virtual Speech

- **Telesphere Mask:** Ainda nos anos 60, Morton Heiling criou o que podemos considerar como o primeiro capacete de RV, conhecido como Telesphere Mask. O dispositivo era utilizado diretamente na cabeça, como um óculos (Fig. 6), e exibia filmes em visão 3D estereoscópica e som estéreo, porém não contava com nenhum tipo de sensor de movimento (Virtual Reality Society, 2020). Hoje em dia com o avanço proeminente das tecnologias de placas gráficas, sensores, telas, entre outros componentes, a tecnologia de RV evoluiu e se tornou popular com o lançamento de outros dispositivos citados abaixo.

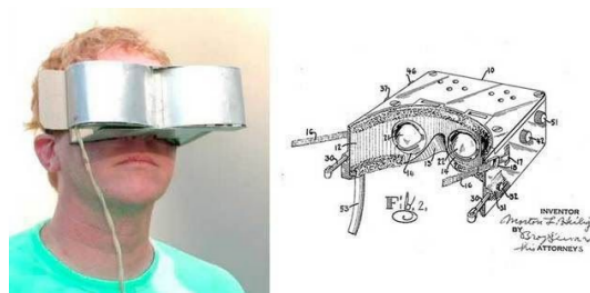


Figura 6: Telesphere Mask. Fonte: Research Gate

- **Oculus Rift:** Inicialmente desenvolvido pela Oculus RV, que fabricou protótipos conhecidos como Developers Kits (DK), foi comprada então pelo Facebook em 2014, e em 2016 foi lançada a versão final destinada à usuários, conhecida como Oculus Rift CV1 (Fig. 7). Esse dispositi-

tivo já conta com sensores de movimento, fones de ouvido estéreo, óculos com uma tela OLED de 1080x1200 pixels de resolução, para cada olho e 2 joysticks para manipulação de elementos virtuais (VR BOUND, 2020).



Figura 7: Oculus Rift. Fonte: Oculus

- HTC Vive: Lançado também no ano de 2016 (Fig. 8). Esse dispositivo compete diretamente com o Oculus Rift como alternativa para o uso de RV, porém por falta de sensores externos como os presentes no Rift, este capacete tem a sua área de jogabilidade menor, sendo indicado ao usuário que fique sentado durante toda a sua experiência o que o torna menos popular que seu competidor (VIVE Developers, 2020).



Figura 8: HTC Vive Fonte: HTC

3.3 Ferramentas

Para a elaboração de um projeto é necessário o levantamento das ferramentas mais adequadas para o seu desenvolvimento. Pois cada uma delas terá uma função específica e de suma importância para o projeto.

O objetivo deste tópico é apresentar as ferramentas usadas durante a construção do trabalho. De forma a melhorar o entendimento, as ferramentas foram organizadas de acordo com a sua importância para o projeto.

3.3.1 Núcleos de Jogos

Conhecidos pelo termo “game engines”, os núcleos de jogos ou motores gráficos em tradução livre, são softwares que tem como objetivo o desenvolvimento de games ou aplicações interativas que usem uma estrutura baseada em jogos digitais. Essas ferramentas são o principal meio para obtenção de projetos de gamificação por sua ampla capacidade de reconhecimento de formatos de arquivos multimídia, algumas configurações pré estruturadas para jogos, além da construção dos projetos para várias plataformas, principalmente computadores e consoles. As engines possuem módulos para lidar com entrada e saída de dados, renderização 3D e 2D e uma física genérica para utilização no mundo do jogo (Jacobson and Lewis, 2002).

3.3.2 Unity3D

A Unity3D é um motor criado pela Unity Technologies para desenvolvimento de jogos onde é possível desenvolver aplicações em 2D ou 3D, dando suporte à diferentes tipos de APIs: Direct3D no Windows e Xbox 360, OpenGL no MacOS e Linux, OpenGL ES no Android e iOS e WebGL na Internet, permitindo assim, o build para diversas plataformas (Unity3D, 2020a).

Para auxiliar no desenvolvimento, o motor dá suporte à importação de diversos formatos de arquivos, tais como: Autodesk (.FBX), 3D Studio Max (.MAX), Blender (.BLEND) e Wavefront (.OBJ) para modelos 3D; JPEG, PNG, GIF e Photoshop (.PSD e .TIFF) para imagens; MP3, OGG e WAV para arquivos de áudio e MP4, MOV, AVI, ASF, MPG e MPEG para arquivos de vídeo. A Unity 3D oferece ferramentas completas que facilitam o uso de códigos para alcançar os objetivos do desenvolvedor que são parte de sua API, contando com rotinas, estruturas de dados e classes utilizadas diretamente como bibliotecas nos scripts (Okita, 2014). Para a criação de experiências em RV o Unity3D dispõe de ferramentas e compatibilidade com API's das principais fabricantes de capacetes de RV do mercado. Algumas ferramentas como pipeline gráfico de alto desempenho, sistemas de partículas e áudio espacial se destacam como principais atrativos para desenvolvimento de uma boa experiência utilizando a engine (Unity3D, 2020b). Considerando a produção de uma aplicação em RV utilizando Unity, a mesma pode objetivar várias plataformas e capacetes, entre eles podemos citar: Google Cardboard, Oculus Rift, HTC Vive, entre outros.

3.3.3 Visual Studio

O Visual Studio é um ambiente de desenvolvimento integrado, conhecidos como IDE's, criado pela Microsoft, que facilita o desenvolvimento de aplicações para meios e plataformas diversos através de uma interface própria intuitiva e destinada à uma boa prática de programação. É um ferramenta multilinguagem, portanto permite que o código possa ser feito em diferentes linguagens de programação. As principais linguagens suportadas pelo Visual Studio são Visual Basic, C, C# e C++. Segundo Johnson (2012), o Visual Studio provém ferramentas e utilidades focadas em aumentar a produtividade dos desenvolvedores e isso o destaca em meio a outras IDE's disponíveis no mercado. Os projetos desenvolvidos neste software são ainda mais abrangentes se levada em consideração a vasta

aplicação de seus produtos em desenvolvimentos voltados à páginas web utilizando ASP.NET, e contando com o .NET Framework, ferramenta focada principalmente em redação de um código único passível de ser executado em qualquer plataforma que use o Framework, já presente nos sistemas Windows, MacOS e Linux.

As funções de depuração, autocomplete de código e interface visual amigável contam como pontos positivos na escolha dessa IDE para grandes projetos. O Visual Studio foi integrado como editor de scripts principal para desenvolvimento com o Unity 3D em 2018, na versão 2018.1, trazendo consigo todas as funcionalidades supracitadas à produção de jogos digitais na plataforma (Microsoft, 2019).

3.3.4 3Ds Studio Max

O 3Ds Max é um programa desenvolvido pela Autodesk para modelagem 3D. Com ele é possível adicionar formas já pré-estabelecidas ou criar novas a partir de ferramentas específicas, podendo alterar a largura, altura e profundidade do próprio objeto ou de sua localização dentro do ambiente. Seu formato de exportação padrão é .3DS, podendo ser exportado para outros formatos universais como o .FBX. Segundo Matossian (2001), o 3Ds Max é um dos programas de design 3D mais poderosos e conhecidos da atualidade com a possibilidade de ser utilizado em aplicações com diversas finalidades sejam comerciais, artísticas, arquiteturais, filmes, jogos e muitas outras (AUTODESK, 2020).

3.3.5 Adobe Photoshop

O Photoshop é uma ferramenta criada pela Adobe para edição e manipulação de imagens. Com seus recursos, é possível selecionar, cortar, duplicar ou excluir partes específicas, podendo substituí-las pela própria textura de fundo. Também é possível alterar o brilho, saturação, entre outras características de luz, alterar a forma de um objeto, corrigir deformidades, fazer sobreposições, criar textos personalizados, redimensionar, entre outras possibilidades. Os principais formatos que o programa suporta são BMP, JPEG, PNG e GIF. Um projeto é normalmente salvo em PSD e exportado para os formatos já conhecidos, além de TIFF, PDF, EPS, entre outros. Considerado como uma referência na geração de imagens pelo seu grande poder, precisão e controle, o Photoshop amplia os limites da edição de imagens digitais (ADOBE, 2009).

4 Métodos

Neste capítulo estão contidos os principais conceitos utilizados para obtenção de resultado do projeto, além da explicação detalhada dos processos executados e suas respectivas particularidades e adições benéficas para o desenvolvimento da ferramenta como foco da pesquisa do projeto.

4.1 Requisitos e Conceitos

Como foco principal para esse projeto, temos a construção de uma ferramenta que pode simular de forma realista a experiência dentro de um laboratório de medicina nuclear servindo como um ponto de treinamento dos profissionais

que exercem funções diretamente relacionadas com esta parte da medicina.

O trabalho é definido como o desenvolvimento do que podemos chamar de Jogo Digital, mais especificamente um “Serious Game” utilizando a ferramenta Unity 3D como principal elemento para elaboração das funções e integração dos elementos gráficos e lógicos necessários para cumprimento dos requisitos.

Além do uso de computadores comuns para execução do projeto, um dos diferenciais é o possível uso de Realidade Virtual como ferramenta de imersão, nesse caso contribuindo para a experiência geral do usuário. Nesse projeto, o dispositivo utilizado para alcançar tal objetivo foi o Oculus Rift.

4.1.1 Estudo e Conhecimento de Normas Oficiais

Para realização de uma boa experiência técnica relacionada às atividades realizadas por profissionais da medicina nuclear, foi necessário um estudo mais aprofundado sobre as normas oficiais e a legislação vigente no tocante à radiação ionizante, para que haja uma proteção aos profissionais e pacientes. Tais normas e legislações são regidas pela Comissão Nacional de Energia Nuclear (CNEN), porém, quando o processo é relacionado à medicina nuclear, que está na área da saúde, também é necessário atender às exigências, legislação e normas da Agência Nacional de Vigilância Sanitária (ANVISA). Quando consultadas as normas e legislações necessárias, é visível que a medicina nuclear exige ambientes especiais e equipamentos específicos para garantir a proteção contra a radiação, o que é observado em diversas resoluções como as citadas a seguir:

- RDC ANVISA Nº 50: Regulamento técnico para planejamento, programação, elaboração e avaliação de projetos físicos de estabelecimentos assistenciais de saúde (ANVISA, 2004);
- NORMA CNEN NN 6.01:: Requisitos para o registro de pessoas físicas para o preparo, uso e manuseio de fontes radioativas (CNEN, 1999);

4.1.2 Obtenção de Referências Visuais

Além de estudar e obter todas as informações relevantes sobre as normas de manipulação de radiofármacos, para um resultado ainda mais fiel à realidade dos laboratórios de medicina nuclear, uma das principais fontes de referências foram fotografias tiradas no laboratório de radiofarmácia do Hospital Universitário da UFRJ. Com o auxílio de profissionais da área, foram tiradas fotos de todos os principais componentes do laboratório. Também foi reconhecido o fluxo que um radiofármaco percorre em ambiente hospitalar, desde sua criação até a injeção no paciente.

4.1.3 Roteiro da Aplicação

Para objetivar e ilustrar o fluxo do radiofármaco dentro do ambiente hospitalar, foi elaborado um roteiro (Fig. 9), passando pelos pontos principais de manipulação da substância até a sua injeção no paciente. Ao iniciar a experiência, o usuário será levado passo a passo através de uma navegação automatizada que segue o roteiro na ordem estabelecida pelos profissionais de medicina nuclear envolvidos na produção do projeto.

Em cada ponto destacado nessa etapa da navegação, há

uma breve explicação do uso do equipamento exposto. Tal explicação está disponível via texto na tela, e também via áudio, narrado para o jogador durante o percurso. Os objetos destacados contam com um indicador virtual com símbolo da letra “I”, que representa um ponto de informação presente no roteiro (Fig. 10). Durante esse percurso, o usuário está privado de se movimentar pelo cenário virtual de forma manual, pois a própria aplicação o “guia” automaticamente aos pontos destacados, porém toda a visualização em estereoscopia está disponível para dar destaque à sensação de imersão. Portanto, é como se o usuário, usualmente sentado em uma cadeira durante a experiência, tenha a sensação de estar sendo levado fisicamente para conhecer todo o laboratório virtual.



Figura 9: Infográfico do roteiro

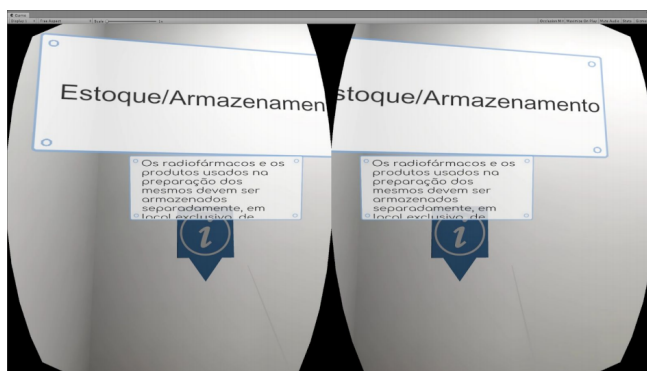


Figura 10: Informação sobre Estoque e Armazenamento

4.2 Desenvolvimento da Aplicação

O simulador da Radiofarmácia é uma aplicação virtual, considerada um Serious Game para ser usada como ferrame

nta de treinamento para profissionais da área de medicina nuclear no sentido de elucidar os principais conceitos operacionais necessários para uma boa prática e conhecimento dos elementos presentes em um laboratório de radiofarmácia. O software pode ser usado de duas formas: através de um computador e monitor convencional, no qual a aplicação exibe o ambiente de uma radiofarmácia com gráficos 3D de alta qualidade visual e com visão em primeira pessoa, conhecida como FPV, ou com o uso de capacetes de RV, que utilizam estereoscopia, sensores de movimento e outros recursos que servem para aumentar a sensação de imersão do usuário na experiência. O game contém etapas diferentes, estruturadas de forma sequencial a fim de seguir um parâmetro didático elucidativo para todos os usuários. Inicialmente há o menu principal, seguido de um modo guiado com uma navegação automatizada e por último uma experiência de utilização livre que conta com a manipulação dos elementos e objetos virtuais.

4.2.1 Menu Principal

A aplicação conta com um menu inicial, sem estar em estereoscopia, para a escolha de língua desejada, tendo disponível Português e Inglês, além de contar com os botões “iniciar” e “créditos”. A utilização do menu tem como principal objetivo anteceder a experiência ao que vai ser executado a seguir, além de disponibilizar o tempo necessário para o usuário preparar o equipamento de RV antes de iniciar a estereoscopia. O menu conta com um logotipo desenhado exclusivamente para o projeto que faz alusão ao uso de pílulas de radiofármacos além da utilização de óculos de Realidade Virtual, utilizando tons de cores fracos remetendo às usadas em ambientes hospitalares.

4.2.2 Navegação Automatizada

A navegação automatizada consiste na automatização do movimento do usuário dentro do ambiente virtual na primeira parte da aplicação. Considerando o roteiro traçado anteriormente, onde a sua ordem é de extrema importância para a compreensão dos elementos da radiofarmácia e o fluxo dos radiofármacos no laboratório, o usuário é inicialmente guiado por todo esse percurso de forma automatizada, como se um guia real estivesse mostrando todo o ambiente para o mesmo.

Para que isso fosse feito de forma automatizada, foi utilizado o componente conhecido como “Animation” do Unity 3D, próprio para elaboração de animações que envolvam os objetos de jogos presentes em cena. Dessa forma, é possível compor o movimento que o personagem principal fará em cada um desses pontos, e o caminho que vai ser traçado para ir de um ponto a outro. O Animation utiliza sistemas de “keyframes”, que são chaves ou pontos em uma linha do tempo que definem a posição do objeto a ser alterado na cena do jogo (Fig. 11). Assim, é possível marcar diferentes keyframes que representam, nesse caso, posições diferentes e fazem uma transição suave entre um e outro, de forma a simular o caminhar do personagem.

A partir da criação de todas as animações de transição entre os pontos, as mesmas animações foram invertidas para que o usuário possa voltar de um ponto a outro sem que isso altere a sequência previamente estipulada. O sistema funciona como o percorrer de estações, sendo possível avançar ou retroceder caso seja necessário revisitar



Figura 11: Linha do tempo da animação

algum elemento já previamente exposto e explicado. Isso facilita o aspecto didático da aplicação. Dessa forma, foi elaborado um código de controle para o avanço e retrocesso dessas estações virtuais (Fig. 12). Durante o período da navegação automatizada, os botões “P” e “O” do teclado conectado ao computador podem ser utilizados para avançar ou retroceder respectivamente. Além desses comandos, os joysticks usados pelo dispositivo de RV Oculus Rift também contém botões dedicados à essas funções, sendo eles “RT” para avançar e “LT” para retroceder (Fig. 13).

```

32 {
33     if (Input.GetKeyDown(KeyCode.P) || OVRInput.GetDown(OVRInput.Button.SecondaryIndexTrigger))
34     {
35         if (ControleLocalIda <= 16)
36         {
37             tocando = true;
38             gameObject.GetComponent<Animation>().clip = anims_ida[ControleLocalIda];
39             gameObject.GetComponent<Animation>().Play();
40             ControleLocalIda++;
41             if (ControleLocalIda > 1)
42                 ControleLocalVolta++;
43         }
44         print("Controle local Ida:" + ControleLocalIda);
45         print("Controle local Volta:" + ControleLocalVolta);
46     }
47
48     if (Input.GetKeyDown(KeyCode.O) || OVRInput.GetDown(OVRInput.Button.PrimaryIndexTrigger))
49     {
50         if (ControleLocalIda > 0)
51         {
52             tocando = true;
53             gameObject.GetComponent<Animation>().clip = anims_volta[ControleLocalVolta];
54             gameObject.GetComponent<Animation>().Play();
55             ControleLocalIda--;
56             if (ControleLocalVolta > 0)
57                 ControleLocalVolta--;
58         }
59         print("Controle local Ida:" + ControleLocalIda);
60         print("Controle local Volta:" + ControleLocalVolta);
61     }
62 }
63
64
65

```

Figura 12: Controle de avanço e retrocesso

Considerando a definição dos elementos, os textos explicativos são expostos em cada uma dessas estações supracitadas, e junto com eles os áudios que foram previamente gravados como narração também são executados. Para que isso seja exposto de forma correta, é necessário uma associação de cada áudio e texto com seu respectivo objeto, portanto foram utilizadas marcas virtuais presentes em espaços específicos do cenário onde o usuário vai realizar as paradas. Esses espaços funcionam como gatilhos, conhecidos no Unity como “Triggers” (Fig. 14). A partir do momento que um gatilho é acionado pela entrada do personagem, por meio de associação, o sistema sabe qual estação está sendo visitada e informa quais áudios e textos devem ser selecionados e executados. Essa lógica exposta foi implementada por meio de código e os elementos foram associados na definição das variáveis, sendo elas do tipo “string” para os textos e do tipo “audioclip” para os áudios, organizados em vetores (Fig. 15). Logo são 32 variáveis, 2 para cada objeto destacado.

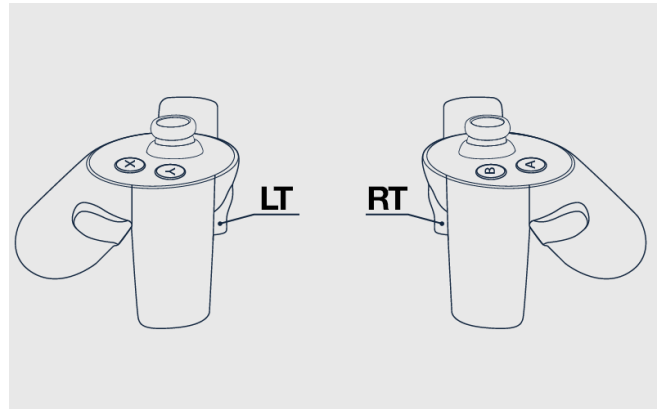


Figura 13: Botões ‘LT’ e ‘RT’ do joystick



Figura 14: Collider dos triggers

Assim que o personagem se posiciona no espaço referido para observação do objeto, surge um painel virtual com o texto associado àquele objeto específico, e o mesmo é exibido em um sistema de rolagem automática (Fig. 16), acompanhando o tempo relativo ao áudio também associado e executado. Dessa forma, tem um objeto painel, que faz parte da definição de UI (Interface de usuário), que segue as mesmas predefinições utilizadas no menu no que se refere à estilização, cores e fontes escolhidas. Para a execução perfeita da rolagem do texto, foi necessário a criação de um código com a classe correspondente ao texto, que define até que altura o mesmo vai ser exibido, para que o conjunto inteiro de caracteres seja exposto sem que haja cortes ou falhas em relação à execução do áudio em segundo plano.

A execução do texto e áudio é equivalente ao tempo que o usuário permanece no gatilho. Quando uma tecla é pressionada para avançar ou voltar uma estação do roteiro, imediatamente o painel é removido da tela até que a outra estação seja alcançada e todo o processo seja feito novamente.

4.2.3 Manipulação de Elementos

A implementação da movimentação foi desenvolvida via código, seguindo as diretrizes de movimentação do personagem no Unity, que usa a variável do tipo “Vector 3”, que é uma matriz de movimentação no espaço do jogo. Assim,

```

public class subida_texto : MonoBehaviour
{
    public float altura_desejada;
    public float velocidade_scroll;
    public int tempo_de_espera;
    public string[] textos_pt, textos_en;
    public AudioClip[] audios_pt, audios_en;

    public static void muda_texto_ativa(string texto_lng, int num_texto, int num_audio) {
        switch (texto_lng) {
            case "pt":
                print("Ativa texto");
                esse_texto.GetComponent<Text>().text = esse_texto.GetComponent<subida_texto>().textos_pt[num_texto];
                esse_texto.GetComponent<AudioSource>().clip = esse_texto.GetComponent<subida_texto>().audios_pt[num_audio];
                esse_texto.GetComponent<AudioSource>().Play();
                break;
            case "en":
                esse_texto.GetComponent<Text>().text = esse_texto.GetComponent<subida_texto>().textos_en[num_texto];
                esse_texto.GetComponent<AudioSource>().clip = esse_texto.GetComponent<subida_texto>().audios_en[num_audio];
                esse_texto.GetComponent<AudioSource>().Play();
                break;
            default:
                esse_texto.GetComponent<Text>().text = esse_texto.GetComponent<subida_texto>().textos_pt[num_texto];
                esse_texto.GetComponent<AudioSource>().clip = esse_texto.GetComponent<subida_texto>().audios_pt[num_audio];
                esse_texto.GetComponent<AudioSource>().Play();
                break;
        }
    }
}

```

Figura 15: Controle de texto e áudio

```

IEnumerator subida_text_func()
{
    print(gameObject.GetComponent<RectTransform>().anchoredPosition);
    yield return new WaitForSeconds(tempo_de_espera);
    while (gameObject.GetComponent<RectTransform>().anchoredPosition.y < altura_desejada)
    {
        if(gameObject.GetComponent<RectTransform>().anchoredPosition.y < 0)
        {
            gameObject.GetComponent<RectTransform>().anchoredPosition = new Vector2(gameObject.GetComponent<RectTransform>().anchoredPosition.x,
            gameObject.GetComponent<RectTransform>().anchoredPosition.y + velocidade_scroll);
        }
        if (gameObject.GetComponent<RectTransform>().anchoredPosition.y == 0)
        {
            gameObject.GetComponent<RectTransform>().anchoredPosition = new Vector2(gameObject.GetComponent<RectTransform>().anchoredPosition.x, 0.00001f);
        }
        else
        {
            gameObject.GetComponent<RectTransform>().anchoredPosition = new Vector2(gameObject.GetComponent<RectTransform>().anchoredPosition.x,
            gameObject.GetComponent<RectTransform>().anchoredPosition.y + velocidade_scroll);
        }
    }
    yield return null;
}

```

Figura 16: Rolagem do texto

a alteração dos valores referentes a cada eixo é relacionada à entrada dos botões, no caso do teclado são “W”, “S”, “A” e “D” (Fig. 17), e no caso do joystick é utilizado o analógico esquerdo, realizando a movimentação do personagem em tempo real. Ao pressionar os botões ou alterar a posição do analógico no caso do joystick, os valores representados na matriz de posição são acrescidos ou diminuídos. Dessa forma as funções correspondem aos seguintes movimentos:

- Forward – Frente
- Back – Trás
- Left – Esquerda
- Right – Direita

As funções citadas podem ser observadas na Fig. 18, que correspondem ao seu uso para movimentação do personagem pelo ambiente.

No uso dos capacetes de RV, a rotação da câmera principal é realizada por meio do movimento da cabeça, devido aos sensores de movimento presentes no hardware. Isso é capturado pelo Unity e transferido diretamente às variáveis de rotação, que são do tipo “Quaternion” no sistema. Essa rotação é definida via código, e funciona como a lógica de movimentação vista anteriormente, a cada movimento feito pelo usuário, seja no uso do capacete ou no uso do mouse, as variáveis que compõem a matriz de rotação do “Quaternion” são acrescidas ou diminuídas, permitindo a movimentação da cabeça do personagem virtual. Através de variáveis definidas diretamente pela API do Oculus, as

```

public virtual void UpdateMovement()
{
    if (HaltUpdateMovement)
        return;

    if (EnableLinearMovement)
    {
        bool moveForward = Input.GetKey(KeyCode.W) || Input.GetKey(KeyCode.UpArrow);
        bool moveLeft = Input.GetKey(KeyCode.A) || Input.GetKey(KeyCode.LeftArrow);
        bool moveRight = Input.GetKey(KeyCode.D) || Input.GetKey(KeyCode.RightArrow);
        bool moveBack = Input.GetKey(KeyCode.S) || Input.GetKey(KeyCode.DownArrow);

        bool dpad_move = false;

        if (OVRInput.Get(OVRInput.Button.DpadUp))
        {
            moveForward = true;
            dpad_move = true;
        }

        if (OVRInput.Get(OVRInput.Button.DpadDown))
        {
            moveBack = true;
            dpad_move = true;
        }

        MoveScale = 1.0f;
    }
}

```

Figura 17: Mapeamento das teclas

```

Quaternion ort = transform.rotation;
Vector3 ortEuler = ort.eulerAngles;
ortEuler.z = ortEuler.x = 0f;
ort = Quaternion.Euler(ortEuler);

if (moveForward)
    MoveThrottle += ort * (transform.lossyScale.z * moveInfluence * Vector3.forward);
if (moveBack)
    MoveThrottle += ort * (transform.lossyScale.z * moveInfluence * BackAndSideDampen * Vector3.back);
if (moveLeft)
    MoveThrottle += ort * (transform.lossyScale.x * moveInfluence * BackAndSideDampen * Vector3.left);
if (moveRight)
    MoveThrottle += ort * (transform.lossyScale.x * moveInfluence * BackAndSideDampen * Vector3.right);

```

Figura 18: Valores de movimentação

movimentações de rotação e translação no âmbito físico são transferidas para o “Quaternion” do personagem e alteradas para reposicionar a cabeça como definido na função “SetHeadPositionModifier” (Fig. 19). Caso o usuário não esteja utilizando o capacete, a rotação da câmera é definida pela posição do ponteiro do mouse na tela, comumente utilizada nos jogos digitais.

```

private Vector3 _headPoseRelativeOffsetRotation;
public Vector3 headPoseRelativeOffsetRotation
{
    get
    {
        return _headPoseRelativeOffsetRotation;
    }
    set
    {
        OVRPlugin.Quatf rotation;
        OVRPlugin.Vector3f translation;
        if (OVRPlugin.GetHeadPoseModifier(out rotation, out translation))
        {
            Quaternion finalRotation = Quaternion.Euler(value);
            rotation = finalRotation.ToQuatf();
            OVRPlugin.SetHeadPoseModifier(ref rotation, ref translation);
        }
        _headPoseRelativeOffsetRotation = value;
    }
}

```

Figura 19: Movimentação da câmera

A partir da movimentação livre do personagem, a manipulação dos objetos foi implementada utilizando os joysticks do capacete de RV, que devido aos sensores, conseguem mapear os movimentos das mãos do usuário e transferir tais movimentos às mãos virtuais. Ao pressio-

nar botões específicos no joystick, a mão virtual responde e executa animações referentes ao novo posicionamento dos dedos. Esse comportamento é definido pelo código exposto na Fig. 20, contido na API do Oculus, no qual a função “UpdateAnimStates” atualiza as mãos virtuais ao pressionar os botões físicos. A função executa a animação referente à uma das 4 posições possíveis:

- Mão aberta (Pose)
- Mão fechada (Flex)
- Dedo indicador apontado (Point)
- Dedo polegar para cima (Thumbs Up)

```
private void UpdateAnimStates()
{
    bool grabbing = m_grabber.grabbedObject != null;
    HandPose grabPose = m_defaultGrabPose;
    if (grabbing)
    {
        HandPose customPose = m_grabber.grabbedObject.GetComponent<HandPose>();
        if (customPose != null) grabPose = customPose;
    }
    // Pose
    HandPoseId handPoseId = grabPose.PoseId;
    m_animator.SetInteger(m_animParamIndexPose, (int)handPoseId);

    // Flex
    // blend between open hand and fully closed fist
    float flex = OVRInput.Get(OVRInput.Axis1D.PrimaryHandTrigger, m_controller);
    m_animator.SetFloat(m_animParamIndexFlex, flex);

    // Point
    bool canPoint = !grabbing || grabPose.AllowPointing;
    float point = canPoint ? m_pointBlend : 0.0f;
    m_animator.SetLayerWeight(m_animLayerIndexPoint, point);

    // Thumbs up
    bool canThumbsUp = !grabbing || grabPose.AllowThumbsUp;
    float thumbsUp = canThumbsUp ? m_thumbsUpBlend : 0.0f;
    m_animator.SetLayerWeight(m_animLayerIndexThumb, thumbsUp);

    float pinch = OVRInput.Get(OVRInput.Axis1D.PrimaryIndexTrigger, m_controller);
    m_animator.SetFloat("Pinch", pinch);
}
```

Figura 20: Movimentação das mãos

Os objetos passíveis de manipulação contêm um componente chamado “Collider”, que nada mais é que um colisor para representar a forma física do mesmo (Fig. 21), definindo os limites e o comportamento físico do objeto na cena. Para simular a física do mundo real em ambiente virtual, foi necessário a associação do componente “Rigidbody” que permite a definição de um objeto como corpo rígido, além da gravidade virtual e a forma como ela atua no elemento.

As mãos virtuais interagem com os objetos através de um comportamento definido via código (Fig. 22). Esse código define que, se o elemento interativo tiver seu collider invadido pela mão e a mão virtual se fechar durante esse processo, o collider será temporariamente desligado e a posição do objeto passa a respeitar a posição relativa às mãos, ação que pode ser observada na chamada da função “MovePosition” e “MoveRotation”, que recebem como parâmetro, justamente a posição e rotação das mãos naquele instante específico. Além disso, a física dos objetos capturados é temporariamente desligada. Dessa maneira, a mão consegue carregar o objeto sem a interferência de outros agentes externos, e a sua posição e rotação no espaço do jogo ficam definidas pelas mãos até que o objeto seja solto novamente, o que faz com que o collider e a física do objeto sejam novamente ligados. Os objetos principais que

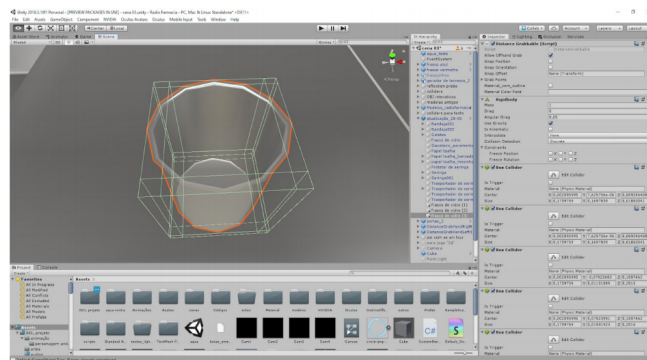


Figura 21: Collider do objeto

podem ser manipulados são:

```
protected override void MoveGrabbedObject(Vector3 pos, Quaternion rot, bool forceTeleport = false)
{
    if (m_grabbedObj == null)
    {
        return;
    }

    Rigidbody grabbedRigidbody = m_grabbedObj.GetComponent<Rigidbody>();
    Vector3 grabbablePosition = pos + rot * m_grabbedObjectPosOff;
    Quaternion grabbableRotation = rot * m_grabbedObjectRotOff;

    if (m_movingObjectHand)
    {
        float travel = m_objectPullVelocity * Time.deltaTime;
        Vector3 dir = grabbablePosition - m_grabbedObj.transform.position;
        if (travel > dir.magnitude)
        {
            dir.Normalize();
            grabbablePosition = m_grabbedObj.transform.position + dir * travel;
            grabbableRotation = Quaternion.LookRotation(m_grabbedObj.transform.rotation, grabbablePosition);
        }
        grabbedRigidbody.MovePosition(grabbablePosition);
        grabbedRigidbody.MoveRotation(grabbableRotation);
    }
}
```

Figura 22: Lógica para manipulação dos objetos

- Tubos de ensaio
- Transportador de seringa
- Gerador de tecnécio
- Frascos de vidro

A fim de proporcionar uma experiência mais completa de imersão e manipulação dos objetos em cena, as maçanetas das portas virtuais podem ser acionadas também com o movimento das mãos. Essa lógica, definida pelo código exposto na Fig. 23, prevê que os objetos da maçaneta podem ser rotacionados, e a partir de um certo grau dessa rotação, é realizada uma chamada à função declarada no código “controle portas”, que executa uma animação de abertura da porta. Após esse acionamento, a rotação da maçaneta volta à sua posição inicial utilizando a função responsável por fazer uma interpolação entre os dois valores da rotação de maneira suavizada.

5 Testes e Resultados

5.1 Simulador da Radiofarmácia

A aplicação radiofarmácia, como ambiente virtual, se mostra como uma forma interessante e realista de representar o ambiente real. Dessa forma, os usuários tiveram uma experiência completa ao navegar pelo ambiente e perceber a presença dos elementos mais importantes de um laboratório e a disposição dos mesmos pela área proposta no simulador. O uso de modelos 3D otimizados resulta-

```

void Update()
{
    if(transform.localEulerAngles.z>100)
    {
        if(entra_lerp==false)
            StartCoroutine(lerpar());
    }
}

IEnumerator lerpar()
{
    if (entra_lerp)
    {
        yield break;
    }
    entra_lerp = true;

    obj = Quaternion.Euler(0, 90, 90);
    float t = 0f;

    yield return new WaitForSeconds(0.1f);
    transform.GetComponentInParent<controle_portas>().cont_porta();
    while (transform.localEulerAngles.z != 90)
    {
        t += Time.deltaTime;
        print("ENTROU");
        transform.localRotation = Quaternion.Lerp(transform.localRotation,obj, t/duracao);

        yield return null;
    }
    yield return new WaitForSeconds(1f);
    entra_lerp = false;
}

```

Figura 23: Valores de movimentação

ram em um desempenho satisfatório e, ao mesmo tempo, com bom aspecto visual. A iluminação contribuiu para o bom resultado gráfico, uma vez que os posicionamentos das luzes virtuais seguiram a mesma disposição visualizada nos laboratórios visitados (Fig. 24). A identificação do ambiente como um todo se tornou bem instantânea, principalmente com testes feitos com usuários profissionais de medicina nuclear e proteção radiológica.



Figura 24: Área de manipulação

5.2 Navegação Virtual (Automatizada)

A implementação do modo guiado resultou em um bom fluxo de navegação durante a execução dos testes e principalmente na percepção dos usuários. A utilização dos comandos do joystick para avançar e retroceder nas estações foi uma ferramenta facilitadora para essa fase do simulador, uma vez que seu posicionamento é de fácil acesso por parte dos usuários, mesmo que estejam usando os capacetes de RV se tornando rapidamente uma forma intuitiva de comandar essa parte da experiência.

A ordem dos elementos da navegação automatizada e as informações destacadas em cada estação tiveram que ser alteradas algumas vezes mediante à orientação de profissionais da área. Dessa forma, foram considerados 17 pontos para compor as paradas que o sistema faz na explicação de componentes (Fig. 25). Os principais pontos são:

- Entrada
- Área de Administração
- Paramentação e Proteção Radiológica
- Contador Geiger-Müller
- Curiômetro
- Transportador de Seringa
- Capela de Fluxo Laminar
- Estoque/Armazenamento
- Dispositivo de Içamento
- Estoque de Fonte
- Controle de Qualidade
- Coletor de Resíduos
- Depósito de Rejeitos
- Carrinho de Transporte
- Passagem dos Radiofármacos
- Sala para a Administração de Radiofármacos
- Protetor de Seringa

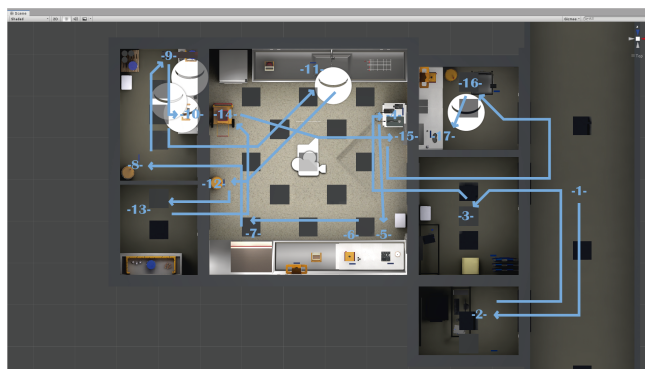


Figura 25: Roteiro das interações

Após as alterações, o funcionamento do sistema ocorreu de forma satisfatória, e a implantação de áudio narrado junto ao texto facilitou ainda mais a compreensão das informações em destaque.

5.3 Funcionalidades

O uso da aplicação como simulador exige a implementação da manipulação dos objetos da cena referente ao processo de manuseio dos radiofármacos. Portanto, ao fim da implementação do modo guiado que segue um caminho pré estipulado, foi desenvolvido o modo considerado “livre”, que inclui a livre circulação do personagem pela cena através dos controles e a manipulação de alguns dos principais objetos expostos. O modo livre dá autonomia ao usuário para controlar a movimentação dentro do ambiente virtual. Essa movimentação foi implementada de forma a utilizar os comandos do teclado no caso de uso sem capacete de RV ou do joystick disponível para o uso do capacete. Os co-

mandos associados à movimentação são: “W” para frente, “S” para trás, “D” para direita e “A” para a esquerda, no caso do teclado, e o uso do analógico esquerdo do joystick no caso de uso do capacete de RV (Fig. 26).



Figura 26: Movimentação no joystick

A escolha desses comandos seguiu um padrão observado pelos jogos desenvolvidos para computador no caso do teclado. E no caso dos joysticks, o analógico esquerdo foi projetado com o propósito de movimentar o personagem.

Além disso, foi utilizado um modelo de mãos virtuais para conceder a sensação de imersão e manipulação de objetos. As mãos virtuais respondem ao pressionamento de botões específicos no joystick, definidos por “LG”, “RG”, “LT” e “RT”. Os botões “LG” e “RG” (Fig. 27), são referentes aos 3 últimos dedos da mão esquerda e direita respectivamente, de forma que, ao serem pressionados, esses 3 dedos virtuais serão fechados, assim como os botões “LT” e “RT” (Fig. 28), são responsáveis pelo fechamento dos dedos indicadores esquerdo e direito respectivamente. Ao pressionar todos esses botões citados, a mão virtual se fecha por completo, realizando o movimento necessário para pegar um objeto da cena virtualmente (Fig. 29).



Figura 27: Botões ‘LG’ e ‘RG’

Essa forma de manipulação e movimentação desenvolvida foi escolhida por funcionar de maneira bem intuitiva, principalmente no caso da manipulação de objetos, uma vez que a forma como o joystick do Oculus Rift se encaixa na mão do usuário tende a facilitar a assimilação de res-



Figura 28: Botões ‘LT’ e ‘RT’



Figura 29: Usuário manipulando o objeto

posta das mãos virtuais ao pressionar os botões de cada posição.

Na aplicação final, o usuário pode manipular tubos de ensaio, transportadores de seringa, gerador de técnico, maçanetas das portas e frascos de vidro. Esses objetos, especificamente, tem uma grande importância nos procedimentos operacionais referentes ao fluxo dos radiofármacos dentro do ambiente laboratorial. Dessa forma, o treinamento pode ser efetuado, inicialmente, utilizando essas técnicas e objetos virtuais como referência evitando a exposição dos profissionais à radiação mas, ao mesmo tempo, dispondo uma noção de como seria a manipulação destes elementos em uma situação real.

Apesar do simulador responder bem às entradas realizadas pelos botões e não apresentar problemas técnicos em relação a isso, alguns usuários tiveram dificuldades para manipular objetos virtuais com os joysticks, por vezes se atrapalhando com os comandos e apresentando um certo nível de frustração ao tentar pegar os objetos virtuais, muitas vezes por não terem experiência com a jogabilidade e com os comandos dos joysticks.

5.4 Requisitos Mínimos

- Placa gráfica: NVIDIA GTX 1050Ti, AMD Radeon RX 470 ou superior
- CPU: Intel i3-6100, AMD Ryzen 3 1200, FX4350 ou superior
- Memória: 8GB de RAM ou mais

- Saída de vídeo: DisplayPort™ 1.2, Mini DisplayPort (com adaptador incluso na caixa)
- Portas USB: Uma porta USB 3.0
- Sistema Operacional: Windows 10

5.5 Público Objetivo

Para validação da ferramenta desenvolvida, foi necessário organizar uma apresentação com grupo de profissionais, que tenham a formação pertinente à área ou atuantes no segmento nuclear.

Os profissionais participantes foram divididos em três grupos, o primeiro grupo é composto por profissionais técnicos da área nuclear que atuam diretamente em radiofarmácias, com a manipulação de radioisótopos, o segundo grupo interface homem/sistema do IEN, foi escolhido com propósito de realizar avaliação da ferramenta, pois possuem conhecimento no uso de sistemas e em cognição humana. O terceiro grupo são servidores da Divisão de Segurança e proteção radiológica (DISPR) do IEN, que realizaram avaliação do ponto de vista da proteção radiológica, do modelo virtual de uma radiofarmácia.

Primeiramente, foram apresentados em slides uma breve introdução aos objetivos, com as principais diretrizes, normas e as legislações vigentes, que são aplicadas em uma radiofarmácia real e as boas práticas pertinentes a MN. Posteriormente em associação com ambiente virtual um roteiro, descrevendo-se os ambientes que compõem o fluxo do radiofarmaco, desde o seu recebimento até administração ao paciente de forma detalhada. Em seguida os profissionais de forma individualizada, foram orientados pela equipe de desenvolvimento sobre o funcionamento do modelo (simulador), a experiência imersiva no ambiente virtual, inicialmente os mesmos seguiram o roteiro pré-estabelecido, exemplificado anteriormente nos slides e ao final da aplicação foi colocada para livre experimentação. Após uso da ferramenta os profissionais, foram convidados a responder o questionário de avaliação.

No questionário de avaliação foram propostas 12 questões sobre a ferramenta virtual, foram elas:

- O ambiente virtual representa as condições adequadas de infraestrutura física, prevista na RDC Anvisa no. 50/02, Normas CNEN-NE-3.05 e CNEN-NE-6.02, referente a uma instalação de Radiofarmácia em um Serviço de Medicina Nuclear.
- O ambiente virtual apresenta estar equipado com o mínimo, de equipamentos e materiais previstos em Norma CNEN-NE-3.05, referente a uma instalação de Radiofarmácia em um Serviço de Medicina Nuclear.
- Um treinamento realizado no ambiente virtual (simulador), apresenta os requisitos mínimos de proteção radiológica previstos em Norma CNEN-NE-3.05, referente a uma instalação de Radiofarmácia em um Serviço de Medicina Nuclear.
- A apresentação de falas e as legendas no ambiente virtual, expressam corretamente a finalidade dos equipamentos e ambiente de uma Radiofarmácia.
- O fluxo estabelecido para prática operacional no ambiente virtual, reflete à prática cotidiana de uma Radiofarmácia em um Serviço de Medicina Nuclear.
- O ambiente virtual como um todo, simula um ambiente

real de uma Radiofarmácia.

- A apresentação dos slides em associação com ambiente virtual em Radiofarmácia, permite que o treinamento aplicado tenha melhor aprendizagem.
- O ambiente virtual auxilia de forma eficiente e dinâmica na compreensão dos elementos presentes em uma Radiofarmácia.
- A criação do ambiente virtual (simulador), utilizando técnicas de realidade virtual, é uma boa ferramenta para a preparação de profissionais que irão atuar em uma Radiofarmácia
- A simulação em ambiente virtual permite diminuição de custos no processo de treinamento.
- Com a utilização da realidade virtual os riscos ao profissional se tornam menos eminentes.
- O ambiente virtual permite que o treinamento tenha menor custo se comparado com o ambiente real.

5.6 Resultados pelo Método Likert

Para a análise segundo a escala de Likert, foram analisados 3 grupos:

- Para o primeiro grupo de Técnicos da Área Nuclear, foram obtidas as respostas de 7 questionários cujos resultados são exibidos na Fig. 30.
- Para o segundo grupo de Interface Homem / Sistema do IEN, foram obtidas as respostas de 4 questionários cujos resultados são exibidos na Fig. 31.
- Para o terceiro grupo de Proteção Radiológica do IEN, foram obtidas as respostas de 8 questionários cujos resultados são exibidos na Fig. 32.

Para estas imagens, vale os seguintes critérios para as respostas:

- DI: Discordo integralmente;
- DP: Discordo parcialmente;
- NCND: Não concordo nem discordo;
- CP: Concordo parcialmente;
- CI: Concordo integralmente.

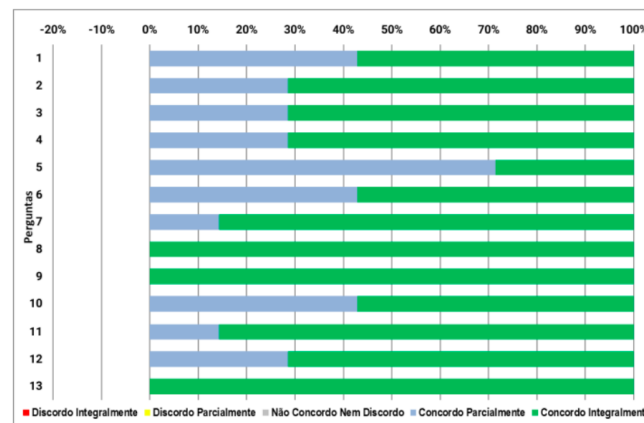


Figura 30: Grupo de Técnicos da Área Nuclear

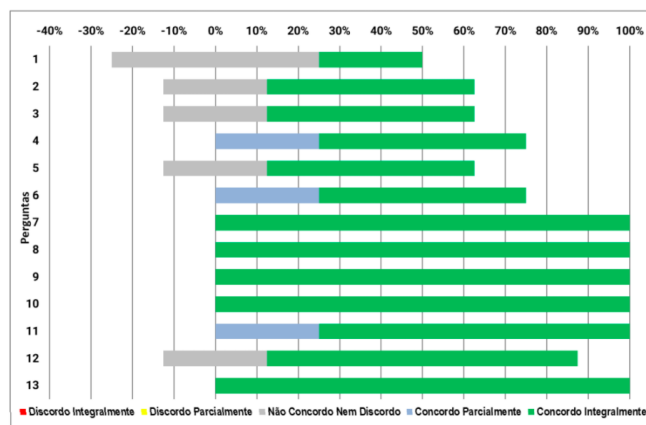


Figura 31: Grupo de Interface Homem / Sistema

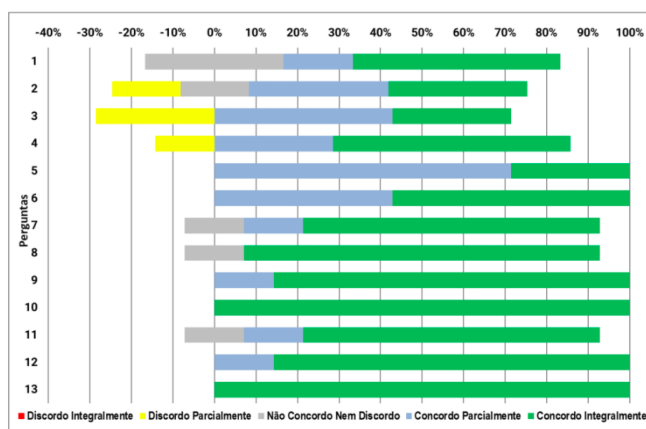


Figura 32: Grupo de Proteção Radiológica

6 CONCLUSÃO E TRABALHOS FUTUROS

O projeto teve como objetivo o desenvolvimento de uma ferramenta que pudesse simular os principais procedimentos operacionais de profissionais da medicina nuclear, utilizando técnicas baseadas em Serious Games e na tecnologia de Realidade Virtual para obter um resultado visual e interativo útil e interessante.

A elaboração do simulador apresenta vantagens relevantes ao apresentar uma solução de treinamento básica sem a utilização de ambientes reais e produtos radioativos, levando em consideração a proteção física dos profissionais que possam atuar na área, promovendo um treinamento seguro, imersivo e reativo às manipulações dos usuários, além de reduzir expressivamente os custos relacionados a um ambiente de treinamento.

A representação visual do laboratório de radiofarmácia foi implementada de forma realista, seguindo métricas rígidas e referências de espaços físicos reais, apresentando um alto nível de qualidade gráfica e promovendo uma sensação de imersão perceptível, principalmente ao usar o capacete de RV durante a experiência.

Os principais objetos utilizados pelos profissionais foram devidamente representados, categorizados e explica-

dos de maneira objetiva e dinâmica, expondo as informações cruciais para a compreensão do fluxo dos radiofármacos em ambiente hospitalar.

A lógica da interação dos usuários com os elementos do cenário e a livre circulação do personagem desenvolvida durante o trabalho contribuíram para a manutenção dos aspectos de treinamento no simulador, respeitando, principalmente, as métricas e as técnicas de utilização do capacete de RV, de forma a proporcionar uma experiência segura e ainda mais acessível a todos os usuários. Além disso, a otimização de códigos e modelos feita durante o processo foi de suma importância para a execução do software na plataforma do Oculus Rift, uma vez que não houve problemas técnicos relacionados ao desempenho da aplicação durante todo o período de utilização.

Os testes realizados durante e após a finalização do projeto mostraram a viabilidade de implementação da experiência em treinamentos de funcionários, uma vez que a ferramenta foi testada e aprovada por profissionais de medicina nuclear e agentes de proteção radiológica, que atestaram a eficácia da aplicação como simulador.

Portanto, conclui-se que é perfeitamente viável o uso de simuladores para o treinamento de profissionais não só da medicina nuclear como de outras áreas. Também conclui-se que a utilização da tecnologia de RV se mostra como uma alternativa ainda mais interessante para promover a imersão nesses ambientes e garantir uma boa experiência para maior parte dos usuários.

Como sugestão de trabalhos futuros, seria interessante a inserção de mais elementos interativos a ponto de todo o funcionamento do laboratório ser simulado de forma virtual. Dessa maneira, abrangendo ainda mais o público-alvo do projeto e promovendo uma experiência de simulação mais completa.

Além disso, a implementação de um sistema de rede em que vários usuários pudessem entrar simultaneamente no mesmo ambiente e interagir no laboratório virtual, elevando a experiência para o próximo nível, contando com a contribuição de outros usuários na ferramenta, dessa forma simulando também como seria o trabalho em grupo em um ambiente real.

Referências

- ADOBE (2009). Adobe photoshop cs3 classroom in a book: Guia oficial de treinamento. 1. ed. EUA: Bookman. p. 1-10.
- AUTODESK (2020). Novos recursos do 3ds max. Disponível em: <https://www.autodesk.com.br/products/3ds-max/features?plc=3DSMAX&term=1-YEAR&support=ADVANCED&quantity=1>.
- Burdea, Grigore C; Coiffet, P. (2003). Virtual reality technology. 2.ed. Canada: John Wiley & Sons, Inc. p.2-4.
- Chelles, D. R. (2017). Uso da realidade virtual no planejamento de repositórios de rejeitos radioativos. IEN, Brasil, v.1, n.1, p.30-50. Disponível em: https://www.ien.gov.br/images/posien/dissertacoes/dissertacao_mestrado_ien_2017_06.pdf.
- Hagita, K., Kodama, Y. and Takada, M. (2020). Simplified virtual reality training system for radiation shielding

- and measurement in nuclear engineering, *Progress in Nuclear Energy* **118**: 103127. <https://doi.org/10.1016/j.pnucene.2019.103127>.
- Jacobson, J. and Lewis, M. (2002). Game engines in scientific research, *Communications of The ACM* **45**: 27–31. Disponível em: <https://www.cse.unr.edu/~sushil/class/gas/papers/GameAIp27-lewis.pdf>.
- Johnson, B. (2012). Professional Visual Studio 2012. 1.ed. Indiana, USA: John Wiley & Sons, Inc. p.1–4.
- Marins, E. R. (2018). Realidade virtual aplicada À proteção física de instalações nucleares. COPPE UFRJ, Brasil, v.1, n.1, p.1–50, jun./2018. Disponível em: <https://www.coc.ufrj.br/pt/documents2/doutorado/2018-1/3168-marins-er-td-18-1>.
- Matossian, M. (2001). 3ds max para windows. 4.ed. São Paulo: Campus, 2001. p.2–4.
- Mazuryk, Tomasz; Gervautz, M. (1996). Virtual reality: History, applications, technology and future. Institute of Computer Graphics, Austria, v.1, n.1, p. 1–4, jan./1996. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.7849>.
- Melo, M. R. and Adamatti, D. (2020). Descoberta de conhecimento em dados não estruturados: Uma análise em um chat de jogo online, *Revista Brasileira de Computação Aplicada* **12**(3): 51–60. <http://dx.doi.org/10.5335/rbca.v12i3.10389>.
- Microsoft (2019). Introdução às ferramentas do Visual Studio para Unity. Disponível em: <https://docs.microsoft.com/pt-br/visualstudio/cross-platform/getting-started-with-visual-studiotools-for-unity?view=vs-2019>.
- O'Connor, M., Stowe, J., Potocnik, J., Giannotti, N., Murphy, S. and Rainford, L. (2020). 3d virtual reality simulation in radiography education: The students' experience, *Radiography* **27**: 208–214. <https://doi.org/10.1016/j.radi.2020.07.017>.
- Okita, A. (2014). Virtual reality technology. 2.ed. Canada: John Wiley & Sons, Inc. p.2–4.
- Tsai, F. (2016). La réalité virtuelle, un outil pour renouer avec la sensorialité?. Hermès, La Revue, France, v.1, n.74, p.188–199.
- Unity3D (2020a). Realidade virtual. Disponível em: <https://unity.com/pt/unity/features/vr>.
- Unity3D (2020b). Unity user manual. Disponível em: https://docs.unity3d.com/Manual/index.html?_ga=2.61060713.477388292.1591042548-1539504131.1589749459.
- Virtual Reality Society (2020). History of virtual reality. Disponível em: <https://developer.vive.com/resources/knowledgebase/vive-specs>.
- VIVE Developers (2020). Vive Specs & User Guide. Disponível em: <https://developer.vive.com/resources/knowledgebase/vive-specs/>.
- VR BOUND (2020). Oculus Rift Specs & Requirements. Disponível em: <https://www.vrbound.com/headsets/oculus/rift>.
- Zyda, M. (2007). Creating a science of games : The same technology that makes interactive 3d games so entertaining in the physical action domain is just as effective in education, training, and other more serious applications. Communications of the ACM, Los Angeles, v.50, n.7, p.27–28,. Disponível em: <http://gamepipe.org/zyda/resources/pubs/CACM-July2007.pdf>.