



SOURCE ENGINE: A INSERÇÃO DE UMA NOVA ALTERNATIVA TECNOLÓGICA AO PROCESSO DE ENSINO DE ENGENHARIA

Andre Newinski – anewciv@urisan.tcche.br

Márcio Antônio Vendruscolo – marven@urisan.tcche.br

Universidade Regional Integrada do Alto Uruguai e das Missões (URI)

Av. Universidade das Missões, 464, Universitário

98.802-470 – Santo Ângelo - RS

Resumo: *A procura por novas soluções nos campos da evolução tem conduzido inovações tecnológicas que se amplificam em uma demanda significativa e intensificada no século XXI, resultando diversificações de elementos tecnológicos e imersões de novos sistemas digitais. Este trabalho tem a finalidade de expor e detalhar uma alternativa tecnológica eficiente frente aos meios convencionais de ensino que podem ser aplicados a engenharia. A alternativa apresentada é encontrada simultaneamente em jogos computacionais, sendo aqui aplicada para o desenvolvimento de ambientes virtuais reservados ao ensino, especialmente sobre áreas tecnológicas como engenharia. Proporcionando uma forte experiência visual e atuando como uma nova e promissora forma de transmissão de conhecimento, acredita-se que esta tecnologia educacional possui maiores vantagens quanto aos processos usuais, podendo confrontar com diversas situações e dificuldades que se manifestam aos estudantes de engenharia. Aliado a isto, o projeto demonstra o processo de execução de ambientes virtuais, o qual é alcançado com a instituída tecnologia Source Engine criada pela empresa norte-americana Valve e com a aplicação da ferramenta de desenvolvimento Source Development Kit (Source SDK).*

Palavras-chave: *Tecnologias, Construções Virtuais, Simulações, 3d*

1 INTRODUÇÃO

O atual projeto revela construções de ambientes virtuais super-realísticos influenciados por importantes avanços evolutivos onde tecnologias recentes são empregadas em desenvolvimento com a engenharia moderna. A preferência da plataforma de trabalho Source SDK produzida pela empresa norte americana VALVE é relacionada à alta qualidade gráfica gerada com um custo benefício excelente. O principal objetivo estabelecido para este projeto foi de verificar a aplicabilidade do software através da construção de partes importantes do Campus da URI de Santo Ângelo, incluindo áreas compostas por prédios e os ambientes que os contornam. Os ambientes virtuais propostos implicaram em diversos tipos de atividades que são de extrema importância a este trabalho, e foram realizados com o máximo de



aproveitamento, seguidos por seus métodos conhecidos através de pesquisas. A qualidade obtida pelo projeto é evidenciada não só pelo que a plataforma nos oferece, mas também pela busca de softwares alternativos. Os resultados esperados pelo projeto foram atingidos com a criação virtual dos prédios onde se encontram os laboratórios das Engenharias da Universidade, promovendo e potencializando o uso dessa aplicação como tecnologia educacional. Por outro lado, a importância deste projeto não é restrita somente ao uso da tecnologia na área de estudos, mas idealiza engajar o projeto através de propostas que se fazem necessárias para divulgação da Universidade. Introduzindo a sociedade, subprodutos retirados dos ambientes virtuais desenvolvidos (imagens e vídeos), os quais favoreçam uma boa aceitação e reconhecimento do projeto.

As pesquisas na área de tecnologias educacionais atravessam um período em que a diversidade tecnológica se destaca. Em especial, o uso da informática na educação descortina um imenso campo de possibilidades, sendo possível destacar um campo específico neste segmento, o dos jogos digitais. O uso de games na educação tem sido objeto de diversos estudos. Sugere-se, a este respeito, que o ensino de ciência e tecnologia, e especificamente, o ensino de engenharia, no Brasil, pode se beneficiar com a pesquisa sobre o potencial desse recurso como apoio ao processo educacional (CINTED-UFRGS, 2006).

Com base nesta justificativa acredita-se que novas alternativas desenvolvidas por estas ferramentas possam contribuir ao progresso educacional de engenharia tanto em cenários regionais, nacionais como internacionais.

2 O AMBIENTE VIRTUAL

A *Source Engine* é uma tecnologia recente e promissora presente em alguns dos mais conhecidos títulos de jogos digitais da atualidade, podendo ser facilmente empregada para visualização de ambientes virtuais em qualquer jogo sobre ela programado. A plataforma de trabalho pode ser definida como um conjunto de ferramentas utilizadas no desenvolvimento de soluções que atuam em compatibilidade com essa tecnologia. A aquisição dos ambos se deu através do sistema comercial *steampowered* da norte-americana VALVE. A plataforma *Source SDK*, é responsável por várias ferramentas para se desenvolver mapas, modelos e outras aplicações de acordo com a tecnologia *Source Engine*, a qual que vem sendo excessivamente usada para modificações (criação de *mod-games*). Pela razão de que todo modelo de *map* (ambiente virtual) a ser criado requer um *mod-game* para ser executado, adquiriu-se o *Counter-Strike: Source*. A aquisição é vantajosa em um ponto de vista no qual não exista necessidade de se desenvolver um *mod-game*, não significando suprimir sua existência, mas sendo uma alternativa rápida e eficiente para acessar o ambiente virtual desenvolvido neste trabalho. O principal *software* presente no *Source SDK* e de influência para esse trabalho é o Valve Hammer Editor, sendo o único envolvido em todas as partes do processo de criação dos ambientes virtuais. Sendo assim o desafio é desenvolver de um ambiente virtual com a proposta de expor todos os procedimentos possíveis e existentes neste processo. Sendo o local de estudo, os Laboratórios de Engenharias da URI Campus de Santo Ângelo.



3 O PROCESSO CONSTRUTIVO DO AMBIENTE VIRTUAL

A situação inicial do projeto conduziu a uma exploração sobre o ambiente físico pelo qual se encontrava os Laboratórios de Engenharia da URI, fazendo o reconhecimento de todos os objetos encontrados e suas respectivas dimensões. O objetivo desta etapa inicial é conservar os padrões geométricos dos elementos posteriormente criados pelo *software*. As sucessivas etapas são aplicadas sobre estes elementos e ao espaço virtual carregado.

O *software Valve Hammer Editor 4.x* fornecido como um componente do *Source SDK* é considerado ferramenta oficial para a construção de ambientes arquitetônicos para games e *mod-games* baseados na *Source Engine* (Figura 1). O processo de geração de ambientes virtuais sobre o *Hammer* não é difícil, porém complexo. A infinidade e diversidade dos elementos visíveis no ambiente real resultam um desenvolvimento considerado e cuidadoso sobre os objetos gerados virtualmente. Com a finalidade de representar melhor isto, adotou-se um sistema geral efetivo objetivando discernir tais elementos. São alguns tipos, objetos estáticos e dinâmicos, materiais simples e compósitos, entidades ambientais que estabelecem interação entre o observador e o ambiente virtual e os elementos especiais que podem ser aplicados a um objeto, não ser um objeto, mas ter influência sobre ele impondo características especiais necessárias.

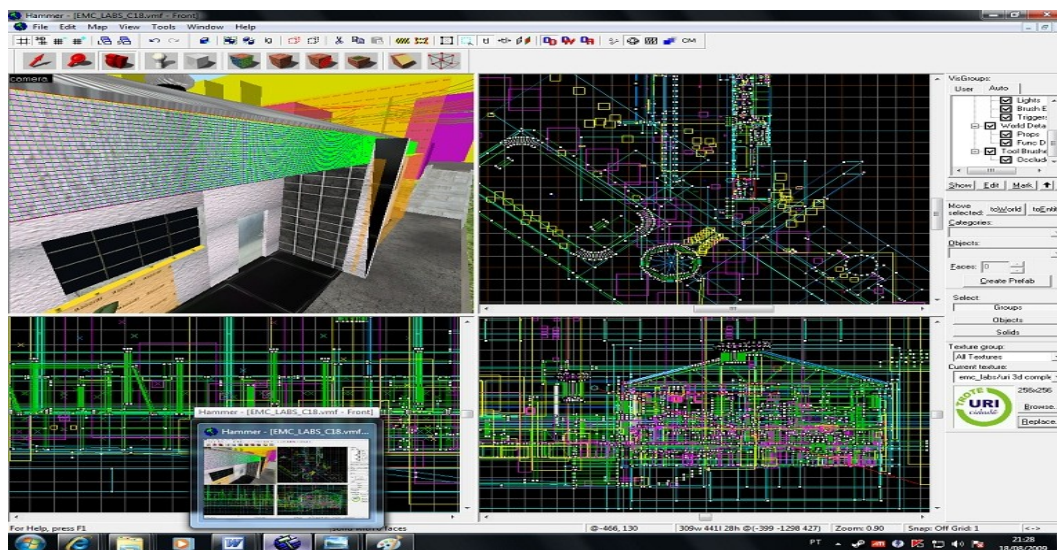


Figura 1 – O *software Valve Hammer Editor 4.x*



3.1 Do Sistema Geral Efetivo

Os materiais simples podem ser definidos como texturas. É sua função atribuir diferentes aspectos aos objetos criados. Neste caso uma textura simples pode ser escolhida através da base de texturas do *Hammer* ou ser uma imagem convertida para a extensão “vmt” através do software *VTF Edit*. Ao longo do estudo foram tiradas fotografias complementares e a partir delas obtidas uma variedade de materiais simples que são aplicados na criação de diversos objetos desenvolvidos no ambiente, são exemplos, estruturas de concreto e aço, equipamentos, tipos de solos e materiais diversos. Os materiais compósitos são texturas que podem causar um efeito de tridimensionalidade, como também capacidade de refletir a iluminação, parecendo muito mais realistas e gerando várias percepções quando o navegador se movimenta aos seus contornos. São aplicadas no *Hammer* pela solicitação da instrução *toggle texture application*.

Os elementos desenvolvidos até então podem integrar sistemas estáticos ou dinâmicos. Os sistemas estáticos podem ser definidos como um conjunto de objetos que podem ser encontrados em estado de inércia. Os objetos dinâmicos por sua vez estão em movimento constante. Sistemas de exaustão e insuflamento de ar são casos visíveis no projeto e estão relacionados ao uso da entidade *func_rotating*, extremamente necessária para fazer o objeto adquirir propriedades de funcionamento idêntica a de um ventilador cuja função é girar em torno do eixo. Objetos com uso de entidades primárias e secundárias exemplificam o uso de portas com sensores. Um caso específico é o uso da entidade *func_trigger* (sensor) que é interligada a entidade *func_door_rotating* (porta) sendo possível acionar o movimento da mesma pela passagem do observador sobre o sensor. Como qualquer entidade presente no *Hammer*, elas podem ser aplicadas pela solicitação da instrução *entity tool*.

Um passo necessário para o ambiente virtual é a definição de uma forma física que limitará o universo do ambiente onde internamente a ele constroem-se os elementos. Essa limitação deve ser feita por um conjunto de objetos sob os quais se aplicam as texturas *skybox*. Isto define à qual atmosfera emerge-se sobre ela a aplicação de entidades ambientais. Essencialmente nesta fase consiste em inserir um sistema de iluminação geral para o ambiente onde é relevante a inserção da entidade *light_environment*, sendo as opções de seu brilho controladas através do sistema *red green blue (rgb)* e um fator multiplicativo alterável nas propriedades da mesma. Como um reforço para isto prossegue o uso da entidade primária que representa o sol, denominada por *env_sun*, onde age paralelamente à entidade secundária *info_target*, a qual referencia a posição na qual o observador tem maior efeito sobre o sol. Pontos de iluminações específicos (artificiais) requerem o uso da entidade *light* localizadas próximas aos objetos que irão receber o efeito desejado. Neste trabalho foram necessárias as aplicações dessa entidade devido à presença de ambientes escuros onde o sistema de iluminação geral não é atingido ou tem efeito reduzido. Finalizando os elementos básicos necessários (Figura 2) para suprir o processo de compilação aplica-se a entidade *info_player_start* a qual define a posição específica e a direção para qual um ou mais observadores se encontram no primeiro momento da simulação 3D.

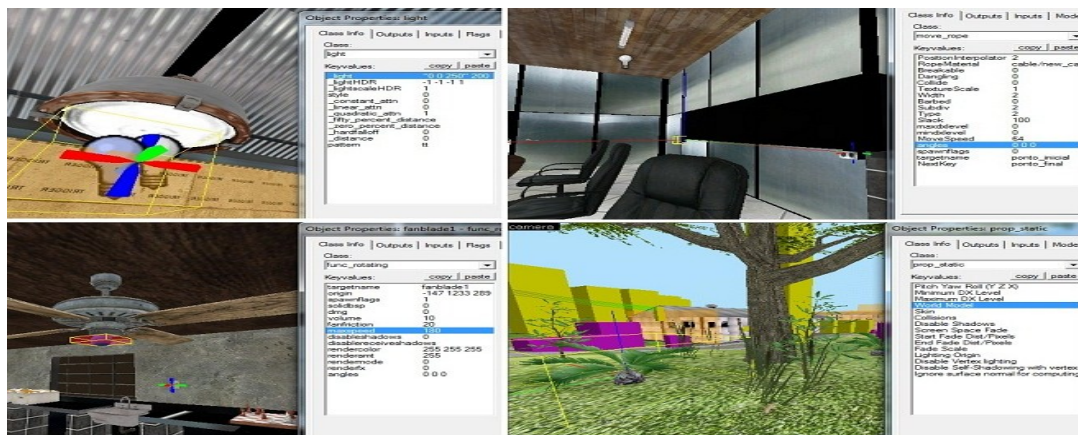


Figura 2 – Representação do sistema geral efetivo dos elementos.

3.2 Processo de Compilação

O processo de compilação resulta da passagem de uma conhecida extensão de arquivo para uma seguinte. Essa nova extensão gerada é interpretada pelo *mod-game* posteriormente. A compilação inicia-se pela solicitação no software Valve Hammer Editor sendo iniciada a transferência da extensão *valve map file (vmf)*, a qual carrega as informações contidas sobre o ambiente virtual que representa os Laboratórios de Engenharia da URI para a criação da nova extensão, a *binary space partition (bsp)*. Sendo a ultima compatível com qualquer *mod-game* programado sobre a *Source Engine*. Ao carregar o ambiente no *mod-game* Counter-Strike: Source verificou-se a presença do novo arquivo compilado possibilitando a visualização do ambiente virtual.

3.3 Otimização Final

Após a execução do *mod-game*, o qual possibilita a exploração do ambiente, existe um conceito de influência no momento da visualização dos elementos e animações que compõem o *level*. Neste trabalho foi necessário compreender os efeitos causados por *frame rate*, uma medida da frequência em que um dispositivo de processamento de imagens processa sobre um determinado período cuja unidade de medida é o *frames per second* ou *fps*.

Os dispositivos de processamento gráficos usados estão presentes na maioria dos *personal computers (pc)* ou sistemas que integram hardwares importantes como uma *central processing unit (cpu)*, *graphics processing unit (gpu)* e *physics processing unit (ppu)*. A taxa de *frame rate* não é somente afetada pela quantidade de elementos e efeitos presentes no ambiente virtual, mas também depende do equipamento a ser



utilizado para processá-la. O propósito de mencionar estas informações tão importantes está conectado a necessidade da taxa de *frame rate* não causar *latency* devido a um dimensionamento falho do ambiente. O termo *latency* em simulações 3D é relacionado a um momento de atraso exibido entre cada *frame* processado pelos dispositivos de processamento quando executam movimentos solicitados pela simulação 3D, indiferentemente de a mesma ser observada em dispositivos de visualização como monitores de vídeo, retro projetores ou televisões digitais.

Fluxos de imagens superiores a uma faixa de 60 *fps* possuem valores de *latency* insignificantes e praticamente impossíveis de perceber a variação, contestada a capacidade de visão humana de enxergar acima dessa frequência, sendo valores otimizados para que as operações de simulações 3D fluam corretamente. Porém, quanto menor esse fluxo de imagens maior será a latência, sendo a aplicação 3D lesada, causando sensações desagradáveis durante a simulação. A verificação do *frame rate* é identificado em duas imagens alusivas aos ambientes virtuais dos laboratórios de engenharias presentes na “Figura 3”, nesta situação os dispositivos de processamento apontam duas taxas (40 *fps* e 60 *fps*).



Figura 3 – Imagens do *frame rate* do ambiente virtual

Constata-se então sobre algumas áreas a incidência de um problema de *latency*, provavelmente originado por imposição em excesso de elementos construtivos no ambiente. Em consequência disso, verificou-se a necessidade de adotar um processo de redimensionamento dos elementos e apresentação de novas medidas frente à sobrecarga causada sobre os dispositivos de processamento. Como alternativa para eliminar alguns efeitos desta ordem foram utilizados alguns métodos envolvendo elementos especiais, os quais são intensamente eficientes na melhora das condições dos fluxos de imagens processados.

3.3.1 Métodos de Aperfeiçoamento

O primeiro método consiste no uso da textura especial *nodraw* que consiste em eliminar o processamento de áreas localizadas em partes não visíveis de construções. A “Figura 4” demonstra que todos os objetos desenvolvidos recebem um efeito de



polígonos sob suas faces. A quantidade de polígonos existentes no ambiente afeta diretamente o desempenho dos dispositivos de processamento gráfico.

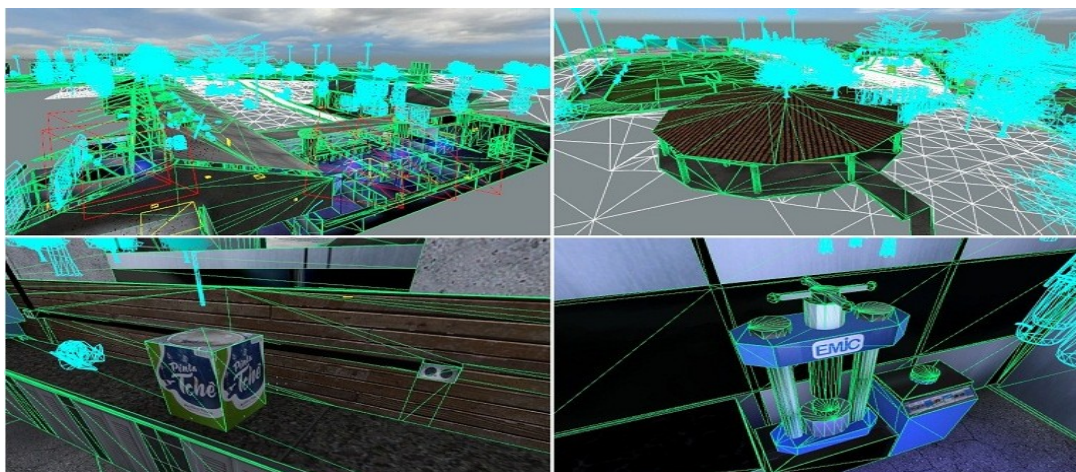


Figura 4 - Efeito dos polígonos sobre um ambiente virtual desenvolvido.

O segundo método tem a função de corrigir o problema causado pelo contato direto entre dois ou mais objetos resultando polígonos excessivos no processamento onde a retificação deve consistir de um espaçamento mínimo entre os mesmos. Um terceiro método que também pode ser empregado para esse caso consiste em atribuir a entidade especial *func_detail* para cada objeto criado no ambiente e otimizando esse efeito. A “Figura 5” evidencia a situação exposta.

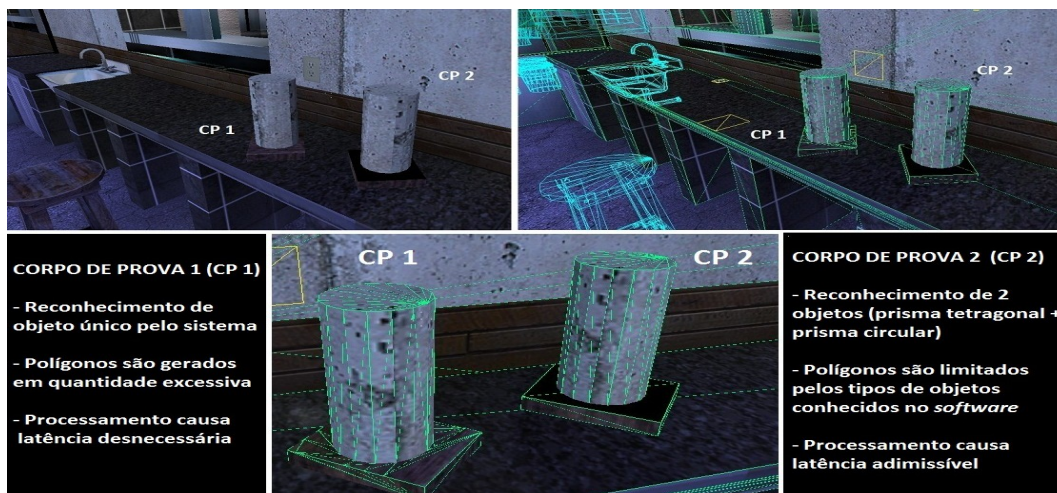


Figura 5 – Método da entidade especial *func_detail*.



Um quarto método consiste em esconder o processamento dos *models* (Figura 6) em áreas que se encontram além do campo de visão do observador. Os *models* são modelos de objetos pré-existentis que são inseridos no cenário por meio das entidades *props*. A aplicação do método é tida pela criação de um elemento geométrico e nele aplicado o uso da entidade especial *func_occluder* com a atribuição da textura *trigger* em suas faces transversais.

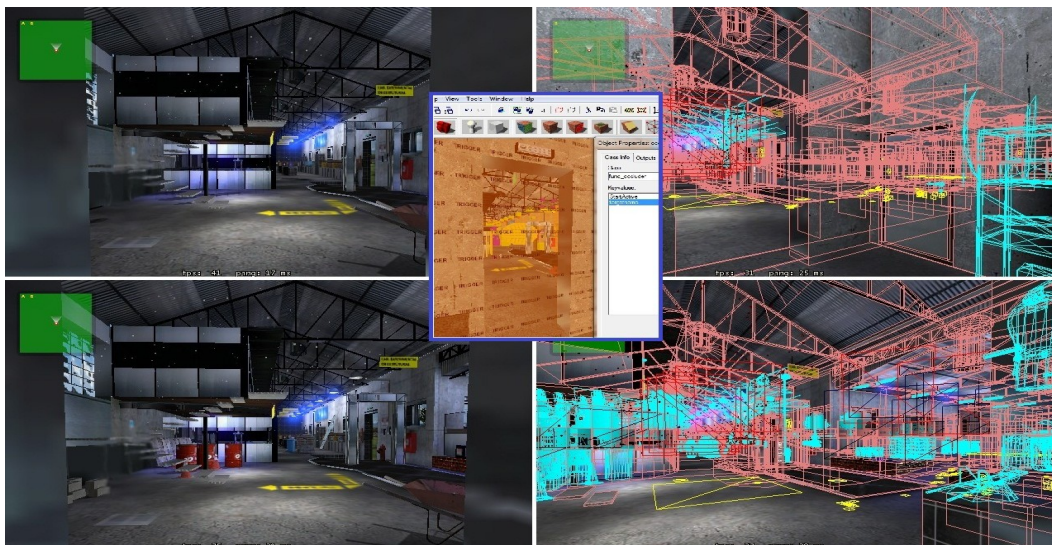
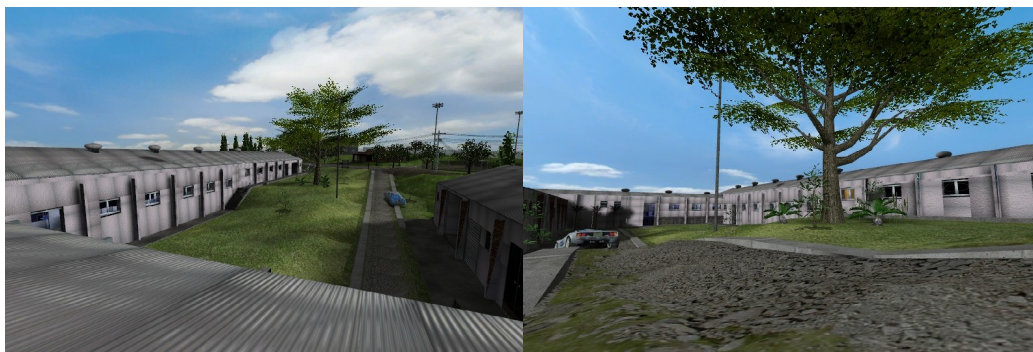


Figura 6 – Eliminação do processamento de *models* em áreas não visíveis do ambiente.

4 RESULTADOS

Subprodutos como imagens dos *levels* desenvolvidos neste trabalho introduzem uma nova forma de transmissão que pode também ser usada para elucidar seu progresso, conseqüentemente, suprimindo alguns dos resultados esperados. A “Figura 7” ilustra este propósito.



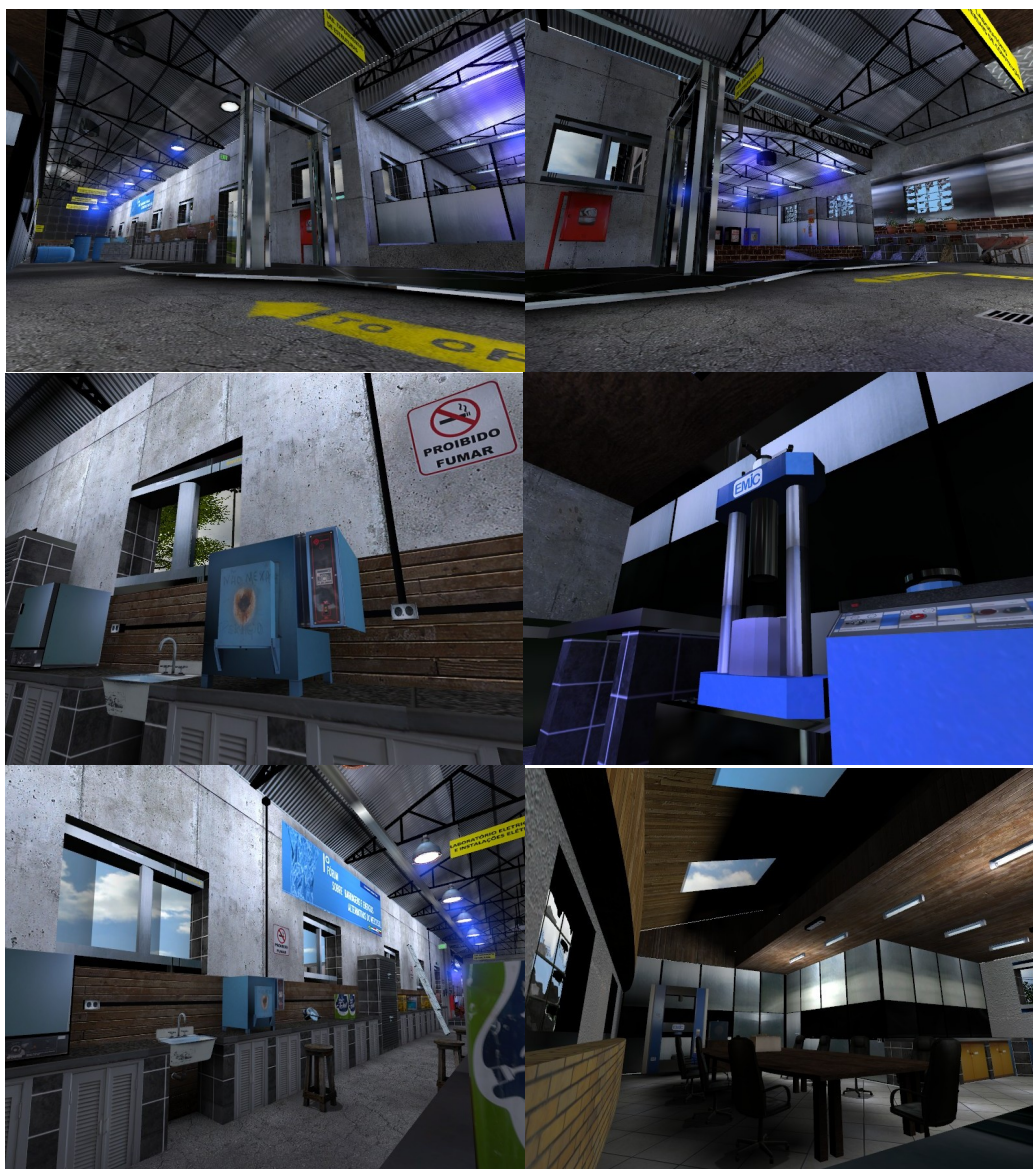


Figura 7 – *Level* final dos Laboratórios de Engenharias da URI - Campus Santo Ângelo

5 CONSIDERAÇÕES FINAIS

Com este trabalho foi possível o desenvolvimento de um ambiente virtual especificadamente para a área de engenharia aplicando-se um software de desenvolvimento de jogos digitais. Esta ferramenta demonstrou-se eficiente para a finalidade proposta onde foi possível reproduzir com grande realismo os *levels* construídos.



REFERÊNCIAS BIBLIOGRÁFICAS

Internet

RIBEIRO, L. O. M.; TIMM, M. I.; ZERO, M. A. **Citação de referências e documentos eletrônicos.** Disponível em: <http://www.cinted.ufrgs.br/renote/jul2006/artigosrenote/a36_21203.pdf> Acesso em: 15 ago. 2008.

Internet

VALVE. **Citação de referências e documentos eletrônicos.** Disponível em: <http://developer.valvesoftware.com/wiki/SDK_Installation> Acesso em: 2 set. 2008.

Internet

VALVE. **Citação de referências e documentos eletrônicos.** Disponível em: <<http://developer.valvesoftware.com/wiki/Category:Hammer>> Acesso em: 17 set. 2008.

Internet

VALVE. **Citação de referências e documentos eletrônicos.** Disponível em: <[http://developer.valvesoftware.com/wiki/Optimization_\(level_design\)](http://developer.valvesoftware.com/wiki/Optimization_(level_design))> Acesso em: 22 fev. 2009.

SOURCE ENGINE: AN INSERTION OF A NEW ALTERNATIVE TECHNOLOGY TO ENGINEERING PROCESS OF INSTRUCTION

Abstract: *The search for new solutions in the fields of evolution leads us to realize that technological innovations are expanding at a significant demand and intensified in the 21st century, resulting diversification of technological elements and embeddings of new digital systems. This article makes up a project with the aim of exposing and detailing an alternative efficient technology compared to the conventional means of teaching that can be applied to engineering. The alternative presented is found both in computer games, and here applied to the development of virtual environments reserved for teaching, especially on technology areas such as engineering. Providing a strong visual experience and acting as a promising new way of transmitting knowledge, believes that educational technology has major advantages when compared with usual procedures, and may confront different situations and difficulties that arise for students of engineering. Allied to this, the project demonstrates the process of developing levels with the establishment of Source Engine technology powered by the U.S Valve Corporation and implementation of its development tool Source Development Kit (Source SDK).*

Keywords: Technologies, Level design, Simulations, 3d