

ESPAÇO VIRTUAL IMERSIVO PARA MULTI USUÁRIOS - APLICAÇÃO PRÁTICA EM REALIDADE VIRTUAL

Davi Belloni Jorge (IC) e Luciano Silva (Orientador)

Apoio: PIBIC Mackenzie

Resumo

O ambiente virtual 3D criado a partir do motor Unreal 4 utilizando conceitos de Realidade Virtual e por isso foi dada ênfase na realidade e na parte visual. Foi utilizado processos para criação de jogos como: a elaboração de um design de fase, a criação de objetos estáticos no espaço, a texturização de todo o espaço e a programação de iluminação e interatividade desse espaço para o(s) usuários. Ao início da pesquisa foi pensado em quais tecnologias deveriam ser usadas e quais elementos utilizados no espaço, após alguns testes foi decidido usar o motor gráfico Unreal pela facilidade em adaptar o espaço 3D para Realidade Virtual. Foi criado um sistema que aceita mais de um usuário no mesmo ambiente virtual promovendo assim a interatividade. Neste projeto é realizado um estudo geral de construção de uma aplicação de Realidade Virtual com base na metodologia de criação de jogos, sendo assim é feito um ambiente utilizável e otimizado para diversas plataformas, claro com a utilização do Oculus Rift. É discorrido também acerca da importância da imersão do usuário no ambiente e como a falta dela pode reduzir a experiência como um todo, assim como caso o ambiente não esteja de acordo com o que já é conhecido do mundo real pelo usuário e seu funcionamento pode acarretar em um estranhamento por parte do usuário

Palavras-Chave: Realidade Virtual, usuário, ambiente virtual, imersão

Abstract

The 3D virtual environment created from the Unreal 4 engine using concepts of virtual reality and so was given emphasis on reality and the visual part. It was used processes for creating games such as the preparation of a designating phase, the creation of static objects in space, the texturing of all the space and lighting programming and interactivity that space for the users. At the beginning of the research was designed in which technologies should be used and which elements used in space, after some tests it was decided to use the Unreal graphics engine for the easy to adapt the 3D space to Virtual Reality. Was created system that accepts more than one user in the same virtual environment thus promoting interactivity was created. In this project we conducted a general study of building an application of virtual reality based games design methodology, so it made a usable environment and optimized for multiple platforms, of course using the Oculus Rift. It discoursed also about the importance of user immersion in the environment and how the lack of it can reduce the experience as a

whole, and if the environment is not in accordance with what is known of the real world by the user and in operation could lead in an estrangement from the user.

Keywords: Virtual Reality, user, virtual environment, immersion

Introdução

O objetivo deste projeto é demonstrar como é possível unir e promover interação social de pelo menos duas pessoas em um ambiente virtual por meio do uso da Realidade Virtual. Utilizando da imersão possível com o Oculus Rift para tornar essa experiência virtual agradável ao usuário.

Esta tecnologia serve não só como experiência social, mas também como tratamento de traumas e maior interação na educação tendo em vista que é esses são muito mais eficazes se feitos com uma experiência mais intensa e controlada, elementos que só um ambiente virtual pode gerar com 100% de eficácia se feitos em um ambiente foto realista.

Com este projeto é esperado verificar a possibilidade de imersão com mais de um usuário no mesmo ambiente e demonstrar uma aplicação de Realidade Virtual com Oculus Rift

Referencial Teórico

Romero Tori, Claudio Kirner e Robson Siscoutto defendem em Fundamentos e Tecnologia de Realidade Virtual e Aumentada (2006) que a Realidade Virtual tem potencial para uma vasta área de aplicação, podendo ser usado por uma única pessoa isoladamente, como por muitas pessoas usando um sistema distribuído.

Ainda é citado que com o auxílio dos Ambientes Virtuais Distribuídos (AVDs) os usuários podem compartilhar o mesmo espaço tridimensional virtual de trabalho e podem interagir entre si, podem agir em conjunto para realizar uma tarefa. Essas pessoas estão nesse espaço sendo representadas por meio de um *avatar* (representante virtual) e compartilham tempo, espaço e presença, tornando possível o trabalho conjunto da mesma forma que em um ambiente real.

Metodologia

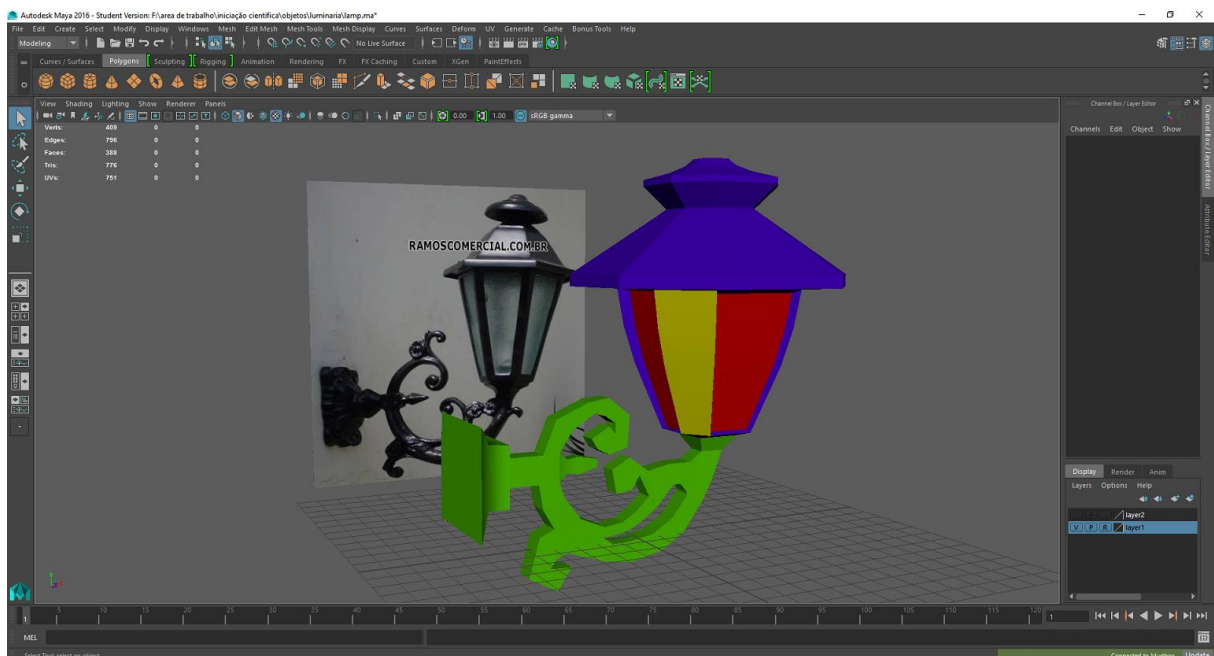
Para o desenvolvimento desse projeto foi realizado uma pesquisa para saber quais ferramentas seriam utilizadas no desenvolvimento do ambiente virtual. Para desenvolver um projeto que envolva um ambiente 3D interativo como há em jogos e simuladores é necessários softwares de apoio para a modelagem dos objetos a serem utilizados, softwares para a montagem desse ambiente e outro para a codificação do programa para que o usuário passa utiliza-lo. Depois de pesquisar foi decidido utilizar o Unreal 4.11.2 como motor gráfico, montagem e programação, Maya 2016 para a modelagem dos objetos low poly¹,

¹ Baixa contagem de polígonos

Mud Box 2016 para a modelagem de objetos high poly² (alta contagem de polígonos) e XNormal para otimização dos objetos para que o programa seja executado com uma taxa de FPS³ maior que 30.

A construção dos objetos foi feita com base em referências de ambientes reais. Analisando as imagens de locais fechados, comecei a construí-los no software Maya. Neste software se constrói os objetos em um formato básico, ou seja, detalhes não se aplicam nesta fase, se mantendo somente a representar as faces dos objetos de forma que o usuário esteja a uma longa distancia desse objeto ainda possa perceber sua silhueta. Exemplo na Figura 1.

Figura 1: Objeto base já com o mapa UV feito

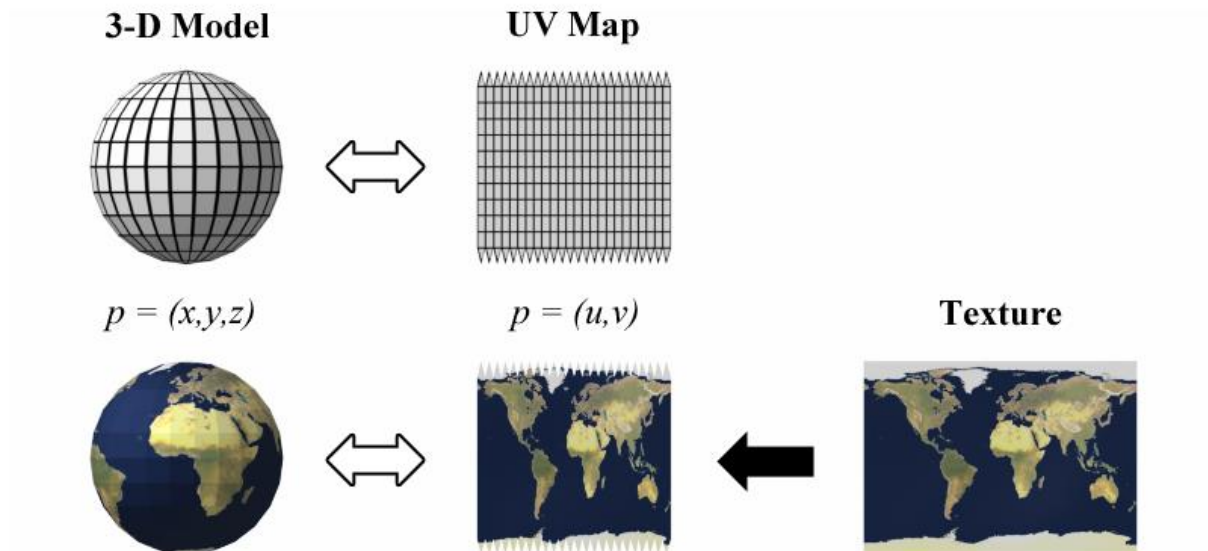


Depois é necessário extrair o chamado mapa UV que é basicamente o objeto aberto em um plano 2D como mostrado na Figura 2. Esse processo é muito importante para a qualidade final dos mapas que serão extraídos posteriormente

² Alta contagem de polígonos

³ Frames por segundo

Figura 2: Demonstração do processo teórico de extração e aplicação do mapa UV

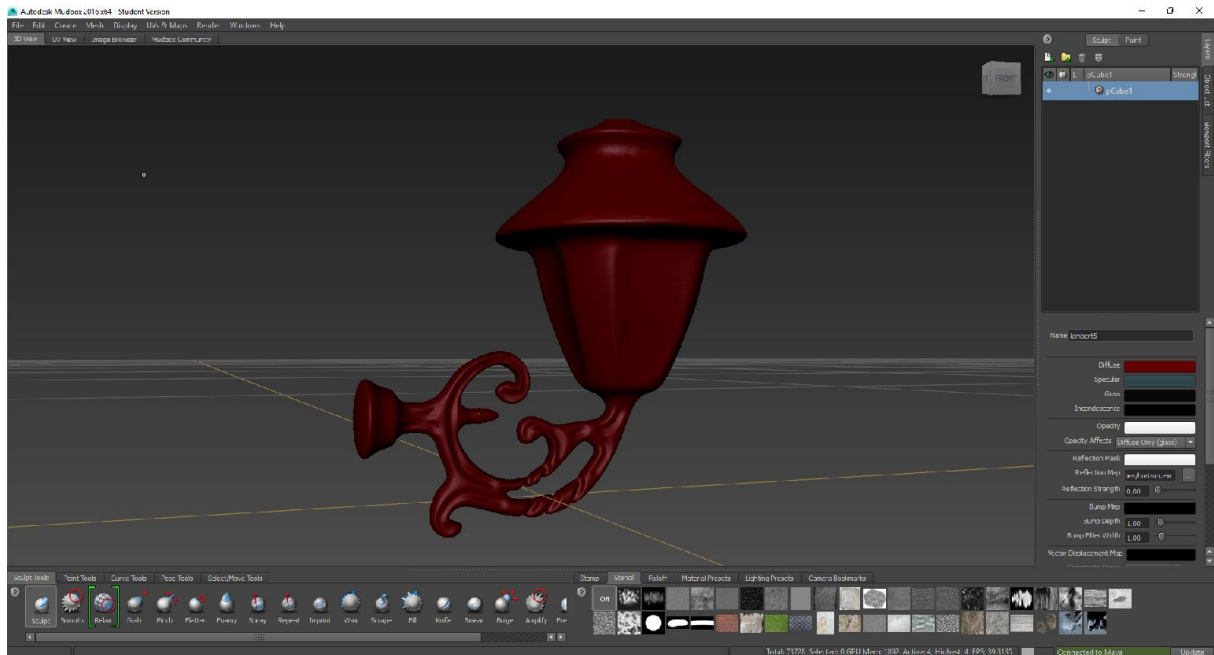


Fonte: https://en.wikipedia.org/wiki/UV_mapping

Após essa construção básica do objeto ele passa por um processo de detalhamento, este processo é feito no software Mud Box, este processo consiste em deixar o objeto mais pesado. O parâmetro para saber se um objeto 3D é utilizável/leve para uso em uma aplicação é a contagem de polígonos, o menor polígono existente é o triângulo, assim sendo a contagem é feita pelo que chamamos de “tris”. O polígono primário feito no Maya é um polígono leve e por isso é chamado low poli. Quanto mais polígonos o objeto trabalhado tem, maior é o nível de detalhe possível nele, porem mais pesado para o computador calcular.

Assim no software Mud Box é aumentada a quantidade de tris do objeto exponencialmente para que possamos dar todos os detalhes que queremos que o usuário veja no objeto em questão, por isso é importante que no objeto base só tenha polígonos de quatro vértices, pois ao aumentar a quantidade de tris o software divide um quadrado em quatro e assim sucessivamente Este processo de detalhamento é o que requer mais noção de arte de toda a produção deste projeto, pois é feito com brush/pincel. Exemplo na Figura 3.

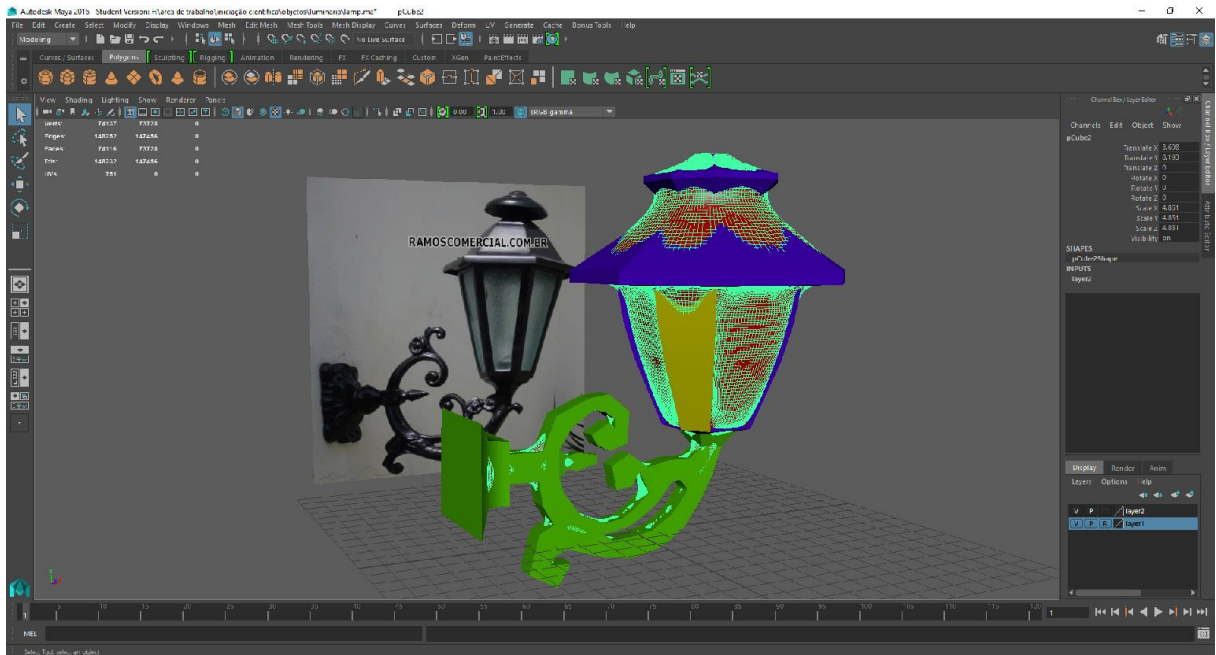
Figura 3: Objeto high poly já com detalhamento



Após dar os detalhes necessários o objeto high poly é enviado novamente para o Maya para realizar um processo chamado retopologia. Este processo consiste em diminuir a quantidade de tris do objeto high poly fazendo um mapeamento dele. É importante que ao final desse processo ele não contenha polígonos com mais de quatro vértices, porém diferente do processo anterior este pode possuir triângulos.

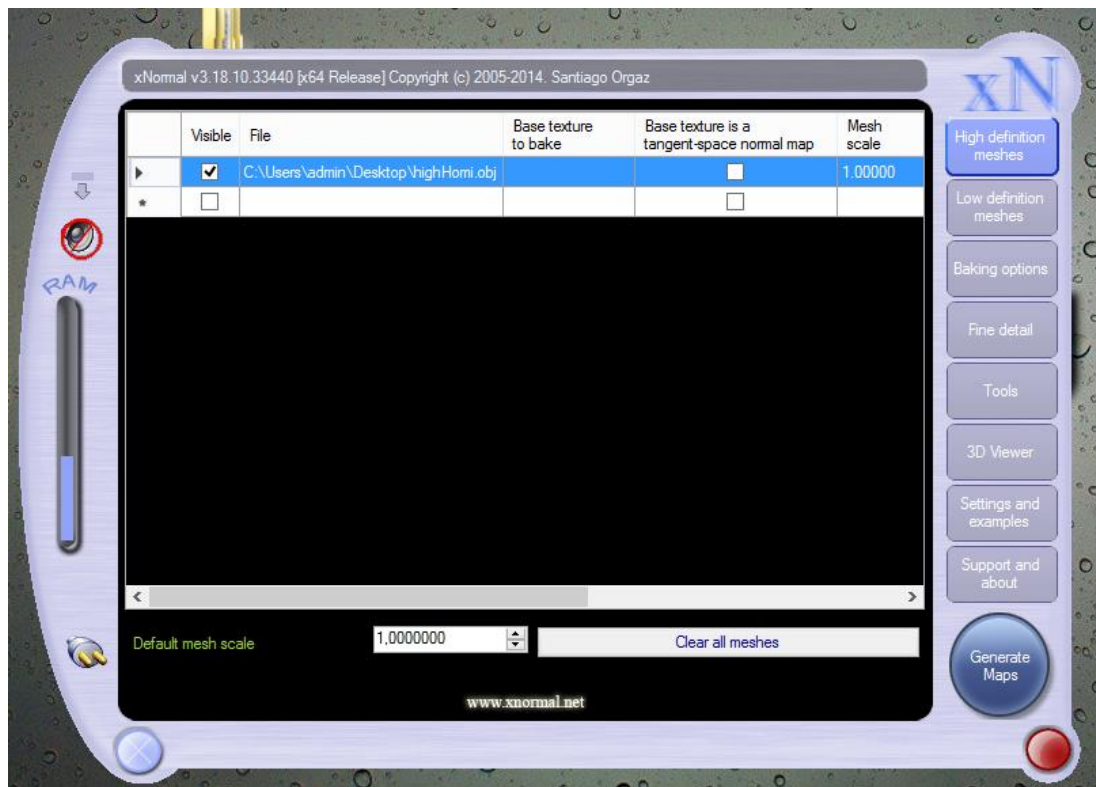
O mapeamento deverá ser parecido com o objeto original (low poly), pois precisa se ajustar formato básico do objeto, para assim extrair os mapas de UV. Agora que temos os dois objetos, precisamos sobrepor um ao outro como mostrado na Figura 4 e exportar cada um em um arquivo diferente com extensão. obj e assim passamos para o próximo passo desse importante processo.

Figura 4: Objetos high poly e low poly sobrepostos.



No software XNormal os objetos serão escaneados com base em suas UV's. Para isso temos que adicionar os objetos ao programa, cada objeto em seu lugar, o objeto high poly na aba de High definition meshes e o low poly na aba de Low definition meshes como mostrado na Figura 5.

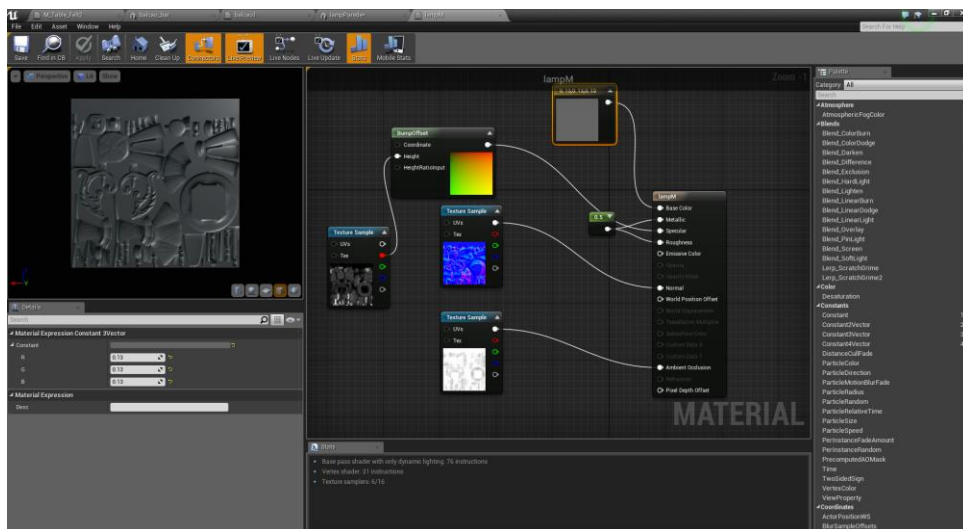
Figura 5: Software XNormal



Na aba de Baking options pediremos para extrair os principais mapas usados em objetos que são Ambient Occlusion e Normal. O programa irá compilar e extrair a maioria dos detalhes do objeto high poly e colocar em dois arquivos .tga (Targa) que são arquivos feitos para imagens 3D. Esses dois mapas serão adicionados ao objeto low poly após ele ser importado pelo Unreal.

Dentro do motor Unreal ainda é necessário criar sistemas chamado de Material que unirão essas texturas criada e aplicará efeitos a elas, melhorando a qualidade final como mostrado na Figura 6. Um único objeto pode ter vários materiais aplicado a ele, porem isso é definido na criação do objeto no Maya.

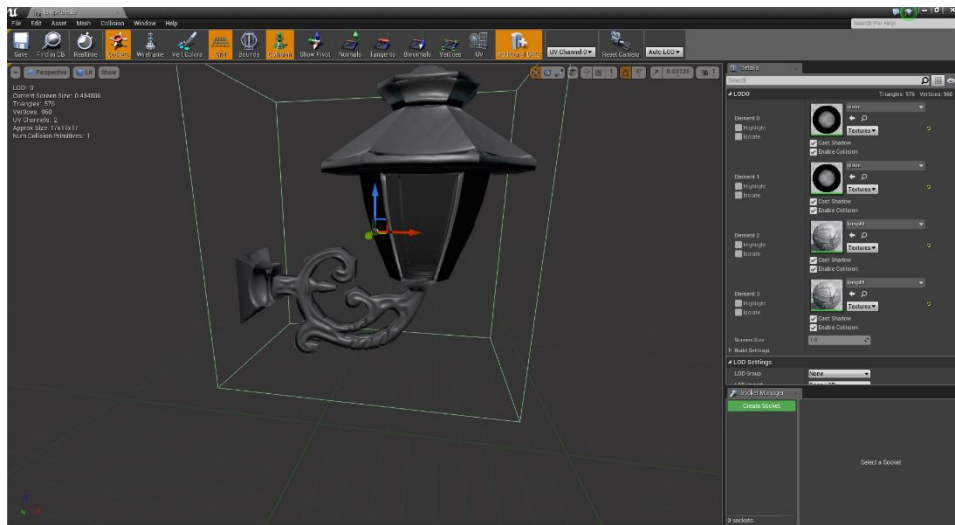
Figura 6: Sistema de texturas criado na Unreal



Dentro do motor gráfico Unreal 4 é necessário um processo que passa desde a blocagem do cenário ate a implementação de luz dinâmica e programação.

O primeiro passo foi a realização da blocagem do cenário que é basicamente pensar na proporção e como o usuário se sentira e se moverá pelo cenário, essa blocagem é feita com objetos como quadrados e esferas já prontas na Unreal. Depois os blocos colocados anteriormente são substituídos pelos objetos que serão finais no projeto, esses objetos precisam receber uma física simples, que é basicamente uma colisão que faz com que o usuário não atravesse esse objeto, Exemplo na Figura 6. As formas das colisões são objetos primitivos: quadrado, esfera, cilindro. Podem pesar tanto quanto o objeto em si dependendo da quantidade de objetos primitivos for criado como colisão.

Figura 7: Objeto final com colisão e materiais aplicados



Após o passo anterior a iluminação tem de ser trabalhada. O motor Unreal possui quatro sistemas de iluminação sendo duas delas utilizadas nesse projeto como mostrado na Figura 8. A primeira e mais importante são as Point Light que são basicamente pontos de luz como uma lâmpada, a luz se propaga para todos os lados a partir de um ponto, essa luz projeta sombras por onde passa, a intensidade das sombras são dependentes da intensidade da luz e do campo de projeção que é uma esfera que circunda o ponto 0 da luz e tem um raio de extensão. Com que qualidade essa luz agirá nos objetos depende principalmente do mapa de luz que determina qual será a resolução/qualidade da luz no objeto, o mapa de luz se encontra em cada objeto e é editável, normalmente a resolução fica entre 64 e 128 (sempre múltiplos de 2).

O segundo sistema de luz utilizado neste projeto é a Spot Light. Esse sistema propaga luz em um cone que possui um alcance, além do alcance essa luz possui intensidade assim como a outra, porém ela é usada geralmente para iluminar pontos específicos no mapa.

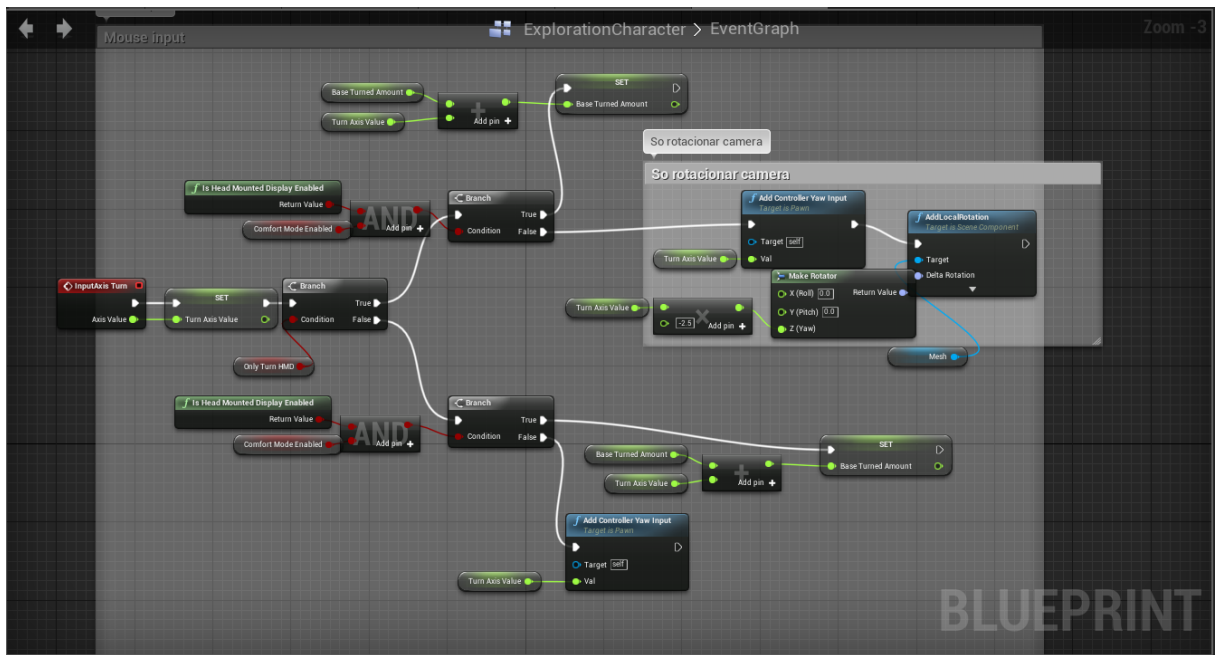
Figura 8: Os dois sistemas de luz agindo em conjunto no ambiente



Após a etapa de luz é necessário fazer uma otimização do ambiente, essa otimização é feita pelo Post Process Volume, o Box Reflection Capture e uma serie de compilações de luz, objeto e level. Destes os mais importantes são as compilações que dependendo das configurações utilizadas podem deixar mais leve ou mais pesado o produto final.

O processo de adaptação de uma simulação 3D normal para a realidade virtual foi um processo mais difícil do que o esperado, a Unreal possui um plugin para reconhecer o Oculus Rift como um hardware de entrada (a rotação) e saída (o visor) porem toda a adaptação do Oculus para a utilização do usuário neste motor gráfico foi feita por meio de linguagem visual, a Blue Print, exemplo na Figura 9. Essa linguagem visual é baseada na linguagem C++ que é vastamente utilizada em todo segmento de jogos e simulações.

Figura 9: Um trecho de código feito na linguagem visual Blue Print



Resultados e Discussão

Os resultados deste projeto são resultados práticos e por isso demonstrados por meio de imagens. Os testes foram realizados em laboratório e os resultados estão abaixo:

Figura 10: Vista geral do ambiente com Realidade Virtual



Figura 11: Vista do bar com Realidade Virtual

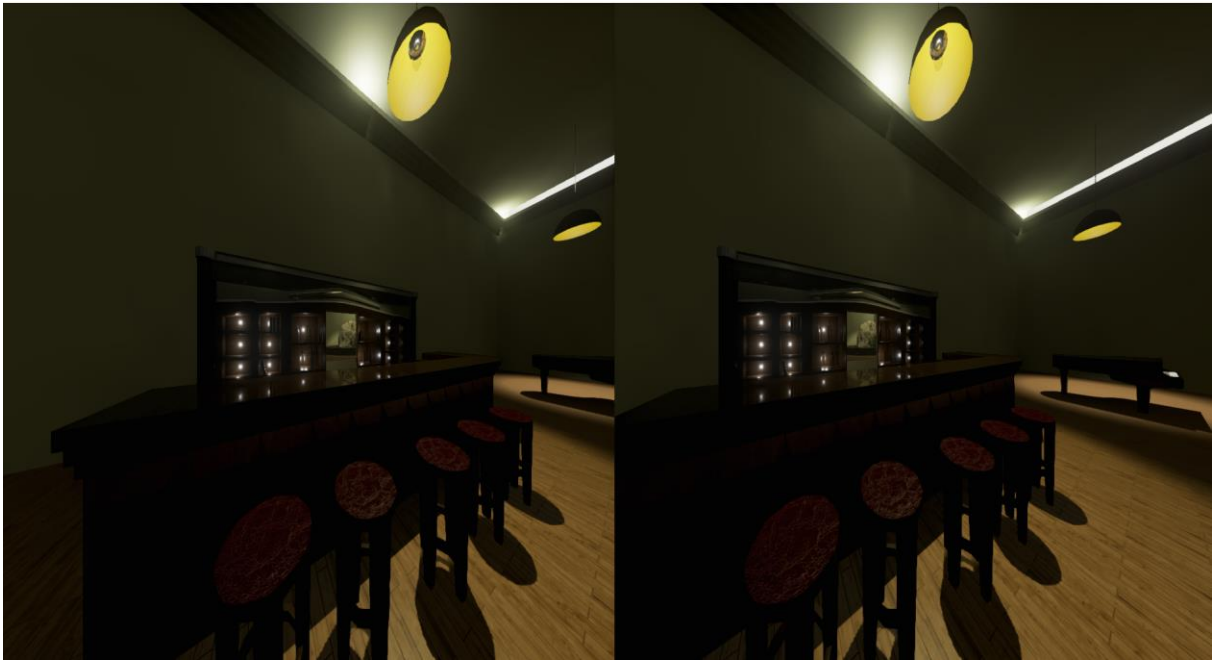


Figura 12: do salão e lustre com Realidade Virtual

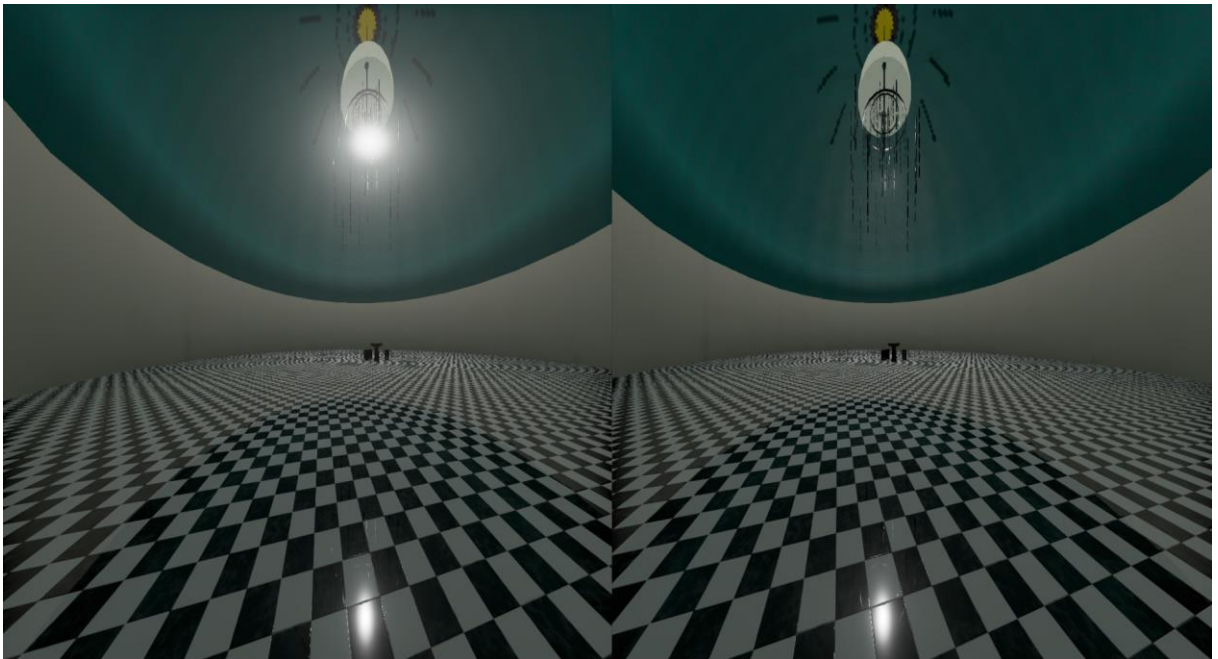


Figura 13: Vista de sofás e mesa com Realidade Virtual

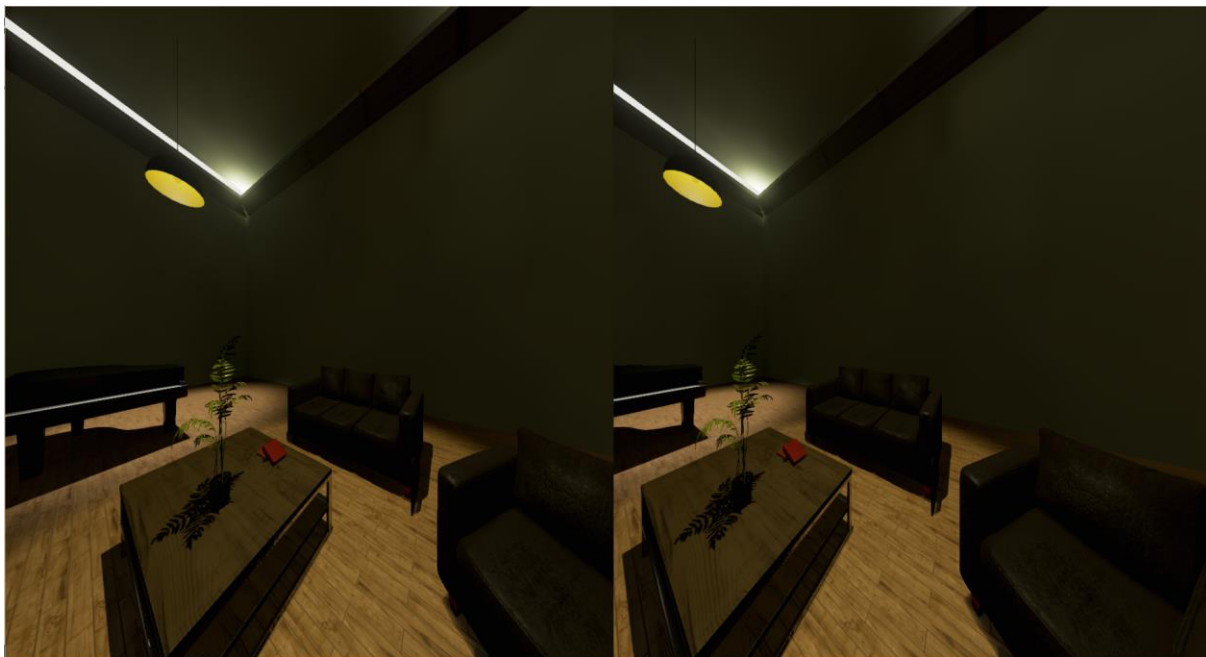
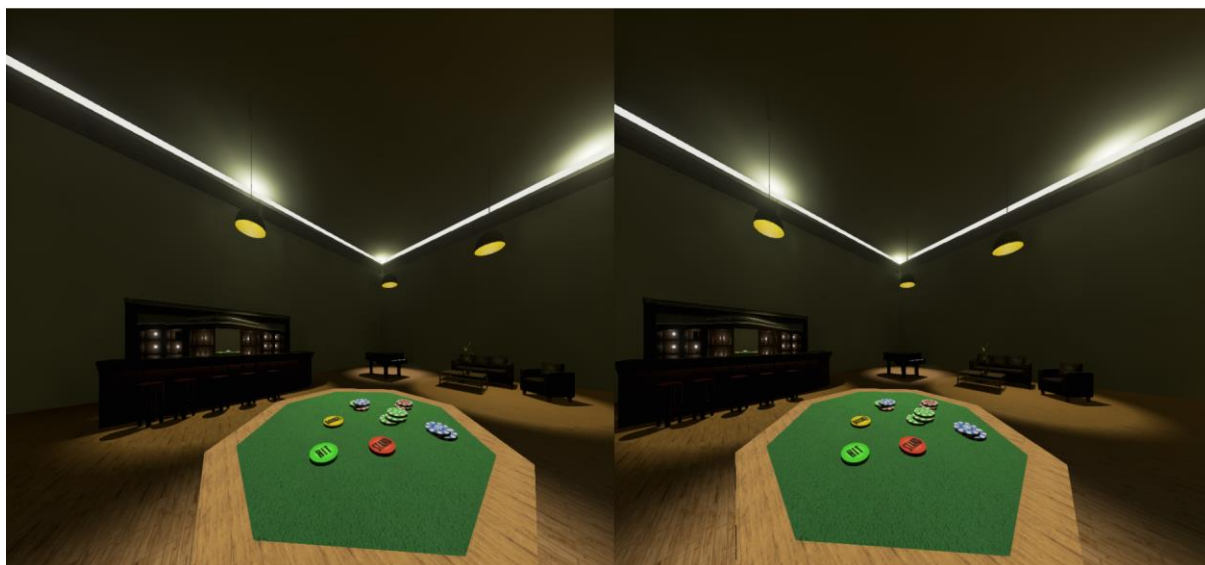


Figura 14: Vista lago e pedra com Realidade Virtual



Figura 15: Mesa de cartas



Considerações Finais

Após a finalização do ambiente e os testes realizados, foi obtido ótimos resultados relacionados a imersão visual, porém para uma imersão de maior profundidade é necessário o estímulo dos demais sentidos. Neste projeto foi concluído com êxito somente o estudo de Realidade Virtual na parte visual, portanto uma imersão parcial do usuário.

Foi concluído em determinado estágio da pesquisa que não seria possível criar um personagem do usuário que seria satisfatório para um ambiente de simulação e que mesmo assim fosse feito só diminuiria a experiência imersiva.

Quanto a parte auditiva, não foi realizado nenhum estudo pratico relacionado a esta área, porem como dito anteriormente o estímulo dos demais sentidos se faz necessário para uma experiência de imersão mais profunda, portanto é necessária para estudos futuros que buscam esse resultado.

Foi possível também que dois ou mais jogadores utilizassem o mesmo ambiente virtual, porem somente de forma local, ou seja, no mesmo computador. Isso torna a possibilidade de um sistema multiplayer⁴ local inviável na maioria dos computadores de mesa, já que a realidade virtual utiliza de pelo menos o dobro de processamento usualmente utilizado em cenas 3D comuns e como foi testado com dois usuários, o ambiente estava muito pesado e lento para tornar a experiência agradável.

⁴ Vários usuários ou jogadores

Referências

SYMPOSIUM ON VIRTUAL REALITY, 8, 2006, Belém. Fundamentos e Tecnologia de Realidade Virtual e Aumentada: Romero Tori, Claudio Kirner, Robson Siscoutto. Porto Alegre: Sbc, 2006. 412 p. Disponível em: <http://www.pcs.usp.br/~interlab/Fundamentos_e_Tecnologia_de_Realidade_Virtual_e_Aumentada-v22-11-06.pdf>. Acesso em: 05 jun. 2015.

ADAMI, Anna. Realidade Virtual. Disponível em: <<http://www.infoescola.com/tecnologia/realidade-virtual/>>. Acesso em: 17 nov. 2015.

AZEVEDO, Eduardo; LOPES, Tiago. Desenvolvimento de Jogos 3D e Aplicações em Realidade Virtual. São Paulo: Abril, 2005.

Estudo do software Maya: <http://simplymaya.com/>

Estudo do software Unreal: <https://docs.unrealengine.com/latest/INT/>

Contato: davi.b.j@hotmail.com