ESTUDO COMPARATIVO DE SOLUÇÕES PARA O DESENVOLVIMENTO DE UM LABORATÓRIO VIRTUAL 3D INTERATIVO

Heitor Silva (UFSCar – heitor.uabitapevi.si2@gmail.com)
Joice Lee Otsuka (UFSCar – joice@ufscar.br)
Diogo Augusto Gonçalves (UFSCar – diogoatgs@gmail.com)
Paulo Roberto Montantaro (UFSCar – montanaro@ufscar.br)

Grupo Temático 4. *Inovação em Educação e Tecnologias Digitais*
Subgrupo 4.4 *Jogos e outras mídias interativas na educação: desafios e possibilidades*

Resumo:
Este trabalho apresenta uma comparação entre duas soluções para o desenvolvimento de laboratórios virtuais 3D interativos: o OpenSimulator e o Unity 3D. O estudo comparativo foi realizado por meio da implementação de um tour virtual e um pequeno experimento em um Laboratório Virtual de Química. São apresentados os desafios, as limitações e as facilidades de cada solução, considerando os requisitos levantados para o Laboratório Virtual de Química.
Palavras-chave*: laboratório virtual, jogos digitais, jogos educacionais, recursos educacionais abertos.*

Abstract:
This paper presents a comparison between two solutions for the development of interactive 3D virtual laboratories: OpenSimulator and Unity 3D. The comparative study was carried out by implementing a virtual tour and a small experiment in a Virtual Chemistry Laboratory. We present the challenges, constraints and features of each solution, considering the requirements raised for the Virtual Chemistry Laboratory.
Keywords*: virtual laboratory, digital games, educational games, open educational resources.*

1. Introdução

O desenvolvimento de atividades práticas de laboratório é um desafio em cursos oferecidos na modalidade a distância e, muitas vezes, também em cursos presenciais. A dificuldade de acesso aos laboratórios físicos frequentemente inviabilizam atividades práticas que requeiram o uso de equipamentos e materiais específicos. Nesse contexto, tornou-se premente a investigação de alternativas para a criação de laboratórios virtuais interativos, que possibilitem a prática virtual de procedimentos antes possíveis apenas em laboratórios reais.

Pesquisas sobre ambientes virtuais altamente interativos, como os laboratórios virtuais, indicam que estes recursos são apropriados e podem ser bastante efetivos para programas formais de aprendizagem (Aldrich 2009, Squire *et al*. 2004). Alguns argumentos

apresentados por Aldrich (2009, p. 5-6) para justificar a efetividade desses recursos são: (i) o emprego de elementos de jogos como ferramenta de aprendizagem, instigando e motivando o envolvimento dos aprendizes; (ii) a criação de contextos significativos e similares aos reais, onde o conhecimento construído poderá ser aplicado; (iii) a possibilidade de participação ativa do estudante, o que é essencial para uma aprendizagem efetiva. Além disso, em um ambiente simulado os estudantes podem explorar os recursos de forma segura, sem medo de acidentes.

A presente pesquisa foi desenvolvida com o intuito de investigar e testar soluções que possam ser utilizadas para o desenvolvimento de um laboratório virtual 3D interativo. A utilização de objetos 3D foi adotada para promover experiências mais imersivas, permitindo a exploração do espaço, bem como todo o universo diegético que o envolve. Dessa forma, os equipamentos modelados em 3D, substâncias e outros instrumentos para o laboratório virtual são objetos de estudos dos participantes, que poderão conhecer as principais características de cada um a partir da interação virtual, assim como da proposta didático-pedagógica criada para este projeto. Este trabalho integra uma proposta mais ampla que vem sendo desenvolvida em parceria com docentes da área de Química dos cursos de Engenharia Ambiental e Tecnologia em Produção Sucroalcooleira da UFSCar (ambos oferecidos na modadalidade EaD) , visando a criação de um laboratório virtual de química, o LabTecA - Laboratório Técnico Analítico de Química.

Este projeto está alinhado com outros projetos desenvolvidos pelo grupo de produção de materiais didáticos interativos, que têm como diretriz o desenvolvimento de recursos educacionais abertos (REA) (Educação Aberta, 2013) (Otsuka et. al. 2012), buscando soluções que favoreçam o reuso, revisão, redistribuição e remixagem dos recursos produzidos. Todos os roteiros, modelos 3D, ilustrações, trilhas e códigos produzidos serão compartilhados com a comunidade sob a licença Creative Commons (Creative Commons, 2014).

Este artigo apresenta os resultados de um estudo que buscou a identificação das características mais relevantes de *engines*[1] para o desenvolvimento de laboratórios virtuais 3D interativos. Na próxima seção é apresentada a metodologia do trabalho e nas seções seguintes são apresentados os principais resultados obtidos nas etapas de levantamento bibliográfico, desenvolvimento de experimentos e análise das soluções.

2. Metodologia

Conforme Stephen Alesse e Stanley Trollip (2009), o uso de jogos pode motivar estudantes, fazendo da aprendizagem uma experiência mais instigante e agradável. A proposta de criação do laboratório Virtual de Química tem por objetivo não só simular um ambiente de laboratório, mas criar um ambiente de Jogo Educacional em que o usuário se sinta desafiado e motivado. Para isto foram seguidas algumas etapas.

A primeira etapa deste trabalho consistiu em um levantamento bibliográfico nas áreas de jogos educacionais, simulações, mundos virtuais, objetos de aprendizagem, recursos educacionais abertos. Alguns resultados desta etapa são apresentados na seção 3. Também houve um trabalho de pré-produção audiovisual, com a formação da equipe responsável, planejamento de áreas e sub-áreas envolvidas nesse processo, tais como criação visual, trilha sonora, roteiros específicos, programação, dentre outras.

[1] Neste artigo estamos considerando os *graphic engines,* que são softwares ou conjunto de bibliotecas especializadas para a renderização de ambientes 3D.

A segunda etapa teve como objetivo o levantamento de requisitos do laboratório, por meio de reuniões com a professora de Química que colabora com este projeto e visitas a laboratórios reais, coletando material para a recriação virtual de espaços, dispositivos e objetos específicos. A partir dos requisitos e especificações de equipamentos e experimentos realizados pela professora, foi desenvolvido um Game Design Document (GDD), documento que detalha as características do jogo (tais como a interface, modelos 3D, arte), unificando todo o processo para a equipe envolvida no projeto. O GDD do LabTecA reúne elementos pedagógicos apresentados pela professora, com elementos de jogos e simulação, visando a criação um recurso pedagógico e lúdico que promova experiências de aprendizagem interessantes e eficientes aos estudantes.

A terceira etapa consistiu na elaboração de modelos 3D para todos os equipamentos, móveis e cenários internos e externos do laboratório. Para a modelagem foi utilizado o software Blender[2]. Também foram elaboradas texturas e aplicados materiais para todos os modelos, permitindo a sua finalização e ajuste gráfico.

Na quarta etapa foram realizados testes de *engines* e ambiente de desenvolvimento para jogos e simulações. E, a partir dos objetos 3D e o GDD, foram desenvolvidos os primeiros testes de implementação do laboratório virtual utilizando duas soluções selecionadas: o Unity 3D[3] e o OpenSimulator[4], definindo também questões estéticas diferentes para cada uma das plataformas. Os principais resultados desta etapa são apresentados na seção 4.

Na quinta etapa foram definidos os critérios para a análise das soluções de desenvolvimento e foram realizadas as análises das duas soluções utilizadas na quarta etapa. Os resultados desta etapa são apresentados na seção 5.

3. Trabalhos relacionados

As últimas três edições do NMC HorizonReport[5] destacam tópicos relacionados a jogos na Educação como tecnologia emergente de grande impacto para a Educação. A Edição 2012 do NMC HorizonReport afirma que jogos podem desenvolver a capacidade de resolver problemas, de colaboração e socialização fazendo com que os estudantes se sintam motivados e se inteirem facilmente do conteúdo da disciplina. É importante ressaltar que grande parte dos atuais estudantes do ensino médio e graduação já chegam a estes cursos com muitas horas de experiência em jogos (Johnson et. al. 2012).

Outras pesquisas com o uso de jogos educacionais têm sido empreendidas para apoiar a formação de disciplinas específicas. Em (Seabra et al. 2012) é apresentada uma ferramenta didática 3D para o ensino e aprendizado de disciplinas com conteúdo de Geometria Descritiva no ensino de Engenharia. Permite a execução de técnicas de construções geométricas, outrora manuais, em um ambiente 3D contribuindo para a aprendizagem de Geometria Descritiva. O uso da ferramenta motivou os estudantes e facilitou a dinâmica de desenho e ensino da disciplina.

Em (Bispo *et al*. 2012) é apresentado um jogo educacional aberto disponibilizado na internet no formato de um REA para o ensino e aprendizado de conceitos de Toxilogia. O

2 http://www.blender.org/
3 http://unity3d.com/pt
4 http://opensimulator.org/
5 Relatório anual elaborado pelo New MediaConsortium em colaboração com EDUCAUSE Learning Initiative, que visa apontar as tecnologias que causarão impacto na educação em um horizonte de 5 anos

jogo, chamado Daphnia World, em que o jogador controla um animal que busca vencer desafios ambientais a fim de chegar a seu local de reprodução. O jogo despertou o interesse e a motivação dos estudantes na assimilação dos conceitos da disciplina de Toxilogia Ambiental.

Em (Machado *et al*. 2011) é apresentado um jogo educacional 3D voltado ao ensino médio que permite o controle de um personagem 3D que ao responder perguntas (de conteúdos que podem ser cadastrados por professores), tem acesso a ambientes onde é possível encontrar "cristais" e acumular pontos. O trabalho demonstrou que é possível integrar uma ferramenta de ensino a um ambiente de jogo 3D colaborativo.

Em (Rocha *et al*. 2010) foram comparadas diversos *engines* 3D open source. As principais funcionalidades esperadas pelos *engines* (Blender, Crystal Space, Delta3D, Irrlicht, jMonkey, Ogre3D, OpenSceneGraph, Panda 3D) foram divididos em grupos (recursos gráficos, não-gráficos, desenvolvimento e recursos de fundo) e informados da sua disponibilidade ou não para cada *engine*. Este trabalho nos mostra que a escolha do *engine* não é uma tarefa fácil, pois há uma grande variedade de aplicações disponíveis e a comparação de qual é a melhor opção deve ser realizada a partir das necessidades do projeto. Porém, deixa claro que é possível, utilizando softwares livres e abertos, ter acesso a recursos e ferramentas fundamentais para a viabilização da construção de jogos.

4. Desenvolvimento de um *tour* virtual e um experimento

Os primeiros testes de implementação do LabTecA foram realizados utilizando duas soluções: o Unity 3D e o Open Simulator, escolhidas por melhor atenderem as características do GDD (reproduzir laboratório Química, controlar personagem em terceira pessoa, simulação/exploração e resolução de problemas). Os testes iniciais envolveram o desenvolvimento de um *tour* virtual no laboratório e de um experimento de química simples.

O *tour* virtual foi implementado nas duas plataformas como uma visita interativa, onde o estudante tem a oportunidade de explorar os principais equipamentos e espaços do laboratório. Esse passeio no ambiente virtual é realizado por meio de um avatar, personagem projetado para ser uma extensão do aluno no mundo virtual, podendo realizar as ações que o aluno executaria em um laboratório real. Neste ponto, há uma questão importante sobre a questão estética e suas implicações reais na experiência do jogador. No caso do OpenSimulator, o avatar segue alguns padrões da plataforma, enquanto uma representação mais realística de um ser humano. Já no Unity 3D, a escolha foi seguir um caminho um pouco diferente, utilizando um personagem mais caricato, com características cartunescas. Esta diferença estética tem por objetivo verificar questões de imersão e identificação do usuário para com o simulador. Durante a visita, o jogador pode interagir e conhecer todos os equipamentos presentes no espaço. Quando o avatar toca em um equipamento, abre-se uma caixa de diálogo contendo um texto com informações sobre o equipamento.

O primeiro experimento que está sendo implementado envolve a interação com três equipamentos: uma forrageira, uma estufa e uma balança. O estudante é desafiado a descobrir a porcentagem de água encontrada em uma porção de cana-de-açúcar e deve utilizar esses três equipamentos para encontrar a solução do problema. Para tanto, foi necessária a implementação dos comportamentos dos equipamentos e materiais (O que

acontece com um material quando é colocado em cada um desses equipamentos? Que tipo de resultado/*feedback* o participante recebe?)

Figura 1. Imagem comparativa entre o LabTecA desenvolvido no OpenSimulator (esquerda) e o Unity 3D (direita)
Fonte: Autoria própria.

Nas próximas subseções são apresentadas considerações sobre as implementações do *tour* e do experimento no OpenSimulator e Unity 3D.

4.1. Considerações sobre tour e experimento desenvolvidos no Open Simulator

O OpenSimulator[6] é um *engine open-source* utilizado para simular mundos virtuais, um conceito que ficou bastante conhecido com o Second Life™ e utiliza como padrão a linguagem LSL. Para a criação do *tour* foi utilizado o OpenSimulator 0.7.5, na configuração de grid (que permite múltiplos acessos simultâneos).

O principal desafio encontrado na implementação utilizando o OpenSimulator foi a importação dos objetos 3D que já haviam sido criados no Blender. Após diversos testes com formatos de exportação diferentes e importação em *viewer*[7] diferentes (tais como o Firestorm e o Imprudence), obtivemos sucesso exportando os objetos 3D no Blender em formato Collada (extensão .dae) e importando-os por meio do *viewer* Firestorm. Alguns objetos, quando importados, perdem ou modificam a renderização, sendo necessário aplicá-las novamente dentro do ambiente OpenSimulator.

O experimento está em fase inicial de implementação e alguns desafios encontrados utilizando o OpenSimulator são: (i) na programação da interatividade dos problemas e desafios propostos pelo GDD, como por exemplo no uso de equipamentos do Laboratório para simular experimentos de Química em primeira pessoa; (ii) em prover a dinâmica do jogo dentro do mundo virtual, como a sequência de usos de diversos equipamentos a fim de completar um experimento; (iii) em disponibilizar o Laboratório de Química aos jogadores (em como se dará a configuração e acesso para cada jogador ao mundo virtual; como deve ser instalado e configurado o lado do servidor, para prover acesso de forma segura, escalável[8] e com custos adequados).

[6] http://opensimulator.org
[7] Viewer são softwares que permitem ao jogador acessar o mundo virtual instalado em um servidor OpenSimulator
[8] Capacidade de lidar com o crescimento dos acessos de forma uniforme

A implementação do experimento mencionado anteriormente apresenta limitações técnicas importantes para viabilizar a interatividade do experimento de Química (Exemplo: pegar porções de cana picada, pesá-las, colocá-las na forrageira, ligar o equipamento, retirar o caldo resultante, pesá-lo etc). Esse conjunto de ações requer uma intensa interatividade do estudante com os objetos e informações, a fim de simular um experimento real, com os detalhes necessários para que os objetivos pedagógicos pretendidos possam ser alcançados. O OpenSimulator, de forma nativa, não oferece recursos de adaptação e programação para implementar a manipulação de objetos com a precisão e interatividade esperada pelo experimento. Para vencer estas limitações está sendo testado o uso de janelas em que seja possível executar instruções WebGL com HTML5. Isto nos permitiria alcançar recursos de programação e interatividade superiores aos oferecidos pelo OpenSimulator. No estágio atual dos testes, é possível criar janelas (como quadros) dentro do mundo virtual que permitam executar instruções web, como carregar uma página da internet, o próximo passo é conseguir a troca de dados entre estes ambientes.

4.2. Considerações sobre tour e experimento desenvolvidos no Unity 3D

O Unity 3D é um software proprietário, cuja utilização tem se expandido entre diferentes empresas da iniciativa privada. Entretanto a desenvolvedora disponibiliza uma versão gratuita, a qual possui muitas ferramentas que possibilitam a produção de jogos e objetos interativos. Existem algumas diferenças entre as duas versões, no entanto, o conjunto de ferramentas disponível na versão livre tem se mostrado suficiente para o desenvolvimento de um jogo completo, tal como o próprio LabTecA, sendo que todas as dinâmicas presentes nesse projeto podem ser executadas a partir desta versão do Unity 3D.

Este *engine* aceita três diferentes tipos de linguagem de programação orientada a objeto (C#, Javascript e Boo). O site do software oferece amplo suporte de scripts, e tutoriais para a comunidade que deseja aprender sobre suas especificidades, além de existir uma infinidade de tutorias livres na internet. O número de desenvolvedores que utiliza esse software é extenso, devido à versão gratuita. O suporte dado ao desenvolvimento do *engine* tem crescido muito nos últimos meses, sendo que a versão livre tem recebido atualizações, podendo até exportar jogos para plataformas que antes eram exclusivas da versão paga, como dispositivos que usam o sistema Android, por exemplo.

O software dá suporte para o desenvolvimento de uma variedade de gêneros de jogos, sendo possível implementar jogos para a web, e jogos em rede. O formato de importação de modelos 3D é o .fbx, presente na maioria dos softwares de modelagem 3D, tal como o Blender, que vem sendo utilizado como o principal software de suporte nos experimentos feitos para este trabalho.

5. Análise das soluções

Na seção 5.1 são apresentados os critérios utilizados na escolha dos *engines* e as características utilizadas nas comparações. Já na seção 5.2 é apresentada a sistemática utilizada para pontuação de cada característica e; finalmente, na 5.3 são informados os resultados da comparação entre os *engines*.

5.1 Análise das soluções

As comparações entre os ambientes buscaram responder a três perguntas:

1. Qual ferramenta propicia o ambiente de desenvolvimento mais produtivo?
   - Que exija o menor esforço de recursos para construção;
   - Que permita um alto grau de reutilização de técnicas, ferramentas ou elementos;
   - Que disponibilize um ambiente de desenvolvimento de fácil aprendizado e/ou amplo suporte técnico.

2. Qual ferramenta viabiliza a construção de um produto final de melhor qualidade?
   - Que consiga reproduzir o ambiente de um laboratório de química o mais próximo do pretendido pelo projeto;
   - Que viabilize a interação do personagem com as instalações físicas (paredes, acessos), móveis e equipamentos;
   - Que permita que sons ambientes sejam reproduzidos ao longo da interação do personagem com o ambiente;
   - Que permita a publicação deste produto em um local acessível ao público-alvo do projeto;
   - Que propicie meios para facilitar a manutenção e evolução posterior do produto.

3. Qual ferramenta é mais apropriada para o desenvolvimento de recursos educacionais abertos?
   - Que permita a distribuição do produto final, bem como de todos os recursos que o compõem (roteiro, código fonte, ilustrações, modelos, texturas etc.) de forma livre e gratuita, possibilitando o seu reuso, remix, revisão e redistribuição;
   - Que seja possível o acesso a todo o ambiente de desenvolvimento de forma livre e gratuita;
   - Que seja de fácil aprendizagem, favorecendo a manutenção e adaptação;

Procurando responder às questões e critérios acima, foram definidos três conjuntos de características a serem avaliadas: (i) eixo da produtividade do ambiente; (ii) eixo da qualidade do produto final; (iii) eixo de adequação à produção de REA.

A partir desses três eixos, foi definido um conjunto de características considerado nas análises, apresentado na Tabela 1.

Tabela 1. Conjunto de características considerado nas análises.

Terreno customizável	Permite a criação de áreas 3D exploráveis por um personagem (avatar) e a criação de áreas externas para montagem de cenários reais (elementos naturais ou paisagens urbanas).
Controle "Avatar"	Se o *engine* já possui de forma nativa e reutilizável recursos controle de um avatar/personagem no ambiente 3D.
Importação de objetos 3D (Blender)	Se o ambiente de desenvolvimento permite a importação de elementos 3D de outras ferramentas. Principalmente dos padrões e formatos exportados pela ferramenta Blender.
Controle de Física	Se o ambiente de desenvolvimento já possui, de forma nativa e reutilizável, o controle de gravidade e colisão.
Objetos reutilizáveis	Se possui recursos que permitam a reutilização de elementos como objetos,

	eventos, *templates* ou qualquer outro elemento de programação.
Recursos de programação	Se o *engine* permite que o ambiente 3D criado possa ser adaptado via programação para situações específicas de interação do avatar.
Confiabilidade	Se a instituição que suporta e fornece o *engine* é confiável, avaliando as condições de manutenção/evolução futura do *engine*.
Produtividade Usabilidade	Avalia a facilidade de instalação do ambiente de desenvolvimento; usabilidade; facilidade de aprendizado.
MMOG (massively multiplayer online)	Se o *engine* permite o desenvolvimento de Jogos no formato MMOG, ou seja, em que múltiplos jogadores estão em um mesmo ambiente on-line.
Browser nativo/plug-in	Se o *engine* permite que o produto final seja executado na web, mesmo que seja necessária a instalação/configuração do navegador.
Qualidade gráfica	Avalia a qualidade gráfica do produto final.
Sons/Musica	Se o *engine* possibilita a inclusão de trilhas e efeitos sonoros.
Atmosfera/Interação/Jogabilidade	Se os recursos esperados de jogabilidade descritos no GDD podem ser implementados por meio do *engine*.
Manutenibilidade	Avalia quais as condições o *engine* oferece para a implementação de boas práticas, que favoreçam a futura manutenção do produto (padronização de codificação, organização da arquitetura, suporte técnico etc.)
Interatividade	Se é possível construir um Objeto de Aprendizagem Interativo.
Integração com o Moodle	Se é possível disponibilizar o produto final integrado ao Moodle.
Portabilidade	Se os elementos e produtos criados podem ser portados para outro sistema operacional ou dispositivo.
Adaptabilidade	Se os elementos e produtos criados podem ser facilmente adaptados para necessidades de outros contextos educacionais.
Apreensibilidade	Se o *engine*, bem como os elementos criados por meio dele, são de fácil assimilação.
Licença	Esta característica avalia se o *engine*, os códigos fontes e o produto final podem ser disponibilizados de forma livre e aberta.

Fonte: Autoria própria.

5.2. Critérios para classificação e pontuação

Os critérios definidos para pontuação foram criados de forma a classificar e explicitar de forma qualitativa, porém concreta:

- Se o *engine* possui ou não a característica;

- Se possui, porém exige um esforço maior de programação, se comparado às outras ferramentas avaliadas;
- Se, dentre os avaliados, tem destaque nessa característica.

Para isto foi criada a seguinte tabela de pontuação: 0 – Não atende, 1 – Atende com grande esforço na solução técnica, 2 – Atende, 3 – Atende com destaque).

Ao final os pontos de cada característica são somados e os *engines* foram classificados em ordem dos que mais receberam pontos (*ranking*), para cada eixo (produtividade do ambiente, qualidade do produto final e adequação à construção de recursos educacionais abertos).

5.3. Resultado das análises

É importante reforçar que a comparação teve como ponto de partida as necessidades e características esperadas para o Laboratório Virtual de Química e que, por conta disto, esta não é uma comparação ampla entre *engines* para desenvolvimento de jogos educacionais e simulações. Além disso, foram avaliados *engines* que se propõem originalmente a objetivos bem distintos: (i) o OpenSimulator visa a criação de mundos virtuais; (ii) o Unity 3D visa a criação de jogos 3D.

Observando os critérios, características e sistemática já citados para comparação dos ambientes, nas tabelas abaixo estão os resultados de cada *engine* avaliado.

Tabela 2. Eixo da produtividade do ambiente.

Engine	**Terreno Adaptável**	**Controle "Avatar"**	**Importação objetos 3D**	**Física**	**Objetos reutilizáveis**	**Programação**	**Confiabilidade**	**Produtividade**
OpenSim	2	2	2	2	2	1	2	2
Unity 3D	3	2	3	3	2	3	3	3

Fonte: Autoria própria.

Os dois *engines* possuem recursos nativos que permitem a construção do terreno, controle do "avatar" e da simulação da física. Porém o OpenSimulator possui limitações para a programação das dinâmicas de jogo, possibilitando apenas interações simples do avatar com os objetos.

Tabela 3. Eixo da qualidade do produto final.

Engine	**MMOG**	**Browser nativo/plug-in**	**Qualidade gráfica**	**Sons**	**Atmosfera/Interação/Jogabilidade**	**Manutenibilidade**
OpenSim	3	1	2	2	2	2
Unity 3D	2	3	3	3	3	3

Fonte: Autoria própria.

Como é nativo para o uso em mundos virtuais, o OpenSimulator já esta pronto para o uso em rede e com múltiplos acessos de jogadores a uma mesmo ambiente (MMOG), o que no Unity 3D precisa ser configurado e programado (embora não exija grande esforço). O

OpenSimulator necessita do software *viewer* instalado na máquina de cada participante, inviabilizando o uso diretamente de um navegador (algo que pode ser implementado no Unity 3D por meio de uma pequena adaptação). A qualidade gráfica, dinâmica de jogo e sons possuem mais recursos e facilidade de adaptação/construção no Unity 3D.

Tabela 4. Eixo da adequação à produção de REA.

Engine	**Interatividade**	**Integração com AVA**	**Apreensiblidade**	**Portabilidade**	**Customização**	**Licença**
OpenSim	2	3	1	1	2	3
Unity 3D	2	2	2	3	2	0

Fonte: Autoria própria.

O OpenSimulator, via Sloodle[9], permite que ações desempenhadas dentro do mundo virtual sejam refletidas como atividades avaliativas no ambiente de aprendizado Moodle[10]. O OpenSimultor destaca-se por possuir livre acesso ao seu código fonte, o que no Unity 3D não é possível, embora a versão básica utilizada tenha o binário (instalação) livre. O Unity 3D se destaca por permitir que o produto final seja executado em ambientes e sistemas operacionais diversos e também pelo volume e qualidade do suporte técnico disponibilizado na Internet.

Tabela 5. Posições/Ranking

Produtividade	**Pontos**
Unity 3D	22
OpenSim	15

Qualidade	**Pontos**
Unity 3D	17
OpenSim	12

REA	**Pontos**
Unity 3D	11
OpenSim	12

Final	**Pontos**
Unity 3D	50
OpenSim	39

Fonte: Autoria própria.

5.4. Justificativa da pontuação por engine e característica

Tabela 6. OpenSim

Terreno adaptável	2 - Possui recursos para importar terras, além de possuir de forma nativa recursos para criação e customização do terreno.
Controle "Avatar"	2 – O controle de "avatar" em primeira pessoal é nativo, pois faz parte do conceito de mundos virtuais.
Importação objetos 3D (Blender)	2 – Permite a importação de objetos 3D em diversos formatos. Não faz isto muito bem, em alguns casos os erros de importação não são amigáveis ou facilmente compreendidos.
Controle de Física	2 – O controle de gravidade / colisão é nativo, pois faz parte

[9] http://www.sloodle.org/
[10] https://moodle.org/

	do conceito de mundos virtuais.
Objetos reutilizáveis (render, elementos...)	2 – Possui templates e modelos de novos elementos 3D.
Customização/programação	1 – Permite o uso de linguagem script para "automação" e interação dos objetos no mundo virtual. Porém é limitado a interação entre os objetos e mundo, não permite grandes adaptações e mudanças de dinâmica do jogo (exemplo: mudança de fases, ciclos ou interações fora do mundo virtual).
Confiabilidade	2 – É suportado por uma comunidade livre, existem tutoriais e suporte. Risco de descontinuidade baixo.
Produtividade	2 – A criação da terra, salas, avatar e ambiente é relativamente rápida. Há problemas quando é necessário "sair" da perspectiva do mundo virtual.
MMOG (massively multiplayer online)	3 – O acesso de múltiplos jogadores em rede é nativo da ferramenta.
Browser nativo/plug-in	1 – Para acessar mundo virtual utiliza uma ferramenta cliente, um viewer (visualizador). As ferramentas viewer estáveis não são suportadas via browse, necessitam ser baixadas e instaladas localmente. Há experiências de uso de viewer via browse, mas incipientes.
Qualidade gráfica	2 – Devido ao uso de clientes locais permite potencializar a qualidade gráfica dependendo do equipamento do usuário.
Sons/Musica	2 - Permite a inclusão de sons. Os sons são acionados através de eventos programáveis (interação entre objetos).
Atmosfera/Interação/Jogabilidade	2 – Atmosfera e jogabilidade adequados. A "imersão" no mundo virtual cria uma dinâmica de jogo instigante.
Manutenibilidade	2 – Manutenção e melhorias são facilitadas pelo fato de ser "Open Source" e contar com o apoio da "comunidade" de desenvolvedores.
Construção objeto aprendizagem interativo	2 – Permite a construção de objetos e ambientes customizáveis e com interação controlada.
Recurso educacional aberto	2 – Permite a disponibilização dos fontes, acessos via internet (não www e sim clientes próprios - viewers) e com amplo acesso a material para customização.
Integração AVA moodle	1 – Há experiências de uso de viewers diretamente de browsers, o que em tese, permitiria o uso de "dentro" do ambiente virtual. Mas é algo que certamente limitaria a desenvolvimento (problemas de renderização e continuidade de suporte do viewer)
Apreensiblidade	2 – Fácil aprendizado, há amplo apoio da comunidade de desenvolvedores.
Portabilidade	2 – Existem versões para Windows, Linux e Mac. O produto desenvolvido também "roda" nestes mesmos ambientes.

1
1

Customização	2 – É amplamente viável customizar o produto final em outros contextos.
Licença	3 – Totalmente livre, inclusive com código fonte do próprio OpenSim disponível.

Fonte: Autoria própria.

Tabela 7. Unity 3D

Terreno adaptável	3 – Possui diversas terras *template* com terreno totalmente customizável. O diferencial são as ferramentas e recursos que facilitam a customização do terreno.
Controle "Avatar"	2 – O controle de "avatar" em primeira pessoal é nativo, há um objeto já "codificado" com estes recursos.
Importação objetos 3D (Blender)	3 – Permite a importação de objetos 3D em diversos formatos. Possui suporte a erros de importação.
Controle Gravidade / colisão	3 – Possui uma ferramenta específica para simular o controle de gravidade e colisão. Permite, de forma simples, customizar os parâmetros de interação.
Objetos reutilizáveis (render, elementos...)	2 – Possui templates e modelos de novos elementos 3D.
Customização/programação	3 – Altamente customizável, foi desenvolvido como plataforma para criação de games.
Confiabilidade	2 – É suportado por uma grande multinacional. Risco de descontinuidade baixo.
Produtividade	3 – É uma ferramenta integrada de alta produtividade para elaboração de jogos.
MMOG (massively multiplayer online)	2 – O acesso de múltiplos jogadores em rede é possível, mas depende de codificação/configuração.
Browser nativo/plug-in	2 – Permite a elaboração de jogos para browser. O usuário final precisa baixar um plug-in para o browser.
Qualidade gráfica	3 – Utiliza o que o mercado considera de alto nível para elaboração de games.
Sons/Musica	3 - Permite a inclusão de sons e musica ambiente. Os sons são acionados através de eventos programáveis (interação entre objetos).
Atmosfera/Interação/Jogabilidade	3 – Atmosfera e jogabilidade são diferenciais, pois permite o uso de "jogos" de câmera e controle que potencializam a dinâmica do jogo.
Manutenibilidade	3 – Possui ampla rede de apoio e suporte ao desenvolvedor.
Construção objeto aprendizagem interativo	2 – Permite a construção de objetos e ambientes customizáveis e com interação controlada.

Recurso educacional aberto	1 – Permite a disponibilização dos fontes, mas não é possível disponibilizar de forma livre o ambiente de desenvolvimento (o próprio Unity).
Integração AVA moodle	2 – Devido a criação de jogos via browse, permite o uso de "dentro" do ambiente virtual. Há casos de uso do moodle + Unity em outra universidade (Universidade Nacional Autonoma de México)
Apreensiblidade	2 – Fácil aprendizado, há amplo apoio da empresa e comunidade de desenvolvedores.
Portabilidade	3 – É um diferencial além de versões para Windows, Linux e Mac., permite criar games para consoles (XBOX, Nintendo)
Customização	2 – É viável customizar o produto final em outros contextos.
Licença	1 – Há uma versão inicial free, o Unity Pro com todas as funcionalidades é pago.

Fonte: Autoria própria.

6. Considerações finais

Neste artigo foi apresentado um estudo comparativo de duas soluções de *engines* para a criação de um laboratório virtual interativo 3D: o OpenSimulator (aplicação de código aberto para a criação de mundos virtuais 3D multi-usuário) e o Unity 3D (*engine* para a criação de jogos 3D). A análise teve como intuito verificar a solução mais adequada para a construção de um laboratório virtual interativo 3D, possibilitando a simulação de interações detalhadas com objetos, além de considerar a implementação de elementos de jogos (desafios, regras) e a disponibilização do produto final como um recurso educacional aberto.

O resultado de nosso estudo mostra que o Unity 3D é a solução que atende melhor os requisitos levantados, destacando-se pela produtividade e melhor qualidade do produto final. No entanto, vale destacar que o OpenSimulator fornece um ambiente virtual rico para a promoção da interação entre os participantes, além de ser um software livre, sendo uma solução bastante adequada, mas que requer um esforço adicional de desenvolvimento.

7. Referências

Aldrich, C. (2009). Learning with games, simulations and virtual worlds: strategies for online instruction. Published by Jossey-Bass, John Wiley & Sons Inc, 2009.

Bispo, Digo M. M.; Zabeu, Márcio S.; Umbuzeiro, Gisela A.; Borges, Marcos A. F. (2012). Desenvolvimento de Jogo Educacional sobre Ecotoxicologia Utilizando HTML5. In: Revista Brasileira de Informática na Educação, 2012.

Creative Commons. **Creative Commons - Sobre as Licenças**. Disponível em: http://creativecommons.org/licenses/. Acesso em: 08 jun. 2014.

Educação Aberta. O que é REA. Recursos Educacionais Abertos (REA): Um caderno para professores. Campinas, SP. Disponível em: <http://www.educacaoaberta.org/wiki>. Último acesso em: 03/10/2012.

Johnson, L., Adams, S., and Cummins, M. (2012).The NMC Horizon Report: 2012 Higher Education Edition. Austin, Texas:The New Media Consortium.

Machado, Alex F. V.; Cazeta, Paôla P.; Figueiredo, Ana Mara O. ; Sant'ana, Leandro dos S.; Junior, Nilson A. da S.; Dutra, Sebastião de F. E.; Clua, Esteban; "*Uma proposta de Jogo Educacional 3D com Questões Didáticas*". In: Anais do XXII SBIE, . Aracaju 2011.

OTSUKA, J. L., BEDER, Delano Medeiros, MONTANARO, P. R., ROCCA, I. Z., GHELARDI, A. Desenvolvimento de jogos educacionais abertos In: Workshop Recursos Educacionais Abertos, 2012, Rio de Janeiro. Anais dos Workshops do CBIE 2012 , 2012.

Rocha, R. V.; Rocha, R. V.; Araújo, R. B. (2010). Selecting the Best Open Source 3D Games *Engines*. In: Proceedings of SBGames, 2010.

Seabra, Rodrigo Duarte; "*Experiencia Inovadora e Contribuições no Ensino de Geometria em uma Ferramenta 3D:um Estudo de Caso na Engenharia*". In: Revista Brasileira de Informática na Educação, 2012.

Squire, K.;Barnett,M.; Grant, J. M.; Higginbotham, T (2004) Electromagnetism superchanged! Learning physics with digital simulation games. In: Proceedings of the 2004 International Conference of the Learning Sciences. Los Angeles: UCLA Press.