

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO**

**PROTÓTIPO DE MUNDO VIRTUAL PARA**  
**RELACIONAMENTO COM PARTICIPANTES DO**  
**INTERAÇÃO FURB**

**FLAVIANO JOSÉ ANGELI**

**BLUMENAU**  
**2010**

**2010/1-11**

**FLAVIANO JOSÉ ANGELI**

**PROTÓTIPO DE MUNDO VIRTUAL PARA  
RELACIONAMENTO COM PARTICIPANTES DO  
INTERAÇÃO FURB**

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Sistemas  
de Informação — Bacharelado.

Prof. Dalton Solano dos Reis, M.Sc - Orientador

**BLUMENAU  
2010**

**2010/1-11**

**PROTÓTIPO DE MUNDO VIRTUAL PARA  
RELACIONAMENTO COM PARTICIPANTES DO  
INTERAÇÃO FURB**

Por

**FLAVIANO JOSÉ ANGELI**

Trabalho aprovado para obtenção dos créditos  
na disciplina de Trabalho de Conclusão de  
Curso II, pela banca examinadora formada  
por:

Presidente: \_\_\_\_\_  
Prof. Dalton Solano dos Reis, M.Sc – Orientador, FURB

Membro: \_\_\_\_\_  
Prof. Francisco Adell Péricas, M.Sc – FURB

Membro: \_\_\_\_\_  
Prof. Oscar Dalfovo, Dr – FURB

Blumenau, 05 de julho de 2010

Dedico este trabalho a minha esposa, amigos,  
e aqueles que ajudaram na realização deste.

## **AGRADECIMENTOS**

A todos aqueles deram apoio nos momentos difíceis na elaboração deste trabalho.

A minha esposa Daiane por acreditar na conclusão deste trabalho e sempre que necessário dispor de carinho e atenção, e quando tudo parecia perdido “injetar” ânimo para seguir em frente.

Ao meu orientador Dalton Solano do Reis e amigos que com muita paciência ajudaram a resolver os mais variados problemas.

Ex ore parvulorum veritas.

Autor Desconhecido

## RESUMO

Este trabalho apresenta a utilização de uma tecnologia, no caso, o OpenWonderland, para criação de mundos virtuais em 3D executado em uma rede local (*Local Area Network* – LAN) ou através da *internet*. O mundo virtual é disponibilizado através de um servidor que permite a conexão de vários usuários simultaneamente onde cada usuário possui uma representação no mundo virtual através de um avatar podendo ser personalizado de acordo com as suas preferências. Para atender o Interação FURB utilizou-se o servidor de mundo virtual OpenWonderland, com algumas alterações em seu código-fonte, permitindo que objetos sejam disponibilizados dinamicamente no mundo virtual e as contas de usuários sejam criadas através do preenchimento de um formulário em uma página web.

Palavras-chave: Simulador. Mundo virtual. Avatar. Interação FURB.

## **ABSTRACT**

This paper presents the use of technology, in case OpenWonderland for creating 3D virtual worlds running on a LAN (Local Area Network - LAN) or via the Internet. The virtual world is provided through a server that allows connection of multiple users simultaneous where each user has a representation in the virtual world using avatars can be customized according to your preferences. To meet the Interação FURB used the virtual world OpenWonderland, with some changes in your source code, allowing objects to be displayed dynamically in the virtual world and the user accounts are created by completing a form on a web page.

**Key-words:** Simulator. Virtual world. Avatar. Interação FURB.



## LISTA DE ILUSTRAÇÕES

Figura 1 - Desenho feito no Google SketchUp .....	17
Figura 2 - Página do Interação FURB .....	19
Figura 3 - Active Worlds .....	21
Figura 4 – OpenSimulator .....	22
Figura 5 – Imagem demonstrando Projeto OpenWonderland.....	24
Figura 6 - Apresentação do módulo do sistema de partículas .....	25
Figura 7 - Arquitetura do servidor OpenWonderland .....	26
Quadro 1 - Requisitos funcionais .....	27
Figura 8 - Diagrama de casos de uso para cadastros .....	28
Figura 9 - Diagrama de casos de uso para manutenção.....	29
Figura 11 - Diagrama de classes para criação de contas de usuários .....	30
Figura 11 - Diagrama de classes para criação de objetos .....	31
Figura 12 - Diagrama de sequência para criação de conta de usuário.....	32
Figura 13 - Diagrama de sequência para criação de objetos .....	33
Quadro 3 - Linhas de comando para iniciar o servidor .....	35
Quadro 4 - Comando para realizar download do servidor.....	35
Quadro 5 - Comando para compilar e empacotar a aplicação.....	35
Quadro 6 - Comandos para download de código-fonte.....	36
Quadro 7 - Linha de comando para compilação de módulos .....	37
Quadro 8 – Parte do código-fonte responsável pela criação de contas de usuários .....	39
Figura 14 - Formulário para criação de conta.....	39
Figura 15 - Informação de conta que já existe.....	40
Figura 16 - Aviso de campo não preenchido .....	40
Quadro 9 - Parte do código-fonte para criação de objetos dinamicamente .....	41
Figura 17 - Formulário para criação de objetos.....	42
Figura 18 - Interface para gerenciamento de grupos .....	43
Figura 19 - Formulário para criação de grupos de usuários .....	43
Figura 20 - Incluindo usuários no grupo .....	43
Figura 21- Páginat inicial para conectar ao mundo virtual.....	44
Figura 22 - Cliente do mundo virtual sendo executado.....	44
Figura 23 - Aplicação solicitando autenticação.....	45

Figura 24 - Usuário conectado ao servidor de mundo virtual .....	45
Figura 25 - Propriedades de um objeto no mundo virtual .....	46
Figura 26 - Adicionando nova funcionalidade para o objeto .....	46
Figura 27 - Editando as permissões do objeto .....	47
Figura 28 - Diagrama apresentado em reunião com supervisora .....	48
Quadro 10 - Comparação entre servidores de mundo virtual.....	51
Quadro 11 - Detalhamento do caso de uso UC01 - Cadastrar objeto.....	57
Quadro 12 - Detalhamento do caso de uso UC02 - Manter objetos .....	58
Quadro 13 - Detalhamento do caso de uso UC04 - Relatório de contas criadas.....	59
Quadro 14 - Detalhamento do caso de uso UC05 - Cadastrar conta de usuário.....	59
Quadro 15 - Detalhamento do caso de uso UC06 - Definir permissões.....	60
Quadro 16 - Detalhamento do caso de uso UC07 - Cadastrar grupos.....	61
Quadro 17 - Procedimentos para obter o código XML de um objeto no OpenWonderland....	62
Quadro 18 - Exemplo de arquivo XML .....	63
Quadro 19 - Troca de mensagens sobre o programa connection-sample .....	64
Quadro 20- Conversa com equipe do OpenWonderland.....	65

## **LISTA DE SIGLAS**

DNS – *Domain Name System*

JAR – *Java Archive*

JNLP – *Java Network Launching Protocol*

MB – *Megabyte*

SVN – *Subversion*

XML - *eXtensible Markup Language*

# SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>13</b>
1.1 OBJETIVOS DO TRABALHO .....	14
1.2 ESTRUTURA DO TRABALHO .....	14
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>15</b>
2.1 MUNDOS VIRTUAIS .....	15
2.1.1 Java Network Launch Protocol (JNLP) .....	16
2.1.2 Google Sketchup .....	16
2.1.3 Serviço de Terminal X11 .....	17
2.1.4 Serviço de diretório Active Directory .....	17
2.2 SISTEMA ATUAL .....	18
2.3 TRABALHOS CORRELATOS .....	20
2.3.1 AWSINOS – Active Worlds .....	20
2.3.2 OpenSimulator .....	21
2.3.3 Projeto OpenWonderland.....	23
<b>3 DESENVOLVIMENTO DO SISTEMA.....</b>	<b>27</b>
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	27
3.2 ESPECIFICAÇÃO .....	29
3.2.1 Criação de contas de usuários .....	29
3.2.2 Criação de objetos no mundo virtual .....	31
3.2.3 Diagramas de Sequência .....	32
3.3 IMPLEMENTAÇÃO .....	33
3.3.1 Técnicas e ferramentas utilizadas.....	33
3.3.2 Operacionalidade da implementação .....	34
3.3.2.1 Implementando e iniciando um servidor de mundo virtual .....	34
3.3.2.2 Implementando e instalado módulos adicionais .....	36
3.3.2.3 Controle de Acesso e segurança das contas de usuário .....	38
3.3.2.4 Criação e atualização de objetos no mundo virtual .....	40
3.3.2.5 Criação e gerenciamento dos grupos de usuários .....	42
3.3.2.6 Utilização do mundo virtual pelos visitantes.....	44
3.3.2.7 Definir ou editar permissões em objetos no mundo virtual.....	46
3.4 RESULTADOS E DISCUSSÃO .....	47

<b>4 CONCLUSÕES</b> .....	<b>52</b>
4.1 EXTENSÕES .....	53
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> .....	<b>54</b>
<b>APÊNDICE A – Detalhamento dos casos de uso</b> .....	<b>57</b>
<b>APÊNDICE B – Procedimentos para obter o código XML de um objeto</b> .....	<b>62</b>
<b>APÊNDICE C – Exemplo de arquivo XML utilizado para criação de objetos</b> .....	<b>63</b>
<b>APÊNDICE D – Conversa com Jonathan Kaplan sobre programa connection-sample</b> .	<b>64</b>
<b>APÊNDICE E – Conversa com a equipe de desenvolvimento do Projeto Wonderland</b> ..	<b>65</b>

## 1 INTRODUÇÃO

A evolução nos meios de comunicação revolucionou a forma com que as pessoas vivem. Barreiras como distância e tempo entre cada mensagem permitem que o oriente e ocidente tenham conversação em tempo real através do uso da internet.

Nos primeiros anos de existência da internet, por volta de 1995, a rede era vista com desconfiança pelos usuários. Para muitos, a internet era uma ferramenta que isolava as pessoas (MARQUES, 2005, p. 1), chegando a ser considerada “[...] apenas um passeio estático e solitário”. (ROSA, 2001, p. 1). Mas sua rápida evolução possibilitou adicionar cores e imagens reunindo uma grande variedade de ferramentas que tornaram possível a criação dos mundos virtuais.

De um simples texto, agora é possível demonstrar expressões, simular a aparência física, degustar de sensações que no mundo real não são possíveis. “Suas formas de representação podem ser através de apelido em formato texto, bidimensional como em uma foto ou em três dimensões como em um videogame, sons ou tato” (CPQD, 2008, p. 1).

Atualmente a Universidade Regional de Blumenau (FURB) realiza o evento Interação FURB, cujo objetivo é aproximar a Universidade com o aluno do Ensino Médio, por meio de informações específicas para o aluno, além de ações de relacionamento que façam o aluno ter motivação em interagir com a FURB.

A principal interface de comunicação da organização do Interação FURB com os alunos envolvidos é realizada através do site do evento (UNIVERSIDADE REGIONAL DE BLUMENAU, 2009b, p. 1). Após a realização do evento o contato com os alunos é mantido através de palestras conforme as escolas solicitam. Não há outras formas de contato ou interação com os alunos. No presente momento a organização do Interação FURB necessita de outras ferramentas que possam manter o contato com os participantes, sejam alunos ou escolas.

No intuito de atender esta carência os mundos virtuais entram como uma forma de ajudar o Interação FURB, pois “Mundos Virtuais visam atingir um senso de realismo e proporcionar experiências imersivas” (ALBUQUERQUE; VELHO, 2001, p. 2), tendo a possibilidade de manter o evento ativo, mesmo após a remoção da estrutura física presente anualmente durante a realização das atrações do Interação FURB.

Desta forma os objetos dispostos no mundo virtual podem conter materiais (textos ou conteúdo multimídia) para os visitantes, ou serem utilizados para promoção de marketing

através de vídeos e figuras que representem cartazes, panfletos ou propagandas de televisão do mundo real.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver um ambiente virtual para interagir com os participantes do Interação FURB sendo representados através de um avatar no mundo virtual.

Os objetivos específicos são:

- a) implementar um mundo virtual que possa ser utilizado pela organização do Interação FURB;
- b) permitir que os participantes tenham disponíveis os materiais apresentados pelo Interação FURB através de objetos dispostos no mundo virtual;
- c) implementar nova ferramenta de veiculação de marketing para a Universidade Regional de Blumenau utilizando os objetos dispostos no mundo virtual;
- d) implementar rotinas que permitam os administradores do mundo virtual adicionar ou remover objetos no mundo virtual.

## 1.2 ESTRUTURA DO TRABALHO

O capítulo dois faz a fundamentação teórica, apresentando os principais tópicos para o desenvolvimento deste trabalho, entre os tópicos que serão abordados estarão conceitos sobre mundos virtuais e JNLP.

O capítulo três demonstra a especificação e implementação do sistema de mundo virtual, abordando tópicos sobre os requisitos necessários para atender o Interação FURB, o processo de implementação assim como as técnicas e ferramentas utilizadas.

No capítulo quatro são apresentadas as conclusões deste trabalho e possíveis extensões em trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Na seção 2.1 é apresentado um breve conceito sobre os mundos virtuais, na seção 2.2 será possível identificar o processo atual que é realizado no departamento de Marketing da Universidade para a organização e realização do Interação FURB, e a seção 2.3 apresentará quais ambientes virtuais já existentes possuem características compatíveis ou semelhantes com a aplicação que se propõe desenvolver com este trabalho.

### 2.1 MUNDOS VIRTUAIS

Os mundos virtuais fazem uso da realidade virtual para poder construir uma abstração do mundo real. Desta forma “a Realidade Virtual baseia-se no uso de interfaces computacionais interativas com o objetivo de criar nos usuários uma sensação de realidade [...]” (MARINS; HAGUENAUER; CUNHA, 2007, p. 2), permitindo com que o usuário desfrute de uma experiência de aprendizado e exploração do ambiente ao qual está conectado, interagindo com outros usuários ou com os objetos dispostos no cenário.

Tratando-se das vantagens em utilizar a realidade virtual, pode-se

“(1) tornar o aprendizado mais interessante e divertido com o objetivo de melhorar a motivação e a atenção; (2) reduzir custos, quando a utilização do objeto e do ambiente real for mais dispendiosa que a simulação; (3) possibilitar que se explorem situações que são impossíveis de serem feitas no mundo real, por exemplo: explorar um planeta como Marte [...]”. (CLARK, 2006, apud MARINS; HAGUENAUER; CUNHA, 2007, p. 3).

Sua utilização é possível em diversos ambientes e direcionado para os mais variados públicos, “[...] um exemplo de mundo virtual é o implementado em jogos conhecidos como *Massively Multiplayer Online Games* (MMOG)[...]” (CPQD, 2008, p. 1) onde o mundo virtual possui determinadas regras e até mesmo pode possuir uma moeda corrente, para a comercialização de objetos ou vantagens no mundo virtual.

Segundo Mesa (2007) os mundos virtuais “[...] emergem não só como lugares para criar e passar o tempo através de um avatar, mas também para desenvolver o comércio.[...]”

Nas empresas, os mundos virtuais podem ser utilizados para reuniões, treinamentos e outros eventos que necessitem de pessoas que geralmente possuem dificuldade para estarem reunidas, normalmente os problemas estão associados com a distância geográfica entre os



participantes da reunião.

### 2.1.1 Java Network Launch Protocol (JNLP)

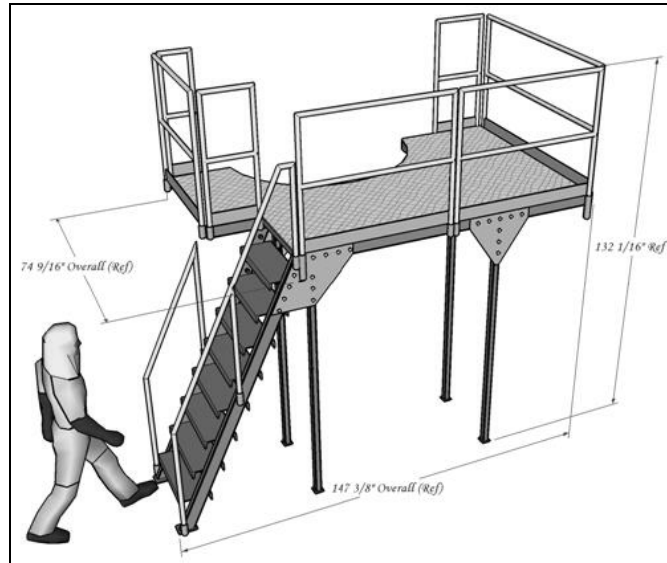
“[...] Este é um mecanismo para distribuição de programas através de um servidor web. Tipicamente iniciado através do browser, estes programas são distribuídos para os clientes e executados fora do escopo do browser. [...]” (ZUKOWSKI, 2010, p. 1, tradução nossa).

A atualização do software distribuído por JNLP também é simplificada devido ao fato de que cada vez que a aplicação é executada ela é comparada com a versão que está no servidor. Caso a do servidor seja diferente da que está na máquina local, automaticamente é iniciado o *download* da nova versão.

### 2.1.2 Google Sketchup

“O Google SketchUp é um software que você pode usar para criar modelos em 3D de qualquer coisa que desejar [...]” (GOOGLE, 2010a, p. 1). Esta ferramenta permite que sejam criados modelos de áreas como engenharia, arquitetura e design, construção civil, educação entre outros.

Entre os vários formatos de arquivos que o Google SketchUp tem suporte, é possível destacar as extensões DWG, 3DS, OBJ, PDF entre outros. A ferramenta atualmente é disponibilizada em versão gratuita e comercial, possuindo diferenças de funcionalidade em cada uma delas. A Figura 1 apresenta um desenho feito no Google SketchUp.



Fonte:Google (2010b)

Figura 1 - Desenho feito no Google SketchUp

### 2.1.3 Serviço de Terminal X11

“[...] Um terminal X pode ser configurado para conectar múltiplos usuários a um mesmo servidor, criando sessões customizadas em uma estrutura de rede simples. [...]” (MUI; PEARCE, 1992, p. 45, tradução nossa).

O serviço de terminal X11 permite que vários usuários conectem-se ao mesmo servidor Linux ou Unix e utilizem as aplicações instaladas no servidor de forma compartilhada tendo todo o processamento e armazenamento de informações centralizadas no servidor.

### 2.1.4 Serviço de diretório Active Directory

Segundo Losano (2010), um “[...] serviço de diretório é um conjunto de Atributos sobre recursos e serviços existentes na rede, isso significa que é uma maneira de organizar e simplificar o acesso aos recursos de sua rede centralizando-os. [...]”. Entre os recursos que o *Active Directory* armazena, pode-se citar informações de contas de usuários, impressoras, computadores, grupos entre outros. Cada elemento no *Active Directory* é considerado um objeto, cada objeto possui propriedades as quais são conhecidas como atributos.

Além de centralizar as informações dos objetos existentes na rede, é possível gerenciar as permissões, autorizando o acesso ou utilização de determinado recurso por grupos ou usuários, garantindo também proteção contra intrusos.

## 2.2 SISTEMA ATUAL

A FURB teve o início de suas atividades em 1964 tendo seu reconhecimento e credenciamento com o Ministério da Educação (MEC) em 1986. A partir de 1995 através da Lei Complementar Municipal N° 80 é reconhecida como Instituição de Ensino Superior e mantida pela Fundação Universidade Regional de Blumenau (UNIVERSIDADE REGIONAL DE BLUMENAU, 2009a, p. 1).

Desde 2007 a FURB tem realizado o evento Interação FURB. Este evento tem como principal objetivo aproximar o aluno do ensino médio por meio de informações específicas para o futuro profissional e educacional, bem como ações que tornem possível ampliar a interação entre a FURB e os alunos.

O evento ocorre uma vez por ano e possui um espaço próprio no site da instituição conforme pode ser verificado na Figura 2. Neste espaço é possível obter mais informações sobre o evento, como funciona e quais as atrações que estarão disponíveis. Também é possível conferir as fotos do evento dos anos anteriores e dicas sobre profissões.



UNIVERSIDADE REGIONAL DE BLUMENAU

Alunos do Ensino Médio terão um dia de aula na Universidade

23 de Setembro

Interação FURB

Você está em: FURB: Interação FURB: Interação FURB

**Inscrições** **DIA DE AULA NA FURB - 23/09**

Fotos Interação 2007-2008

Fotos Interação 2006

Fale Conosco

Organização

Dicas

Vestibular FURB

Saiba mais sobre a FURB

No dia 23 de setembro os alunos de Ensino Médio terão um Dia de Aula na FURB. Serão oferecidos workshops em cada curso de Graduação. Para participar é necessário inscrever-se.

**O período de inscrições é de 03 a 31 de agosto.** As vagas são limitadas!

Neste dia, o estudante que participar dessa grande oportunidade conversará com profissionais de diversas áreas de atuação, obterá informações sobre o mercado de trabalho, conhecerá a estrutura dos cursos que está interessado e vivenciará um dia na Universidade.

[Clique aqui](#) para ouvir o Jingle do Interação.

orkut Interação FURB twitter

Copyright © 2004 FURB - Universidade Regional de Blumenau. Todos os direitos reservados.

Fonte: Universidade Regional de Blumenau (2009b).

Figura 2 - Página do Interação FURB

Os organizadores do Interação FURB realizam visitas às escolas que possuem ensino médio, em especial as com alunos matriculados no terceiro ano do ensino médio. Estas visitas ocorrem entre os meses de março e agosto, onde é realizada uma conversa com os alunos. No momento destas conversas é entregue um questionário a fim de cadastrar os alunos e é aproveitada a oportunidade para demonstrar a estrutura da FURB (cursos, intercâmbios, entre outras informações), quais as formas de ingresso e um vídeo que apresenta cada curso da graduação. Também é disponibilizado para cada sala de aula do terceiro ano uma faixa com informações sobre a escolha de uma profissão.

Próximo à realização do evento, é disponibilizado um *link* para que os interessados e que ainda não tenham entrado em contato com a organização do Interação FURB possam se cadastrar e recebam informações até a data de realização do evento.

Na maior parte do tempo, a página do Interação FURB não recebe atualizações e a quantidade de informações é pequena, impossibilitando os interessados de terem uma visão ampla sobre o evento. Também não há ferramentas que tornem o espaço mais dinâmico e interessante, pode-se citar, por exemplo, de que não é possível promover um debate virtual com os alunos que já participaram do evento.

Após a realização do Interação FURB é mantido contato com as escolas através de telefone, email ou correspondência. O contato com as escolas permanece o mesmo todos os anos, sendo necessária a visita na escola e a apresentação do evento aos novos alunos do terceiro ano. Além das visitas, também são disponibilizadas palestras, de acordo com o tema

solicitado pela escola. A organização do Interação FURB verifica com os professores da Universidade a disponibilidade de realizar palestras na escola, permitindo assim ampliar o relacionamento que o Interação FURB possui com a escola e seus alunos.

## 2.3 TRABALHOS CORRELATOS

No mercado existem diversas tecnologias e produtos que permitem a criação e manutenção de mundos virtuais, entre eles é possível mencionar o AWSINOS, OpenSimulator e o Projeto OpenWonderland. Estes projetos possuem características semelhantes ao sistema que se deseja desenvolver e por este motivo são apresentados nas seções a seguir.

### 2.3.1 AWSINOS – Active Worlds

O Active Worlds UNISINOS é um mundo virtual no qual os participantes são autores, convocados a experimentar o processo de aprendizagem em ação. A construção do AWSINOS exigiu a realização de um planejamento, delineando a proposta e contemplando os interesses profissionais dos envolvidos. Ficou estabelecido então que o mundo seria uma cidade baseada em bairros temáticos, dentre os quais: alemão, inglês e contos, para trabalhar respectivamente com línguas estrangeiras, educação infantil e séries iniciais (GRUPO DE PESQUISA EM EDUCAÇÃO DIGITAL GP E-DU UNISINOS/CNPQ, 2009, p. 1).

O AWSINOS faz uso da tecnologia de mundos virtuais Active Worlds. Cada mundo virtual está compreendido em um universo, e neste universo existe a possibilidade de estarem presentes diversos mundos, sejam ou não do mesmo proprietário. A conexão com o servidor é feita através de um programa cliente desenvolvido pela própria Active Worlds, a fim de conectar e ter acesso aos vários mundos virtuais. Dependendo da configuração estabelecida para cada mundo, o usuário pode construir objetos ou apenas interagir com a estrutura já existente conforme se pode notar na Figura 3.



Fonte: Active Worlds (2009).

Figura 3 - Active Worlds

Para realizar a conexão com o mundo virtual é necessário realizar um cadastro na Active Worlds. Este cadastro nada mais é do que uma lista de conexões com diversos servidores de mundos virtuais, assim que a Active Worlds registra o novo servidor, todos os clientes são atualizados e apresentam a possibilidade de conexão com os mundos virtuais já existentes anteriormente e os novos.

A Active Worlds mantém a lista de mundos virtuais disponíveis para conectar e todos os programas necessários para colocar em operação um novo mundo virtual, no entanto, este serviço não é gratuito. Instituições de ensino possuem preço diferenciado através de descontos, quem não possui infraestrutura suficiente a fim de comportar um mundo virtual pode contratar os serviços da Active Worlds, alugando servidores com os softwares já instalados e prontos para utilização, suporte técnico, redundância de servidores e conexão com a internet para manter o mundo virtual disponível vinte e quatro horas por dia.

### 2.3.2 OpenSimulator

OpenSimulator pode ser utilizado para criar um ambiente virtual, sendo acessado por vários tipos de programas clientes e com protocolos de comunicação diferentes. Uma das principais características do OpenSimulator é de simular o mesmo ambiente do Second Life, incluindo a compatibilidade com o programa cliente. Outras características podem ser atribuídas ao servidor simplesmente carregando módulos adicionais, como suporte para conversação por áudio.

Seu código fonte é disponibilizado sobre a licença *Berkeley Software Distribution* (BSD), permitindo que o código fonte seja aberto e comercializado. As aplicações que rodam dentro do mundo virtual disponibilizado pelo OpenSimulator podem ser escritas nas linguagens LindenLab Script, C# ou Visual Basic .NET, dando flexibilidade para os desenvolvedores escolherem a linguagem que estão mais familiarizados.

Outra característica do OpenSimulator é o fato de permitir a conexão de vários mundos. Para exercer esta função, a equipe de desenvolvimento do OpenSimulator criou uma tecnologia chamada de hypergrid, “[...] o hypergrid é uma extensão do OpenSimulator que permite conectar o OpenSimulator com outros OpenSimulators” (HYPERGRID, 2009, p. 1, tradução nossa),

Dentro da topologia do *hypergrid* existem diversas formas de conectar os mundos virtuais, sendo possível, por exemplo, que vários mundos diferentes sejam conectados e permitam que o usuário seja transportado de um mundo para outro ou fazendo com que cada parte conectada seja uma determinada localização de um mundo ainda maior.

A Figura 4 apresenta um ambiente virtual utilizando o OpenSimulator.



Fonte: Opensimulator (2009).

Figura 4 – OpenSimulator

Para utilizar o OpenSimulator é necessário possuir um Servidor de Banco de Dados (SGBD) que pode ser MySQL ou Microsoft SQL (MSSQL). Conforme a necessidade de disponibilizar os serviços do OpenSimulator é possível escolher se a configuração será do tipo *standalone* (todos os serviços em um único servidor) ou *grid mode* (cada serviço é executado em um servidor diferente).

O software necessário para colocar o mundo virtual em funcionamento está disponível para ser utilizado nas plataformas Microsoft Windows e distribuições Linux, sendo possível também realizar o download do código fonte e compilar o mesmo por conta própria.

A configuração em modo *grid* permite atender uma demanda muito maior devido ao fato de que cada serviço que o OpenSimulator necessita estar sendo executado em servidores diferentes, bem como caso necessário, pode-se ativar ou desativar somente partes do mundo virtual conforme necessidade.

É possível utilizar como exemplo uma instituição que possua vários mundos interligados, caso necessário é possível desligar os mundos que estão com pouca demanda e manter ativos somente os com maior demanda. O hardware utilizado nos servidores varia conforme a quantidade de clientes que se deseja atender.

Para realizar a conexão com o mundo virtual da instituição é necessário que o usuário realize o download do software cliente e após sua instalação na máquina cliente realizar a configuração da conexão com o servidor de mundos virtuais. O software cliente é fornecido por outros projetos que não necessariamente estejam vinculados com o OpenSimulator.

Como utilizador do OpenSimulator é possível destacar o Second Life (LINDEN LAB, 2010), sendo seu principal colaborador com o desenvolvimento da tecnologia do OpenSimulator.

### 2.3.3 Projeto OpenWonderland

OpenWonderland é um projeto desenvolvido inteiramente na linguagem Java. O principal objetivo é possibilitar a criação de mundos virtuais 3D para ambientes empresariais, onde os usuários podem comunicar-se, compartilhar aplicações, documentos ou realizar reuniões virtuais. A tecnologia utiliza a licença *General Public License*(GNU) versão 2.0, e possui código fonte aberto, sendo responsável pelo seu desenvolvimento a Sun Microsystems. A Figura 5 apresenta um ambiente utilizando esta tecnologia.





Fonte: Project Wonderland (2009).

Figura 5 – Imagem demonstrando Projeto OpenWonderland

O ambiente de desenvolvimento, onde o código-fonte do OpenWonderland pode ser modificado, ou o servidor, onde o mundo virtual será executado pode ser baseado em Windows, Linux/Unix e MacOS, sendo a única obrigatoriedade a instalação do Java com no mínimo a versão 1.6.

É importante ressaltar que o projeto OpenWonderland ainda não é um produto pois ainda está em desenvolvimento pela Sun Microsystems e está disponível para todos os que possuírem interesse em utilizar ou até mesmo alterar o código fonte.

O OpenWonderland não necessita que o usuário do mundo virtual instale o software cliente no computador que está utilizando, necessitando apenas iniciar a execução de uma aplicação JNLP. A utilização de uma aplicação JNLP torna simples e fácil a distribuição da aplicação para os usuários do mundo virtual independente de estarem utilizando o mundo virtual de suas casas ou em um computador compartilhado em uma *lanhouse* ou na biblioteca da universidade.

“No projeto OpenWonderland v0.5 você pode executar cada componente em servidores diferentes a fim de realizar o balanceamento de carga [...]”. (PROJECT WONDERLAND, 2010b, p. 1, tradução nossa). Para executar os componentes em servidores separados é necessário apenas que seja realizada a alteração de um parâmetro em um arquivo de configuração em cada servidor que irá fazer parte do balanceamento de carga.

Para disponibilizar um mundo virtual utilizando o OpenWonderland é necessário possuir um ou mais servidores físicos onde a configuração do hardware irá variar de acordo com a demanda que a instituição possuir. O sistema operacional do servidor necessita estar sobre a plataforma de 32 bits. A demanda do *link* de acesso à internet também varia conforme a demanda de clientes.

Ao colocar em funcionamento o mundo virtual em um servidor é necessário apenas disponibilizar aos usuários um *link* para a página que possui o software cliente. Não é necessário realizar nenhum tipo de cadastro ou informar à Sun Microsystems sobre o mundo virtual a fim de permitir o seu funcionamento ou liberação de acesso.

Entre os principais utilizadores do OpenWonderland está a própria Sun através do projeto MPK20 (ORACLE, 2009, p. 1) onde os colaboradores da Sun possuem aplicações para compartilhamento e colaboração de conteúdo.

Uma das características do OpenWonderland é a possibilidade de instalar módulos adicionais aumentando as capacidades do mundo virtual, sendo que estes módulos podem ser desenvolvidos por qualquer pessoa que tenha interesse. Na Figura 6 é apresentado o módulo que permite ao mundo virtual representar um sistema de partículas, neste caso está sendo realizada a simulação de neve.



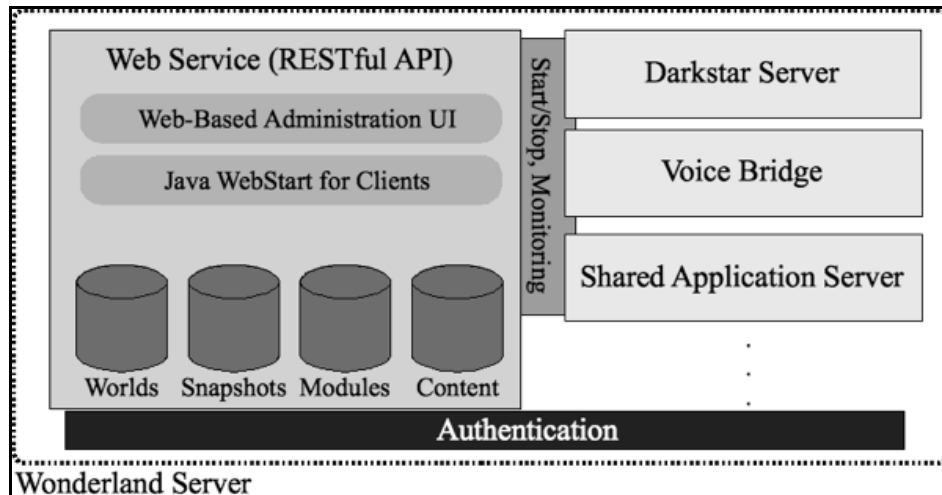
Fonte: Babiuch (2010).

Figura 6 - Apresentação do módulo do sistema de partículas

A configuração padrão do OpenWonderland é disponibilizada com alguns módulos já pré-configurados disponíveis como aplicações para objetos em 2D, entre os exemplos estão um “[...] quadro branco, um visualizador de apresentações, e um bloco de notas[...]” (OPENWONDERLAND, 2010e, p. 1, tradução nossa).

Por padrão, ao obter o código-fonte, o mundo virtual não possui controle de autenticação, todas as ferramentas de gerenciamento estão disponíveis aos usuários e visitantes, e o módulo de não autenticação está habilitado.

Através da Figura 7 é possível verificar a arquitetura do OpenWonderland e compreender como os módulos e demais componentes são integrados.



Fonte: Project Wonderland (2010c).

Figura 7 - Arquitetura do servidor OpenWonderland

Em Project Wonderland (2010a) é descrito o que é necessário para realizar a instalação do módulo de autenticação e desativar o módulo padrão denominado *security-session-noauth*. “[...] Os passos para configurar a autenticação consistem basicamente em remover o módulo padrão sem autenticação e substituí-lo por um que suporte autenticação [...]” (PROJECT WONDERLAND, 2010a, p. 1, tradução nossa). Este módulo de autenticação grava as contas de usuários no banco de dados do servidor que executa o mundo virtual.

O módulo padrão não é a única forma existente de autenticação, é possível utilizar o módulo de autenticação denominado *Ldaplogin*. Este módulo realiza a verificação das contas de usuários existentes em um servidor *Lightweight Directory Access Protocol* (LDAP).

A aparência do mundo virtual pode ser alterada a qualquer momento, sua aparência não está relacionada com nenhum módulo instalado no servidor, sendo possível, por exemplo, utilizar a ferramenta Google Sketchup para desenhar uma pequena cidade e disponibilizá-lo no OpenWonderland.

### 3 DESENVOLVIMENTO DO SISTEMA

Neste capítulo são descritas as particularidades do sistema proposto, a apresentação dos requisitos funcionais e não funcionais e os diagramas de caso de uso, os diagramas de classes e os diagramas de sequência utilizados em seu desenvolvimento.

#### 3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

Nesta seção são apresentados os requisitos funcionais, não funcionais e os respectivos casos de uso. O Quadro 1 apresenta os requisitos funcionais previstos para o sistema e sua rastreabilidade, ou seja, vinculação com os casos de uso associados.

<b>Requisitos Funcionais</b>	<b>Caso de Uso</b>
RF01: O sistema deverá permitir ao administrador o cadastramento de objetos que serão apresentados no mundo virtual.	UC01
RF02: O sistema deverá permitir ao administrador atualizar ou excluir os objetos existentes no mundo virtual.	UC02
RF03: O sistema deverá permitir ao usuário autenticar-se utilizando nome de usuário e senha.	UC03
RF04: O sistema deverá permitir a geração de relatório com estatísticas de contas criadas	UC04
RF05: O sistema deverá permitir ao visitante criar uma conta de usuário para acessar o mundo virtual.	UC05
RF06: O sistema deverá permitir definir permissões de acesso aos objetos que estão no mundo virtual	UC06
RF07: O sistema deverá permitir criar e manter grupos de usuários	UC07

Quadro 1 - Requisitos funcionais

No Quadro 2 são apresentados os requisitos não funcionais previstos para o sistema.

<b>Requisitos Não Funcionais</b>
RNF01: O sistema deverá utilizar banco de dados Apache Derby.
RNF02: O sistema deverá ser executado em um servidor que possa ser acessado através da internet.
RNF03: Os módulos de gerenciamento do sistema deverão possuir interface web.
RNF04: O servidor deverá estar instalado no sistema operacional Linux ou Solaris

Quadro 2 - Requisitos não funcionais

A Figura 8 e a Figura 9 representam os diagramas de casos de uso para os requisitos funcionais do sistema já apresentados no Quadro 1, sendo os seus detalhamentos apresentados no Apêndice A.

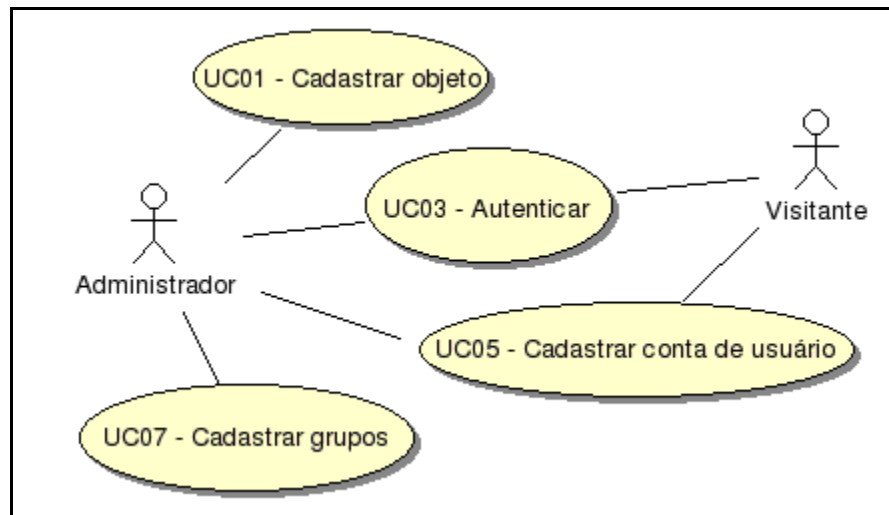


Figura 8 - Diagrama de casos de uso para cadastros

Através da Figura 9 é possível identificar o diagrama de casos para manutenção do sistema.

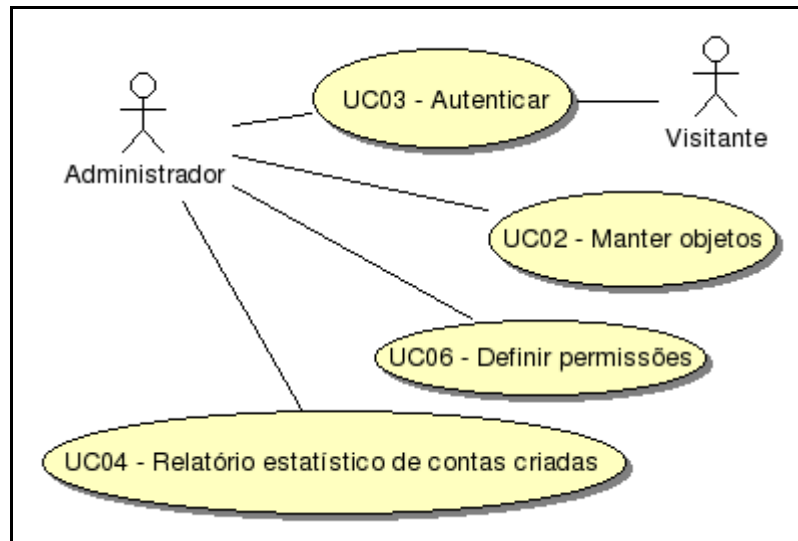


Figura 9 - Diagrama de casos de uso para manutenção

## 3.2 ESPECIFICAÇÃO

Para este trabalho foi necessário realizar algumas alterações no código-fonte original do OpenWonderland, de modo que permitisse o funcionamento de todo o sistema sem comprometer os vários módulos conectados ao mundo virtual. Partes das rotinas foram duplicadas para evitar comprometimento da estrutura principal do sistema.

Entre as alterações necessárias são destacadas duas principais alterações. Primeiro a alteração na forma de criação das contas de usuários, permitindo que o próprio visitante possa realizar o registro de sua conta através de um formulário web. Segundo a criação de objetos de forma dinâmica sem a necessidade de um usuário conectado ao mundo virtual.

A segunda alteração necessita de um arquivo XML, seu formato não está documentado pela equipe de desenvolvimento do OpenWonderland, no Apêndice B há trechos de uma conversa com Jonathan Kaplan, que descreve alguns procedimentos a serem realizados para obter exemplos dos arquivos XML necessários para a criação de objetos.

### 3.2.1 Criação de contas de usuários

O processo de criação de contas de usuário está implementada no servidor de mundo virtual. As modificações no código-fonte original do OpenWonderland resultaram na criação

de novas classes e rotinas a fim de evitar ou até mesmo prejudicar outros processos existentes no sistema e que dependem das rotinas originais do sistema. Na Figura 10 é possível verificar as classes `FindUser` e `CreateUser` que foram desenvolvidas para atender a nova forma de criação de contas. Estas classes fazem parte do módulo de autenticação de usuários `wonderland-security-session-auth`, o qual possui várias outras classes.

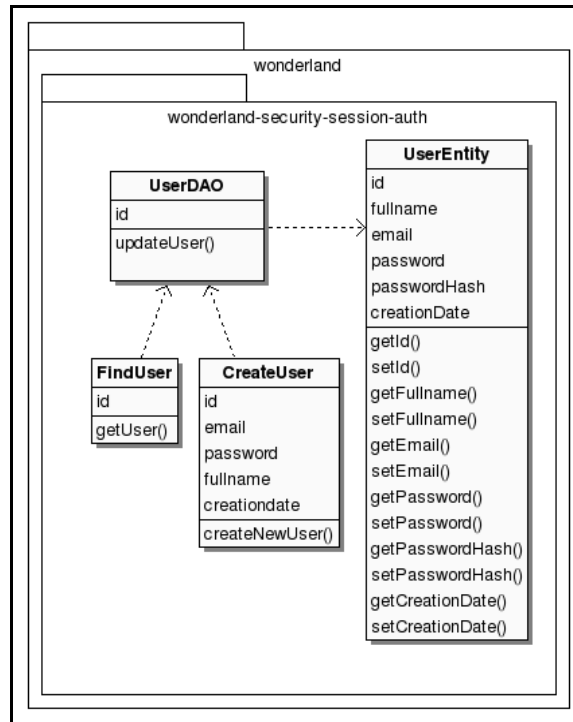


Figura 10 - Diagrama de classes para criação de contas de usuários

O pacote `wonderland-security-session-auth` possui as classes necessárias para a criação ou atualização de contas e faz parte do pacote principal `wonderland`, através da classe `UserDAO`, o método `updateUser` verifica o banco de dados e define se existe a necessidade de apenas atualizar informações em uma conta existente ou se é necessário a criação de uma nova conta.

A classe `UserEntity` é responsável por manter as informações da conta do usuário, seus métodos são utilizados para obter informações de uma determinada conta, ou para configurar uma nova conta. O atributo `creationDate` foi acrescentado afim de armazenar a data de criação da conta do usuário e permitir o desenvolvimento do relatório estatístico de contas criadas. Os métodos `getCreationDate()` e `setCreationDate()` foram implementados para definir ou buscar a informação do atributo `creationDate` na conta do usuário.

A implementação da classe `FindUser` tornou possível verificar se determinada conta de usuário existe no banco de dados do sistema. O método `getUser()`, foi implementado para

determinar se o usuário está tentando cadastrar uma identificação (`login`) exclusiva, ou seja, que ainda não foi cadastrada, retornando um resultado em formato binário (0 para identificação não existente e 1 para já existente).

### 3.2.2 Criação de objetos no mundo virtual

O formato original para criar objetos no mundo virtual necessita que o usuário esteja conectado ao mundo virtual e utilize as ferramentas disponibilizadas pelo cliente de acesso. Este procedimento foi contornado implementando-se outras rotinas que realizam a criação de objetos sem estar conectado ao mundo virtual, a Figura 11 apresenta o diagrama das classes `CreateObject`, `EditObject` e `Object` que foram implementadas para permitir a criação de objetos dinamicamente no mundo virtual.

A classe `wonderland-web-front` possui várias classes que atendem outros módulos do `OpenWonderland` e faz parte do pacote `wonderland`.

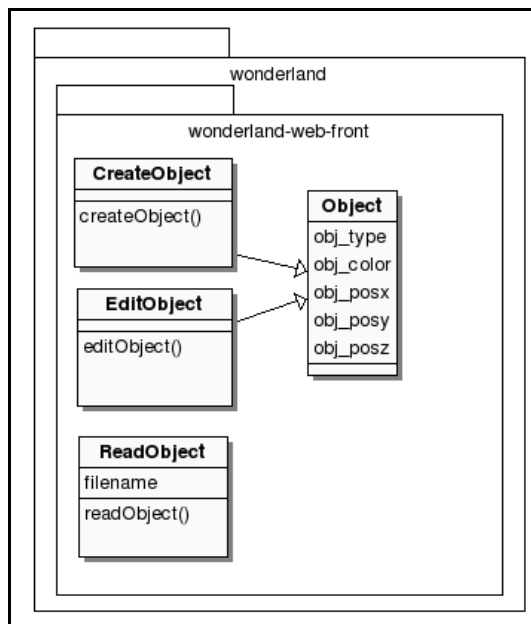


Figura 11 - Diagrama de classes para criação de objetos

Através da implementação da classe `CreateObject` foi possível gerar arquivos no formato XML contendo os parâmetros necessário para a criação de objetos. O arquivo é armazenado em um diretório específico do servidor de mundo virtual e posteriormente é executada a leitura do arquivo para a criação do objeto.

A edição das propriedades dos objetos foi possível com a implementação da classe `EditObject`, esta classe possui o método que é capaz de realizar alterações nos arquivos



XML gerados pela classe `CreateObject`.

A implementação da classe `ReadObject` teve como finalidade simplesmente realizar leitura das informações dos arquivos gerados pela classe `CreateObject` ou alteradas pela classe `EditObject`. Esta classe não possui nenhum método que seja capaz de realizar alterações nos arquivos.

O processo de criação de objetos é dividido em duas partes. A primeira parte consiste em o usuário administrador criar o objeto, definindo algumas configurações e gravando um arquivo em formato XML em um diretório específico do servidor. A segunda parte é executada no servidor, onde uma aplicação irá ler o arquivo XML gerado e criar o objeto no mundo virtual.

### 3.2.3 Diagramas de Sequência

O diagrama da Figura 12 apresenta o processo de criação de contas conforme é executado no servidor.

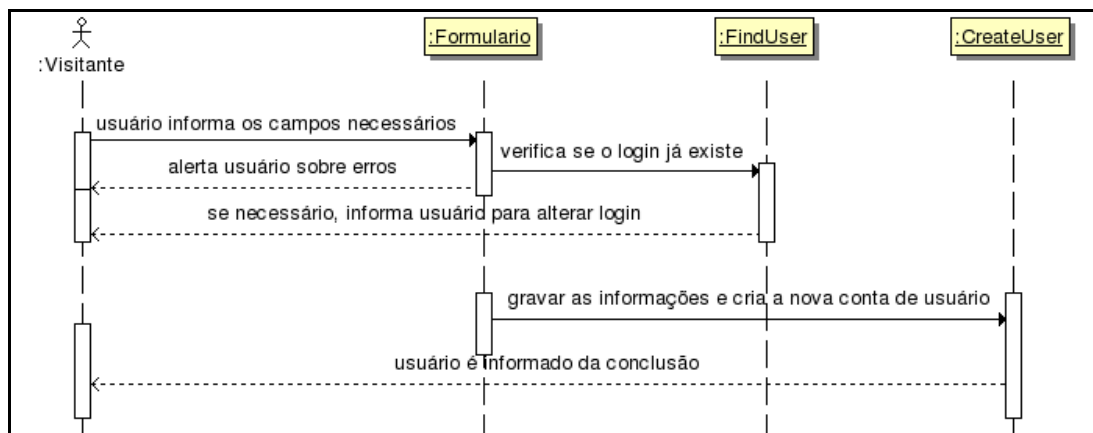


Figura 12 - Diagrama de sequência para criação de conta de usuário

O usuário, também conhecido como visitante, informa os dados necessário para a criação da conta, o sistema verifica se a conta ainda não existe no banco de dados, e caso necessário informa ao usuário para alterar a identificação. Se tudo estiver correto, a conta de usuário é gravada no banco de dados e o usuário é informado da conclusão do processo.

A Figura 13 apresenta a primeira parte do processo de criação de objetos.

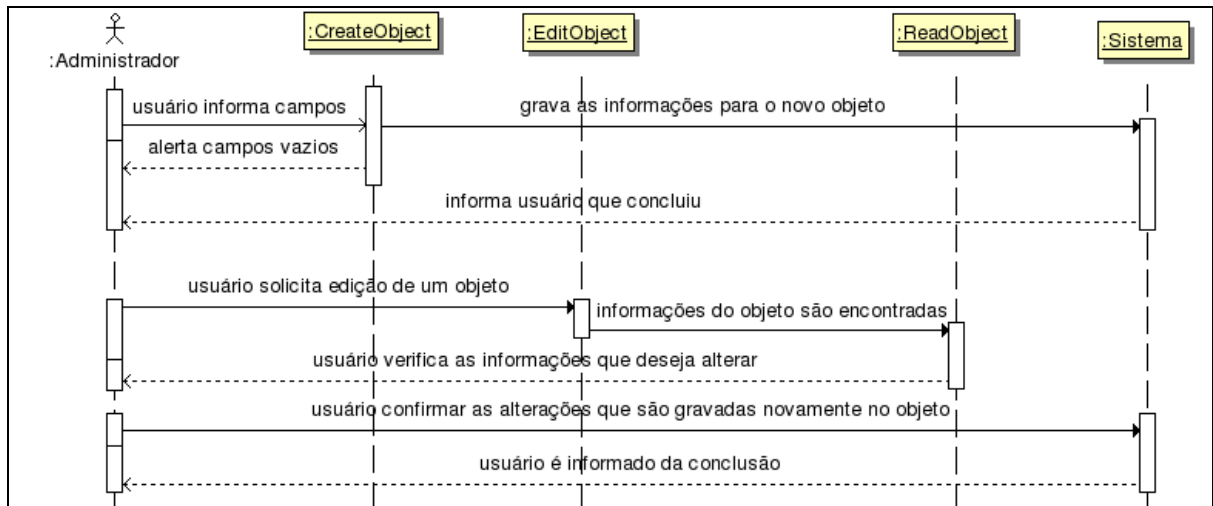


Figura 13 - Diagrama de sequência para criação de objetos

O usuário acessa o formulário que contém as informações necessárias de parametrização do objeto. Em seguida o sistema verifica se algum campo não está preenchido, se houver algo de errado, o usuário é alertado. Se tudo estiver correto, o sistema realiza a gravação de um arquivo, este arquivo contém as informações necessárias para o servidor de mundo virtual criar o objeto, por fim o usuário é informado sobre a conclusão do processo.

### 3.3 IMPLEMENTAÇÃO

Nessa seção são apresentados os aspectos a respeito da implementação e operacionalidade do servidor de mundo virtual, bem como as técnicas e ferramentas utilizadas. Também é apresentado na seção “3.3.2.6 - Utilização do mundo virtual pelos visitantes” descreve como os usuários podem realizar o acesso a este servidor.

#### 3.3.1 Técnicas e ferramentas utilizadas

A implementação do servidor com o software necessário para executar o mundo virtual é baseada na documentação desenvolvida pela equipe do OpenWonderland, que pode ser encontrada em OpenWonderland (2010c). Nesta página é possível encontrar outros documentos referentes ao servidor de mundo virtual, como por exemplo, os procedimentos de como deve ser efetuado o download do código-fonte para desenvolvedores realizarem

alterações, ou os passos para realizar a instalação de módulos adicionais.

A versão do OpenWonderland que foi utilizada é a *0.5 userpreview 4 build 4407*. O servidor utilizado para testes foi instalado com o sistema operacional Sun Solaris 10 10/09 x86, por se tratar do servidor, a única obrigação é que a máquina virtual Java versão 1.6 ou superior esteja instalada.

Para alteração do código-fonte do mundo virtual e alguns módulos, utilizou-se a ferramenta Netbeans IDE *6.5build 091101* em um computador com sistema operacional Linux kernel versão 2.6.31.12, a versão do Java utilizado é o *JDK 1.6.0update20*.

### 3.3.2 Operacionalidade da implementação

Nas seções abaixo são apresentados os procedimentos necessários para realizar o download do código-fonte dos módulos e sua instalação e como iniciar o servidor de mundo virtual.

#### 3.3.2.1 Implementando e iniciando um servidor de mundo virtual

Em Openwonderland (2010a) são descritos os procedimentos necessários para implementar um servidor de mundo virtual utilizando o OpenWonderland. A lista abaixo descreve resumidamente os passos e o quadro 3 apresenta as linhas de comandos necessárias para iniciar o servidor:

- a) realize o download do binário *Wonderland.jar*. Não renomeie o arquivo, o arquivo *jar* é grande e seu download pode demorar vários minutos;
- b) abra um terminal ou uma janela de comandos;
- c) vá até o diretório onde o arquivo *Wonderland.jar* foi salvo.

O servidor de mundo virtual, após iniciado, irá atender às requisições através da porta 8080, através desta porta será possível acessar sua console de gerenciamento, o acesso é realizado utilizando um navegador web informando o endereço IP ou nome DNS do servidor. Para que o servidor seja acessado através da internet, é importante verificar as configurações de *firewall* existentes na máquina que está hospedando o servidor de mundo virtual e a infraestrutura de rede a qual está conectada.

Linux, Solaris e OpenSolaris execute o comando

```
exemplo@localhost: />$ java -jar Wonderland.jar
```

Mac OS X

```
exemplo@localhost: />$ java -server -jar Wonderland.jar
```

No Windows é necessário substituir `_XX` pela versão que você possui instalado

```
C:\>"\Program Files\Java\jdk1.6.0_XX\bin\java" -jar Wonderland.jar
```

Quadro 3 - Linhas de comando para iniciar o servidor

Os passos apresentados são para servidores de mundo virtual sem qualquer tipo de personalização, ou seja, o programa está em seu formato original. A utilização do OpenWonderland sem nenhuma personalização não atende os requisitos necessários para o Interação FURB tornando obrigatório o download do código-fonte do servidor, no entanto esses passos são necessários para posteriormente executar o servidor de mundo virtual com as alterações.

O download do código-fonte do servidor pode ser executado utilizando uma ferramenta com suporte à SVN. O quadro 4 apresenta o comando necessário para esta tarefa utilizando um computador com o sistema operacional Linux instalado.

```
linux@localhost: /src>$ svn checkout \
http://openwonderland.googlecode.com/svn/trunk wonderland
```

Quadro 4 - Comando para realizar download do servidor

A conclusão deste comando pode demorar vários minutos, devido ao tamanho do código-fonte do servidor ser extenso (600 MB). Ao término do download é criada uma pasta no disco rígido nomeado de `trunk` e um subdiretório nomeado `wonderland` contendo o código-fonte.

Com o código-fonte do servidor em mãos é possível realizar as alterações necessárias para atender os requisitos do Interação FURB. O projeto do código-fonte do servidor esta no formato compreendido pela ferramenta Netbeans, no entanto o desenvolvedor está livre para realizar suas alterações em qualquer outro ambiente ou ferramenta que esteja familiarizado.

Para testar as alterações no código-fonte do servidor ou até mesmo disponibilizar uma nova versão é necessário realizar a compilação e empacotamento da aplicação. Este procedimento pode ser realizado de forma automatizada (compilação) no Netbeans ou manualmente através da linha de comando conforme apresentado no quadro Quadro 5.

```
linux@localhost: /src/trunk/wonderland>$ ant package
```

Quadro 5 - Comando para compilar e empacotar a aplicação

O comando apresentado no Quadro 5 irá gerar um arquivo nomeado `wonderland.jar`,

considerando que o comando foi executado no diretório `/src/trunk/wonderland`, o binário resultante da compilação estará em `/src/trunk/wonderland/dist`. Este arquivo pode ser utilizado para iniciar o servidor com as novas alterações conforme os procedimentos apresentados no início desta seção.

### 3.3.2.2 Implementando e instalado módulos adicionais

As modificações realizadas no servidor e nos módulos não seguem o padrão normal de funcionamento do mundo virtual do OpenWonderland, tornando necessário um programa externo chamado `connection-sample`. O programa é dividido em duas partes, a primeira parte trata-se de um módulo que deve ser instalado no servidor de mundo virtual. Este módulo será responsável por tratar as solicitações que serão realizadas pela segunda parte do `connection-sample`.

A segunda parte é um programa que é executado externamente no servidor. O principal desenvolvedor do programa `connection-sample` faz parte da equipe que mantém o código-fonte do OpenWonderland. Seu funcionamento permite que sejam criados objetos a partir de arquivos gravados no disco rígido do servidor.

Os arquivos necessitam apenas estar em formato XML e parametrizados de acordo com o objeto que se deseja criar. O Apêndice C possui como exemplo a estrutura utilizada para a criação de um bloco de notas.

Para que o módulo `connection-sample` seja utilizado no Interação FURB algumas alterações em seu código-fonte foram necessárias, o que obrigou a compilação e instalação do módulo de forma manual. Em OpenWonderland (2010d) são apresentados os passos para a instalação de módulos no servidor, estes procedimentos podem ser utilizados para instalar o módulo `connection-sample` que foi personalizado para o Interação FURB. A versão original do módulo é obtida no comando apresentado no Quadro 6.

Os procedimentos para obter o código-fonte dos módulos são semelhantes com os procedimentos que são utilizados para o servidor. No Linux, em um terminal, o Quadro 6 apresenta o comando para realizar o download do código-fonte dos módulos.

```
linux@localhost:/src>$ svn checkout \  
    http://openwonderland-modules.googlecode.com/svn/trunk/0.5\  
wonderland-modules
```

Quadro 6 - Comandos para download de código-fonte

A conclusão do comando pode levar vários minutos, visto que a quantidade de código-fonte é extensa (141 MB). Ao término do download os códigos-fonte estarão disponíveis no diretório `wonderland-modules` que será criado dentro da pasta `trunk`.

A partir deste ponto, é possível que o desenvolvedor realize alterações no código-fonte dos módulos. O projeto dos módulos é disponibilizado no formato da ferramenta Netbeans. Conforme já comentado sobre o código-fonte do servidor, pode ser utilizada qualquer outra ferramenta que o desenvolvedor esteja familiarizado.

Quando uma alteração é realizada em algum módulo, é necessário realizara compilação e instalação do módulo no servidor. Os módulos que são certificados para funcionarem com o OpenWonderland, são instalados automaticamente ao compilar o código-fonte do servidor de mundo virtual.

As alterações realizadas para o Interação FURB utilizam o módulo `connection-sample`, o qual não faz parte do conjunto oficial distribuído pelo OpenWonderland. A utilização deste módulo tornou necessária a instalação do módulo `connection-sample` manualmente.

Para realizar a instalação do módulo `connection-sample` em um terminal do Linux, é necessário ir até o diretório quem contém o código-fonte do módulo que foi alterado. No diretório utiliza-se o comando que pode ser verificado no Quadro 7. A única observação necessária é que o servidor de mundo virtual deve estar em execução para que a instalação do módulo `connection-sample` seja realizada.

```
lnx@loc:/src/trunk/wonderland-modules/unstable/connection-sample>$ ant \
deploy
```

Quadro 7 - Linha de comando para compilação de módulos

Após a instalação do módulo, o servidor passa a ter suporte para as instruções que são fornecidas pelo programa `connection-sample`.

A execução do programa `connection-sample` necessita que seja fornecido um usuário válido no mundo virtual, e também é necessário a execução da linha de comando `ant run` no diretório onde está o código fonte do módulo.

O processo é necessário devido à grande quantidade de bibliotecas que o módulo utiliza para funcionar. Este procedimento não é obrigatório e pode ser executado de outra forma, porém mais complicada. O Apêndice D apresenta uma conversa com o desenvolvedor do módulo `connection-sample` e qual a dificuldade em instanciar as várias bibliotecas necessárias para o programa funcionar corretamente.

### 3.3.2.3 Controle de Acesso e segurança das contas de usuário

A princípio utilizar o módulo de autenticação LDAP chamado `ldaplogin` foi desencorajado, pois sua versão atual está em fase de testes, não sendo recomendada para utilização em ambientes de produção. O estado de desenvolvimento do módulo pode ser encontrado em OpenWonderland (2010b).

Uma forma mista para permitir a integração com os usuários que já possuem vínculo com a FURB seria a possibilidade de integrar o módulo padrão de autenticação do mundo virtual com um utilitário que realize a importação das contas de usuários do LDAP da FURB. O utilitário regularmente sincronizaria a base de dados de usuários da FURB com o catálogo existente no mundo virtual. O desenvolvimento deste utilitário não foi iniciado, pois seria necessário recorrer a diversos utilitários de proteção contra intrusos, não sendo este o foco deste estudo.

Também não seria possível garantir a integridade do servidor que executa o mundo virtual, visto que o mesmo está acessível através da internet, tendo a possibilidade de o servidor ser invadido e as contas de usuários serem furtadas. A opção por utilizar o módulo padrão tem o menor impacto em caso de falha de segurança, visto que o vazamento de informações irá afetar unicamente o ambiente do mundo virtual.

Em contrapartida criou-se uma nova base de controle de acesso, não sendo este um problema expressivo, já que a maioria dos usuários que irão utilizar o sistema não possui vínculo ativo com a FURB, sendo em grande parte alunos de escolas ainda não matriculadas na FURB.

Com o intuito de facilitar o processo para acessar o mundo virtual, surgiu a opção de desenvolver uma rotina que permita aos usuários registrarem as suas próprias contas, visto que o mecanismo padrão do OpenWonderland permite apenas aos administradores do sistema registrarem as contas e definir a senha.

Para compreender a arquitetura do banco de dados e as rotinas envolvidas para registrar as contas de usuários, houve a necessidade de entrar em contato com a equipe de desenvolvimento do OpenWonderland. O Apêndice E contém parte da troca de mensagens realizada com os desenvolvedores. Através da mensagem é explicado onde devem ser procuradas as rotinas responsáveis pela manutenção de contas de usuários no código-fonte do OpenWonderland. Através do Quadro 8 é possível verificar parte do código-fonte que é sugerido pelos desenvolvedores do OpenWonderland para ser verificado.

```

155 public UserEntity updateUser(final UserEntity user) {
...
166     UserEntity cur = em.find(UserEntity.class, user.getId());
167     if (cur == null) {
168         // find the current value (if any) for this entity
169         em.persist(user);
170         return user;
171     }
172
173     // the entity exists - copy the password hash in if the
174     // passed in value is null
175     if (user.getPasswordHash() == null) {
176         user.setPasswordHash(cur.getPasswordHash());
177     }
178
179     //update the entity in persistence
180     em.merge(user);
181     return user;
182 }

```

Quadro 8 – Parte do código-fonte responsável pela criação de contas de usuários

No código-fonte é possível verificar que através do parâmetro `user` na linha 155, a linha 166 realiza uma busca no banco de dados buscando algum registro que tenha o mesmo `id` fornecido pelo parâmetro `user`, não obtendo resultado positivo, ou seja, a linha 167 possui resultado nulo. A linha 169 realiza a gravação da nova conta de usuário utilizando o método `persist`. A linha 175 é utilizada quando uma conta de usuário já existe no banco de dados e o atributo `passwordHash` está nulo, então é realizada a atualização do atributo com o método `setPasswordHash` e a gravação da atualização através da linha 180 com o método `merge`.

Com a criação dos procedimentos necessários para gravar a nova conta de usuário no banco de dados houve a necessidade de desenvolver uma interface para interagir com o usuário, conforme pode ser verificado na Figura 14.

**Informe os dados para criar uma nova conta.**

Identificação (login):  \*

Nome completo:  \*

Email:  \*

Senha:  \*

Campos obrigatórios: \*

Figura 14 - Formulário para criação de conta

A interface desenvolvida conforme a Figura 14 realiza validações verificando se algum campo não foi informado e se o nome identificação (`login`) é exclusivo, ou seja, não consta no banco de dados do mundo virtual. Esse processo é executado antes de permitir ao usuário registrar a nova conta.



A Figura 15 apresenta a tentativa de criar uma conta de usuário com uma identificação que já exista no banco de dados.

Informe os dados para criar uma nova conta.

**Identificação já existe!**

Identificação (login): admin \*

Nome completo: \*

Email: \*

Senha: \*

Registrar Limpapar formulário

Campos obrigatórios: \*

Figura 15 - Informação de conta que já existe

Outra validação não permite que as contas sejam criadas sem informar os campos obrigatórios, como exemplo, ao tentar realizar o registro da conta sem informar o campo *e-mail*, é apresentada uma mensagem de erro, conforme a Figura 16.

Informe os dados para criar uma nova conta.

**O campo [Email] não foi preenchido!**

Identificação (login): exemplo \*

Nome completo: Usuário de Exemplo \*

Email: \*

Senha: ●●●●●● \*

Registrar Limpapar formulário

Campos obrigatórios: \*

Figura 16 - Aviso de campo não preenchido

#### 3.3.2.4 Criação e atualização de objetos no mundo virtual

A fim de atender os requisitos do Interação FURB, é necessário que o mundo virtual possua componentes que permitam interação com ou entre os usuários. Tais objetos podem ser quadros de avisos, imagens, vídeos, entre outros, sendo que muitos deles podem necessitar que o usuário inicie algum evento para obter o resultado.

Para esta situação verificou-se a necessidade de implementar rotinas que tornassem possível criar dinamicamente objetos no mundo virtual, ou até mesmo atualizar as informações dos objetos que já existem.

No Quadro 9 é possível visualizar parte do código necessário para criar objetos no mundo virtual.

```

...
75     // initialize the login object
76     login = newProgrammaticLogin<WonderlandSession>(serverURL);
...
81     WonderlandSession session = login.login(username, passwordFile);
...
86     session.connect(ccc);
10    ...
100    //Create objects on virtual world
101    for (int i = 0; i <files.length; i++) {
102        if (!files[i].isDirectory()) {
103            //Get XML code from file
104            StringBuilder contents = new StringBuilder();
105            try {
106                input = newBufferedReader(
107                    newFileReader(files[i]));
107                while ((line = input.readLine()) != null) {
108                    contents.append(line);
109                    contents.append(System.getProperty(
110                        "line.separator"));
110                }
111                xml = contents.toString();
112                //Create cell according XML.
113                final CellIDcellID = ccc.createCell(xml);
114                System.out.println("created cell id: " cellID + " - " +
115                                    files[i].toString());
115            } catch (Exception ex) {
116                Logger.getLogger(ConnectionClientMain.class.getName()).
117                Log (Level.SEVERE, null, ex);
117            }
...

```

Quadro 9 - Parte do código-fonte para criação de objetos dinamicamente

A linha 76 do código-fonte apresentado no Quadro 9 inicia o processo para conectar ao servidor de mundo virtual utilizando o método `ProgrammaticLogin`, a linha 86 apresenta o programa iniciando conexão com o servidor de mundo virtual através do método `connect`. Após realizar a conexão, é iniciado o processo de criação de objetos, a alteração desta parte do código-fonte permitiu a leitura dos arquivos XML armazenados no servidor, a implementação pode ser verificada entre as linhas 100 e 113.

O método `createCell` na linha 113 foi alterado para recebe as informações que estavam no arquivo XML do disco rígido do servidor conforme código entre as linhas 107 e 111, criando o objeto no mundo virtual. O código existente entre as linhas 115 e 117 são utilizadas para exibição de erros. A Figura 17 apresenta o formulário que foi desenvolvido para ser utilizado na criação de um novo objeto a ser disponibilizado no mundo virtual.

Formulário para criação/edição de objetos no mundo virtual	
Tipo de objeto:	Post-it/Stick-Note ▾
Texto:	Este é o texto do bloco de post-it
Posição X:	10
Posição Y:	10
Posição Z:	0
Cor (RGB):	255:255:120
Proprietário (id):	admin
Usuários estão autorizados a modificar o objeto?:	<input type="radio"/> Sim <input checked="" type="radio"/> Não
<input type="button" value="Gravar!"/>	

Figura 17 - Formulário para criação de objetos

### 3.3.2.5 Criação e gerenciamento dos grupos de usuários

Através dos grupos de usuários é possível personalizar configurações de forma a permitir ou bloquear acesso a determinados objetos no mundo virtual para os usuários que fazem parte de um determinado grupo, como por exemplo, alunos de cursos diferentes. A Figura 18 apresenta a interface disponível para o administrador gerenciar os grupos de usuários.

## Manage Groups

Name	Members	Actions
admin	4	edit delete
students	3	edit delete

Add group

Figura 18 - Interface para gerenciamento de grupos

Ao clicar na opção `Addgroup` o sistema exibe o formulário para realizar o cadastro de um novo grupo, conforme pode ser verificado na Figura 19.

## Group Name

Group name:

Edit Group Members

Figura 19 - Formulário para criação de grupos de usuários

Após informar o nome do grupo, o administrador necessita apenas clicar na opção `EditGroupMembers` para ter acesso ao formulário que permite adicionar os usuários no grupo conforme pode ser visualizado na Figura 20.

## Group Members for teachers

Name	Owner	Actions
<input type="text" value="pernalonga"/>	<input checked="" type="checkbox"/>	delete
<input type="text" value="patolino"/>	<input type="checkbox"/>	delete
<input type="text" value="frajola"/>	<input type="checkbox"/>	delete
<input type="text" value="gaguinho"/>	<input type="checkbox"/>	delete

Cancel Add Group Member Save

Figura 20 - Incluindo usuários no grupo

### 3.3.2.6 Utilização do mundo virtual pelos visitantes

Nesta seção são apresentados os passos necessários para os visitantes realizarem a conexão com o servidor que executa o mundo virtual

A Figura 21 apresenta a página web que foi modificada para iniciar o programa cliente que será responsável por conectar o usuário ao mundo virtual. Esta aplicação é desenvolvida em linguagem Java e distribuída por JNLP.

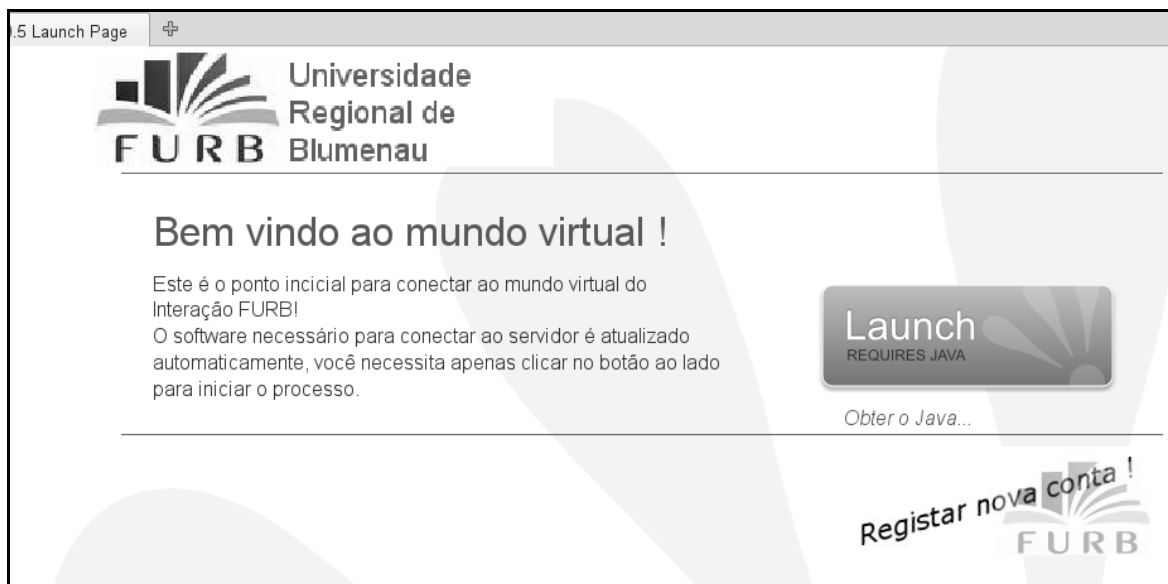


Figura 21- Página inicial para conectar ao mundo virtual

Ao clicar no botão `Launch` é iniciado o download do cliente conforme a Figura 22.

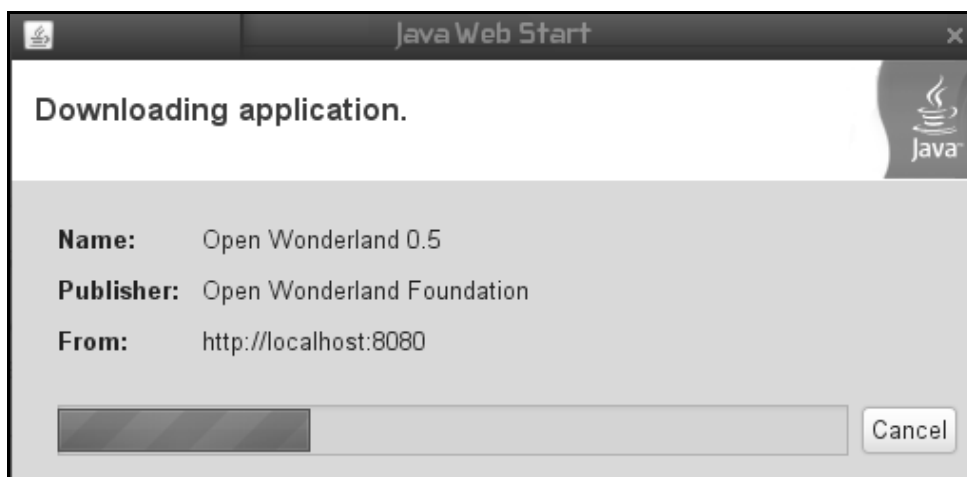


Figura 22 - Cliente do mundo virtual sendo executado

Devido ao fato de a aplicação ser distribuída por JNLP, não existe a necessidade de instalação, desta forma após o download estar completo a aplicação é iniciada automaticamente. A Figura 23 apresenta a aplicação já iniciada e solicitando a autenticação do usuário, a tela de autenticação foi personalizada para simplificar as informações que o

usuário necessita para realizar a conexão com o servidor de mundo virtual.

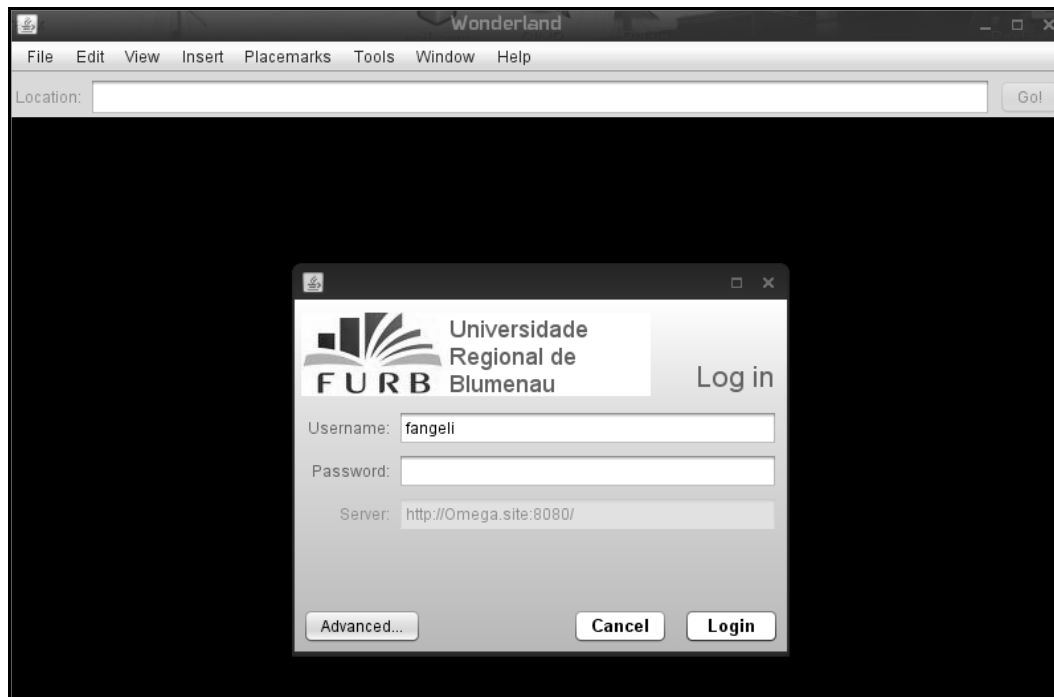


Figura 23 - Aplicação solicitando autenticação

Após o usuário estar identificado, a aplicação inicia uma conexão com o mundo virtual. E é realizado o download dos componentes necessários para que o cliente possa funcionar corretamente e atualizado, este processo é transparente para o usuário. A Figura 24 apresenta um usuário já conectado ao servidor.



Figura 24 - Usuário conectado ao servidor de mundo virtual

### 3.3.2.7 Definir ou editar permissões em objetos no mundo virtual

Ao disponibilizar o mundo virtual, o usuário administrador pode definir o que os usuários poderão modificar ou visualizar neste mundo. A configuração de permissões garante que não seja adicionado nenhum tipo de conteúdo que seja impróprio para o foco que o servidor de mundo virtual está direcionado, permitindo seu funcionamento de forma mais apropriada.

Após definir as configurações de segurança, somente o usuário administrador poderá realizar alterações na configuração. A segurança de um objeto é definida acessando suas propriedades clicando com o botão direito do mouse sobre o objeto e selecionando a opção *Properties*, conforme apresenta a Figura 25.



Figura 25 - Propriedades de um objeto no mundo virtual

A tela de propriedades do objeto será exibida. Caso o objeto não possua a funcionalidade de segurança, é necessário adicioná-la ao objeto conforme a Figura 26.

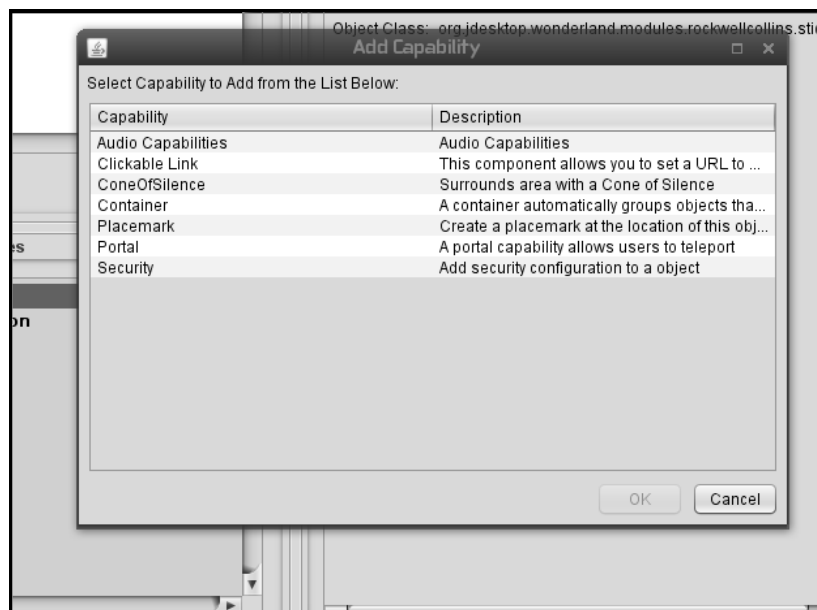


Figura 26 - Adicionando nova funcionalidade para o objeto

Selecionando a opção *Security* e clicando no botão *OK* o objeto passa a ter propriedades que são responsáveis pela segurança, definindo o que pode ser feito pelos usuários e quem é o proprietário do objeto. A Figura 27 apresenta as propriedades de um objeto que está com as configurações de segurança sendo editadas.

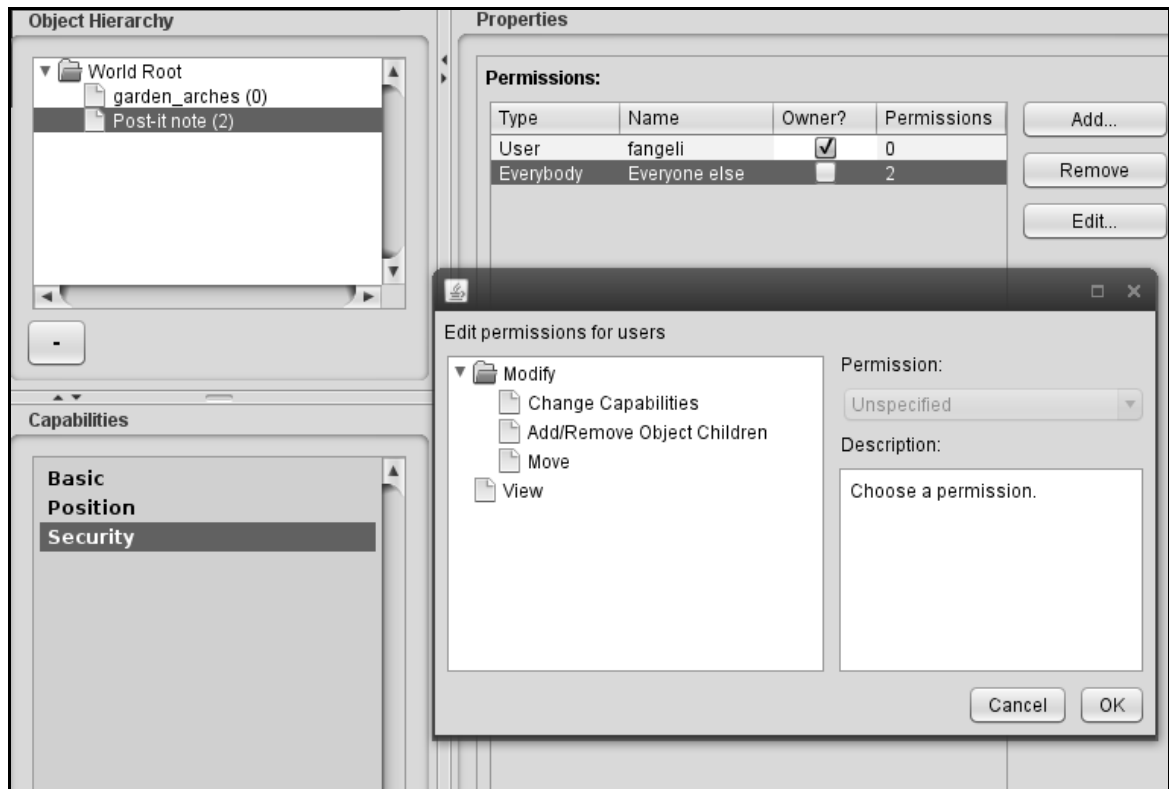


Figura 27 - Editando as permissões do objeto

Desta forma o ambiente virtual disponibilizado permite a representação do visitante através de um avatar, os materiais podem ser visualizados por objetos gráficos (como exemplo o bloco de notas conhecido como post-it) neste mundo virtual.

### 3.4 RESULTADOS E DISCUSSÃO

Através da realização deste trabalho foi possível identificar algumas das tecnologias existentes para simulação de mundos virtuais (ver seções 2.3.1, 2.3.2 e 2.3.3). Entre as possibilidades optou-se por utilizar o OpenWonderland. Este software possui código-fonte aberto e não necessita de registro na empresa que fornece o software para ser acessado por usuários que estão fora da rede da instituição.

Em relação ao software cliente de acesso ao mundo virtual, pode ser personalizado e



não necessita acessar nenhum servidor do distribuidor para obter a lista de servidores disponíveis para realizar conexão. Para permitir interatividade com o mundo virtual, é possível criar objetos e disponibilizá-los pelo ambiente do mundo virtual. A criação de objetos no mundo virtual é realizada de forma fácil e rápida, o que apontou como um fator favorável na adoção do OpenWonderland.

Em relação aos outros softwares de simulação de mundos virtuais, até a presente data o OpenWonderland é o único que permite adicionar novas funcionalidades ao mundo virtual simplesmente adicionando novos módulos. As alterações no código-fonte principal do servidor raramente necessitam de alterações.

De forma a compreender as necessidades do Interação FURB e qual o papel do mundo virtual, realizou-se reuniões com a supervisora do trabalho Priscila, funcionária do departamento de marketing da FURB. Estas reuniões tinham o intuito de alinhar o que o mundo virtual poderia fornecer e o que era esperado por parte da organização do Interação FURB. Através da Figura 28 é possível verificar um diagrama que apresenta o foco inicial que foi dado a este trabalho.

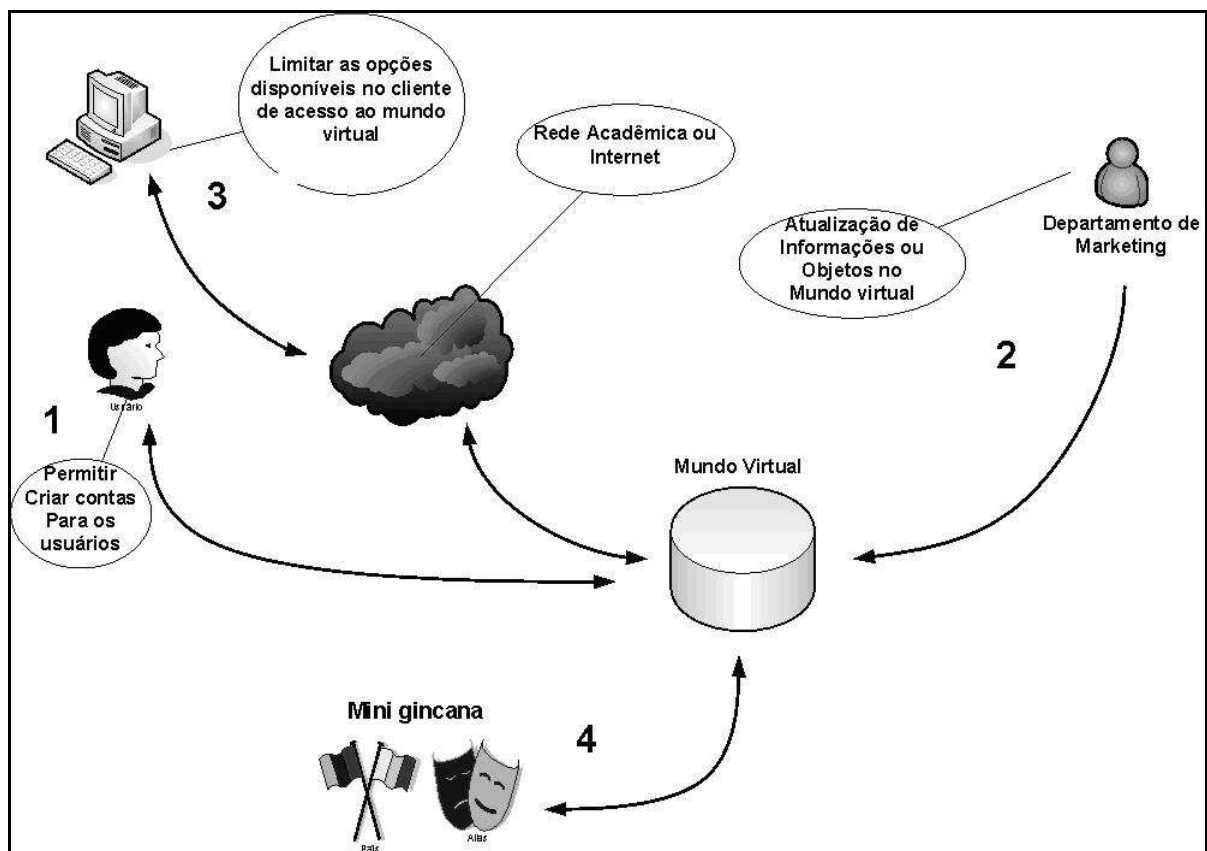


Figura 28 - Diagrama apresentado em reunião com supervisora

Através da Figura 28 é possível identificar quatro funcionalidades. A primeira funcionalidade é o controle de acesso de usuários, permitindo que o próprio visitante realize o

cadastro de sua conta de usuário, o processo original do OpenWonderland permite apenas que o administrador realize o cadastro de contas, e a segunda é a criação de objetos no mundo virtual de forma dinâmica, permitindo que o administrador realize o cadastro de objetos através de uma interface web, os objetos poderão ser materiais de ensino para os visitantes ou material utilizado em marketing, como panfletos ou cartazes, o formato original do OpenWonderland obriga ao administrador conectar ao mundo virtual e realizar a edição ou inclusão do objeto no mundo virtual. Já a terceira é a remoção de algumas opções do cliente de acesso, deixando somente os itens realmente necessários, o cliente padrão utilizado para conexão ao mundo virtual, possui diversas opções nos menus, uma parte destas opções não são úteis para os visitantes, realizar a remoção de alguns destes itens irá deixar o cliente utilizado para acesso ao mundo virtual com uma interface mais limpa. Por fim, a quarta funcionalidade seria a ideia em criar uma mini gincana, onde o usuário necessita clicar em algum objeto no mundo virtual e disponibilizar esta informação para utilização na distribuição de brindes, por exemplo.

No caso, o controle de acesso disponibilizado no mundo virtual permitia apenas ao administrador do servidor cadastrar as contas e definir a senha para conta. Este processo não seria viável para o Interação FURB, visto que a quantidade de participantes, normalmente ultrapassa 1.000 visitantes. Uma solução que poderia ser adotada para resolver este problema seria importar as contas do AD da FURB, utilizando um módulo do servidor de mundo virtual.

No entanto as implicações com a segurança da informação seriam muito críticas, havendo necessidade de implementar diversos mecanismos de proteção, visto que o servidor pode ser acessado através da internet e sofrer uma invasão. Para resolver este problema implementou-se uma rotina alternativa, a fim de facilitar a criação de contas para os visitantes (ver seção 3.3.2.3).

A implementação deste processo foi dificultada devido a grande extensão do código-fonte do OpenWonderland. Houve a necessidade de entrar em contato com os desenvolvedores do projeto a fim de solicitar ajuda para identificar onde o servidor de mundo virtual realiza a manutenção das contas de usuários (ver Apêndice E).

A segunda funcionalidade, que é a criação de objetos no mundo virtual, por padrão necessita que o usuário administrador esteja conectado ao mundo virtual. Identificou-se que este processo pode ser um pouco complicado e demorado para usuários que não tenham experiência com a interface do OpenWonderland.

Outra questão importante é que o computador onde está sendo feito o acesso deveria

realizar o download do software cliente para acessar o mundo virtual. Para facilitar este processo, verificou-se a possibilidade de desenvolver uma interface acessível através de qualquer navegador web, onde o usuário simplesmente preenche alguns campos em um formulário, e o objeto é criado no mundo virtual (ver seção 3.3.2.4).

Durante o desenvolvimento desta etapa, foram encontrados alguns problemas, pois a rotina padrão do OpenWonderland obriga a conexão com o mundo virtual e a utilização das ferramentas disponibilizadas pelo software cliente para a criação de objetos no mundo virtual.

Chegar a solução deste problema foi possível após uma conversa com um dos integrantes da equipe que desenvolve o OpenWonderland (ver Apêndice D). Esta conversa resultou na apresentação de um programa e módulo adicional com o nome de `connection-sample` para o servidor de mundo virtual.

O programa e módulo `connection-sample` faz uso de técnicas utilizadas durante as primeiras versões do OpenWonderland, onde o mundo virtual era criado a partir de arquivos dispostos no servidor. Após verificar o código-fonte e obter algumas orientações do desenvolvedor do `connection-sample` foi possível alterar sua estrutura e permitir que arquivos em formato XML fossem a base para a criação de objetos no mundo virtual.

Atualmente a modificação no `connection-sample` realiza a leitura dos arquivos que estão dispostos em um diretório no servidor, criando os objetos no mundo virtual. Este processo é realizado apenas durante a inicialização do programa e necessita ser aprimorado.

A remoção de itens da interface do cliente de acesso ao servidor de mundo virtual não foi possível de ser implementada. A quantidade de código-fonte a ser revisada é muito grande e necessita de mais tempo para ser completada.

Após definir um ambiente inicial no mundo virtual foram necessárias apenas algumas personalizações para iniciar uma mini gincana. As personalizações foram alguns objetos dispostos pelo ambiente e a configuração de um endereço para uma página web que registra o nome do usuário que a acessou. Esta funcionalidade pode ser utilizada no Interação FURB para, por exemplo, ao fim do evento disponibilizar brindes para os visitantes que conseguiram chegar na página ou concluíram alguma tarefa no mundo virtual.

Na reunião com a supervisora Priscila foi possível identificar pontos positivos em relação ao mundo virtual apresentado, os quais foram:

- a) conceito inicial de que o mundo virtual era estático, ou seja, que não haveria um avatar capaz de caminhar em um ambiente simulando uma área livre ou até mesmo uma sala ou prédio;
- b) a representação gráfica dos materiais para os visitantes e possíveis cartazes ou

panfletos para veiculação de marketing através de objetos cadastrados por uma interface web;

- c) facilidade de navegação pelo mundo virtual.

Durante a reunião também foram apontados alguns pontos a serem melhorados ou até mesmo implementados futuramente, conforme a aumenta de demanda do mundo virtual, as melhorias apontadas foram:

- a) disponibilizar outros matérias, figura, áudio, vídeo, arquivos PDF;  
 b) estender a mini gincana, de forma a conter mais pontos para serem visitados e aumentar a dificuldade para chegar aos objetos.

Com base em alguns testes que foram realizados no servidor e no cliente utilizado para acessar o servidor de mundo virtual foi possível identificar:

- a) a quantidade de memória utilizada para iniciar o servidor foi de aproximadamente 670 MB, a conexão de clientes ao servidor, aumentou o uso de memória em aproximadamente 170 MB por cliente conectado;  
 b) a transferência de dados entre cliente e servidor durante o processo de conexão foi de aproximadamente 30 MB utilizando um cenário de mundo virtual com 1.6 MB em uma rede ethernet 100 Mbps na primeira conexão, reduzindo a quantidade de dados transferida para aproximadamente 400 KB nas conexões posteriores;  
 c) a quantidade média de FPS registrada no cliente foi de 28,6 fps utilizando como cliente um computador com o sistema operacional Windows XP SP3;  
 d) a quantidade de FPS registrada no cliente foi de 29 fps utilizando como cliente um computador similar ao do item c com o sistema operacional Linux OpenSuSe 11.2.

Em relação aos trabalhos correlatos, o Quadro 10 apresenta uma comparação entre as funcionalidades existentes entre cada servidor de mundo virtual.

	OpenWonderland	OpenSimulator	AWSinos
Servidor de mundo virtual pode ser disponibilizado sem registro com o fornecedor da aplicação	Atende	Atende parcialmente	Não Atende
Cliente utilizado para acessar o mundo virtual dispensa instalação prévia para acessar o servidor	Atende	Não atende	Não atende
Permite executar o servidor em diferentes sistemas operacionais (Windows/Linux/Mac OS)	Atende	Atende	Atende parcialmente
Sem custos de licenciamento	Atende	Atende	Não atende
Código fonte aberto	Atende	Atende	Não atende

Quadro 10 - Comparação entre servidores de mundo virtual

## 4 CONCLUSÕES

Utilizar uma tecnologia de mundo virtual no Interação FURB, poderá permitir aos estudantes que ainda não possuem vínculo com a Universidade desfrutarem de um novo ambiente e terem contato com materiais educativos que possam ajudá-los a identificar qual curso seguir ao ingressar no ensino superior.

As tecnologias de mundo virtuais existentes são imensas, neste trabalho foi possível identificar algumas delas. Definir qual tecnologia utilizar foi um grande desafio, pois cada tipo possui um conjunto particular de características e funcionalidades, como por exemplo, o banco de dados utilizado na implementação, a possibilidade de criação de conteúdo no mundo virtual, ou a interoperabilidade entre sistemas operacionais.

A implementação do servidor que executa o mundo virtual com o OpenWonderland é muito simples se comparado com as outras tecnologias apresentadas neste trabalho. Sua única exigência é que o sistema operacional já esteja instalado com no mínimo a versão do Java 1.6, o administrador realiza a cópia de um arquivo com extensão JAR e o executa.

Durante o desenvolvimento identificou-se que o servidor de mundo virtual possui o seu banco de dados embarcado na aplicação. Ao contrário do que se imaginava, não existe a necessidade de um SGDB separado. Identificar o funcionamento deste banco de dados foi uma grande barreira. A troca de mensagens com a equipe que desenvolve o OpenWonderland ajudou a superar alguns dos obstáculos.

Entre as modificações que foram efetuadas, a que mais apresentou dificuldade foi à rotina para criação de objetos no mundo virtual. O procedimento padrão existente no OpenWonderland exige que a pessoa tenha conhecimento de como funcionam as ferramentas de edição existentes no cliente de acesso ao servidor e que o usuário esteja conectado ao mundo virtual.

Em relação aos objetos que podem ser disponibilizados aos visitantes no mundo virtual e utilizados pelo Interação FURB ou o departamento de marketing estão vídeos, textos, imagens, documentos em formato PDF e quadros brancos (similar aos quadros das salas de aula). Para cada tipo de objeto que se pretende disponibilizar no mundo virtual, é necessário o arquivo em formato XML com as propriedades que correspondem ao objeto (ver Apêndice B).

Desta forma, o desenvolvimento deste trabalho permitiu demonstrar uma forma de como a tecnologia de mundos virtuais pode ser utilizada a fim de receber os futuros

acadêmicos que estão chegando à universidade. A utilização do OpenWonderland permitiu a agregação de novos conhecimentos, principalmente na utilização da linguagem de programação Java unindo diversos programas com desenvolvimento separado e posteriormente convertidos em módulos, até o momento não havia participado no desenvolvimento de aplicações deste porte e com este nível de padronização. As técnicas utilizadas pela equipe que desenvolve o OpenWonderland ajudaram a acrescentar uma camada extra de dificuldade tornando o desenvolvimento deste trabalho ainda mais interessante.

#### 4.1 EXTENSÕES

Sugere-se as seguintes extensões para a continuação deste trabalho:

- a) realizar a integração entre o banco de dados do OpenWonderland e a base de dados de usuários da FURB, através do módulo `ldaplogin` existente em OpenWonderland (2010b). Permitindo que usuários que possuam vínculo ativo com a FURB não necessitem criar uma nova conta de usuário para o servidor de mundo virtual;
- b) verificar em Project Wonderland (2010c) o que é necessário para converter o método atual utilizado para criação de contas de usuários para o Interação FURB em módulo diminuindo o impacto na alteração de código-fonte do servidor de mundo virtual;
- c) ampliar a quantidade objetos disponíveis para criação no mundo virtual (atualmente é possível somente criar blocos de notas);
- d) modificar o cliente utilizado para conectar ao mundo virtual removendo as opções que não são necessárias;
- e) modificar o programa `connection-sample` de forma que ele possa realizar a atualização dos objetos criados no mundo virtual sem a necessidade de que o programa seja reiniciado.

## REFERÊNCIAS BIBLIOGRÁFICAS

ACTIVE WORLDS. **Active Worlds and Education**. Las Vegas[2009]. Disponível em: <<http://www.activeworlds.com/edu/index.asp>>. Acesso em: 28 maio 2010.

ALBUQUERQUE, Antonia Lucinelma; VELHO, Luiz. **Mundos Virtuais e Jogos por Computador: Pong – Um Estudo de Caso**. Instituto de Matemática Pura e Aplicada, Setembro 2001. Disponível em: <[http://www.visgrafimpa.br/Data/RefBib/PS\\_PDF/pong2001/reportech.pdf](http://www.visgrafimpa.br/Data/RefBib/PS_PDF/pong2001/reportech.pdf)>. Acesso em: 15 ago 2009.

BABIUCH, Ryan. **Particle Capability**. [2010]. Disponível em: <<http://blogs.openwonderland.org/2010/02/24/particle-capability/>>. Acesso em: 20 maio 2010.

CPQD. **Mundos virtuais: ferramenta com vantagens reais**. [2008?]. Disponível em: <<http://www.cpqd.com.br/highlights/341-mundos-virtuais-ferramenta-com-vantagens-reais.html>>. Acesso em: 28 maio 2010.

GOOGLE. **Google Sketchup 7**. [2010a] Disponível em: <<http://sketchup.google.com/product/gsu.html>>. Acesso em: 20 fev 2010.

\_\_\_\_\_. **Sketchup for mechanical design**. [2010b?]. Disponível em: <[http://docs.google.com/Doc?id=dc837t9h\\_117hgr2twgb](http://docs.google.com/Doc?id=dc837t9h_117hgr2twgb)>. Acesso em: 20 maio 2010.

GRUPO DE PESQUISA EM EDUCAÇÃO DIGITAL GP E-DU UNISINOS/CNPQ. **AWSINOS**. [2009?]. Disponível em: <<http://www.unisinos.br/pesquisa/educacao-digital/>>. Acesso em: 28 maio 2010.

HYPERGRID. **The OpenSim Hypergrid**. [2009] Disponível em: <<http://opensimulator.org/wiki/Hypergrid>>. Acesso em: 20 ago 2009.

LINDEN LAB. **Second Life**. [2010?]. Disponível em: <<http://secondlife.com>>. Acesso em: 22 fev 2010.

LOSANO, Monique. **Introdução ao Active Directory - Parte 1**. [2010?]. Disponível em: <<http://technet.microsoft.com/pt-br/library/cc668412.aspx>>. Acesso em: 21 abr 2010.

MARINS, Vânia; HAGUENAUER, Cristina; CUNHA, Gerson. **Realidade Virtual em Educação Criando Objetos de Aprendizagem com VRML**. Colabor@ - A revista digital da CVA-RICESU, v. 4, n. 15, p. 1-11, set 2007.

MARQUES, Renato. **Pessoas se encontram e se relacionam em MMOs**. Jogos firmam a Internet como novo espaço da sociedade para relacionamentos. Universia, 02 dez. 2005. Disponível em: <<http://www.universia.com.br/materia/materia.jsp?materia=9395>>. Acesso em: 20 ago 2009.

MESA, Alycia de. **Branding em mundos virtuais: uma realidade?**. 2007. Disponível em: <[http://www.cbm.rn.gov.br/contentproducao/aplicacao/sesed\\_cbm/arquivos/artigos/branding%20em%20mundos%20virtuais%20-%20marketing.pdf](http://www.cbm.rn.gov.br/contentproducao/aplicacao/sesed_cbm/arquivos/artigos/branding%20em%20mundos%20virtuais%20-%20marketing.pdf)>. Acesso em: 20 maio 2010.

MUI, Linda; PEARCE, Eric. **X Window System Administrator's Guide**. Sabastopol[1992]. O'Really & Associates INC, v. 8, p. 45, 1992.

OPENSIMULATOR. **ScreenShots**. [2009]. Disponível em: <[http://opensimulator.org/wiki/Image:OpenSim\\_001.jpg](http://opensimulator.org/wiki/Image:OpenSim_001.jpg)>. Acesso em: 20 ago 2009.

OPENWONDERLAND. **Binary Download**. [2010a?]. Disponível em: <[http://openwonderland.org/index.php?option=com\\_content&view=article&id=70&Itemid=64](http://openwonderland.org/index.php?option=com_content&view=article&id=70&Itemid=64)>. Acesso em: 18 abr 2010.

\_\_\_\_\_. **LDAP Login**. [2010?b]. Disponível em: <[http://openwonderland.org/download/modules/doc\\_details/189-ldap-login&cat=add\\_ons&Itemid=87&Itemid=87](http://openwonderland.org/download/modules/doc_details/189-ldap-login&cat=add_ons&Itemid=87&Itemid=87)>. Acesso em: 20 maio 2010.

\_\_\_\_\_. **Open Wonderland Documentation Wiki**. [2010c?]. Disponível em: <<http://code.google.com/p/openwonderland/wiki/OpenWonderland>>. Acesso em: 12 abr 2010.

\_\_\_\_\_. **Open Wonderland v0.5: Download, Build and Deploy Modules**. [2010d?]. Disponível em: <<http://code.google.com/p/openwonderland/wiki/DownloadBuildModules05>>. Acesso em: 24 maio 2010.

\_\_\_\_\_. **Using Open Wonderland for Enterprise Collaboration**. [2010e?]. Disponível em: <<http://www.openwonderland.org/resources/enterprise-users>>. Acesso em: 20 maio 2010.

ORACLE. **MPK20: Sun Virtual Workplace**. [2009?]. Disponível em: <<http://research.sun.com/projects/mc/mpk20.html>>. Acesso em: 20 ago 2009.

PROJECT WONDERLAND. **Project Wonderland v0.5: Configuring Authentication**. [2010a?]. Disponível em: <<http://wiki.java.net/bin/view/Javadesktop/ProjectWonderlandAuthentication05>>. Acesso em: 12 abr 2010.

\_\_\_\_\_. **Project Wonderland v0.5: Web-based Server Administration**. [2010b?]. Disponível em: <[http://wiki.java.net/bin/view/Javadesktop/ProjectWonderlandServerAdministration05#Advanced\\_Running\\_the\\_server\\_comp](http://wiki.java.net/bin/view/Javadesktop/ProjectWonderlandServerAdministration05#Advanced_Running_the_server_comp)>. Acesso em: 09 fev 2010.



\_\_\_\_\_. **Project Wonderland v0.5: Working with modules.**[2010c?]. Disponível em: <<http://wiki.java.net/bin/view/Javadesktop/ProjectWonderlandWorkingWithModules05>>. Acesso em: 04 jun 2010.

\_\_\_\_\_. **Project Wonderland: Toolkit for Building 3D Virtual Worlds.** [2009?]. Disponível em: <<https://lg3d-wonderland.dev.java.net/>>. Acesso em: 20 ago 2009.

ROSA JR., Onivaldo. **Ambientes Virtuais Cooperativos LRVCHAT3D, um estudo de caso.** In: SBC Symposium on Virtual Reality, 4., 2001, Florianópolis. Anais. Florianópolis: UFSC: SBC. p. 1-11.

UNIVERSIDADE REGIONAL DE BLUMENAU. **História.** Blumenau, [2009a]. Disponível em: <<http://www.furb.br/site/interna.php?secao=94>>. Acesso em: 20 ago 2009.

UNIVERSIDADE REGIONAL DE BLUMENAU. **Interação FURB.** Blumenau, [2009b]. Disponível em: <<http://www.furb.br/interacao>>. Acesso em: 31 jul 2009.

ZUKOWSKI, John. **Deploying Software with JNLP and Java Web Start.**[2010?]. Disponível em: <<http://java.sun.com/developer/technicalArticles/Programming/jnlp/>>. Acesso em: 09 fev 2010.

## APÊNDICE A – Detalhamento dos casos de uso

No quadro 10 é apresentado o detalhamento do caso de uso UC01 – Cadastrar objeto.

**Objetivo:**

Permitir ao administrador realizar o cadastro de objetos no servidor de mundo virtual.

**Pré-condições:**

- O usuário deverá estar logado.
- O usuário deverá possuir privilégio de administrador.

**Pós-condição:**

- Um arquivo contendo as informações fornecidas pelo administrador sobre o objeto deve estar salvo no disco rígido do servidor em formato XML.

**Cenário principal:**

- 1) O administrador acessa a página web responsável pela administração do servidor.
- 2) O sistema solicita autenticação.
- 3) O administrador fornece as credenciais necessárias.
- 4) O administrador clica na opção "NewObject".
- 5) O sistema exibe a tela com os campos necessário para efetuar o registro.
- 6) O administrador fornece as informações nos devidos campos.
- 7) O administrador clica na opção "Gravar!".
- 8) O sistema realiza a gravação do arquivo contendo as informações fornecidas pelo administrador e exibe a mensagem "Congratulations, object created successfully!".

**Cenário de exceção:**

- 1) No 6º passo caso o administrador deixe de fornecer alguma das informações necessárias, é exibida a mensagem "O campo XYZ não foi preenchido!".

Quadro 11 - Detalhamento do caso de uso UC01 - Cadastrar objeto

No Quadro 12 é apresentado o detalhamento do caso de uso UC02 – Manter Objetos.

**Objetivo:**

Permitir ao administrador realizar a manutenção dos objetos cadastrados no servidor de mundo virtual.

**Pré-condições:**

- O usuário deverá estar logado.
- O usuário deverá possuir privilégio de administrador.

**Pós-condição:**

- A lista de objetos disponíveis para serem utilizados no mundo virtual deve estar atualizada.

**Cenário principal:**

- 1) O administrador acessa a página web responsável pela administração do servidor.
- 2) O sistema solicita autenticação.
- 3) O administrador fornece as credenciais necessárias.
- 4) O administrador clica na opção "Manage Objects".
- 5) O sistema exibe uma lista com os objetos disponíveis.
- 6) O administrador clica na opção "Edit" do objeto que deseja atualizar.
- 7) O sistema exibe a tela com os campos necessário para efetuar a manutenção do registro.
- 8) O administrador fornece as informações nos devidos campos.
- 9) O administrador clica na opção "Gravar!".
- 10) O sistema realiza a gravação do arquivo contendo as informações fornecidas pelo administrador
- 11) O sistema retorna para a lista de objetos disponíveis.

**Cenário de exceção:**

- 1) No 6º passo caso o administrador clique na opção "Delete", o objeto será excluído do servidor e o sistema irá diretamente para o passo 11.
- 2) No 8º passo caso o administrador deixe de fornecer alguma das informações necessárias, é exibida a mensagem "O campo XYZ não foi preenchido!".

Quadro 12 - Detalhamento do caso de uso UC02 - Manter objetos

No Quadro 13 é apresentado o detalhamento do caso de uso UC04 – Relatório de contas criadas.

**Objetivo:**

Permitir ao administrador obter informações estatísticas sobre a criação de contas no servidor de mundo virtual.

**Pré-condições:**

- O usuário deverá estar logado.
- O usuário deverá possuir privilégio de administrador.

**Pós-condição:**

- O relatório deverá ser exibido na tela.

**Cenário principal:**

- 1) O administrador acessa a página web responsável pela administração do servidor.
- 2) O sistema solicita autenticação.
- 3) O administrador fornece as credenciais necessárias.
- 4) O administrador clica na opção "Created accounts Report".
- 5) O sistema exibe o relatório gerado.

Quadro 13 - Detalhamento do caso de uso UC04 - Relatório de contas criadas

No Quadro 14 é apresentado o detalhamento do UC05 – Cadastrar conta de usuário.

**Objetivo:**

Permitir ao visitante realizar o cadastro de uma conta de usuário para acessar o mundo virtual.

**Pós-condição:**

- Uma conta de usuário deve estar cadastrada no sistema.

**Cenário principal:**

- 1) O visitante acessa a página web responsável pelo cadastro de contas de usuários.
- 2) O sistema exibe a tela com os campos necessários para realizar o cadastro da conta.
- 3) O visitante fornece as informações necessárias.
- 4) O visitante clica na opção "Registrar".
- 5) O sistema realizar o registro da nova conta.
- 6) O sistema exibe a mensagem "Conta criada com sucesso!".

**Cenário de exceção:**

- 1) No 3º passo caso o visitante deixe de fornecer alguma das informações necessárias, é exibida a mensagem "O campo XYZ não foi preenchido!".

Quadro 14 - Detalhamento do caso de uso UC05 - Cadastrar conta de usuário

No Quadro 15 é apresentado o detalhamento do UC06 – Definir permissões.

**Objetivo:**

Permitir ao administrador definir permissões aos objetos que fazem parte do mundo virtual.

**Pré-condições:**

- O usuário deverá estar logado.
- O usuário deverá possuir privilégio de administrador.

**Pós-condição:**

- Os objetos devem ter suas configurações de segurança atualizadas.

**Cenário principal:**

- 1) O administrador acessa a página web responsável por conectar ao mundo virtual.
- 1) O sistema solicita autenticação.
- 2) O administrador fornece as credenciais necessárias.
- 3) O administrador clica em um objeto que está no mundo virtual e clica na opção "Properties".
- 4) O sistema exibe uma lista contendo as propriedades do objeto.
- 5) O administrador clica na opção "+" da aba "Capabilities", seleciona a opção "Security" e clica no botão "Ok".
- 6) O sistema retorna para a tela de propriedades do objeto.
- 7) O administrador clica na opção "Security" na aba "Capabilities" e modifica as permissões conforme necessário.
- 8) O sistema realiza a gravação das alterações feitas pelo administrador.

Quadro 15 - Detalhamento do caso de uso UC06 - Definir permissões

No Quadro 16 é apresentado o detalhamento do caso de uso UC07 – Cadastrar grupos.

**Objetivo:**

Permitir ao administrador realizar o cadastro de grupos de usuários no sistema.

**Pós-condição:**

- O usuário deve estar logado.
- O usuário deve possuir permissões de administrador.

**Cenário principal:**

- 1) O administrador acessa a página web responsável pela administração do servidor.
- 2) O sistema solicita autenticação.
- 3) O administrador fornece as credenciais necessárias.
- 4) O administrador clica na opção "Manage Groups".
- 5) O sistema exibe uma lista com os grupos já cadastrados.
- 6) O administrador clica na opção "Add group".
- 7) O sistema exibe a tela com os campos necessário para efetuar o registro do grupo.
- 8) O administrador fornece o nome desejado para o grupo e clica na opção "Edit Group Members".
- 9) O sistema exibe uma lista de usuários vazia.
- 10) O administrador clica na opção "Add group member".
- 11) O sistema exibe a tela com os campos necessários para informar o usuário que irá fazer parte do grupo.
- 12) O administrador clica na opção "Save".
  
- 13) O sistema realiza a gravação do grupo contendo as informações fornecidas pelo administrador
- 14) O sistema exibe a lista de grupos disponíveis.

**Cenário de exceção:**

- 1) No 6º passo caso o administrador clique na opção "Edit" de qualquer grupo, o sistema irá para o passo 9 e irá exibir uma lista com os usuários que fazem parte do grupo sendo possível excluir ou adicionar usuários do grupo.

Quadro 16 - Detalhamento do caso de uso UC07 - Cadastrar grupos

## APÊNDICE B – Procedimentos para obter o código XML de um objeto.

On 12 April 2010 21:16, <flaviano.ang...@gmail.com> wrote:

Hi people, I read source code of connection-sample. Runed, tested, and ok works.

Question: If I need to create for example PDFViewer e StickNote, different from connection-samples that create a colored cube.

Is possible ? I noted PDFViewer and StickNote is separated modules. Is possible to use these modules on connection-sample ?

I'm a little lost on how to make this kind of integration.

----- Forwarded message -----

From: Jonathan Kaplan <jonathan...@gmail.com>

Date: 12 abr, 22:55 Subject: Creating Objects To: Open Wonderland Forum

You are on the right track with the connection sample and WFS. Here is how I would do it:

1. have your web app generate a WFS XML file. The XML file will be Different for each cell type you want to create. What I would do is start with An empty world, then add the cell type you want (like a whiteboard) to the world using the normal mechanisms in Wonderland. After the cell is in world, create a snapshot in the "Manage Worlds" area of the web UI. Now go into ~/.wonderland-server/0.5-dev/wfs/snapshots/<snapshot-name>/world-wfs/ directory and look for the -wlc.xml file corresponding to the object you just took a snapshot of. It will look something like this:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<whiteboard-cell-svg>
  <name>Whiteboard</name>
  <metadata/>
  <position-component>
    <translation>
      <x>-7.2930264</x>
      <y>4.2708335</y>
      <z>-7.888997</z>
    </translation>
```

Quadro 17 - Procedimentos para obter o código XML de um objeto no OpenWonderland

## APÊNDICE C – Exemplo de arquivo XML utilizado para criação de objetos.

No Quadro 18 é apresentado o conteúdo de um arquivo XML para criar no OpenWonderland um bloco de notas.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<stickynote-cell>
<name>Bloco de Notas</name>
<metadata/>
<position-component>
<translation>
<x>10</x>
<y>10</y>
<z>10</z>
</translation>
<bounds>
<type>SPHERE</type>
<x>1.0</x>
<y>1.0</y>
<z>1.0</z>
</bounds>
<scale>
<x>1.0</x>
<y>1.0</y>
<z>1.0</z>
</scale>
<rotation>
<x>0.0</x>
<y>-1.0</y>
<z>0.0</z>
<angle>1.9471655</angle>
</rotation>
</position-component>
<pixelScaleX>0.01</pixelScaleX>
<pixelScaleY>0.01</pixelScaleY>
<preferredWidth>640</preferredWidth>
<preferredHeight>480</preferredHeight>
<pixelScaleX>0.01</pixelScaleX>
<pixelScaleY>0.01</pixelScaleY>
<noteType>GENERIC</noteType>
<noteText>Reservado para texto</noteText>
<noteName>Nome do componente</noteName>
<noteAssignee>Person</noteAssignee>
<noteDue>Today</noteDue>
<noteStatus>Not yet</noteStatus>
<noteColor>255:255:153</noteColor>
</stickynote-cell>
```

Quadro 18 - Exemplo de arquivo XML



## APÊNDICE D – Conversa com Jonathan Kaplan sobre programa connection-sample

O Quadro 19 apresenta parte da troca de mensagens com Jonathan Kaplan, que desenvolveu o programa connection-sample.

On Fri, Apr 16, 2010 at 8:19 AM, Flaviano<flaviano...@gmail.com>wrote:

Hi people !

I try to use connection-sample.jar on line command (java -jar connection-sample.jar).

But I receive message:

Failed to load Main-Class manifest attribute from connection-sample.jar

When start program on netbeans, works ok.

Is need another config ?

----- Forwarded message -----

From: Jonathan Kaplan <jonat...@gmail.com>

Date: 19 abr, 00:43 - Subject: connection-sample

To: Open Wonderland Forum

You can use "ant run" from the command line in the connection-sample/ directory to run the connection sample standalone client. The client cannot run using "java -jar" because it relies on a huge number of libraries in the classpath. If you really want to run it without ant, you can see the command that ant runs with "ant -v run". The command is:

java -classpath<huge\_classpath>  
org.jdesktop.wonderland.modules.connections.sample.main.ConnectionClientMainhttp://localhost:8080/admin

On my system, <huge\_classpath> is:

'/Users/jkaplan/proj/wl-latest/wonderland/core/ext/jme20/target/  
jme.jar:/Users/jkaplan/proj/wl-latest/wonderland/core/ext/jme20/target/  
jme-gamestates.jar:/Users/jkaplan/proj/wl-latest/wonderland/core/ext/  
jme20/target/jme-awt.jar:/Users/jkaplan/proj/wl-latest/wonderland/core/  
latest/wonderland/core/ext/jme20/target/jme-effects.jar:/Users/jkaplan/  
jkaplan/proj/wl-latest/wonderland/core/ext/jme20/target/jme ...

Quadro 19 - Troca de mensagens sobre o programa connection-sample

## APÊNDICE E – Conversa com a equipe de desenvolvimento do Projeto Wonderland

No Quadro 20 é possível verificar a orientação da equipe de desenvolvimento do OpenWonderland referente qual parte do código-fonte do mundo virtual é responsável pela manutenção das contas de usuários.

<http://forums.java.net/jive/thread.jspa?messageID=387490&#387490>

Create user ... Posted: Feb 18, 2010 12:35 PM

Hello everybody,

I've been looking for the source code as is the process of creating users, but could not identify.

Someone has a little light to guide me?

Re: Create user ...

Posted: Feb 19, 2010 11:20 AM in response:

There are a bunch of ways to approach this. The user management code is in the wonderland-modules project in the stable/security-session-auth module.

You could look there to see how user accounts are created and maintained in the wonderland database. ...

Quadro 20- Conversa com equipe do OpenWonderland