



Universidade Federal do Ceará

Mestrado em Ciência da Computação
Departamento de Computação

Dissertação de Mestrado

VirTraM: Um Framework para o Desenvolvimento de Treinamentos Utilizando Realidade Virtual em Dispositivos Móveis

Edgar Marçal de Barros Filho

Fortaleza – CE, 2005

Edgar Marçal de Barros Filho

VirTraM: Um Framework para o Desenvolvimento de Treinamentos Utilizando Realidade Virtual em Dispositivos Móveis

Dissertação apresentada ao Mestrado de Ciência da Computação da Universidade Federal do Ceará (UFC), como requisito parcial para a obtenção do título de Mestre em Ciência da Computação

Orientador:
José Riverson Araújo Cysne Rios

Co-orientadora:
Rossana Maria de Castro Andrade

Fortaleza – CE, 2005

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus pela força que tem me dado dia após dia ao longo dessa caminhada.

Aos meus pais e irmãos, pelo apoio fundamental para conseguir terminar essa dissertação.

À minha Mel, por seu amor e paciência nessa reta final.

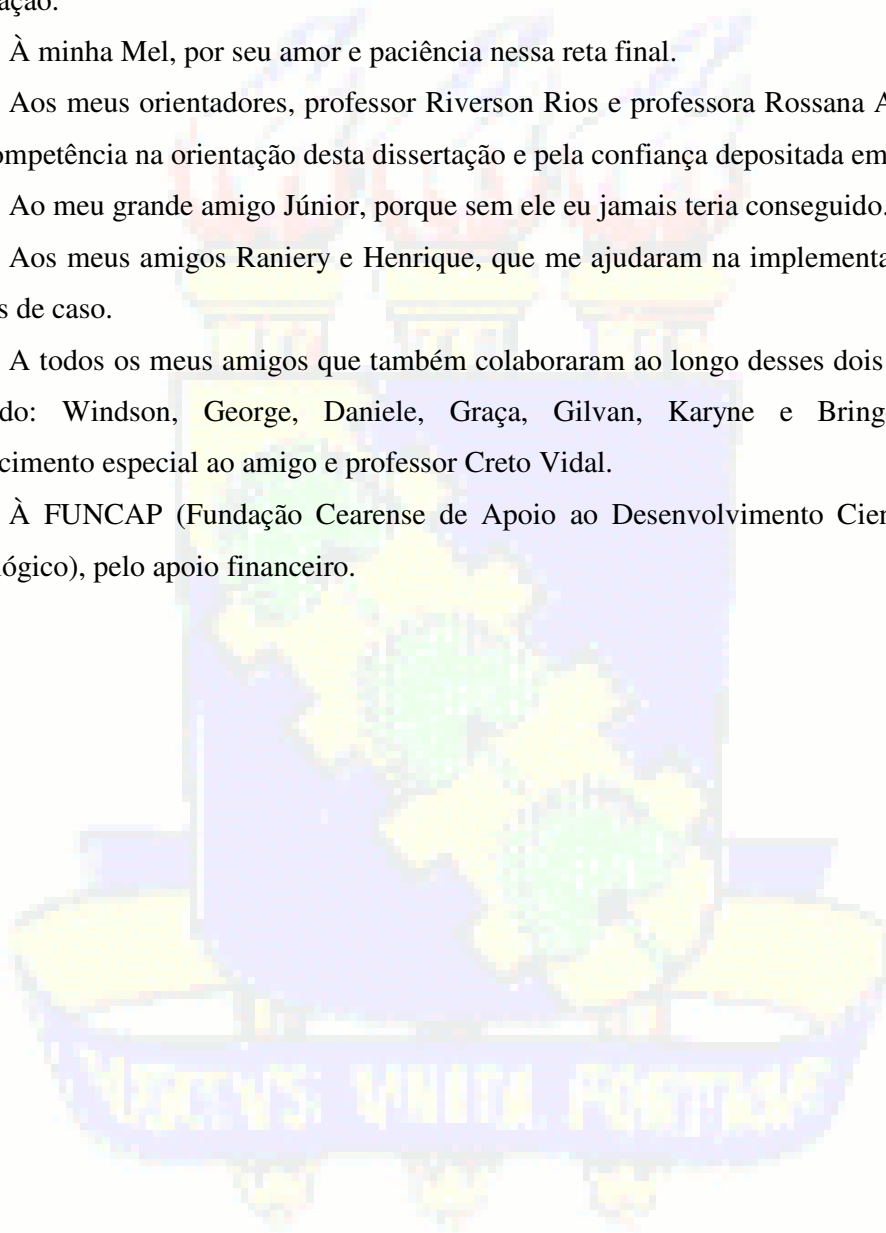
Aos meus orientadores, professor Riverson Rios e professora Rossana Andrade, pela competência na orientação desta dissertação e pela confiança depositada em mim.

Ao meu grande amigo Júnior, porque sem ele eu jamais teria conseguido.

Aos meus amigos Raniery e Henrique, que me ajudaram na implementação dos estudos de caso.

A todos os meus amigos que também colaboraram ao longo desses dois anos de mestrado: Windson, George, Daniele, Graça, Gilvan, Karyne e Bringel. Um agradecimento especial ao amigo e professor Creto Vidal.

À FUNCAP (Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico), pelo apoio financeiro.



RESUMO

A computação móvel tem se destacado na área educacional como uma tecnologia inovadora. O termo Mobile Learning ou m-Learning tem sido empregado para tratar da utilização de dispositivos de comunicação sem fio, de forma transparente e com alto grau de mobilidade, como parte de um modelo de aprendizado integrado. Educação pode ser vista como um processo de exploração, descoberta e construção de conhecimento, através de ambientes tridimensionais. Quando são projetadas aplicações educacionais para dispositivos móveis com recursos tridimensionais, uma série de questões desafiadoras emergem: Como produzir um cenário 3D motivador, juntamente com imagens e informações nas reduzidas telas dos telefones celulares? Como possibilitar formas de interação eficientes? Como contornar a limitação de memória e processamento dos dispositivos móveis?

Nesta dissertação, é apresentado um *framework* para construção de aplicações educacionais em dispositivos móveis com recursos de realidade virtual: o VirTraM. Este *framework* facilita o desenvolvimento de aplicações, ao permitir que o programador trabalhe com menos elementos e abstraia-se dos detalhes de implementação para concentrar-se nos aspectos relacionados ao treinamento. Para a validação do VirTraM, foram desenvolvidos dois estudos de caso que demonstram características como interatividade, usabilidade e portabilidade, além de possibilitar uma experimentação da integração das tecnologias de computação móvel e realidade virtual.

Uma aplicação, construída a partir do VirTraM, fornece ao usuário uma ferramenta inovadora e interativa para auxiliá-lo com informações complementares no momento e no local em que ele precisar. Dentre os benefícios das aplicações baseadas no VirTraM destacam-se: apoiar e motivar o aprendizado; melhorar a compreensão sobre determinados objetos; permitir a simulação e a análise de experiências em campo; possibilitar a demonstração do funcionamento de equipamentos; e fornecer meios para o desenvolvimento de métodos inovadores de ensino e de treinamento.

ABSTRACT

Mobile computing has emerged as an innovative technology for education. For example, a new terminology, called Mobile Learning or m-Learning, has been introduced to designate the use, in a transparent way and with a high mobility degree, of devices with wireless communication, as part of a model of integrated learning. On the other hand, education can be also seen as a process of exploration, discovery and construction of knowledge, using three-dimensional environments. However, different challenges are faced when designing educational applications for mobile devices with three-dimensional resources, such as: How to produce a 3D scenario together with images and information in the reduced screens of cellular phones? How to make possible efficient ways of interaction? How to overcome the limitation of memory and processing of the mobile devices?

In this master's thesis, a framework for the development of educational applications with virtual reality resources in mobile devices is introduced, called VirTraM. This framework facilitates the development of such applications, allowing a programmer to work with little elements, abstracting the implementation details and emphasizing the aspects concerning the training. For the validation of VirTraM, two case studies have been developed. These case studies demonstrate such characteristics as interaction, usability and portability, besides making possible the integration of mobile computing and virtual reality technologies.

An application, constructed from VirTraM, offers to the user an innovative and interactive tool to help with complementary information at the moment and place where his necessary. Amongst the benefits of the applications based on VirTraM, the following are distinguished: to support and to motivate the learning; to improve the understanding on definitive objects; to allow the simulation and analysis of experiences in the field; to make possible the demonstration of the equipment execution; and to provide ways for the development of innovative methods of education and training.

ÍNDICE

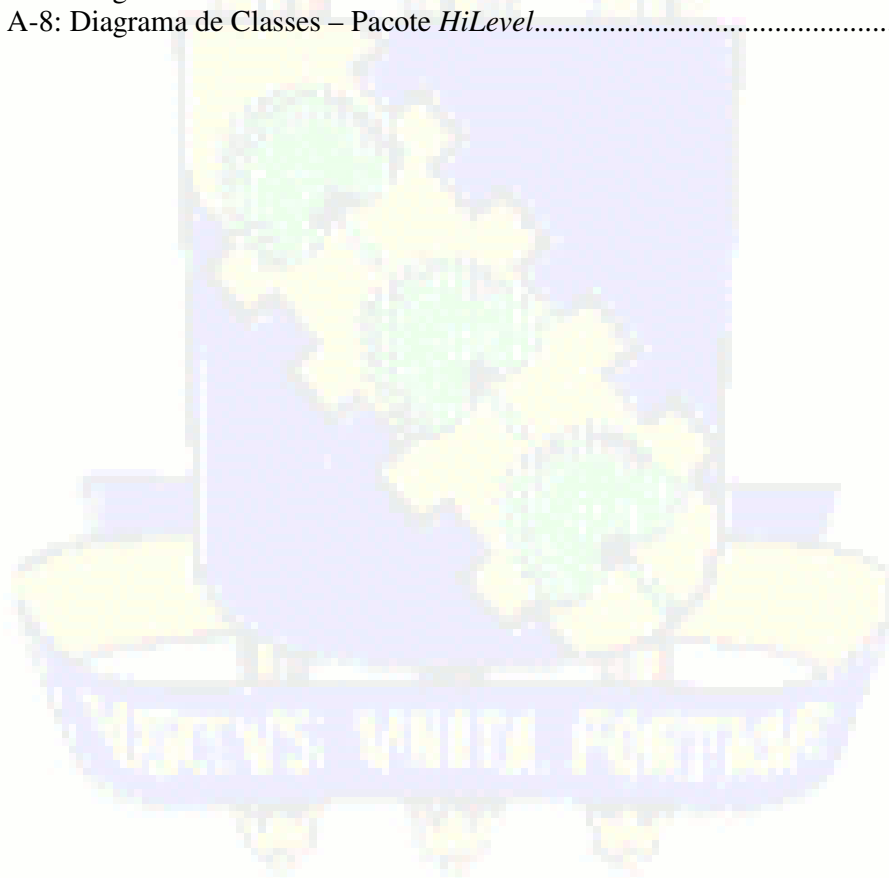
Lista de Figuras	8
Lista de Tabelas	10
CAPÍTULO 1	11
1.1. Motivação	11
1.2. Objetivos	13
1.3. Estrutura da Dissertação	14
CAPÍTULO 2 Mobile Learning	16
2.1. Introdução	16
2.1.1. Visão Geral da Computação Móvel	16
2.1.2. Tecnologias de Comunicação Sem Fio	19
2.1.2.1. Bluetooth	19
2.1.2.2. IEEE 802.11	20
2.1.2.3. Redes Celulares	21
2.2. Paradigmas de Ensino	22
2.3. Aprendizagem Utilizando Dispositivos Móveis	23
2.3.1. Utilização de Dispositivos Móveis para o Ensino de Ciências	29
2.3.2. <i>Palmtops</i> Aplicados ao Estudo de Doenças	30
2.3.3. Computação Móvel Aplicada ao Ensino de Genética	31
2.3.4. Dispositivos Móveis na Educação Física	32
2.3.5. Computação Móvel em Museus	33
2.3.6. Um Jogo Educativo Utilizando PDAs em um Campus Universitário	35
2.4. Requisitos das Aplicações	36
2.5. Considerações Finais	38
CAPÍTULO 3 A Realidade Virtual na Educação	40
3.1. Introdução	40
3.1.1. Realidade Virtual Imersiva e Não Imersiva	41
3.1.2. Simulação Tridimensional	43
3.1.3. Realidade Virtual Passiva, Exploratória ou Interativa	44
3.1.4. Ambientes Virtuais Interativos	45
3.2. Aplicações da Realidade Virtual na Aprendizagem	47
3.2.1. Ensino através de Experiências	47
3.2.2. Ensino de Física	48
3.2.3. Realidade Virtual e Realidade Aumentada	50
3.2.4. Treinamento para Segurança no Trabalho para a Área de Petróleo	52
3.2.5. Treinamento em Medicina	53
3.2.6. Aprendizagem de Idiomas	54
3.3. Requisitos	56
3.3.1. Interatividade	56
3.3.2. Disponibilização de Objetos 3D	56
3.3.3. Portabilidade e Flexibilidade da Aplicação	57
3.3.4. Suporte a Linguagens de Script	57
3.3.5. Realismo	57
3.3.6. Manipulação do Conteúdo	58

3.4. Considerações Finais	59
CAPÍTULO 4 O Framework VirTraM	61
4.1. Introdução	61
4.2. Metodologia de Desenvolvimento Utilizada	62
4.3. Análise de Domínio	64
4.3.1. Objetivos, Benefícios e Limites	64
4.3.2. Requisitos Não Funcionais	66
4.3.3. Requisitos Funcionais	67
4.4. Projeto do Framework.....	68
4.4.1. Pacote <i>Interaction</i>	69
4.4.2. Pacote <i>Persistence</i>	70
4.4.3. Pacote <i>Optional</i>	71
4.4.4. Pacote <i>Log</i>	72
4.4.5. Pacote <i>Visual</i>	73
4.4.6. Pacote <i>Content</i>	74
4.4.7. Pacote <i>3D</i>	77
4.4.8. Pacote <i>HiLevel</i>	80
4.4.9. Resumo dos Pacotes do VirTraM	83
4.5. Aspectos da Implementação	84
4.5.1. <i>Mobile 3D Graphics API - M3G</i>	84
4.5.2. <i>Over-The-Air provisioning (OTA)</i>	85
4.6. Considerações Finais	86
CAPÍTULO 5 Estudos de Caso	88
5.1. O museum	88
5.1.1. Descrição	90
5.1.2. Detalhes de Projeto e Implementação	91
5.1.2.1. Modelagem 3D	91
5.1.2.2. Conteúdo	94
5.1.2.3. Desenvolvimento da Aplicação	95
5.2. mPrinter	98
5.2.1. Descrição	99
5.2.2. Detalhes de Projeto e Implementação	102
5.2.2.1. Modelagem 3D	102
5.2.2.2. Conteúdo	103
5.2.2.3. Desenvolvimento da Aplicação	104
5.3. Considerações Finais	106
CAPITULOS 6. Conclusões	108
6.1. Resultados Alcançados	108
6.2. Trabalhos Futuros	110
Referências Bibliográficas	112
Anexo A – Diagramas de Classes Completos dos Pacotes do VirTraM	122

LISTA DE FIGURAS

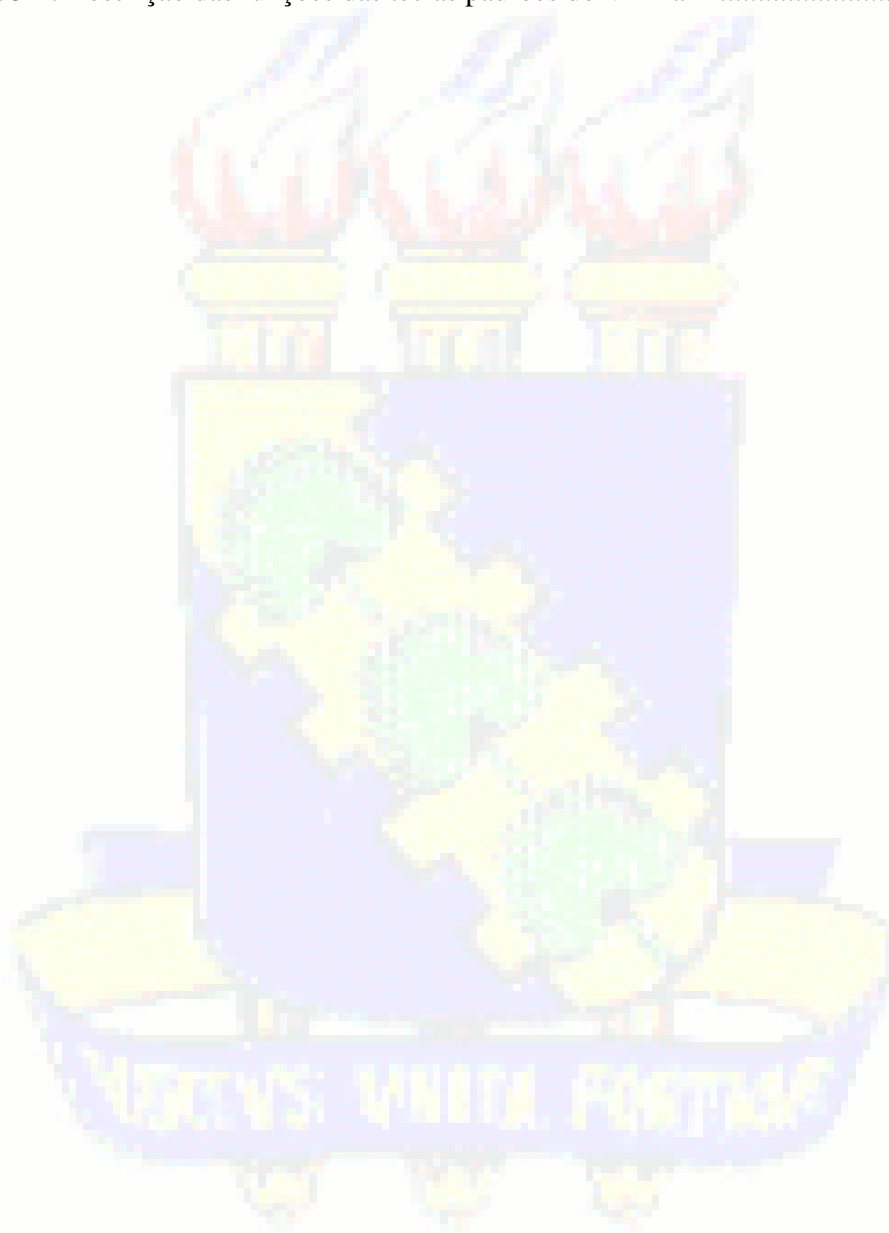
Figura 2-1: Exemplo de um <i>smartphone</i>	19
Figura 2-2: Exemplo de uma piconet	20
Figura 2-3: Exemplo de uma WLAN	21
Figura 2-4: Modelo de comunicação do protocolo WAP	22
Figura 2-5: Usuário estudando com um PDA	25
Figura 2-6: Arquitetura proposta por Trifonova (2004)	26
Figura 2-7: Telas do programa Cooties	30
Figura 2-8: Telas de aplicações de educação física em <i>palmtops</i>	33
Figura 3-1: Ambiente de realidade virtual imersiva	41
Figura 3-2: Equipamentos de realidade virtual não imersiva	42
Figura 3-3: Crianças plantando em um jardim virtual	48
Figura 3-4: Crianças em ambiente virtual no museu da FHW	49
Figura 3-5: Interface do C-VISions	49
Figura 3-6: Aplicação em realidade aumentada para área médica	50
Figura 3-7: Equipamentos de RA combinados com dispositivos móveis	51
Figura 3-8: Exposição em museu virtual	52
Figura 3-9: Tela do sistema ACaMPE	53
Figura 3-10: Tela do módulo cliente de treinamento	54
Figura 3-11: Ambiente virtual modelado	55
Figura 3-12: Apresentação de conteúdo dentro do mundo 3D	58
Figura 3-13: Apresentação de conteúdo de forma estática	59
Figura 4-1: Interseção das tecnologias do VirTraM	64
Figura 4-2: Representação do VirTraM	67
Figura 4-3: Pacotes de classes do VirTraM	69
Figura 4-4: Diagrama de classes do pacote <i>Interaction</i>	71
Figura 4-5: Diagrama de classes do pacote <i>Persistence</i>	72
Figura 4-6: Diagrama de classes do pacote <i>Optional</i>	72
Figura 4-7: Diagrama de Classes do Pacote <i>Log</i>	73
Figura 4-8: Diagrama de Classes do Pacote <i>Visual</i>	74
Figura 4-9: Exemplo de arquivo de conteúdo do VirTraM	75
Figura 4-10: Diagrama de atividades com os processos de manipulação dos dados	76
Figura 4-11: Diagrama de Classes do Pacote <i>Content</i>	77
Figura 4-12: Diagrama de atividade com o processo de manipulação do objeto 3D	79
Figura 4-13: Diagrama de Classes do Pacote <i>3D</i>	81
Figura 4-14: Diagrama de Classes do Pacote <i>HiLevel</i>	82
Figura 4-15: Diagrama de instalação de aplicações com OTA	85
Figura 5-1 a) Uma visão ampla de uma das salas do museu virtual	92
Figura 5-1 b) Tela de introdução do museuM	92
Figura 5-1 c) Uma visão aproximada de um quadro e o menu associado	92
Figura 5-2 a) Uma tela com o conteúdo da obra selecionada	93
Figura 5-2 b) Uma visão aproximada da escultura virtual	93
Figura 5-2 c) Uma visão aproximada da escultura virtual rotacionada	93
Figura 5-2 d) Uma visão de dois quadros com um rotacionado e transladado	93
Figura 5-3: Trecho do arquivo de conteúdo do museuM	94
Figura 5-4: Representação resumida do Diagrama de Classes do museuM	95
Figura 5-5: Teclado destacando as teclas padrões do VirTraM	97

Figura 5-6: Trecho de código do museuM	98
Figura 5.7 a) Uma visão aproximada da impressora virtual	100
Figura 5.7 b) Tela de introdução da mPrinter	100
Figura 5-7 c) Visão da parte traseira da impressora virtual	100
Figura 5-8 a) Uma visão aproximada da impressora com a tampa aberta	101
Figura 5-8 b) Uma visão com a impressora aproximada e rotacionada	101
Figura 5-8 c) Uma visão da impressora mostrando o menu com as opções	101
Figura 5-8 d) A tela de conteúdo antes da animação para levantar a tampa	101
Figura 5-9: Trechos do arquivo XML da mPrinter	103
Figura 5-10: Representação resumida do Diagrama de Classes da mPrinter	104
Figura 5-11: Trecho de código da mPrinter	106
Figura A-1: Diagrama de Classes – Pacote <i>Interaction</i>	122
Figura A-2: Diagrama de Classes – Pacote <i>Persistence</i>	123
Figura A-3: Diagrama de Classes – Pacote <i>Optional</i>	124
Figura A-4: Diagrama de Classes – Pacote <i>Log</i>	125
Figura A-5: Diagrama de Classes – Pacote <i>Visual</i>	125
Figura A-6: Diagrama de Classes – Pacote <i>Content</i>	126
Figura A-7: Diagrama de Classes – Pacote <i>3D</i>	127
Figura A-8: Diagrama de Classes – Pacote <i>HiLevel</i>	128



LISTA DE TABELAS

Tabela 2-1: Diferenças entre o instrucionismo e o construtivismo.....	23
Tabela 2-2: Formas de interação identificadas	34
Tabela 4-1: Resumo dos pacotes do VirTraM	83
Tabela 5-1: Tabela comparativa do grau de fixação da informação dos usuários	89
Tabela 5-2: Descrição das funções das teclas padrões do VirTraM	97



CAPÍTULO 1

Introdução

Este capítulo apresenta na Seção 1.1 uma visão geral sobre computação móvel, realidade virtual imersiva e semi-imersiva e suas aplicações na educação, bem como a caracterização do problema que conduz à motivação do trabalho. A Seção 1.2 descreve os objetivos principais da dissertação. Por fim, na Seção 1.3, está detalhada a organização do restante do trabalho.

1.1. Motivação

As tecnologias de computação móvel encontram-se atualmente em franca evolução, parecendo destinadas a transformar-se no novo paradigma dominante da computação (Myers & Beigl, 2003). Em paralelo ao seu crescente uso, por um lado, surgem novos dispositivos móveis com os mais variados recursos, tais como câmera digital, sensores de movimentos, localizador GPS etc, e por outro lado, os processadores estão cada vez mais rápidos, a memória cada vez maior e os visores com melhor definição, alguns até com placa aceleradora gráfica. Além disso, a capacidade de os telefones celulares aceitarem programas desenvolvidos por terceiros proporciona a esses dispositivos uma enorme aplicação prática. Assim, o dispositivo móvel transforma-se em uma opção interessante para a disponibilização de conteúdo, principalmente por permitir ao usuário obter conhecimento em vários momentos, não necessariamente com local e hora pré-estabelecidos.

A utilização de dispositivos móveis na educação criou um novo conceito, denominado Mobile Learning ou m-Learning. Seu grande potencial encontra-se na utilização da tecnologia móvel como parte de um modelo de aprendizado integrado, caracterizado pelo uso de dispositivos de comunicação sem fio, de forma transparente e com alto grau de mobilidade (Ahonen et al., 2003; Syvänen et al., 2003). Essa oferta de serviços de telecomunicações e de artefatos computacionais, capazes de prover mobilidade aos diferentes participantes de projetos educacionais, apresenta uma oportunidade para o

desenvolvimento de pesquisas no campo da computação móvel aplicadas à educação (Meirelles et al., 2004).

A educação pode ser vista como um processo de exploração, descoberta e construção de conhecimento. Assim, a introdução dos chamados Sistemas de Realidade Virtual (SRV) na educação pode modificar significativamente o papel dos educadores (Ferreira, 2004). Esses ambientes, onde o indivíduo pode se movimentar, ouvir, ver e manipular objetos, como o faz no mundo real, representam interessantes oportunidades à disposição dos educadores.

Várias aplicações podem ser desenvolvidas ao se unirem recursos de realidade virtual e de computação móvel. Em diversas situações, profissionais em campo precisam de conhecimento técnico específico ou mesmo de cursos práticos e eficientes sob demanda para a realização de demonstrações, manutenções, montagens ou reparações. A obtenção imediata desses recursos incrementa de forma significativa a produtividade e a qualidade no atendimento. Por exemplo, um técnico em impressoras, que não dispõe de um manual técnico específico, e está em dúvida sobre o funcionamento de uma determinada peça poderia ser apoiado por uma representação tridimensional com conteúdo auto-explicativo, disponibilizada através de um dispositivo móvel. Outra aplicação é a possibilidade de se utilizarem cenários tridimensionais em dispositivos móveis para simular ambientes reais e auxiliar a aprendizagem, por exemplo, em visitas a museus ou a locais históricos.

Portanto, os recursos de realidade virtual nas aplicações de m-Learning podem propiciar um incremento na aprendizagem ao fornecerem benefícios, tais como:

- Apoiar e motivar o aprendizado em excursões, fornecendo cenários virtuais semelhantes aos reais acrescidos de informações complementares (Bricken, 1993).
- Melhorar a compreensão sobre determinada obra ou experimento, através de maior aproximação e de visualização sob diferentes ângulos (Erickson, 1993).
- Permitir a simulação e a análise de experiências recém vivenciadas, seja na própria sala de aula, no laboratório ou em passeios educativos.

- Possibilitar a demonstração do funcionamento de equipamentos, através de simulações, para auxiliar na resolução de problemas técnicos no momento do atendimento, funcionando como um manual virtual portátil.

Entretanto, alguns desafios aparecem ao se pensar em aplicações educacionais para dispositivos móveis com recursos de realidade virtual:

- Como produzir um cenário virtual motivador, juntamente com imagens e informações nas reduzidas telas dos telefones celulares?
- Como possibilitar formas de interação eficientes nesses dispositivos?
- Como contornar a limitação de memória e processamento dos dispositivos móveis?
- Como otimizar a implementação, facilitando o trabalho do desenvolvedor?
- Como atender a diferentes fabricantes e diferentes modelos de dispositivos?

Inúmeros trabalhos têm sido desenvolvidos com enfoque na utilização de dispositivos móveis para educação e treinamento (Ratto et al., 2003; Sukimoto, 2003; Kynaslahti et al., 2002). Porém, constatou-se a inexistência de um mecanismo que auxiliasse a implementação de treinamentos para esses dispositivos utilizando recursos tridimensionais oriundos de modelos de realidade virtual semi-imersiva. Neste caso particular, a idéia de um *framework* que permita o reaproveitamento de um conjunto de recursos comuns, aplicáveis a vários softwares, surge como uma solução para facilitar o desenvolvimento de aplicações educacionais. Permite-se, assim, que o programador trabalhe com menos elementos, abstraindo-se dos detalhes de implementação relativos à disponibilização de recursos tridimensionais em dispositivos móveis, e concentrando-se nas reais necessidades do treinamento.

1.2. Objetivos

Esta dissertação tem como objetivo o desenvolvimento de um *framework* orientado a objetos para possibilitar o desenvolvimento de aplicações para treinamento em dispositivos móveis utilizando ambientes tridimensionais interativos, baseados em modelos de realidade virtual semi-imersiva, denominado VirTraM (*Virtual Training for Mobile Devices*). Desta forma, este trabalho, ao optar pela construção de um *framework*, pretende

fornecer uma arquitetura reutilizável e extensível para a construção de aplicações educacionais interativas e inovadoras para dispositivos móveis.

Esta dissertação também apresenta como objetivo complementar a implementação de dois estudos de caso, construídos a partir do VirTraM, em áreas distintas: o museu virtual – denominado museuM; e a impressora virtual – a mPrinter. Estes protótipos têm por objetivo validar os requisitos fornecidos pelo *framework*.

Para atingir os objetivos citados anteriormente, diferentes aplicações educacionais são analisadas, tanto na área da computação móvel quanto na área da realidade virtual. A partir deste levantamento, um subconjunto de características presentes em ambas as áreas é definido, para constituir o conjunto de requisitos do *framework* VirTraM.

1.3. Estrutura da Dissertação

Esta dissertação está dividida em mais cinco capítulos e um anexo. Estes descrevem os detalhes da pesquisa e desenvolvimento deste trabalho.

No Capítulo 2, são abordados os aspectos da computação móvel aplicada à educação. São apresentados uma visão geral da área de computação móvel, aspectos educacionais, aplicações educacionais em dispositivos móveis e um levantamento dos requisitos em comum nessas aplicações.

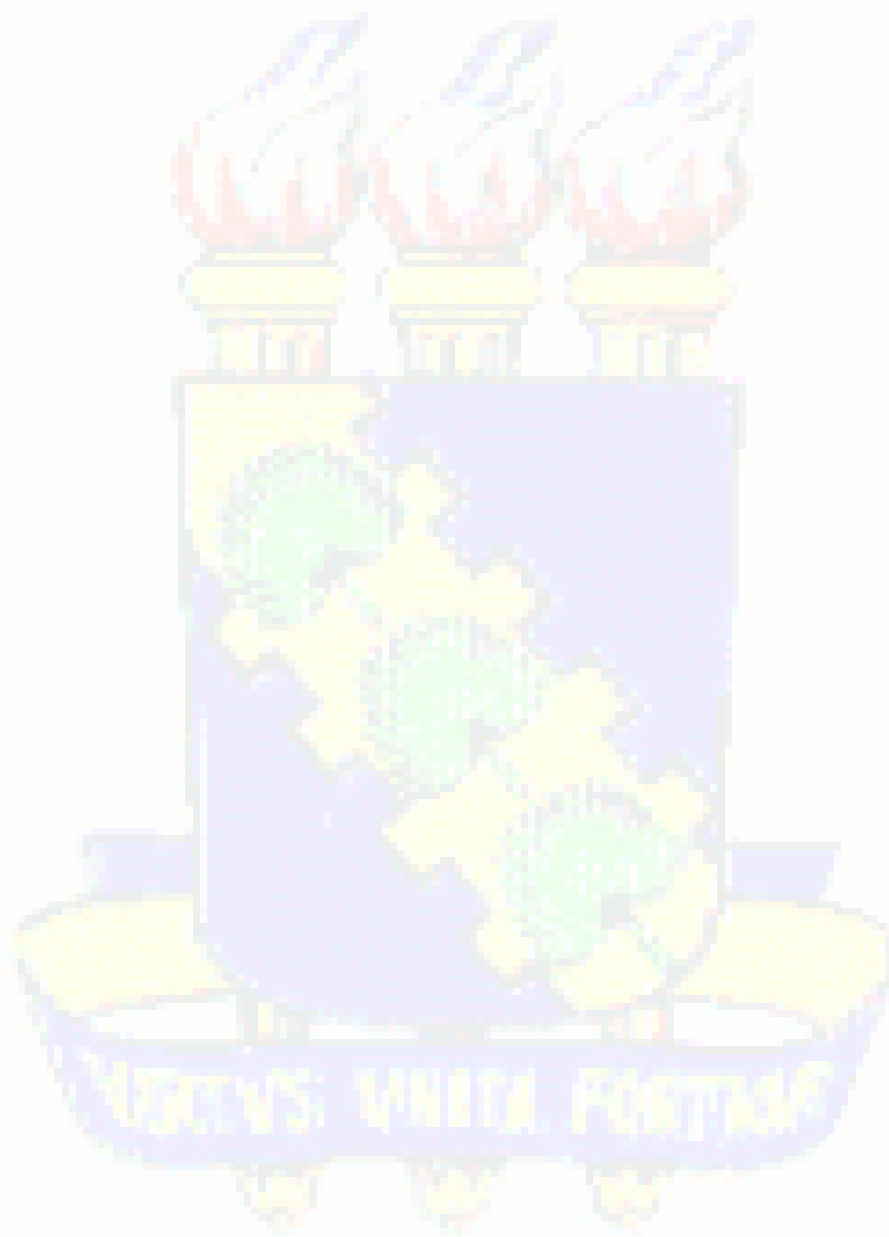
O Capítulo 3 mostra o emprego da realidade virtual na educação. É feita uma introdução à realidade virtual, um levantamento de aplicações que utilizam a realidade virtual na educação e, por fim, são relacionadas características encontradas em comum nessas aplicações.

O Capítulo 4 trata da descrição do VirTraM. São apresentados os aspectos da implementação, uma introdução aos *frameworks* orientados a objetos e o detalhamento do VirTraM, sendo descritos todos os detalhes das classes envolvidas.

No Capítulo 5, são apresentados dois estudos de caso para validação do VirTraM. Para cada um deles são descritos os aspectos relacionados ao tema da aplicação escolhida, os benefícios decorridos da utilização do *framework* e os detalhes da implementação, como por exemplo as classes utilizadas.

O Capítulo 6 conclui a dissertação, apresentando os resultados alcançados com o trabalho e as possibilidades para extensão da dissertação através de trabalhos futuros.

Finalmente, o Anexo A detalha o VirTraM, apresentando seus diagramas de classes completos, incluindo todos os métodos e atributos utilizados.



CAPÍTULO 2

Mobile Learning

Neste capítulo será abordada a aplicabilidade da computação móvel na aprendizagem. A Seção 2.1 apresenta o conceito de *Mobile Learning*, mostrando uma visão geral sobre computação móvel e apresentando os principais paradigmas educacionais. A Seção 2.2 mostra exemplos de aplicações utilizando dispositivos móveis para aprendizagem. A Seção 2.3 relaciona requisitos importantes em aplicações de *Mobile Learning*. Por fim, a Seção 2.4 apresenta considerações finais sobre o assunto.

2.1. Introdução

Esta seção apresenta uma introdução sobre a computação móvel e descreve modelos e tecnologias de comunicação sem fio.

2.1.1. Visão Geral da Computação Móvel

A computação móvel pode ser definida como a utilização de dispositivos computacionais portáteis com capacidade de acessar informações, aplicações e serviços em qualquer lugar e a qualquer momento. Por isso, a computação móvel está fortemente associada à mobilidade de hardware, de software e da informação. As tecnologias de computação móvel abrangem não somente os computadores móveis designados por PDAs (Personal Digital Assistants), *handhelds* ou *palmtops*, mas também outros dispositivos que refletem a convergência entre os computadores móveis e os telefones celulares. Estes últimos designam-se normalmente por telefones inteligentes (*smartphones*), já que a ênfase destes recai nas potencialidades para as comunicações remotas, às quais se adicionam as funcionalidades típicas de um dispositivo móvel. As seguintes limitações também caracterizam esses dispositivos: poder de processamento reduzido, tela pequena (às vezes monocromática), energia limitada (dependente de baterias) e comunicação com taxas de transmissão geralmente menores do que as das redes fixas.

Os dispositivos móveis permitem que os usuários tenham acesso a serviços independentemente de onde estão situados ou de mudanças de localização, o que caracteriza a mobilidade. Dessa forma a computação móvel amplia o conceito tradicional de computação distribuída. Isso é possível graças à comunicação sem fio, que elimina a necessidade de o usuário manter-se conectado a uma infra-estrutura fixa e, em geral, estática. Com a diminuição dos custos desses dispositivos, a computação móvel tornar-se-á viável não somente para o segmento empresarial como também para os usuários finais.

Algumas características importantes que distinguem os dispositivos móveis de outros tipos de computadores são:

- A utilização de telas sensíveis ao toque, combinadas com a utilização de canetas especiais, como meio principal de entrada da informação. O uso da caneta permite selecionar informação no visor e escrever símbolos que são interpretados como letras.
- A inclusão de botões de navegação rápida que dão acesso às aplicações mais utilizadas, como agenda e calculadora.
- A disponibilização de uma gama abrangente de opções de comunicação, principalmente sem fio, incluindo a utilização de cabos seriais e USB, infravermelho e Bluetooth (descrito na seção 2.1.3.2) para a troca de dados dos dispositivos entre si ou com um microcomputador, e IEEE 802.11 (descrito na seção 2.1.3.3) para integrar o dispositivo móvel a uma rede local sem fio. Além disso, merece destaque a possibilidade de transmissão de dados a longas distâncias através da rede de telefonia celular.
- A capacidade de sincronizar o conteúdo da memória do dispositivo móvel com um computador de mesa ou com um laptop. Permitindo, assim, armazenar e manter uma cópia de segurança das informações existentes no dispositivo móvel. O processo de sincronização funciona de modo a assegurar que, ao final da operação, tanto o dispositivo móvel quanto o computador possuam a mesma informação.
- A utilização de cartões de memória, que permite expandir a capacidade de armazenamento limitada, tais como os cartões Secure Digital (SD), Compact

Flash (CF) e Memory Stick (MS). Estes cartões permitem ainda transferir informação de um dispositivo móvel para outro, ou diretamente para um computador de mesa ou um laptop.

- A utilização de baterias recarregáveis que, dependendo da tecnologia, permitem tempos de utilização eficientes que variam de algumas horas a alguns dias.
- A utilização de sistemas operacionais especificamente desenvolvidos para dispositivos móveis, que gerenciam com mais eficiência o consumo de energia, bem como as características específicas do hardware, tais como o Palm OS (Palm, 2004), o Microsoft Windows Mobile (Windows Mobile, 2004), Symbian OS (Symbian, 2004) e o Linux OS (Montavista Linux, 2004).
- A capacidade de instalar e executar aplicações com características específicas para dispositivos móveis, isto é, que sejam concebidas para operar dentro das limitações de processamento, display, memória e energia.
- A utilização de câmeras digitais acopladas, que permitem o registro de eventos, através de fotos ou vídeos.

A combinação, no mesmo dispositivo, das várias tecnologias mencionadas acima, proporciona oportunidades únicas para o desenvolvimento de diferentes aplicações de gestão de informação, com muita utilidade nas mais variadas áreas da atividade humana, incluindo a área empresarial, o entretenimento, a saúde e o ensino. A utilização de dispositivos móveis torna-se ainda mais indicada em ambientes cujas características dificultam a utilização de laptops, tais como as situações que exigem um constante deslocamento físico do usuário, como em um hospital, em uma indústria ou em um restaurante.

A Figura 2-1 apresenta o exemplo de um dispositivo móvel (um *smartphone*), destacando seus recursos. É importante observar, através da imagem, a convergência de diferentes recursos em um único dispositivo, o que permite, além da comunicação por voz, a transmissão de dados remota (e-mail) ou local (infravermelho), som, a captura de imagem (câmera) e a entrada de dados através de um teclado.



Figura 2-1: Exemplo de um *smartphone* (adaptada de Palm, 2004).

2.1.2. Tecnologias de Comunicação Sem Fio

Uma dos principais elementos da computação móvel é a comunicação sem fio, seja entre dispositivos próximos ou geograficamente dispersos. Nas subseções seguintes, são descritas algumas das principais tecnologias de comunicação utilizadas atualmente por dispositivos móveis.

2.1.2.1. Bluetooth

Bluetooth é um padrão proposto pelo Bluetooth SIG (Special Interest Group) (Bluetooth, 2004), um consórcio de empresas de telecomunicações e computação, entre elas: Ericsson, IBM, Intel, Microsoft, Motorola, Nokia, e Toshiba. O padrão opera na faixa ISM (Industrial, Scientific, and Medical) de 2,4 GHz e tem como princípio propor uma tecnologia de baixo custo para conectividade sem fio. Inicialmente, o padrão foi projetado como uma solução para substituição de cabos usados na comunicação de periféricos por

transmissão via rádio. No entanto, ele permite a conexão entre diferentes tipos de dispositivos, a curtas distâncias, possibilitando a formação de redes *ad-hoc*. O Bluetooth é voltado para a comunicação entre dispositivos com baixo poder de processamento e pouca energia, substituindo completamente as conexões físicas.

A estrutura básica de comunicação no Bluetooth é chamada de piconet. Esta tem a característica de ser uma rede onde um nó central, definido como mestre, se comunica ponto-a-ponto com os outros nós chamados de escravos. Os dispositivos se alternam nas funções de mestre e escravo. As piconets formam pequenas redes pessoais, conhecidas como PAN – Personal Area Network. A Figura 2-2 mostra o exemplo de uma piconet.



Figura 2-2: Exemplo de uma piconet.

2.1.2.2. IEEE 802.11

O padrão de comunicação IEEE 802.11 foi criado para suportar a comunicação em redes locais sem fio (WLANs – Wireless Local Networks). A especificação define uma camada de acesso ao meio, camada MAC (Media Access Control), e diferentes camadas físicas, tornando possível acessar o meio de três formas: FHSS (Frequency Hopping Spread Spectrum), DSSS (Direct Sequence Spread Spectrum) e infra-vermelho.

As taxas de comunicação variam de acordo com a versão da especificação técnica do padrão IEEE 802.11. Nas primeiras versões, existem duas taxas de comunicação: 1 e 2 Mbps. Os padrões 802.11a, 802.11b e 802.11g alteraram a especificação para prover taxas

maiores: até 54 Mbps para 802.11a, até 11 Mbps para 802.11b; e até 54 Mbps para 802.11g.

O padrão 802.11 é chamado muitas vezes de “Ethernet sem fio”, por ser uma extensão natural do padrão Ethernet (IEEE 802.3), conforme mostrado na Figura 2-3.

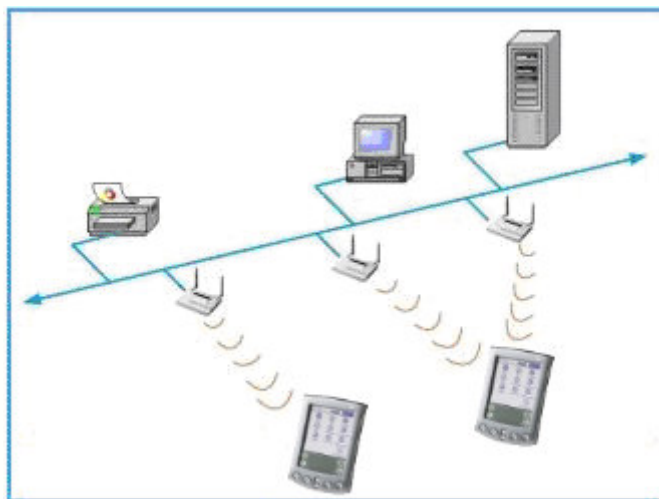


Figura 2-3: Exemplo de uma WLAN.

A popularidade das redes locais sem fio usando o padrão IEEE 802.11b tem crescido a cada dia. Hoje elas já estão presentes em diversos locais, como universidades, aeroportos e restaurantes, e em equipamentos variados, como laptops, PDAs e telefones celulares.

2.1.2.3. Redes Celulares

Os sistemas celulares são os mais populares sistemas sem fio. O nome sistema móvel celular (SMC) advém de sua estrutura em células. Uma célula é uma área geográfica atendida ou coberta por um transmissor de baixa potência, uma ERB (Estação Rádio Base). Uma ERB é uma ou mais antenas fixas, instaladas em torres que têm como objetivo atender a demanda originada pelas estações ou unidades móveis, ou usuários, dentro de sua área de cobertura. A unidade móvel é o equipamento manipulado pelo usuário do SMC. Notadamente os aparelhos portáteis ditos telefones celulares. As redes de telefonia celular possibilitam a troca de informações a longas distâncias para usuários dispersos geograficamente. Para suportar a demanda por tráfego de dados, os sistemas celulares

implementaram serviços específicos com este fim, tais como: GPRS (*General Packet Radio Service*) e EDGE (*Enhanced Data Rates For Global Evolution*).

O WAP (Wireless Application Protocol) é um protocolo de transmissão de dados na Internet para dispositivos móveis, padronizado pelo WAP Forum (WAP Forum, 2004). O WAP foi projetado para suportar comunicações sem fio e para ser independente do dispositivo e do sistema celular utilizado.

O modelo de comunicação WAP é similar ao modelo de comunicação Web. Isto significa que ele provê vários benefícios para a comunidade desenvolvedora de aplicações, incluindo um modelo de programação familiar e a capacidade de reutilização das ferramentas atuais, como os servidores Web, ver Figura 2-4. Atualmente, o protocolo WAP é muito usado para transmissão de dados na telefonia celular.

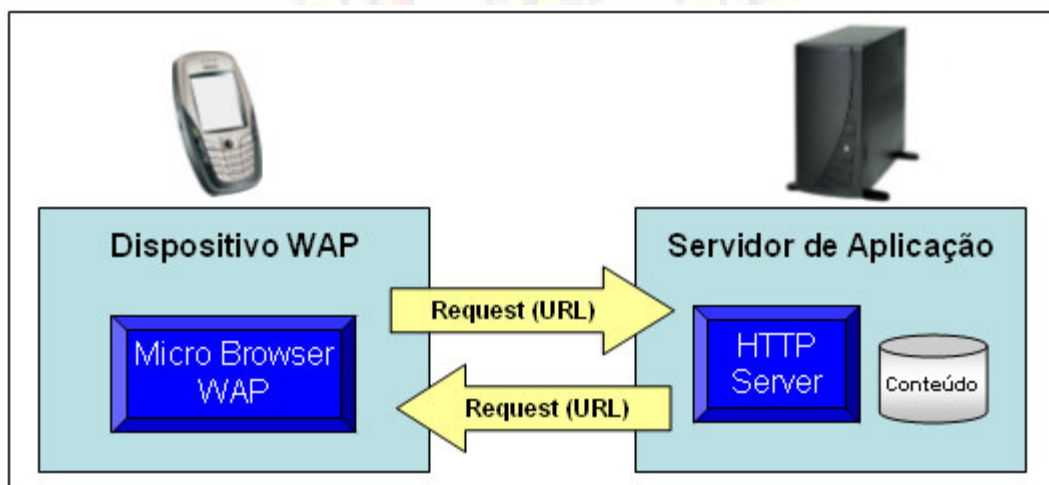


Figura 2-4: Modelo de comunicação do protocolo WAP.

2.2. Paradigmas de Ensino

Antes de apresentar as aplicações de computação móvel voltadas para aprendizagem, é importante descrever e diferenciar os dois principais paradigmas de ensino: o instrucionista e o construtivista.

O primeiro segue uma abordagem behaviorista (Skinner, 1968; Hull, 1943) e enfatiza os objetivos do ensino. Após a formulação dos objetivos, eles são sequenciados em hierarquias de aprendizagem, geralmente uma progressão do nível mais baixo ao mais alto, e a instrução é projetada para o alcance dos objetivos em sequência. Os instrucionistas

consideram os testes como uma forma importante de medir aprendizagem. Esta abordagem utiliza métodos como aulas expositivas e provas, avaliações do conhecimento, sendo dada ênfase à aprendizagem individual (Schrock, 1996).

O segundo modelo de ensino destacado é o construtivismo (Bruner, 1990). Este paradigma ressalta a primazia das intenções dos alunos, suas experiências e estratégias cognitivas. Na abordagem construtivista, os alunos constroem diferentes estruturas de cognição baseadas em conhecimento prévio (Piaget, 1975). A aprendizagem se dá através da exploração das possibilidades, promovendo a experimentação, e a ênfase é dada para a colaboração e o trabalho em grupo.

A Tabela 2-1 apresenta um resumo das principais diferenças entre o instrucionismo e o construtivismo.

Tabela 2-1: Diferenças entre o instrucionismo e o construtivismo.

Instrucionismo	Construtivismo
Aprendizagem através de seqüências de instruções	Aprendizagem através de problemas e na exploração das possibilidades.
Importância no esforço individual	Importância na aprendizagem em grupo
Objetivos claros e específicos	Objetivos abrangentes
Exames através de provas	Avaliações através de trabalhos de pesquisa e solução de problemas práticos

A próxima seção mostra a utilização de dispositivos móveis na educação, apresentando projetos relacionados. Durante a descrição destes, características dos paradigmas construtivista e instrucionista serão evidenciadas.

2.3. Aprendizagem Utilizando Dispositivos Móveis

A utilização de dispositivos móveis na educação criou um novo conceito, o chamado *Mobile Learning* ou m-Learning. Este novo paradigma surge aproveitando-se da disponibilidade de dispositivos móveis e considerando necessidades específicas de educação e treinamento (Nyiri, 2002).

As aplicações de m-Learning dividem-se segundo dois grupos de usuários principais: crianças e trabalhadores externos (profissionais que trabalham fora da empresa).

Dispositivos móveis fornecem um novo e entusiasmante paradigma de interação para as crianças. Várias iniciativas têm sido desenvolvidas nessa área, por exemplo: um projeto da Philips desenvolveu um organizador portátil para educar crianças de 7 a 12 anos (Oosterholt et al., 1996); e, no Canadá, *palmtops* da plataforma Palm OS foram usados para ensinar genética às crianças (Danesh et al., 2001). No caso de trabalhadores externos, a preocupação é fornecer um ambiente de aprendizado que acompanhe sua rotina dinâmica. Às vezes esses profissionais estão viajando, outras horas estão visitando um cliente ou realizando uma demonstração em outro local, sempre precisando da informação mais atualizada possível.

Desta forma, o m-Learning surge como uma importante alternativa de ensino e treinamento, na qual podem ser destacados os seguintes objetivos:

- Melhorar os recursos para o aprendizado do aluno, que poderá contar com um dispositivo computacional para execução de tarefas, anotação de idéias, consulta de informações via Internet, registro de fatos através de câmera digital, gravação de sons e outras funcionalidades existentes.
- Prover acesso aos conteúdos didáticos em qualquer lugar e a qualquer momento, de acordo com a conectividade do dispositivo.
- Incrementar e incentivar a utilização dos serviços providos por uma instituição, através do aumento das possibilidades de acesso ao conteúdo.
- Expandir as estratégias de aprendizado disponíveis, através de novas tecnologias que dão suporte tanto à aprendizagem formal como à informal.
- Expandir os limites internos e externos da sala de aula ou da empresa, principalmente de uma forma ubíqua.
- Fornecer meios para o desenvolvimento de métodos inovadores de ensino e de treinamento, utilizando os novos recursos de computação e de mobilidade.

Em particular, dispositivos de comunicação sem fio correspondem a uma extensão natural da educação a distância via computadores (Lehner et al., 2002), pois contribuem para a facilidade de acesso ao aprendizado, por exemplo, na obtenção de conteúdo específico para um determinado assunto, sem hora e local pré-estabelecidos. A Figura 2-5 mostra um usuário estudando, utilizando um PDA com recursos de som.

Ao longo dos dez últimos anos, pesquisadores em educação têm mapeado a aprendizagem como uma atividade constante e vitalícia. Criou-se então o conceito CoLL (Contextual Life-long Learning) (Sharples et al., 2002), cujas principais idéias estão diretamente associadas a *Mobile Learning*, entre elas:

- A aprendizagem não está restrita a lugares e horários pré-estabelecidos, acontece quando há uma quebra no fluxo da rotina e a pessoa reflete sobre uma situação para solucionar um problema, compartilhar uma idéia ou obter algum conhecimento;



Figura 2-5: Usuário estudando com um PDA (Sharples et al., 2002).

- A educação formal não fornece ao aluno o conhecimento e as habilidades necessárias ao longo da vida.

Assim, o m-Learning torna-se um meio para a implementação deste novo conceito de aprendizagem, possibilitando ao aluno: acessar conteúdos remotamente; executar experiências em campo; capturar e armazenar eventos (com som e imagem) do dia-a-dia; e comunicar e compartilhar pesquisas com colegas e especialistas em todo o mundo.

Trifonova & Ronchetti (2004) definem uma arquitetura para aplicações de *Mobile Learning*. Esta teria um sistema de gerenciamento de aprendizado móvel (mSGA) que

estaria entre o sistema de gerenciamento de aprendizado (SGA) (Almendra, 2003) e o dispositivo móvel (Figura 2-6). O mSGA seria composto por três módulos:

- Descoberta de contexto, que é responsável por localizar o conteúdo associado àquele usuário naquele momento. Por exemplo, quando o usuário solicita uma informação, este módulo automaticamente identifica o usuário e o seu dispositivo. Assim é possível identificar o perfil do usuário (por exemplo, aluno ou professor), analisar seus direitos e decidir quais serviços serão oferecidos.
- Gerenciamento do conteúdo e adaptação, que é responsável pela apresentação da informação ao usuário, adaptando-a de acordo com as limitações (de hardware e de software) do dispositivo. Após receber as informações do servidor e de acordo com os requisitos obtidos pelo módulo de descoberta de contexto, este módulo monta e apresenta o conteúdo.
- Sincronização, que é o módulo que prepara a aplicação para períodos de desconexão, possibilitando que o aluno não pare de estudar nessas circunstâncias. Este módulo salva, localmente no dispositivo móvel, informações relacionadas ao conteúdo estudado, que poderão ser consultadas em caso de perda de comunicação.

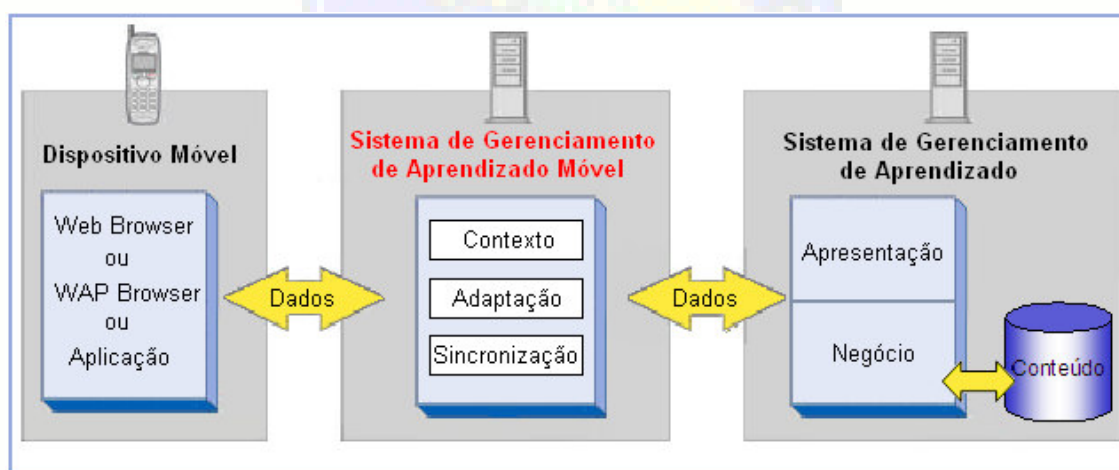


Figura 2-6: Arquitetura proposta por Trifonova & Ronchetti (2004).

Syvänen et al. (2003) definiram um modelo para aplicações de m-Learning fundamentado nos seguintes requisitos:

- Continuidade e adaptabilidade: para possibilitar uma aprendizagem contínua independentemente das variações de contexto;
- Aplicações personalizadas de acordo com o perfil do usuário, suas habilidades e direitos de acesso;
- Gerenciamento da aprendizagem: para permitir um auto-monitoramento do usuário;
- Interação entre usuários do mesmo ambiente de aprendizagem.

Na busca de um modelo pedagógico adequado para um ambiente informatizado com suporte à mobilidade, parece também ser oportuno correlacionar estilos cognitivos com características funcionais dos dispositivos computacionais móveis. Segundo Reif (Geller, 2004), para se determinar um estilo individual de aprendizagem, fundamentado, sobretudo, na atividade individual do sujeito, deve-se observar:

- Como o aluno detecta informações com maior facilidade (de forma visual, tátil, auditiva e cinestésica);
- Como o aluno organiza e processa tais informações;
- Quais condições são necessárias para compreensão e armazenamento das informações (social, emocional, física e ambiental).

Assim, é possível perceber que, em condições ideais, a computação móvel poderá proporcionar um amplo cenário para as dimensões dos estilos cognitivos.

O instituto de pesquisa SRI (*Stanford Research Institute*) realizou uma análise sobre a utilização de dispositivos móveis nas escolas (Crawford & Vahey, 2002). Foram pesquisadas mais de 100 instituições de ensino nos Estados Unidos, do ano de 2000 a 2002. O estudo concluiu que os dispositivos móveis podem oferecer benefícios únicos aos alunos.

Os professores participantes dessa análise demonstraram uma grande aceitação da computação móvel em suas salas de aula. A seguir são apresentados alguns números que demonstram essa aceitação:

- 89% disseram que descobriram nos dispositivos móveis eficientes ferramentas de ensino;

- 93% acreditam que os dispositivos móveis podem ter um impacto positivo na aprendizagem dos alunos;
- 90% pretendem continuar a utilizar os dispositivos móveis em suas aulas;
- 75% dos professores que permitiram que os alunos levassem os dispositivos móveis para casa, constataram um aumento na conclusão dos trabalhos de casa;
- Entre os alunos, 66% acharam confortável o uso do dispositivo móvel.
- Quase a totalidade dos professores afirmou que a utilização de softwares educativos apropriados e acessórios foi de fundamental importância na aprendizagem, ao complementar os recursos básicos dos dispositivos móveis. Por exemplo: alunos podem utilizar sondas acopladas a PDAs para medir e registrar a qualidade da água dos rios.

Em geral, a maioria dos professores afirmou que a introdução da computação móvel na sala de aula tanto aumentou a motivação dos estudantes para aprender, quanto aumentou a colaboração e a comunicação entre eles.

Esta seção apresentou uma visão geral da utilização dos dispositivos móveis na educação, mostrando benefícios, fundamentos e resultados de análises. Apesar do estágio inicial das pesquisas na área de *Mobile Learning*, importantes características já foram observadas, conforme pode ser observado ao longo desta seção. Goh et al. (2004) resumem importantes características encontradas em aplicações de m-Learning:

- As aplicações normalmente são simples. Utilizando recursos como mensagens SMS, textos, imagens, e-mail, web browser e Wap browser.
- A maioria das aplicações foi desenvolvida diretamente e exclusivamente para o dispositivo móvel. Poucas são oriundas da conversão de versões originárias de computadores de mesa.
- Uma grande variedade de dispositivos estão sendo utilizados, tais como: Nokia communicator, HP IPAQ e *palmtops* PalmOne.
- A maior parte das aplicações troca informações com um microcomputador, enviando e recebendo dados.
- As redes WLAN estão sendo bastante utilizadas.

Nas próximas seções, serão apresentadas algumas aplicações, de diferentes áreas do conhecimento, que utilizam dispositivos móveis para auxiliar o aprendizado. Nestas seções, serão mostradas vantagens e desvantagens das aplicações, o que servirá de base para a definição dos requisitos do *framework* VirTraM.

2.3.1. Utilização de Dispositivos Móveis para o Ensino de Ciências

Luchini et al. (2004) apresentam o projeto Mobile Learning Tools for Science (MaLTS). Este trabalho tem como principal objetivo apresentar orientações para o desenvolvimento de soluções educacionais utilizando-se dispositivos móveis. O projeto foi dividido em três fases.

Na primeira fase, foi analisado material de pesquisa relacionado a softwares educativos em geral. Inicialmente, foram estudadas as estratégias cognitivas dos alunos de ciências. Depois, ferramentas educacionais para computadores pessoais foram analisadas. E, por último, foram pesquisados requisitos de usabilidade em aplicações para dispositivos móveis. Os estudos identificaram duas características principais em aplicações com alto grau de usabilidade em dispositivos móveis: aplicações sucintas, que apresentam somente a informação necessária e de forma altamente acessível; e, aplicações automatizadas, que reduzem a necessidade do usuário entrar textos ou executar comandos. Entretanto, as principais contribuições desta fase são as duas orientações para projetos de soluções educacionais em dispositivos móveis, que são:

- Telas baseadas em tarefas. A atividade da aprendizagem deve ser decomposta em tarefas e para cada uma deve ser elaborada uma tela contendo as informações necessárias específicas.
- Elementos com propósito duplo. As telas devem incluir componentes de interface que tenham funções educacionais e operacionais, como imagens que apresentam um conteúdo e funcionam como *links* para outras telas, por exemplo: uma imagem de um corpo humano onde cada parte abriria uma outra tela com informações relacionadas à parte selecionada.

A segunda fase apresentou três softwares educativos desenvolvidos para handhelds Pocket PC, que foram construídos utilizando-se as conclusões da fase anterior. E na fase

três, foram apresentados os resultados da aplicação das soluções desenvolvidas em alunos da oitava série. A avaliação didática da solução foi baseada nos critérios apresentados por Quintana et al. (2003). O trabalho destaca o sucesso na utilização de softwares educativos móveis baseados em tarefas, seguindo a linha instrucionista, porém alerta que atividades complexas podem dificultar a usabilidade e comprometer a aprendizagem.

2.3.2. *Palmtops* Aplicados ao Estudo de Doenças

A Universidade de Michigan desenvolveu um programa para *palmtops* que auxilia as crianças a compreender as implicações biológicas e sociais das doenças, denominado Cooties® (Cooties, 2004).

O Cooties® é um jogo de simulação no qual um grupo de estudantes examina a evolução de doenças. Inicialmente, os alunos definem seus próprios personagens em seus *palmtops*. Enquanto isso, o professor secretamente determina qual o estudante será infectado com o vírus, o nível de imunidade dos outros e o tempo de incubação do vírus. Depois os alunos trocam informações via infravermelho, podendo transmitir o vírus entre si. Quando o personagem de um aluno ficar doente, ele deve sentar-se e os outros ficam analisando a evolução da doença entre eles, com a ajuda do programa instalado no PDA. A Figura 2-7 apresenta telas do programa, para o professor (2-7a) e para o aluno (2-7b).

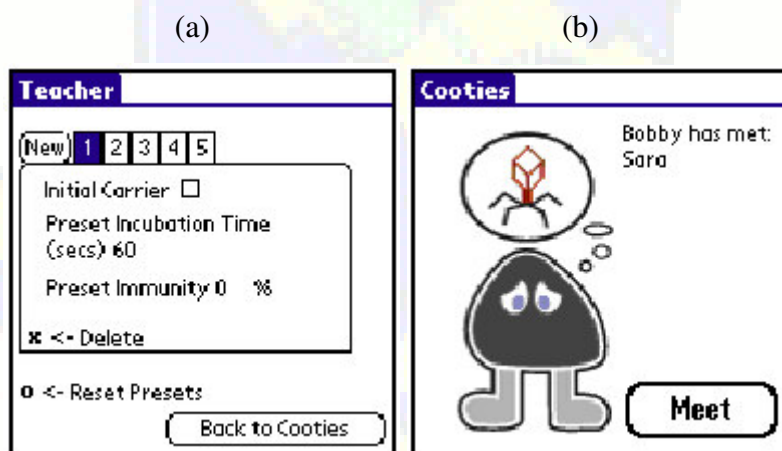


Figura 2-7: Telas do programa Cooties®.

De acordo com Shields & Poftak (2002), os benefícios do Cooties® são:

- Capitalizar a característica das crianças de trabalharem em pequenos grupos, com uma abordagem construtivista;
- Ensinar brincando;
- Fornecer um interessante meio de discussão de assuntos polêmicos da vida real, como doenças sexualmente transmissíveis.

Entretanto, algumas escolas que utilizaram o Cooties[®] relataram os seguintes problemas com relação à utilização de dispositivos móveis em sala de aula:

- Os alunos utilizavam o palmtop como controle remoto para operar a televisão da sala de aula;
- Os alunos ficavam jogando durante as aulas;
- Os alunos ficaram trocando mensagens pessoais durante as aulas.

Este trabalho apresenta contribuições importantes ao apresentar vantagens e pontos negativos vivenciados na prática dos alunos.

2.3.3. Computação Móvel Aplicada ao Ensino de Genética

O laboratório EDGE (Exploring Dynamic Groupware Enviroments) da Universidade Dalhousie no Canadá desenvolveu um programa para auxiliar as crianças a explorarem os conceitos de genética, denominado Geney[®] (Danesh, 2001).

Os pesquisadores queriam investigar como atividades de estudo em grupo poderiam ser estruturadas utilizando-se dispositivos móveis. Então surgiram algumas dúvidas:

- Como os PDAs poderiam ser adaptados para serem usados por crianças, tendo em vista terem sido projetados para adultos?
- Como os dispositivos móveis poderiam ser utilizados em atividades colaborativas?

O Geney[®] foi desenvolvido em um processo iterativo e centrado no usuário. O programa simula uma população de peixes representando um reservatório. Os peixes são distribuídos entre os PDAs, de forma que cada aluno/PDA fica com um tanque de peixes. Os peixes mais velhos morrem com o tempo. Os alunos podem cruzar os peixes e, conseqüentemente, obter novos peixes. Os alunos também podem trocar peixes com os colegas via infravermelho, possibilitando um aumento da variedade genética no seu tanque.

O objetivo deste jogo é provocar uma cooperação entre os alunos a fim de gerar uma espécie de peixe com características particulares. Estas características são definidas no início da aprendizagem. Os alunos devem cooperar entre si, porque somente assim eles podem alcançar o objetivo principal.

Os pesquisadores desenvolveram um trabalho aprofundado que envolveu os seguintes processos:

- Análise de requisitos usando cenários e modelos em tamanho real.
- Validação dos requisitos juntamente aos usuários, na qual se analisou como as crianças interagiriam com um palmtop.
- Desenvolvimento de protótipos.
- Teste dos protótipos com os usuários.
- Elaboração de uma especificação completa da solução.

Mandryk et al. (2001) destacam como principal característica do projeto Geney[®] a grande interação promovida, aumentando sensivelmente o grau de motivação dos alunos.

2.3.4. Dispositivos Móveis na Educação Física

Computadores de mesa têm um papel importante ao auxiliar o ensino, porém limitam-se ao laboratório ou à sala de aula onde estão instalados. Esta limitação se torna mais crítica ao se tratar da educação física, onde os alunos podem estar aprendendo em ginásios ou pistas de corrida. Nestes casos, os dispositivos móveis surgem como uma alternativa interessante.

Aplicações em educação física permitem a coleta de dados, análise de informações e visualização de exames em campo. Essas possibilidades podem ser tanto usadas pelos professores para monitorar seus alunos como para o próprio estudante fazer um auto-monitoramento. A Figura 2-8 apresenta telas de aplicações de educação física em dispositivos móveis (Palmgear, 2004).

As telas apresentadas na Figura 2-8 mostram informações que podem auxiliar professores e alunos de educação física e instrutores pessoais (*personal trainers*) tanto na aprendizagem como na execução de suas atividades.

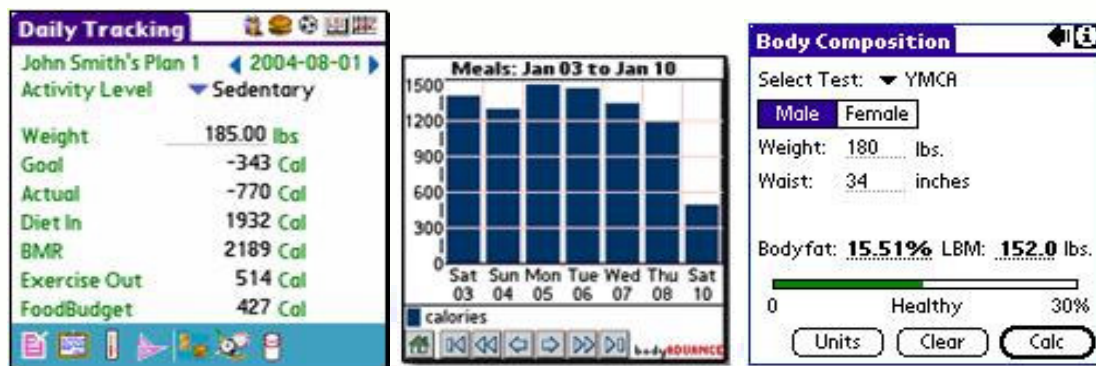


Figura 2-8: Telas de aplicações de educação física em *palmtops* (Palmgear, 2004).

2.3.5. Computação Móvel em Museus

Hsi (2002) apresenta os resultados de um projeto de pesquisa sobre a utilização de dispositivos móveis em visitas a museus.

O projeto foi desenvolvido no museu Exploratorium (Exploratorium, 2004) em São Francisco nos Estados Unidos. Este é um museu especial onde os visitantes interagem com as exposições, realizando experimentos. O objetivo geral do projeto é disponibilizar PDAs aos visitantes para melhorar a compreensão das experiências vivenciadas durante a visita. Mais especificamente, pretende-se:

- Fornecer informação adicional sobre as exposições;
- Capturar as experiências vividas, registrando-as para análises futuras;
- Possibilitar a simulação das exposições para aumentar a compreensão sobre os fenômenos demonstrados na exposição.

Os PDAs são equipados com tags RFID para identificar os visitantes e poder rastreá-los durante a excursão. Ao aproximar-se de uma determinada exposição, o PDA recebe a URL relacionada através de infravermelho. Desta forma, através de uma WLAN, o web browser do dispositivo móvel apresenta informações variadas sobre aquela exposição que está visitando.

Os pesquisadores identificaram importantes interações dos visitantes entre si, com os professores e com o sistema, que são descritas na Tabela 2-2.

Tabela 2-2: Formas de interação identificadas.

Interação	Usuário	Sistema
Exploração/Localização	Aproxima-se da exposição	Envia a URL para o visitante próximo
Requisição de informação	Solicita informações	Responde com informações
Documentação	Registra informações	Armazena as informações em um banco de dados central
Recomendação	Aproxima-se da exposição	Sugere formas do visitante melhor compreender a exposição
Simulação	Solicita uma simulação	Simula o experimento da exposição
Comunicação	Comunica-se com outros visitantes	
Avaliação	Pode se auto-avaliar, avaliar outros visitantes ou ser avaliado pelo professor	

O dispositivo móvel utilizado pelo projeto foi o HP Jornada 690, devido a sua tela grande (640 x 240 pixels), suporte ao IEEE 802.11b, bateria de boa duração e peso aceitável. A única aplicação utilizada foi o web browser do dispositivo, que era responsável por apresentar textos, áudios e vídeos, registrar as anotações e permitir a comunicação com outros usuários. As informações podiam ser obtidas tanto na intranet do museu como externamente, na Internet.

O sistema foi aplicado com guias, visitantes e professores e a maioria percebeu um aumento na aprendizagem das exposições. Também mereceu destaque a possibilidade de registrar as experiências. A possibilidade de simular as exposições no próprio PDA aumentou a motivação para executá-las realmente e despertou o raciocínio. Os professores participantes relataram a importância de se utilizar o dispositivo móvel antes, durante e

depois da visita ao museu. Alguns participantes do estudo relataram a sensação de isolamento durante a visita com os PDAs. Também foi percebida a necessidade de um sincronismo entre a simulação no dispositivo móvel e a exposição no mundo real.

2.3.6. Um Jogo Educativo utilizando PDAs em um Campus Universitário

Uma das aplicações com grande disseminação em dispositivos móveis são os jogos. Göth et al (2004) descrevem uma aplicação que une jogos baseados em localização (Benford et al., 2003) e *Mobile Learning*, utilizando PDAs com recursos de comunicação sem fio.

O jogo funciona como uma forma divertida de excursão pelo campus de uma universidade. Os alunos são agrupados de acordo com seus perfis, previamente preenchidos, e cada grupo recebe um PDA. As equipes devem realizar algumas tarefas relacionadas a locais (como bibliotecas, livrarias ou lanchonetes), eventos (como peças, cursos ou palestras) ou funcionários da universidade (como o reitor, o coordenador do curso ou um determinado professor). O grupo só pode passar para uma nova tarefa após executar a anterior. A posição atual do grupo é mostrada em um mapa digital no dispositivo móvel. Os mapas vão se alternando entre o exterior e o interior dos prédios. Outros alunos podem acompanhar a disputa pela Internet através de web browser. O jogo fundamenta-se na teoria construtivista ao encorajar, através das tarefas, os alunos a formarem seu próprio conhecimento sobre a universidade.

A arquitetura do sistema foi implementada utilizando o modelo cliente-servidor e é formada por três componentes: aplicação cliente no PDA, web browser e aplicação no servidor. A aplicação cliente foi desenvolvida em Java e comunica-se com o servidor através da RMI¹ (*Remote Method Invocation*). O servidor também foi implementado em Java (Java, 2004) e o banco de dados utilizado foi o MySQL (MySQL, 2004). A localização dos grupos é identificada de duas maneiras:

¹ RMI é um conjunto de classes e interfaces em Java que encapsulam vários mecanismos de troca de dados.

- Dispositivo GPS acoplado ao PDA é utilizado nas áreas fora dos prédios. Porém ocasiona alguns problemas de mapeamento devido à imprecisão do sistema GPS.
- WLANs são instaladas no interior dos prédios da universidade. Desta forma, obtém-se uma precisão com variação de um a três metros do posicionamento real.

Um protótipo foi aplicado na Universidade de Koblenz, na Alemanha, e as seguintes lições foram aprendidas:

- Os jogadores afirmaram que não era possível se movimentar e navegar no PDA. E os pesquisadores concluíram que uma boa solução para esse problema seria utilizar recursos de áudio para auxiliar o posicionamento, evitando a necessidade de parar freqüentemente para olhar a tela do dispositivo móvel. Fones de ouvido bluetooth poderiam ser utilizados, mantendo a sensação de mobilidade sem fio.
- Os estudantes destacaram a boa precisão obtida dentro dos prédios, principalmente quando comparada à obtida do lado de fora, com o sistema GPS. Porém os pesquisadores analisam a possibilidade de utilizar etiquetas RFID para aumentar a precisão, diminuindo a margem de erro para menos de um metro.
- Uma grande facilidade apontada pelos alunos foi a utilização de botões na tela sob o mapa digital. Assim, com um ou dois cliques o usuário executava uma determinada função. A utilização de ícones no lugar de textos nos botões permitiu um ganho de espaço na interface.
- Os alunos também apontaram uma dificuldade em responder perguntas subjetivas, preferindo questões de múltipla escolha.

2.4. Requisitos das Aplicações

Ao se pensar nas implicações didático-pedagógicas da computação em todo lugar e durante todo o tempo, novos desafios se apresentam no campo da Informática na Educação. Desta forma, baseado no exposto na seção anterior, importantes requisitos devem ser identificados em aplicações de *Mobile Learning*.

O primeiro requisito que surge é a aplicação poder ser usada em qualquer lugar e a qualquer momento, sempre que for preciso. Nesse caso, o ambiente do usuário pode ser muito variado. Por exemplo, pode não ter uma mesa ou uma cadeira disponível ou ele pode estar em movimento. Desta forma, a aplicação não pode ser dependente de recursos de comunicação, como uma rede sem fio, ou de acessórios, como um teclado portátil.

Apesar de promoverem uma organização mais eficiente da informação, as tecnologias de computação móvel devem ser utilizadas em função das tarefas a realizar, e não em função de si mesmas. Isto significa que é um requisito importante saber distinguir qual informação é essencial manter e organizar, e descartar a informação supérflua.

A aplicação deve ser motivadora, combinando interfaces ricas em detalhes, como áudio e vídeo, com variadas formas de interação. Portanto, a facilidade de uso é outro relevante aspecto de motivação. É sempre importante lembrar que o usuário não tem como atividade fim utilizar aquela aplicação móvel, e sim através dela atingir uma determinada finalidade. Algumas técnicas que facilitam o uso da aplicação são:

- Evitar ao máximo a entrada de texto, dando preferência às questões objetivas e utilizando valores tabulados;
- Evitar uma grande profundidade na telas da aplicação, ou seja, uma tela que chama outra que chama outra e assim por diante.
- Utilizar imagens como atalho para a chamada de rotinas ou de outras telas.

Desta forma, programas simples, intuitivos e de fácil assimilação terão uma menor rejeição por parte do usuário.

Um outro requisito é que a aplicação deve ser adaptada às características do usuário. Por exemplo, um usuário do tipo aluno não poderá acessar informações nem executar funções restritas a usuários do tipo professor, e vice-versa. Outro exemplo são aplicações com características variadas de acordo com a faixa etária do usuário. Uma alternativa para se implementar esses requisitos é a utilização de um cadastro hierárquico de usuários. Este também pode armazenar o estágio atual do usuário, podendo assim identificar qual o último e o próximo passo. Esta é uma importante funcionalidade, tendo em vista as interrupções que o dispositivo móvel pode sofrer, como uma ligação telefônica durante a execução da aplicação.

Outro requisito importante para aplicações de m-Learning é a possibilidade de aprendizagem em grupo, através da colaboração ou competição entre os alunos. Os recursos de comunicação sem fio facilitam a interação dos aprendizes e professores durante a prática educacional, motivando a troca de idéias e experiências, e incrementando a aprendizagem.

A energia é um recurso limitado em dispositivos móveis e o seu gerenciamento é um sério desafio a ser tratado tanto pelo hardware quanto pelo software. Meios de transmissão de dados sem fio, como o Bluetooth e o IEEE 802.11, têm um alto consumo de bateria. Portanto, a aplicação deve utilizar de forma otimizada esse recurso de comunicação, o que constitui um importante requisito. Uma medida para o controle do consumo de energia via software é ativar as conexões apenas no momento da transmissão e desativar posteriormente. Outro fator importante que deve ser analisado é a apresentação do conteúdo. Recursos de áudio e vídeo têm elevado consumo de energia e devem ser utilizados apenas quando forem realmente necessários, devendo ser substituídos por textos e imagens estáticas.

A comunicação sem fio é uma característica dos ambientes de computação móvel, porém nem sempre está presente. Um dos requisitos principais no projeto de uma aplicação para usuários móveis é o tratamento de operações no modo *offline* (desconectado). Sendo assim, a aprendizagem não deve ser interrompida em caso de queda ou ausência da conexão. Algumas das questões a serem analisadas são: que informações devem ser carregadas para a memória do computador móvel antes de haver a desconexão? E como é feita a atualização no servidor das informações modificadas localmente?

Uma característica indispensável em aplicações educacionais em dispositivos móveis é a capacidade de registrar informações, seja através de respostas a questionários ou através de anotações. Esses dados tanto podem ser utilizados para avaliações quanto para consultas futuras. É importante salientar que esse registro pode ser feito através de texto comum, gravações de som, fotografias ou filmagens, variando de acordo com as características do dispositivo móvel.

2.5. Considerações Finais

A oferta de serviços de telecomunicações e de artefatos computacionais, capazes de prover mobilidade aos diferentes atores envolvidos em projetos educacionais na modalidade *online* enseja o desenvolvimento de pesquisas no campo da computação móvel. Os temas de investigação poderão ser bastante variados, entre os quais, estudos que busquem relacionar estilos cognitivos e de aprendizagem com as funcionalidades dos dispositivos móveis e de sistemas gerenciadores de aprendizagem *online*.

Este capítulo mostrou diferentes aplicações de m-Learning, suas vantagens e desvantagens e relacionou importantes requisitos encontrados em aplicações educacionais em dispositivos móveis.

A proliferação de novos dispositivos computacionais portáteis juntamente com a oferta de serviços de comunicação sem fio apresenta um cenário propício para o desenvolvimento de projetos na área de *Mobile Learning*. Desta forma, mecanismos que auxiliem a construção de aplicações nesta área tornam-se essenciais.

O próximo capítulo abordará os aspectos da utilização da realidade virtual na educação, seus benefícios, limitações e exemplos de aplicação.

CAPÍTULO 3

A Realidade Virtual na Educação

Este capítulo introduz o conceito de realidade virtual (RV) e apresenta sua aplicação na área educacional. A Seção 3.1 apresenta conceitos introdutórios relacionados à realidade virtual. A Seção 3.2 apresenta os benefícios da realidade virtual para a aprendizagem e alguns exemplos de aplicações de RV na educação. A Seção 3.3 relaciona importantes requisitos encontrados em ambientes virtuais interativos voltados para aprendizagem. E a Seção 3.4 apresenta as considerações deste capítulo.

3.1. Introdução

Segundo Hamit (1993), o termo “realidade virtual” foi criado, de acordo com alguns relatos, no Massachusetts Institute of Technology – MIT, no final da década de 70, para expressar a idéia de presença humana em um espaço gerado por computador. De uma maneira simplificada, a realidade virtual pode ser definida como sendo a forma mais avançada de interface do usuário de computador até agora disponível (Hancock, 1995). Agrupando algumas outras definições de realidade virtual (RV) (Burdea et al., 1994; Jacobson, 1991; Krueger, 1991), pode-se dizer que realidade virtual é uma técnica avançada de interface, onde o usuário pode realizar imersão, navegação e interação em um ambiente sintético tridimensional gerado por computador, utilizando canais multi-sensoriais (Kirner, 2004).

A interface em RV envolve um controle tridimensional altamente interativo de processos computacionais. O usuário entra no espaço virtual das aplicações e visualiza, manipula e explora os dados da aplicação em tempo real, usando seus sentidos, particularmente, os movimentos naturais tridimensionais do corpo. Usualmente um ambiente virtual, ou mundo virtual, é um espaço tridimensional (3D) gerado por computador no qual uma pessoa pode ser inserida e com o qual ela pode explorar e interagir. Na prática, a RV permite que o usuário navegue e observe um mundo tridimensional, em tempo real e com seis graus de liberdade (rotação e translação nas três

dimensões). Isso exige a capacidade de o software definir, e a capacidade do hardware de reconhecer, seis tipos de movimento: para frente/para trás, para cima/para baixo, para a esquerda/para a direita, inclinação para cima/para baixo, angulação à esquerda/à direita e rotação à esquerda/à direita. Algumas definições são importantes.

3.1.1. Realidade Virtual Imersiva e Semi-Imersiva

A realidade virtual normalmente está associada a equipamentos sofisticados, como HMD (*Head-Mounted Display* – tipo especial de capacete que possui um mini-monitor e, geralmente, também um par de autos-falantes) e *datagloves* (luvas especiais que atuam como interface direta entre a mão de um indivíduo e sua representação no interior de um ambiente de realidade virtual) (Figura 3-1). Este é um exemplo de realidade virtual imersiva, no qual o participante recebe três tipos de estímulos: visual, auditivo e háptico/sinestésico. A realidade virtual imersiva é atualmente mais utilizada em simulações militares, em projetos industriais e em visualização científica.



Figura 3-1: Ambiente de realidade virtual imersiva.

A realidade virtual semi-imersiva, baseia sua interação principalmente nos estímulos visuais e auditivos, oferecendo um conjunto mais restrito de sensações (Leston, 1996). De acordo com Ellis (1991), a realidade virtual semi-imersiva situa-se no nível “espaço virtual”, fazendo uso dos monitores dos computadores de mesa para a simulação dos mundos virtuais. Desta forma, esta modalidade de realidade virtual é também

conhecida como realidade virtual de mesa (*desktop virtual reality*) (Robertson, 1993). A Figura 3-2 apresenta os equipamentos necessários para uma aplicação de realidade virtual semi-imersiva.

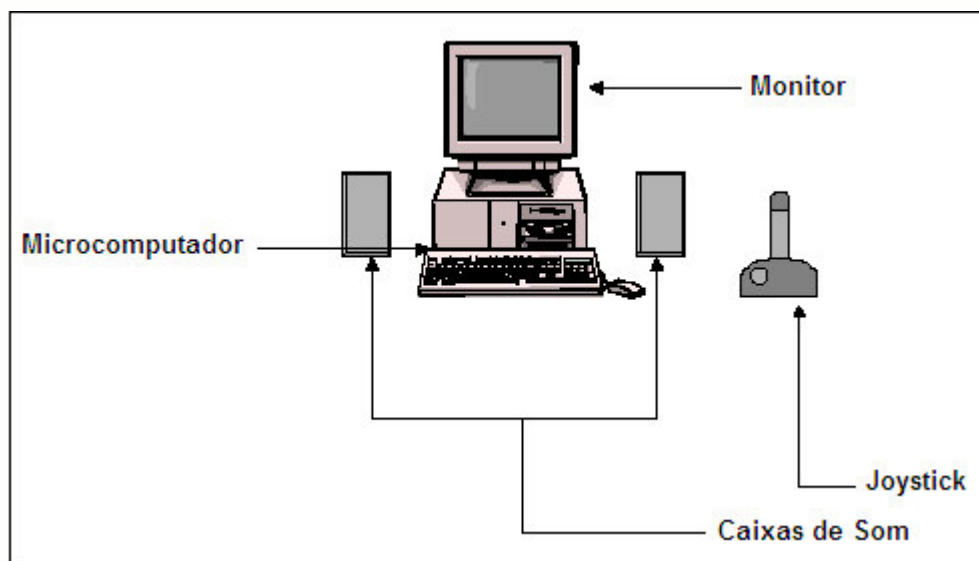


Figura 3-2: Equipamentos de realidade virtual não imersiva.

Embora a realidade virtual imersiva tenha evoluído e seja considerada quase ideal, a realidade virtual com monitor apresenta, ainda assim, pontos positivos, tais como: utiliza plenamente todas as vantagens da evolução da indústria de computadores pessoais, apresentando um menor custo em relação a modelos imersivos; apresenta facilidade de uso e alta disponibilidade de equipamentos. Estas características possibilitam uma maior expansão das aplicações de RV semi-imersiva nas mais diversas áreas do conhecimento (Fraser et al., 2000). Nestes sistemas, o usuário precisa estar constantemente olhando para o monitor ou tela, e utilizar algum dispositivo de entrada (geralmente *mouse* ou teclado) para controlar sua movimentação pelo mundo virtual.

Uma das áreas de maior aplicação da realidade virtual semi-imersiva é a apresentação de produtos, sejam protótipos ou produtos acabados (Dachselt, 2000). É possível não apenas visualizar um produto, como também vê-lo em funcionamento e interagir com ele. Outro importante exemplo de realidade virtual semi-imersiva bastante utilizado é o jogo de computador em três dimensões.

3.1.2. Simulação Tridimensional

Simulação é o uso do software de computador para modelar e analisar o comportamento de elementos existentes no mundo real (Adams, 1994). A computação gráfica, através dos ambientes em três dimensões, enriquece as simulações através do incremento do realismo.

As simulações tridimensionais possuem diversas aplicações. Quando são baseadas em conhecimento técnico se transformam em interessantes ferramentas de treinamento. Assim, a realidade virtual possibilita a aprendizagem baseada em simulações bem próximas a experiências reais (Johnson & Rickel, 1998).

A simulação pode ser aplicada a elementos naturais ou fabricados. Um dos mais clássicos exemplos de simulação tridimensional é o simulador de vôo. Este software modela e simula um avião e seus comportamentos, como decolagem e aterrissagem. Um simulador biológico, por outro lado, modela e simula elementos naturais, cromossomos, por exemplo.

Um conceito importante relacionado a simulações é o canal de simulação: a cadeia de funções de software que produz a simulação em tempo real. Este baseia-se principalmente em funções 3D e instrumentos de animações.

Para aperfeiçoar a precisão e o realismo, a simulação pode ser baseada em alguns métodos científicos. Esta base pode incluir a teoria dos jogos, tabelas de decisão, probabilidades, leis da física, leis da química, entre outros.

Outros conceitos relacionados a simulações tridimensionais são:

- Universo, que é um ambiente 3D particular, incluindo as entidades que ele contém e o script para controlar a simulação;
- Entidade, que inclui os atores, os adereços ou os elementos de cenário que existem no ambiente virtual;
- Gerenciador de simulação, que é o componente responsável pelo controle da simulação, executando as funções de atualização da imagem, de aplicação das regras da simulação, de detecção das ações do usuário e de movimentação das entidades.

- Regras da simulação, que são leis que governam o funcionamento da simulação;
- Graus de liberdade, que se referem ao eixo de rotação e translação proporcionado por um dispositivo de entrada, conforme visto anteriormente.

3.1.3. Realidade Virtual Passiva, Exploratória ou Interativa

Um aplicativo de RV pode proporcionar uma sessão sob três formas diferentes: Passiva, Exploratória ou Interativa (Adams, 1994).

Uma sessão de RV passiva proporciona ao usuário uma exploração do ambiente automática e sem interferência. A rota e os pontos de observação são explícitos e controlados exclusivamente pelo software. O usuário não tem controle algum, exceto talvez, para sair da sessão.

Uma sessão de RV exploratória proporciona uma exploração do ambiente dirigida pelo usuário. O participante pode escolher a rota e os pontos de observação, mas não pode interagir de outra forma com entidades contidas na cena.

Uma sessão de RV interativa proporciona uma exploração do ambiente dirigida pelo usuário e, além disso, as entidades virtuais do ambiente respondem e reagem às ações do participante. Por exemplo, se o usuário move o ponto de observação em direção à porta, esta pode aparecer abrindo-se, permitindo ao participante passar por ela. A navegação em ambientes virtuais é controlada pelo posicionamento do observador no mundo virtual, o chamado ponto de observação (*viewpoint*). A navegação é, provavelmente, a forma mais simples de interação encontrada em muitas aplicações de RV. Virtualmente, toda técnica de navegação pode ser deduzida a partir de um único modelo, que assume uma câmera montada em um “carro” virtual que, às vezes, é referenciado como *Flying Carpet Model* (modelo do tapete voador). Algumas das técnicas de navegação mais usuais são:

- *Point-and-fly*, em que o usuário move o carro virtual apontando em uma determinada direção com os dispositivos de navegação (por exemplo, uma luva), e faz um certo gesto ou aperta determinado botão. Se a luva estiver sendo usada, a velocidade de movimentação pode ser controlada. Se o rastreador de cabeça estiver ativo, o mesmo controlará a câmera virtual. Essa

técnica é uma das mais usadas e sugere um modo mais sofisticado. Neste caso, o usuário aponta para um objeto desejado e o sistema computa o desvio que posicionará o usuário em frente ao objeto, sendo que a velocidade pode também ser controlada. Algumas vezes, é desejável restringir o carro a uma certa altura, por exemplo, ao nível dos olhos, e o usuário pode então se mover ao redor da cena, nessa altura fixa;

- *Eyeball-in-hand* é um paradigma implementado pela realimentação do sistema de rastreamento (por exemplo, a posição de um sensor eletromagnético), diretamente para o ponto e a direção de observação, enquanto o ponto de observação permanece fixo. Essa técnica é muito apropriada para a análise de um único objeto a partir de diferentes pontos de observação, por exemplo, no projeto de interiores. Em resumo, todas as ações são carregadas/atualizadas a partir do ponto de observação do usuário.
- *Scene-in-hand* é paradigma oposto ao anterior. A técnica mapeia os movimentos dos dispositivos de entrada 3D dentro de um mundo virtual, mantendo o ponto de observação do usuário e movendo o mundo em volta.

Algumas vezes é necessário controlar o ponto de observação sem o uso das mãos. Nesse caso, o reconhecimento de voz pode ser empregado para mover o carro virtual, proferindo comandos simples como: à esquerda, pare, etc.

Para aumentar a flexibilidade, é altamente desejável que diferentes modos de navegação sejam mapeados para diferentes configurações de dispositivos de entrada, já que certas combinações de modos de navegação e configurações de dispositivos de entrada tendem a ser usadas com mais frequência do que outras. Existem muitos parâmetros que afetam a representação do usuário e a navegação, como velocidade de navegação, tamanho da mão, escala dos movimentos da cabeça (*head motion*) e separação dos olhos.

3.1.4 Ambientes Virtuais Interativos

A partir de modelos tridimensionais, é possível representar uma grande variedade de situações, voltadas para diversas áreas de aplicação, tais como mundos virtuais reais (museus, terrenos, etc) ou imaginários, objetos (carros, máquinas, etc) ou personificações

de seres reais (homem, animal, etc) ou imaginários (alienígenas). Percebe-se então que as possibilidades de interação disponíveis em ambientes de realidade virtual estão bem relacionadas com as áreas de aplicações às quais se destinam.

A interação em ambientes virtuais é realizada com o objetivo de efetuar operações de seleção, manipulação e navegação no ambiente tridimensional. Ações (estímulos motores) realizadas pelo usuário são concretizadas através de técnicas de interação, que correspondem a métodos através dos quais o usuário especifica comandos e dados para o sistema computacional, visando a execução de tarefas. A resposta do sistema se dá através de dispositivos de saída que estimulam os sentidos do usuário (normalmente tato, visão e audição). Tais estímulos são processados e assimilados pelo cérebro resultando o que se denomina de percepção. A percepção e as expectativas criadas com base em experiências anteriores definem como as pessoas interagem com o ambiente. Alguns dos atributos utilizados no processo perceptivo são cor, textura, tamanho, perspectiva, oclusão, sombra, movimento do observador (Nedel et al., 2003).

Em ambientes de visualização científica, por exemplo, onde o objetivo é a exploração visual de representações tridimensionais de dados (terrenos, equipamentos industriais, etc), a interação não se baseia em processos de manipulação humanos, visto que a total liberdade de manipulação do objeto de estudo é o grande atrativo desse modelo de realidade virtual. Assim, esses ambientes podem fornecer recursos de interação impossíveis de serem reproduzidos em um mundo real (por exemplo, observar a estrutura interna de objetos complexos).

Já em ambientes de simulação, onde o objetivo é reproduzir, dentro de mundos virtuais, situações reais do cotidiano, os processos de interação disponíveis são baseados nas possibilidades e limitações do mundo real. Assim, as pessoas são levadas a interagir com o mundo virtual da mesma forma que interagiriam em um mundo real. Algumas das áreas de aplicação de ambientes de simulação são: o treinamento militar, a aviação civil, a educação e o entretenimento. Johnson (1993) apresenta um exemplo de ambiente virtual interativo voltado para treinamento industrial, onde a representação realística dos equipamentos e do local de trabalho é fator decisivo para o sucesso do treinamento.

3.2. Aplicações da Realidade Virtual na Aprendizagem

Buscando novas formas de obter uma melhoria organizacional das empresas e propiciar a modernização do sistema produtivo, alguns pesquisadores estão propondo a utilização de *software* baseado em ambientes 3D para simulação de equipamentos, treinamento de funcionários, validação de planejamento de produção, visualização de *layouts* de fábricas e prototipação de produtos (Johnson, 1993; McCarty, 1994; Owen, 1995; Ressler, 1997).

Vários autores relatam o uso de simulações tridimensionais no ensino e treinamento, comprovando a eficiência desta tecnologia para tal fim (Byrne et al., 1995; Dede et al., 1994; Rose et al., 1995). Através de um *software* de desenvolvimento de ambientes 3D, é possível modelar maquinários, veículos e dispositivos, e simular o comportamento real do equipamento. Isso pode economizar dinheiro e ciclos de desenvolvimento, e permitir sessões de treinamento e validação feitas com o produto virtual.

Com a realidade virtual, o aluno pode controlar ativamente o ambiente e observar diretamente o comportamento de objetos no interior do mundo virtual. Desta forma, incrementa-se o significado de fenômenos multidimensionais e proporciona-se uma melhor compreensão em termos qualitativos (Erickson, 1993). Alunos são estimulados por suas interações em ambientes virtuais educacionais, induzindo-os a passar a dar mais atenção e utilizar-se de uma maior concentração em uma tarefa (Bricken et al., 1993).

Nas próximas sub-seções serão apresentados exemplos da aplicação da realidade virtual no aprendizado. Estes têm como principal propósito o ensino, apesar de alguns alcançarem outros objetivos como o entretenimento.

3.2.1. Ensino através de Experiências

Em Roussou (2004) são apresentados exemplos de aplicações de realidade virtual voltadas para o ensino de crianças através de experiências interativas. O principal aspecto abordado nestes exemplos é a interatividade. Roussou apresenta o projeto NICE (Narrative-based, Immersive, Constructionist/Collaborative Environments), onde as crianças podem colaborar entre si, plantar um jardim e construir histórias em um ambiente virtual imersivo (Figura 3-3).

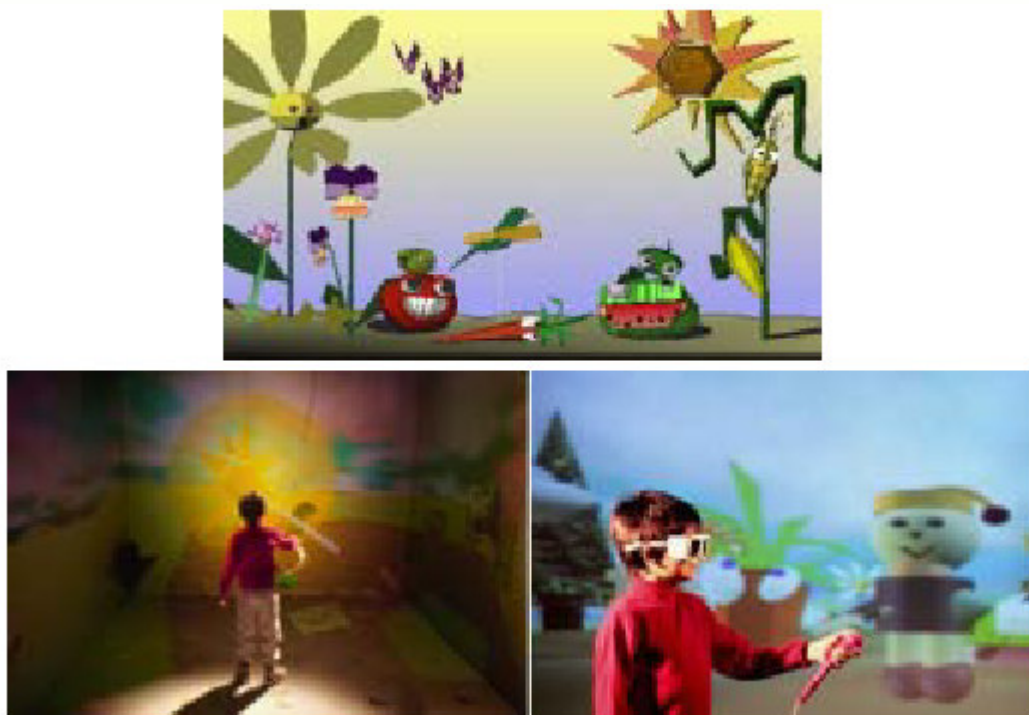


Figura 3-3: Crianças plantando em um jardim virtual (Roussou, 2004).

O projeto NICE combinou uma abordagem construtivista, colaboração através de tele-presença e ferramentas interativas para ajudar as crianças a cultivar o jardim virtual.

Um outro exemplo apresentado por Roussou é um museu virtual interativo na Fundação do Mundo Grego (Foundation of the Hellenic World - FHW). A realidade virtual é usada tanto como ferramenta educacional/recreativa como instrumento para pesquisa histórica. A Figura 3-4 apresenta um grupo de crianças interagindo no ambiente virtual gerado. As exposições de RV foram abertas ao público em 1999 e estão entre as principais atrações do museu da FHW.

Roussou ainda apresenta outros exemplos, sempre destacando o potencial da interatividade em ambientes virtuais na educação formal e informal.

3.2.2. Ensino de Física

C-VISions é um ambiente virtual colaborativo em rede (AVC – Ambiente Virtual Colaborativo) voltado à aprendizagem (Wang & Chee, 2001). Através da Internet, em computadores pessoais comuns, os alunos podem estudar em grupo fenômenos físicos como queda livre, movimento de projéteis e conservação de momento (Figura 3-5).



Figura 3-4: Crianças em ambiente virtual no museu da FHW (Roussou, 2004).

O C-VISions fornece muitas ferramentas para suportar a aprendizagem em grupo, como bate-papo textual, áudio conferência, vídeos e quadro de anotações compartilhado, permitindo assim, que os alunos partilhem idéias e recursos. A aprendizagem também pode se dar através de gráficos, que mostram a atuação das forças nos objetos, como um gráfico de velocidade versus tempo durante a queda de um objeto. O desenvolvimento e os testes visam investigar modelos de interfaces que promovam uma coordenação dos processos cooperativos de maneira mais eficiente.



Figura 3-5: Interface do C-VISions (Wang & Chee, 2001).

3.2.3. Realidade Virtual e Realidade Aumentada

Realidade aumentada (RA) é uma área da RV que utiliza tecnologias específicas para aumentar o desempenho humano na realização de tarefas. A RA permite combinar imagens geradas no mundo virtual com imagens do mundo real por meio de um capacete parcialmente transparente provido de sensores. O objetivo é suplementar um cenário real com informações geradas pelo computador. Desta forma, é possível enxergar-se, por exemplo, um objeto real com o seu detalhamento interno gerado por realidade virtual (Figura 3-6). Este conceito foi primeiramente introduzido por Sutherland (1968). Essa tecnologia pode ser aplicada em diversas áreas tais como manutenção e reparo (Feiner et al., 1997; Lipson et al., 1998), montagem de portas de carros (Reiners et al., 1999) e assistência em cirurgia (State et al., 1996).



Figura 3-6: Aplicação em realidade aumentada para área médica (State et al., 1996).

Em Feiner et al.(1997) é apresentado um trabalho que tem como principal objetivo a utilização de realidade aumentada combinada com dispositivos móveis para auxiliar interações diárias. A solução apresentada é composta por um capacete 3D conectado a um notebook, dentro de uma mochila, um handheld e um equipamento GPS, também conectado ao notebook (Figura 3-7). Este sistema foi projetado para auxiliar um usuário em uma excursão em um campus universitário. À medida que o usuário vai se deslocando no campus, o mundo virtual é gerado, sobrepondo o real. Sobre cada prédio do campus é exibido um *link* que, se selecionado, através de um *trackpad*, carrega informações no

handheld. Desta forma, o handheld, conectado à intranet da universidade, através de uma rede Wi-Fi, funciona como uma biblioteca de referências para apoiar a excursão. O pacote gráfico utilizado para a implementação do ambiente virtual foi o Obliq-3D (Najork & Brown, 1995).



Figura 3-7: Equipamentos de RA combinados com dispositivos móveis(Feiner et al., 1997).

Wojciechowski et al. (2004) apresenta um sistema que permite aos museus construir e gerenciar exposições em realidade virtual e aumentada. O sistema utiliza modelos pré-projetados para permitir uma maior eficiência na criação do ambiente virtual. A exibição em realidade virtual pode se dar tanto dentro do museu, através de monitores *touch-screen*, como via Internet. A apresentação baseada em realidade aumentada permite aos visitantes interagirem com um conteúdo de uma forma mais intuitiva e excitante. A Figura 3-8 ilustra uma exposição virtual construída baseando-se em um museu real. Um dos principais objetivos desse sistema é apresentar os artefatos do museu de uma forma atrativa, fornecendo às pessoas, especialmente às crianças, um incentivo ao estudo da herança cultural.

3.2.4. Treinamento para Segurança no Trabalho para a Área de Petróleo

O ACAmPE (Ambiente Virtual Colaborativo para a Área de Petróleo) (Cadorin & Oliveira, 2004) trata-se de um AVC (Ambiente Virtual Colaborativo) para uso em treinamentos de técnicas de segurança em plataformas de petróleo, a fim de otimizar as ações de funcionários em situações críticas reais.

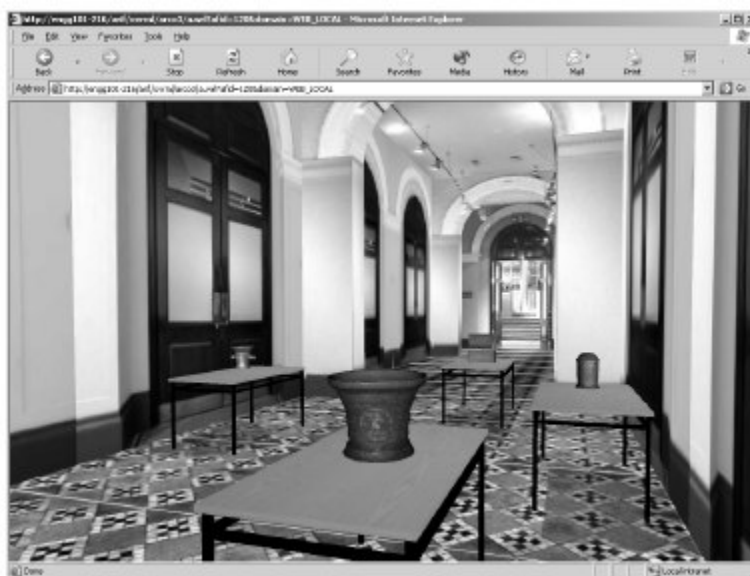


Figura 3-8: Exposição em museu virtual (Wojciechowski et al., 2004).

O ambiente virtual do ACAmPE é uma plataforma 3D. Cada funcionário, ao entrar no sistema, é representador por um avatar tridimensional, com o qual poderá interagir com o ambiente ou com outros funcionários presentes no mesmo espaço virtual. Desta forma, o usuário pode navegar pela plataforma virtual, visualizando os avatares dos outros funcionários, as máquinas e os demais objetos 3D presentes no ambiente.

A navegação do usuário pela plataforma para o seu reconhecimento é uma etapa importante no treinamento. Durante o treinamento, o funcionário deverá executar determinadas tarefas em situações críticas de acordo com a sua função, por exemplo: em uma situação de incêndio, alguns funcionários devem se dirigir para o local e agir para controlar o incêndio, enquanto que outros devem apenas encontrar o caminho para fugir.

O sistema usa realidade virtual semi-imersiva e foi implementado utilizando Java3D (Java 3D, 2004) e VRML (*Virtual Reality Modeling Language*) (VRML, 2004). O usuário se desloca pela plataforma 3D através do uso *mouse*. O diálogo entre os usuários se dá

através de mensagens de texto. A tela principal do ACAmPE é dividida entre o ambiente 3D e o painel de treinamento, que apresenta os objetivos do usuário e seu tempo de execução, conforme pode ser visualizado na Figura 3-9.



Figura 3-9: Tela do sistema ACAmPE (Cadorin & Oliveira, 2004).

O ACAmPE mostra-se como uma ferramenta interessante para o treinamento colaborativo, destacando-se alguns aspectos: a determinação de funções específicas por usuário; a utilização de tempo para controle dos objetivos do treinamento; e a divisão da interface utilizada.

3.2.5. Treinamento em Medicina

Existem inúmeras aplicações da realidade virtual na área médica e Ferro & Paiva (2003) apresentam um ambiente virtual semi-imersivo para o treinamento de técnicos em radiologia. O sistema é baseado em uma sala virtual que disponibiliza ao usuário a possibilidade de posicionar um avatar representando um paciente virtual de acordo com as prescrições de um exame radiográfico, além de calibrar a máquina de raio-x e de simular a radiografia obtida. Adicionalmente, o sistema pode ser usado como uma referência digital para os técnicos em radiologia, fornecendo a posição e os parâmetros da máquina corretos para um determinado exame. A modelagem do sistema foi feita utilizando a linguagem UML (*Unify Modeling Language*) (Rumbaugh et al., 1998). O sistema segue o modelo cliente-servidor, sendo seus módulos:

- Cliente, que é o responsável pela apresentação da simulação tridimensional, composta por um raio-x e um humanóide (Badler, 1997) em uma sala;
- Servidor, que recupera as informações no banco de dados dos exames.

O módulo cliente possui uma interface de controle que é responsável por capturar as interações do usuário, seja para navegar no ambiente ou operar o raio-x virtual. A Figura 3-10 apresenta uma tela do módulo cliente.

O sistema utiliza páginas HTML para apresentar a informação textual, como o resultado dos exames e recomendações necessárias. O sistema também simula uma radiografia, através de uma representação tridimensional de um crânio. Este sistema tem sido testado e até o momento tem apresentado resultados satisfatórios, correspondendo às expectativas.

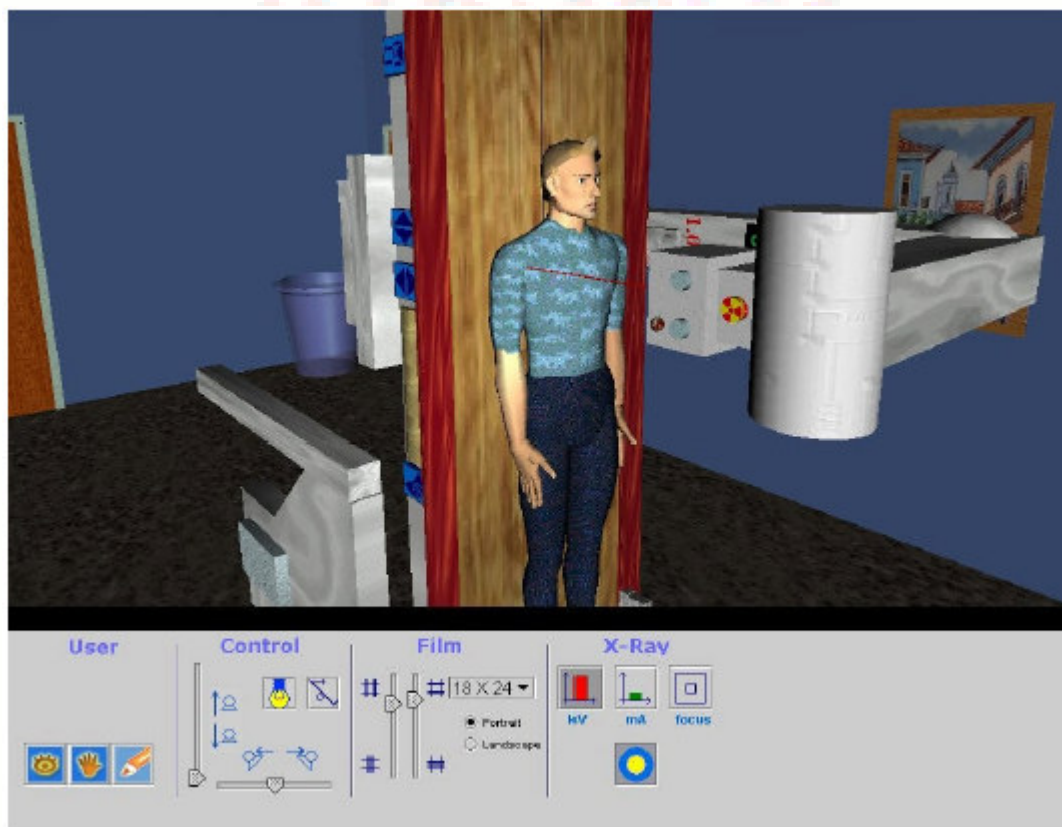


Figura 3-10: Tela do módulo cliente de treinamento (Ferro & Paiva, 2003).

3.2.6. Aprendizagem de Idiomas

Em Vidal et al. (2003) são descritos alguns resultados de pesquisas na utilização de ambientes virtuais cooperativos, via Internet, para o ensino de línguas estrangeiras. Um curso foi desenvolvido para treinar os guias de turismo utilizando ambientes virtuais que simulem suas práticas diárias. Os principais objetivos deste curso são:

- Imersão em situações comunicativas, interagindo com turistas estrangeiros em um ambiente tridimensional;
- Interação entre os alunos, permitindo o desenvolvimento das habilidades necessárias à comunicação;
- Interação entre o aluno e o computador em situações pré-estabelecidas.

O treinamento foi feito em grupo, onde os alunos alternavam-se nas condições de guias ou de turistas, tendo que realizar as tarefas específicas de seus papéis dentro do ambiente a cada turno. O curso foi desenvolvido utilizando a abordagem baseada em tarefas (Borges, 2001), para apresentar o conteúdo didático. O guia deveria realizar algumas tarefas dentro dos ambientes virtuais, tais como ajudar os turistas a comprar produtos de artesanato ou orientá-los sobre a localização de serviços. Para a simulação, foram modelados cenários tridimensionais representando pontos turísticos reais. A Figura 3-11 apresenta um ambiente virtual utilizado. A comunicação entre os usuários se dá através de bate-papo textual ou através de gestos dos personagens virtuais. Um outro recurso importante do sistema é o agente tradutor. Um personagem virtual controlado pelo computador que incorpora uma ferramenta de glossário, auxiliando os usuários na tradução de palavras.



Figura 3-11: Ambiente virtual modelado (Borges, 2001).

3.3. Requisitos

Existem diversas aplicações em realidade virtual semi-imersiva voltadas ao ensino e algumas características são importantes para fornecer um ambiente de aprendizagem eficiente, conforme apresentado nas seções anteriores. A seguir serão apresentados os principais requisitos identificados em mundos virtuais semi-imersivos para educação, baseados no exposto na seção anterior.

3.3.1. Interatividade

Muitos acreditam que a interatividade é a *raison d'être* de um mundo virtual (Roussou, 2004). No contexto da realidade virtual, a interatividade está associada à habilidade de o usuário escolher um caminho dentro do ambiente, podendo navegar livremente, e manipular os objetos tridimensionais. Para tanto, as aplicações em RV devem fornecer recursos de entrada de dados, como teclado, mouse e *joystick*, que possibilitem a navegação do usuário no mundo virtual. As interações devem fornecer ao usuário um retorno imediato (Shaw, 1993). Para possibilitar avaliações do aluno torna-se de fundamental importância o armazenamento das interações, como movimentos, erros e acertos.

3.3.2. Disponibilização de Objetos 3D

Um ambiente virtual pode ser composto por inúmeros elementos, tornando-se necessário fornecer ao usuário ações como seleção e manipulação de um ou mais objetos 3D. Entre essas ações, estão, por exemplo, ações para mover (translação) e girar (rotação), com as quais pode-se posicionar os objetos no mundo virtual, e a ação de escala, que permite alterar as proporções dos objetos em relação ao ambiente, entre outras (Burdea & Coiffet, 1994). Objetos que o usuário pode somente observar têm um nível de complexidade menor que aquele com o qual o usuário pode interagir, modificar, ajustar etc. Os objetos tridimensionais podem ser criados por ferramentas gráficas externas. Desta forma, a atividade de modelagem 3D pode ser desenvolvida em programas específicos para tal fim, aumentando a qualidade visual da aplicação. Também deve ser possível a utilização de objetos 3D simples, através de primitivas básicas, possibilitando a construção de aplicações mais “leves”. Objetos dinâmicos podem estar vinculados entre si, formando

agrupamentos hierárquicos (Vidal et al., 2004). No entanto, é possível configurar alterações que, mantendo a estrutura do agrupamento, afetem elementos individuais do grupo. Alguns objetos dinâmicos possuem comportamentos, aos quais são associadas animações. Há necessidade, portanto, de uma ferramenta de auxílio tanto para a definição dos comportamentos (animações) quanto para atribuí-los aos elementos.

3.3.3. Portabilidade e Flexibilidade da Aplicação

As aplicações de realidade virtual devem ser portáteis. Ou seja, deverão ter facilidade para execução em diversas plataformas, exigindo no máximo uma re-compilação do código. Já o ambiente de desenvolvimento deve ser flexível. Muitas vezes a aplicação é desenvolvida num ambiente e executada em outro. O sistema deve ter a flexibilidade para permitir a utilização de ambientes de desenvolvimento diferentes, bem como a execução de testes com outros dispositivos, com o mínimo de alteração do código (Góes, 2003).

3.3.4. Suporte a Linguagens de Script

Os sistemas que dispõem de personagens virtuais controlados por computador (simulóides, também chamados de *autonomous virtual humans*) são exemplos típicos de sistemas de RV que necessitam de manutenção. Nesses sistemas, é comum a necessidade de atribuir novos comportamentos aos simulóides ou de modificar comportamentos já implementados. Em alguns deles, tarefas dessa natureza requerem que alterações sejam feitas diretamente em partes do código do sistema. Quando o código fonte do sistema não é totalmente aberto, as possibilidades de modificação são nulas ou limitadas (por exemplo, a simples alteração de alguns parâmetros que determinam o comportamento do simulóide). Uma forma de superar as limitações de manutenção dos sistemas de RV é basear os serviços oferecidos na execução de scripts (Maia, 2003).

3.3.5. Realismo

O ambiente virtual deve manter um nível aceitável de fidelidade entre o mundo virtual e o mundo real. E, para tal fim, Styztz (1996) definiu alguns requisitos:

- O ambiente virtual deve ser o mais realista possível;

- Os objetos 3D devem apresentar proporções corretas entre si e ser movimentados em velocidades dentro da realidade;
- O comportamento de possíveis personagens controlados por computadores deve simular as ações desempenhadas por personagens controlados por humanos face às mesmas circunstâncias.

Dainese (2003) propõe que um ambiente virtual fiel à realidade deva fornecer ao usuário as seguintes possibilidades:

- Perceber uma imensa quantidade de dados;
- Utilizar uma vasta quantidade de conhecimento.

3.3.6. Manipulação do Conteúdo

A aplicação de RV deve possuir uma forma de consulta e distribuição dos conceitos relacionados ao conteúdo a ser desenvolvido (utilizado na simulação). Bowman et al. (2003) define este conteúdo como: Informação Abstrata. As formas de apresentação dessas informações podem variar de acordo com a aplicação, por exemplo: embutida no próprio ambiente virtual como um objeto 3D (Figura 3-12); ou estática, fixa na mesma tela do mundo tridimensional ou em outra tela (Figura 3-13).

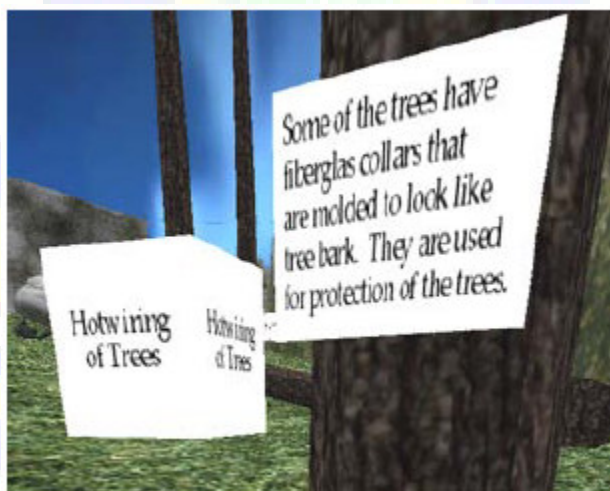


Figura 3-12: Apresentação de conteúdo dentro do mundo 3D (Bowman et al., 2003).

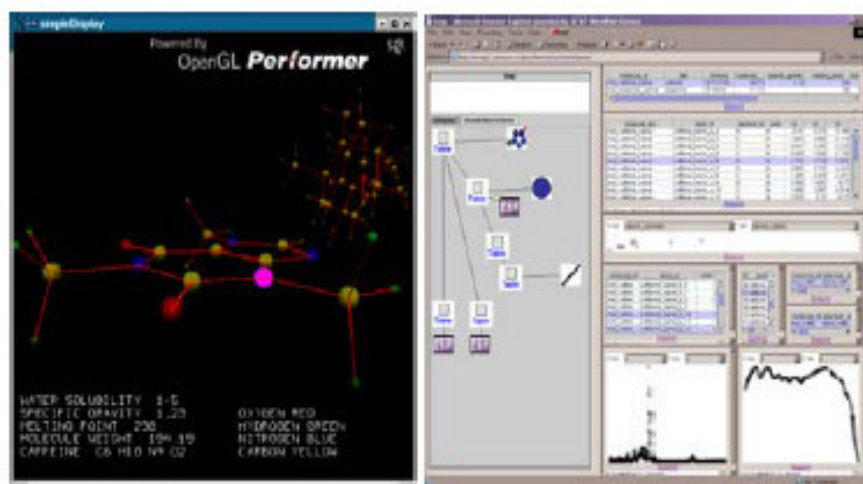


Figura 3-13: Apresentação de conteúdo de forma estática (Bowman et al., 2003).

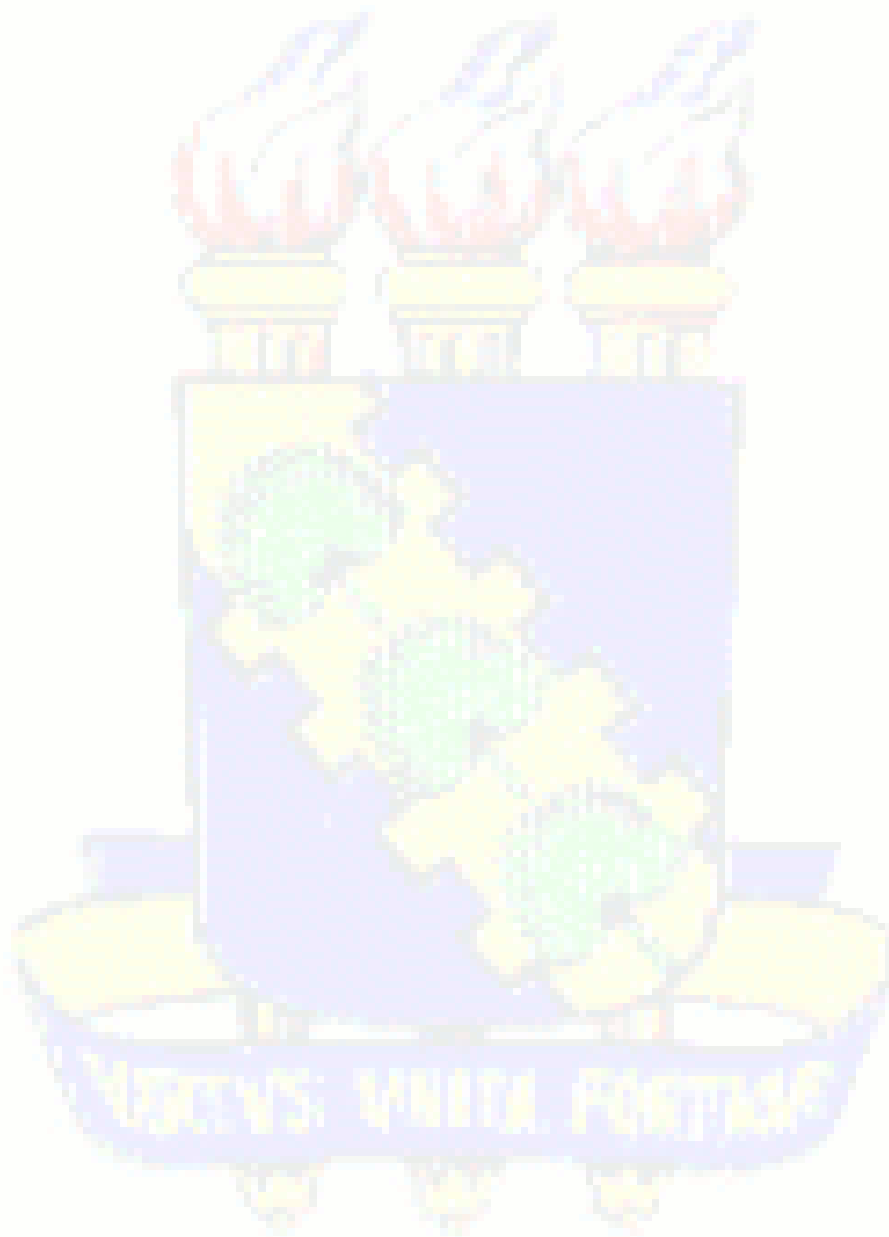
3.4. Considerações Finais

Devido à sua interdisciplinaridade, as aplicações desenvolvidas para realidade virtual são vastas, desde jogos até arquitetura, passando pela educação, medicina, engenharia, treinamento militar, etc. Além disso, é importante ressaltar que na utilização de certas aplicações há perspectivas vantajosas que não são possíveis no mundo real, como, por exemplo, interagir com moléculas de DNA e simular viagens a planetas desconhecidos.

Por utilizar-se de recursos não disponíveis nas demais mídias educacionais, tradicionalmente encontradas nos sistemas multimídias (texto, som e imagem), a realidade virtual tem se destacado por apresentar aspectos positivos para o seu uso na educação. Ela possibilita a criação de novos modelos de ferramentas educacionais, incluindo-se também instrumentos para o ensino a distância, devido à sua capacidade ímpar de permitir ao aluno que explore determinadas situações. Assim, os alunos, motivados pela interatividade fornecida pelo mundo virtual, passam a dar mais atenção na realização de uma tarefa, segundo os experimentos mencionados anteriormente. Além disso, por se valerem de respostas a estímulos físicos sensoriais, os ambientes de aprendizado construídos a partir de recursos dessa tecnologia podem proporcionar ao estudante um rápido desenvolvimento do sentimento de domínio do conteúdo (Leite Júnior, 2000).

Este capítulo apresentou ainda importantes requisitos encontrados em aplicações educacionais que fazem uso da realidade virtual. No próximo capítulo, estes requisitos,

juntamente com os identificados no Capítulo 2, fundamentarão a análise de domínio do framework proposto nesta dissertação.



CAPÍTULO 4

O Framework VirTraM

Este capítulo apresenta o VirTraM (*Virtual Training for Mobile Devices*), um *framework* para a construção de aplicações voltadas ao treinamento utilizando dispositivos móveis com recursos de realidade virtual semi-imersiva. Primeiramente, na Seção 4.1, é apresentada uma introdução sobre o conceito de *frameworks* orientados a objetos. Na Seção 4.2 é descrita a metodologia utilizada para a construção do VirTraM. A Seção 4.3 descreve o VirTraM, incluindo os objetivos, os requisitos e a arquitetura. Finalmente, a Seção 4.4 apresenta as considerações finais deste capítulo.

4.1. Introdução

Um *framework* orientado a objetos é um modelo reutilizável representado por um conjunto de classes que colaboram entre si e constituem a arquitetura principal de uma aplicação, ou sub-sistema (Matsson, 1996). A idéia principal de um *framework* é permitir que um conjunto de recursos comuns seja reaproveitado para cada novo software criado.

O primeiro *framework* largamente utilizado foi o Smalltalk-80 (Sandberg, 1988), chamado de *Model-View-Controller* (MVC) (Matsson, 1996). Nesse *framework*, o modelo (model) consiste da lógica e dos dados da aplicação; a visualização (view) é responsável por mostrar a informação ao usuário; e o controlador (controller) manipula a interação com o usuário e a mapeia para as classes do framework.

Vale ainda ressaltar que um *framework* determina a arquitetura da aplicação e predefine parâmetros de projeto, permitindo ao projetista concentrar-se nos detalhes específicos da aplicação tais como as regras de negócio. Para isso, um *framework* tem uma representação física em termos de classes e métodos onde não apenas a implementação é reutilizável, como também a sua estrutura. As partes extensíveis e especializáveis são denominadas *hot spots* (Pree, 1999), os quais tornam o *framework* flexível.

Os principais benefícios da utilização de *frameworks* orientados a objetos são a

arquitetura modular, a reutilização de código, a extensibilidade e a inversão de controle que eles fornecem aos desenvolvedores (Fayad et al., 1999).

Um conceito importante relacionado a *frameworks*, que vale a pena mencionar, é o de padrão de projeto, que Gamma et al. (1995) definiram da seguinte forma: “Um padrão de projeto nomeia, abstrai e identifica os aspectos-chaves de uma estrutura de projeto comum, tornando-se útil para criar um projeto orientado a objetos reutilizável. O padrão de projeto identifica as classes participantes e instâncias, seus papéis, suas colaborações e a distribuição de responsabilidades”.

Freqüentemente *frameworks* e padrões são confundidos, porém Gamma et al. (1995) apresentam as seguintes diferenças entre eles:

- Padrões de projetos são elementos arquiteturais que podem estar contidos em *frameworks*, mas o inverso não é possível.
- Padrões de projeto são menos especializados. *Frameworks* sempre pertencem a um domínio em particular, enquanto que os padrões poderiam, em princípio, ser usados em qualquer tipo de aplicação.
- Padrões de projeto são mais abstratos, apenas exemplos podem ser incorporados em código. Por outro lado, os *frameworks* orientados a objetos podem ser diretamente codificados.

Durante o desenvolvimento do VirTraM foram utilizados diferentes padrões de projeto.

4.2. Metodologia de Desenvolvimento Utilizada

O processo de desenvolvimento e utilização de um *framework* orientado a objetos compreende, de forma genérica, as seguintes macro-atividades (Markiewicz & Lucena, 2000):

- Análise de domínio, cujo papel é identificar o conhecimento sobre uma determinada classe de problemas;
- Projeto de *framework*, cuja função é especificar as partes estáveis e variáveis do *framework*;

- Instanciação do *framework*, que visa implementar a aplicação utilizando as classes do *framework*.

Existem diversas metodologias para o desenvolvimento de *frameworks*. Pree (1999), Schmid (1999) e Johnson (1993) iniciam com um modelo de aplicação já pronto, que consiste de um conjunto de classes de uma aplicação pertencente ao domínio, e incluem a flexibilidade desejada mais tarde. Em Bosh et al. (1999) a abordagem inicia com um modelo da análise de domínio, o que torna a detecção dos *hot spots* mais previsível. Pree (1999) e Schmid (1999) recomendam o uso de padrões de projeto. Uma característica comum encontrada em todos esses trabalhos é a importância ressaltada para a documentação do *framework* e sua utilização.

Esta dissertação propõe o desenvolvimento de um *framework* utilizando a abordagem de Bosh (1999) pelos seguintes motivos:

- A ênfase encontra-se na análise do domínio, através da captura dos requisitos e da identificação dos conceitos relacionados ao escopo do problema;
- As etapas apresentam-se claramente divididas em análise, projeto e implementação.
- A documentação ocupa um papel importante no processo de construção do *framework* VirTraM.

As etapas propostas por Bosh são:

1ª) Análise de domínio para descrever o assunto coberto pelo *framework*, capturar os requisitos e os principais conceitos;

2ª) Projeto arquitetural do *framework*, que tem como entrada o levantamento da etapa anterior;

3ª) Refinamento do projeto do *framework* e adição de novas classes;

4ª) Implementação das classes do *framework*;

5ª) Teste do *framework* através do desenvolvimento de aplicações testes;

6ª) Documentação do *framework*, descrevendo com mais detalhes a forma de utilização do *framework*.

4.3. Análise de Domínio

A análise de domínio fornece a entrada para o desenvolvimento do *framework*. Trata-se de um processo através do qual a informação a ser usada no desenvolvimento do sistema é identificada, capturada e organizada. Pode ser vista como uma análise ampla, onde são definidos os objetivos, benefícios, limites e requisitos do *framework*.

4.3.1. Objetivos, Benefícios e Limites

O principal objetivo do VirTraM é disponibilizar ao desenvolvedor uma estrutura flexível de classes com a capacidade de adaptar-se a um conjunto de diferentes aplicações voltadas ao treinamento, utilizando-se mundos tridimensionais em dispositivos móveis. Ao integrar recursos de computação móvel e realidade virtual semi-imersiva, o VirTraM apresenta-se como uma alternativa inovadora para o desenvolvimento de aplicações de ensino. Optou-se pela utilização da realidade virtual semi-imersiva devido às limitações de hardware dos dispositivos móveis, principalmente considerando-se os telefones celulares. Desta forma, os ambientes virtuais construídos a partir do VirTraM constituem-se basicamente de cenários tridimensionais interativos sem a necessidade de acessórios adicionais.

A Figura 4-1 ilustra uma representação do domínio do VirTraM. Este pode ser observado como uma interseção entre as tecnologias de computação móvel e realidade virtual, ao implementar parte de seus recursos, conforme será evidenciado ao longo deste capítulo.

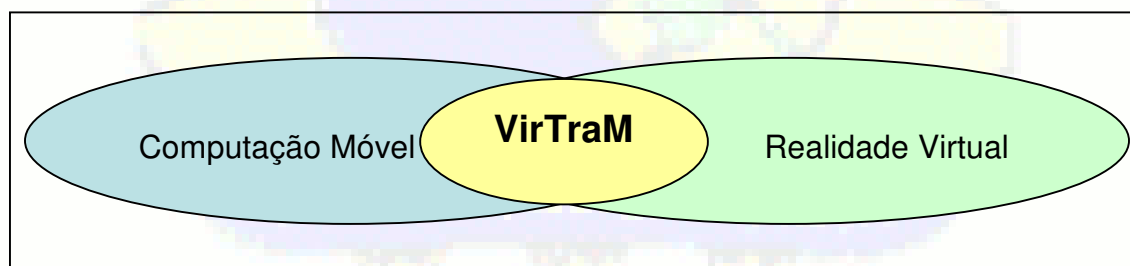


Figura 4-1: Interseção das tecnologias do VirTraM.

Os principais benefícios da utilização do VirTraM são destacados a seguir.

Reutilização de código. Através do reuso das classes do *framework*, o desenvolvedor pode se concentrar nos aspectos da aplicação educacional, abstraindo-se dos detalhes de

implementação. Evita-se também a recriação e revalidação de rotinas comuns a aplicações no domínio do VirTraM, diminuindo a quantidade de linhas de código e o tempo de testes.

Produtividade. O reuso da arquitetura do VirTraM possibilita uma economia de tempo na construção de treinamentos para dispositivos móveis, ao eliminar a necessidade da implementação de funcionalidades já prontas. A manutenção das aplicações também é otimizada, tendo em vista que as correções e melhorias no *framework* serão propagadas para as aplicações.

Extensibilidade. Além do VirTraM possibilitar o desenvolvimento de aplicações com características padronizadas, também é possível a implementação de outros recursos, com o objetivo de permitir a construção de aplicações com novos requisitos e estender as possibilidades do *framework*.

Para possibilitar a conclusão da implementação do VirTraM e manter o foco centrado no desenvolvimento de aplicações com recursos tridimensionais em dispositivos móveis, alguns limites foram estabelecidos ao *framework*.

O VirTraM limita-se à construção de aplicações que serão executadas localmente nos dispositivos móveis, não fornecendo classes de transmissão remota de dados. Isto, porém, não impede a aprendizagem a distância, visto que as aplicações construídas podem ser instaladas remotamente. E também é possível o desenvolvimento de funcionalidades para transmissão de dados, porém estas funcionalidades não estão previstas nas classes do *framework*. Neste caso, utilizam-se diretamente os recursos da linguagem de programação. Esta limitação deve-se à complexidade envolvida na implementação de recursos de comunicação e por não ser o foco desta dissertação.

Outra limitação, relacionada à restrição de memória do dispositivo, é a impossibilidade de manipulação de modelos tridimensionais relativamente complexos. Devido ao tamanho elevado dos arquivos dos modelos 3D, estes comprometem o desempenho da aplicação e, em muitos casos, podem ultrapassar os recursos do dispositivo móvel.

Uma última limitação, decorrente da utilização de recursos tridimensionais, é a execução apenas em equipamentos com biblioteca gráfica 3D. Isto se deve ao fato de o

VirTraM não implementar funções tridimensionais, tratando-se de uma camada entre o motor 3D do dispositivo móvel e a aplicação educacional.

4.3.2. Requisitos Não Funcionais

Os Capítulos 2 e 3 apresentaram requisitos desejáveis respectivamente a aplicações educacionais em dispositivos móveis e em aplicações de realidade virtual na aprendizagem. O VirTraM busca atender um conjunto desses requisitos, possibilitando a construção de aplicações que integrem as melhores características de ambas as tecnologias.

O treinamento deve poder ser realizado em qualquer lugar e a qualquer momento, e de uma forma ininterrupta. Para tanto, as aplicações devem ser instaladas com todos os recursos no dispositivo móvel. Desta forma, não existe dependência de conexão remota para o acesso ao conteúdo ou a outros componentes da aplicação. Para a instalação desta, o usuário pode utilizar as diferentes tecnologias para transmissão dos dados existentes nos dispositivos móveis, tais como: cabo USB, infravermelho, *bluetooth*, *Wi-Fi* e internet.

Inerente tanto a aplicações de *Mobile Learning* quanto a de Realidade Virtual, a motivação do usuário é uma das principais características existentes nas aplicações baseadas no VirTraM. Além do ambiente tridimensional, o treinamento poderá fazer uso de imagens e sons para estimular a aprendizagem do usuário.

A interatividade é outra característica importante presente nos treinamentos desenvolvidos, nos quais o aluno tem participação ativa. E a navegação no ambiente virtual destaca-se como a forma de interação principal, que poderá ser realizada através da utilização das teclas do dispositivo móvel ou de dispositivo apontador.

Uma importante característica da realidade virtual e também presente nos treinamentos baseados no VirTraM é a possibilidade de simulação de situações do mundo real. O sentimento de navegação em um ambiente virtual incentiva o aluno a participar ativamente, a ter autonomia, a desenvolver imaginação e a encontrar soluções (Roussou, 2004). O ato de explorar as alternativas, descobrir por si só, faz com que o aprendiz se sinta parte daquele mundo virtual. A aplicação permite um aprendizado baseado em experiências, onde o aluno pode controlar ativamente o ambiente e observar diretamente o comportamento dos objetos virtuais. Por exemplo, um técnico de manutenção pode ver as

consequências da troca de uma peça, sem necessariamente ter que arriscar danificar o equipamento.

A aplicação baseada no VirTraM pode ser executada em diferentes plataformas, sendo compatível tanto com telefones celulares quanto com PDAs e suportando diferentes sistemas operacionais. Esta portabilidade é uma característica presente tanto em aplicações de computação móvel como de realidade virtual (Luchini et al., 2004; Marson et al., 2003).

Outro requisito importante, presente nas aplicações baseadas no VirTraM, é a facilidade de uso. Segundo Avellis et al. (2003), a aplicação deve ser simples e fácil de usar, pois o usuário não tem como atividade fim a utilização da aplicação móvel, e sim obter através dela uma determinada finalidade. Isso é importante principalmente quando é levada em consideração a complexidade de aplicações com cenários tridimensionais. Sugere-se o emprego das seguintes técnicas para possibilitar uma maior facilidade de uso para a aplicação:

- Fornecer informações formatadas, para diminuir a entrada de texto pelo usuário;
- Evitar muitos níveis de profundidade nas telas da aplicação (uma tela que chama outra tela que por sua vez chama outra tela e assim por diante);
- Utilizar imagens para diminuir a quantidade de texto.

4.3.3. Requisitos Funcionais

O VirTraM pode ser representado como uma camada entre a aplicação e a linguagem de programação, que na implementação apresentada nesta dissertação é o J2ME. A Figura 4-2 ilustra uma forma de representação do VirTraM.

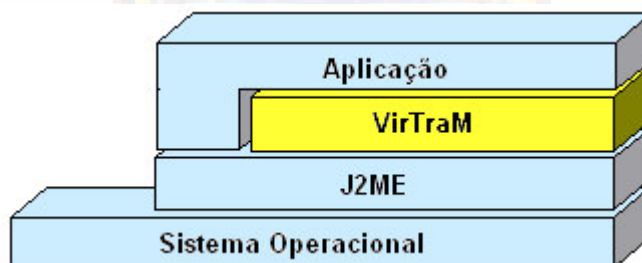


Figura 4-2: Representação do VirTraM.

O *framework* é composto por classes voltadas para o desenvolvimento de aplicações educacionais com recursos 3D. Para a implementação das funcionalidades não cobertas pelo VirTraM, o desenvolvedor deve utilizar diretamente os recursos da linguagem de programação ou estender os recursos do *framework*.

Para o desenvolvimento de aplicações educacionais que satisfaçam as características requeridas na seção anterior e nos capítulos 2 e 3, o *framework* deve atender a determinados requisitos funcionais:

Criar Interfaces Visuais. Consiste em definir como são as telas da aplicação que apresentam o conteúdo ao usuário.

Criar Ambiente 3D. Consiste em determinar qual é o ambiente 3D utilizado no treinamento.

Criar Base de Conteúdo. Consiste em definir a base com as informações que são utilizadas na aplicação.

Criar Formas de Interação. Consiste em determinar as opções de que o usuário pode dispor para interagir com a aplicação.

Criar Recursos Opcionais. Consiste em definir quais recursos, além das telas de conteúdo e do ambiente 3D, o usuário pode utilizar para auxiliar o treinamento, tais como: som, gerenciamento da memória e registro das ações do usuário.

4.4. Projeto do *Framework*

A partir da análise de domínio realizada na Seção 4.4, foram implementados oito pacotes que compõem a arquitetura do VirTraM e contêm as classes necessárias para o desenvolvimento das aplicações, representados na Figura 4-3.

Para facilitar uma melhor compreensão do VirTraM, em função da sua arquitetura modular e da quantidade de classes, a descrição do projeto do *framework* foi dividida de acordo com seus pacotes. As próximas seções descrevem os detalhes de cada pacote, apresentando seus objetivos, casos de uso, principais problemas e soluções utilizadas. Utilizou-se a UML - *Unified Model Language* (Rumbaugh et al., 1998) como notação para os diagramas que serão apresentados.

A descrição dos pacotes é mostrada nas próximas seções na seguinte sequência: *Interaction*, *Persistence*, *Optional*, *Log*, *Visual*, *Content*, *3D* e *HiLevel*. A Seção 4.5.9 resume em uma tabela todas as classes e seus respectivos pacotes.

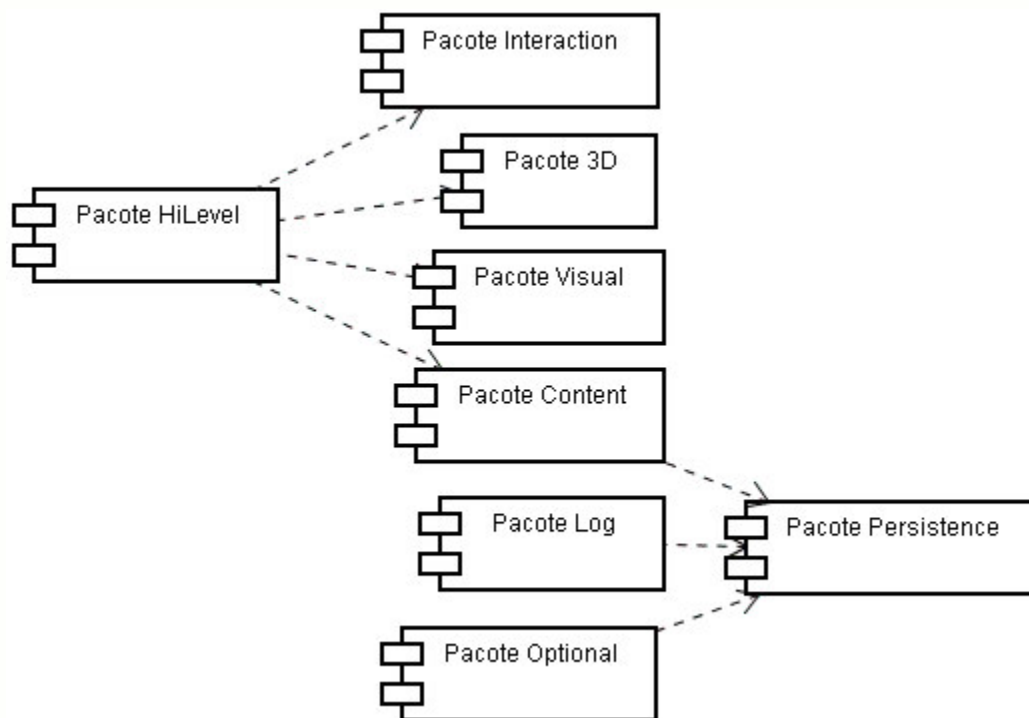


Figura 4-3: Pacotes de classes do VirTraM.

4.4.1. Pacote *Interaction*

Este pacote trata as formas de interação do usuário com a aplicação, através dos meios de entrada do dispositivo móvel. As seguintes definições são feitas relacionadas com as formas de interação proporcionadas pelo VirTraM:

Operação 3D. Função gráfica interativa dentro do ambiente 3D. No caso das aplicações desenvolvidas com o VirTraM, as operações 3D são: rotação e translação.

Elemento Interativo. Componente do treinamento que sofre alteração com a interação do usuário. O VirTraM possui três elementos: o modelo 3D, a câmera e o apontador (*pointer*). Este último trata-se de um círculo sobre o ambiente 3D que será utilizado para selecionar um modelo 3D dentro do ambiente.

Painel de Interação. Componente visual no ambiente 3D que auxiliará o usuário na navegação no ambiente tridimensional, indicando qual o elemento interativo selecionado,

sua posição e qual a operação 3D atual. O framework apresenta três opções de painéis diferentes e possibilita ao desenvolvedor estender e customizar o seu próprio.

O Pacote *Interaction* apresenta os seguintes casos de uso: determinar quais teclas do dispositivo móvel executa as operações 3D; indicar qual a operação 3D selecionada (rotação ou translação); definir a operação que é executada em caso de interação com o dispositivo apontador; determinar o elemento interativo; e selecionar o modelo 3D para manipular.

Este pacote é composto por uma classe principal de interação, a *VTM_Interaction*; uma classe para definir as teclas no dispositivo móvel para interação do usuário, a *VTM_KeyControls*; e uma classe responsável pelo painel de interação, a *VTM_VisualPanel*. Esta última possui três subclasses para apresentar o painel em diferentes tipos de dispositivos móveis, respectivamente para *palmtop*, *pocket pc* e *celular*: *VTM_PanelPalm*, *VTM_PanelPocket* e *VTM_PanelCell*.

Em qualquer parte do código, o desenvolvedor deve ter acesso à classe *VTM_Interaction*; e esta deve possuir apenas uma única instância. Para solucionar este problema é utilizado o padrão de projeto **Singleton** (Hannemann & Kiczales, 2002). A classe de interação também deve tratar os eventos gerados a partir de algum dispositivo de entrada e informar outras classes sobre as interações. Para realizar esse tratamento, implementa-se o padrão **Observer** (Hannemann & Kiczales, 2002), que notifica os objetos relacionados em caso de mudança de estado. A Figura 4-4 apresenta o diagrama de classes do Pacote *Interaction*.

4.4.2. Pacote *Persistence*

Este pacote fornece ao desenvolvedor um conjunto de classes para a persistência dos dados no dispositivo móvel, com operações básicas de inserção, atualização, exclusão e busca dos dados. Assume papel fundamental no VirTraM pelo fato de suas classes serem utilizadas para persistir a informação necessária para outros pacotes, como por exemplo: armazenar o conteúdo do treinamento ou a ação do usuário.

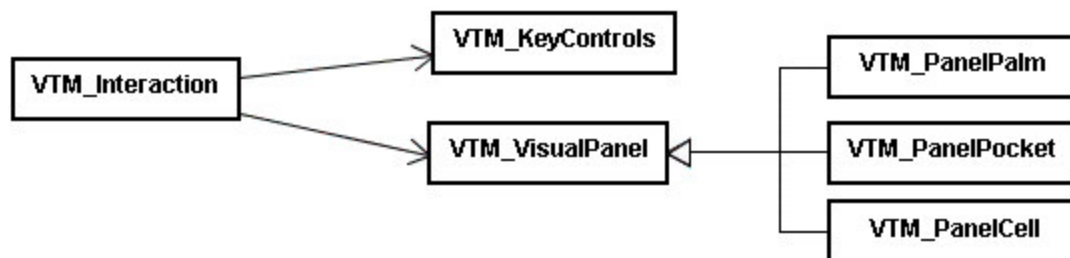


Figura 4-4: Diagrama de classes do pacote *Interaction*.

Algumas definições são feitas com relação à persistência em dispositivos móveis (Magalhães et al., 2004) através do VirTraM.

Registro. Uma estrutura equivalente ao conceito de registro na área de Banco de Dados.

Base de Dados. Trata-se de um conjunto de Registros, sendo uma estrutura equivalente à tabela na área de Banco de Dados. Uma aplicação pode conter várias Bases de Dados.

O Pacote *Persistence* apresenta os seguintes casos de uso: criar base de dados; apagar base de dados; incluir registro; excluir registro; consultar registro; modificar registro; e, navegar pelos registros. Assim, não se faz necessária a utilização de aplicações de banco de dados para armazenamento do conteúdo do treinamento.

Este pacote é composto por duas classes principais: uma classe que é responsável pela execução das funcionalidades de persistência do VirTraM, a *VTM_RecordBase*; e uma classe auxiliar para tratar a forma na qual os Registros se tornarão persistentes no dispositivo móvel, a *VTM_ObjectPersist*. O pacote ainda contém outras três classes que herdam de *VTM_RecordBase* (*VTM_BaseContent*, *VTM_BaseLog* e *VTM_BaseConfig*), que são especializadas para serem utilizadas em outros pacotes do VirTraM. A Figura 4-5 apresenta o Diagrama de Classes desse pacote.

4.4.3. Pacote *Optional*

Este pacote contém classes relacionadas a recursos complementares e não essenciais aos treinamentos. Estas classes tratam de características mais específicas dos dispositivos, podendo, assim, não estar disponíveis em determinados equipamentos.

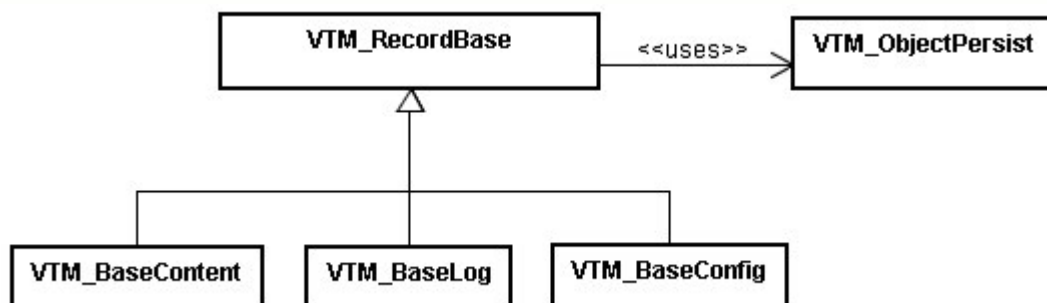


Figura 4-5: Diagrama de classes do pacote *Persistence*.

O Pacote *Optional* apresenta os seguintes casos de uso: consultar memória livre do dispositivo; consultar o nível de bateria atual; obter a velocidade do processador; e obter o tamanho da tela. Este pacote contém também as classes que são responsáveis por permitir a utilização de recursos de som, através dos seguintes casos de uso: consultar suporte a som e tocar som. Ainda é possível vibrar o dispositivo, caso ele suporte esta funcionalidade.

Este pacote é composto pelas seguintes classes: *VTM_Monitor*, que realiza as consultas aos recursos do sistema; *VTM_Sound* e *VTM_CollectionSound*, responsáveis pelos recursos de som do VirTraM; e *VTM_Vibration*, responsável por vibrar o dispositivo. A Figura 4-6 apresenta o diagrama de classes desse pacote.

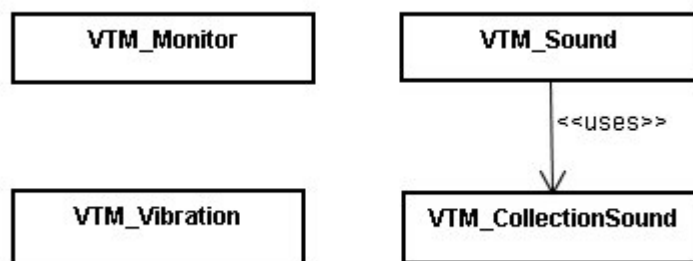


Figura 4-6: Diagrama de classes do pacote *Optional*.

4.4.4. Pacote *Log*

Este pacote tem como principal função fornecer ao desenvolvedor recursos que possibilitem uma avaliação do desempenho do usuário no treinamento, através do registro de suas ações.

Através do Pacote *Log*, é possível registrar o início e o fim das ações do usuário no treinamento, registrar o ponto no ambiente 3D onde acontece a execução de determinada

ação e consultar os tempos e os locais, no cenário 3D, das execuções das ações. Os casos de uso desse pacote são: registrar início da ação; registrar fim da ação; registrar ponto de execução; e retornar tempos de execução; retornar pontos de execução. O programador deve determinar, de acordo com as características do treinamento, em quais momentos as informações devem ser armazenadas.

Este pacote contém duas classes que herdam diretamente das classes do Pacote *Persistence*: *VTM_Log*, responsável pelas funcionalidades do pacote; e *VTM_ObjectLog*, classe auxiliar para persistência das informações. A Figura 4-7 apresenta o diagrama de classes do Pacote *Log*.

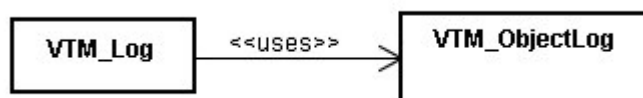


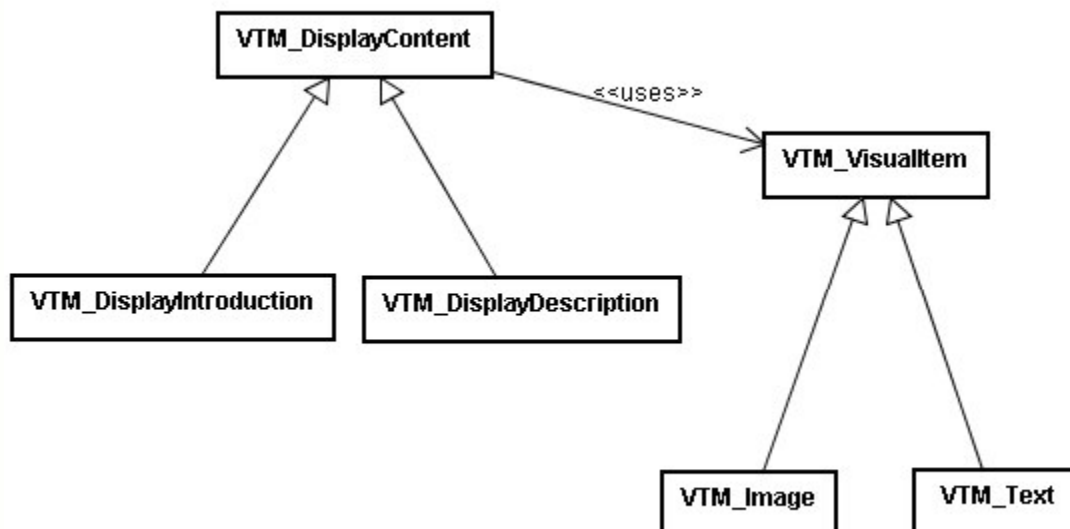
Figura 4-7: Diagrama de Classes do Pacote *Log*.

4.4.5. Pacote *Visual*

Este pacote contém classes responsáveis por criar a interface com o usuário, exceto o ambiente 3D. Estas incluem as telas padrões para apresentação do conteúdo dos treinamentos baseados no VirTraM.

A principal função deste pacote é dispor, de forma estruturada, o conteúdo do treinamento no visor do dispositivo móvel. Desta forma, o desenvolvedor não necessita montar componente a componente (imagens e textos) as telas de introdução ao treinamento ou de descrição do conteúdo, obtendo-se produtividade e padronização na construção das aplicações. Caso o desenvolvedor não queira utilizar as telas padrões, ele pode estender as classes do pacote e utilizá-las no *framework*, normalmente. Os casos de uso desse pacote são: apresentar a introdução do treinamento; e apresentar a tela de conteúdo.

Este pacote contém duas classes principais que constituem as interfaces para o usuário, *VTM_DisplayIntroduction* e *VTM_DisplayDescription*. Estas herdam de *VTM_DisplayContent*. Para compor as telas são utilizadas as classes *VTM_Image* e *VTM_Text* que herdam de *VTM_VisualItem*. A Figura 4-8 apresenta o diagrama de classes do Pacote *Visual*.

Figura 4-8: Diagrama de Classes do Pacote *Visual*.

4.4.6. Pacote *Content*

O Pacote *Content* contém as classes responsáveis por obter, armazenar e disponibilizar o conteúdo do treinamento. Ele define a forma como o material descritivo do treinamento é associado à aplicação, mostrando como o desenvolvedor tem acesso ao conteúdo para poder apresentá-lo. Este pacote contém classes que herdam do Pacote *Persistence*.

A principal função deste pacote é fornecer ao desenvolvedor as classes de acesso às informações do treinamento. Os casos de uso desse pacote são: conectar a base de dados; ler arquivo de conteúdo; processar arquivo de conteúdo; persistir informações do arquivo de conteúdo; apagar arquivo de conteúdo; consultar dados da base; e gravar dados na base.

O conteúdo para ser utilizado nas aplicações baseadas no VirTraM deve estar armazenado no dispositivo de forma persistente. Para isso, os dados são obtidos através da importação de um arquivo XML, que contém os detalhes do treinamento e, posteriormente, armazenados de forma persistente. A utilização deste arquivo XML estabelece uma forma fácil e estruturada para geração do conteúdo da aplicação.

Os principais benefícios decorrentes da utilização de arquivos XML são as possibilidades de o usuário definir novos elementos e de poder alterar o arquivo

diretamente (sem necessitar de ferramentas de edição), tendo em vista que se trata de um arquivo texto. Após ser importado, o arquivo é apagado (removido do dispositivo móvel) e as informações são acessadas utilizando-se os métodos de manipulação de dados persistentes, herdados do Pacote *Persistence*. A Figura 4-9 apresenta um exemplo de arquivo XML de conteúdo para um treinamento baseado no VirTraM.

```
- <Model3D>
- <parts>
- <part>
  <id>82345</id>
  <title>Power</title>
  <image>poweron.png</image>
  <description>Botao de power com a camera
    ligada.</description>
  <image>poweroff.png</image>
  <description>Botao de power com a camera
    desligada.</description>
</part>
</parts>
- <animations>
- <animation>
  <id>99987</id>
  <title>Troca da Bateria</title>
  <image>trocab.png</image>
  <description>Procedimento da troca da bateria. A maquina
    deve estar desligada e a tampa deve ser levantada para
    cima, conforme animacao.</description>
</animation>
</animations>
</Model3D>
```

Figura 4-9: Exemplo de arquivo de conteúdo do VirTraM.

O arquivo da Figura 4-9 descreve as informações de componentes de uma câmera fotográfica digital. O arquivo de conteúdo está diretamente associado ao ambiente 3D e o conteúdo e o objeto tridimensional são associados através do código identificador (id).

O processo completo de tratamento do conteúdo do VirTraM é feito da seguinte forma: inicialmente o desenvolvedor tenta conectar na base de dados do treinamento; caso a base não esteja criada, o arquivo XML deve ser processado e suas informações armazenadas no dispositivo móvel; uma vez tornados persistentes os dados, o

desenvolvedor pode consultar ou gravar informações. A Figura 4-10 mostra este processo, através de um diagrama de atividades UML.

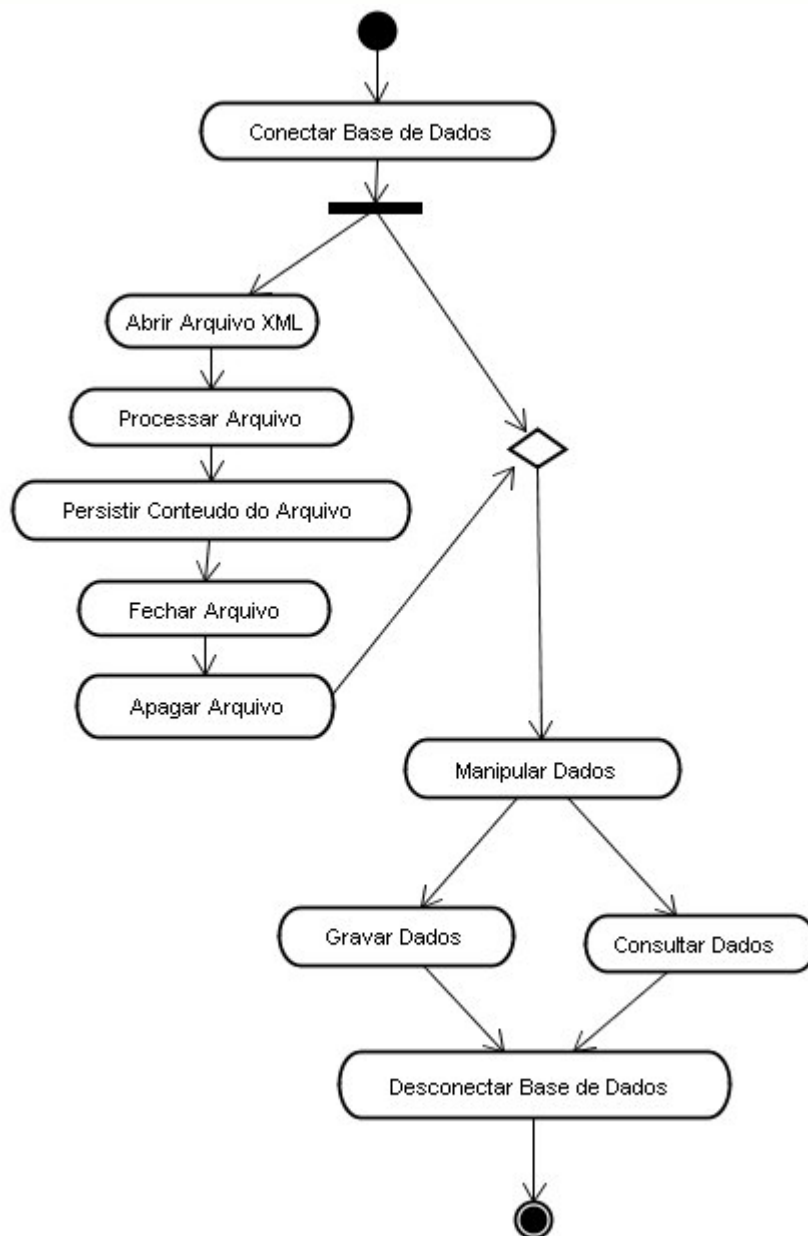


Figura 4-10: Diagrama de atividades com os processos de manipulação dos dados.

Este pacote contém uma classe principal responsável pelo conteúdo, *VTM_ContentTraining*. O treinamento deve possuir uma única instância dessa classe e deve poder usá-la em qualquer parte do código. Para isso, implementa-se o padrão *Singleton*. A *VTM_ContentTraining* utiliza as outras classes do pacote para realizar as operações de

tratamento do arquivo de conteúdo e persistência dos dados. A classe *VTM_ContentFile* tem a função de abrir, ler, fechar e apagar arquivos texto. As classes *VTM_ParserShow* e *VTM_ParserQuestion*, herdadas de *VTM_ContentParser*, são as responsáveis por processar os dados do arquivo XML, retornando objetos formatados. As classes *VTM_ContentExibithion* e *VTM_ContentQuestion*, herdadas de *VTM_Content*, definem o formato no qual os dados serão armazenados. A Figura 4-11 apresenta o diagrama de classes do Pacote *Content*.

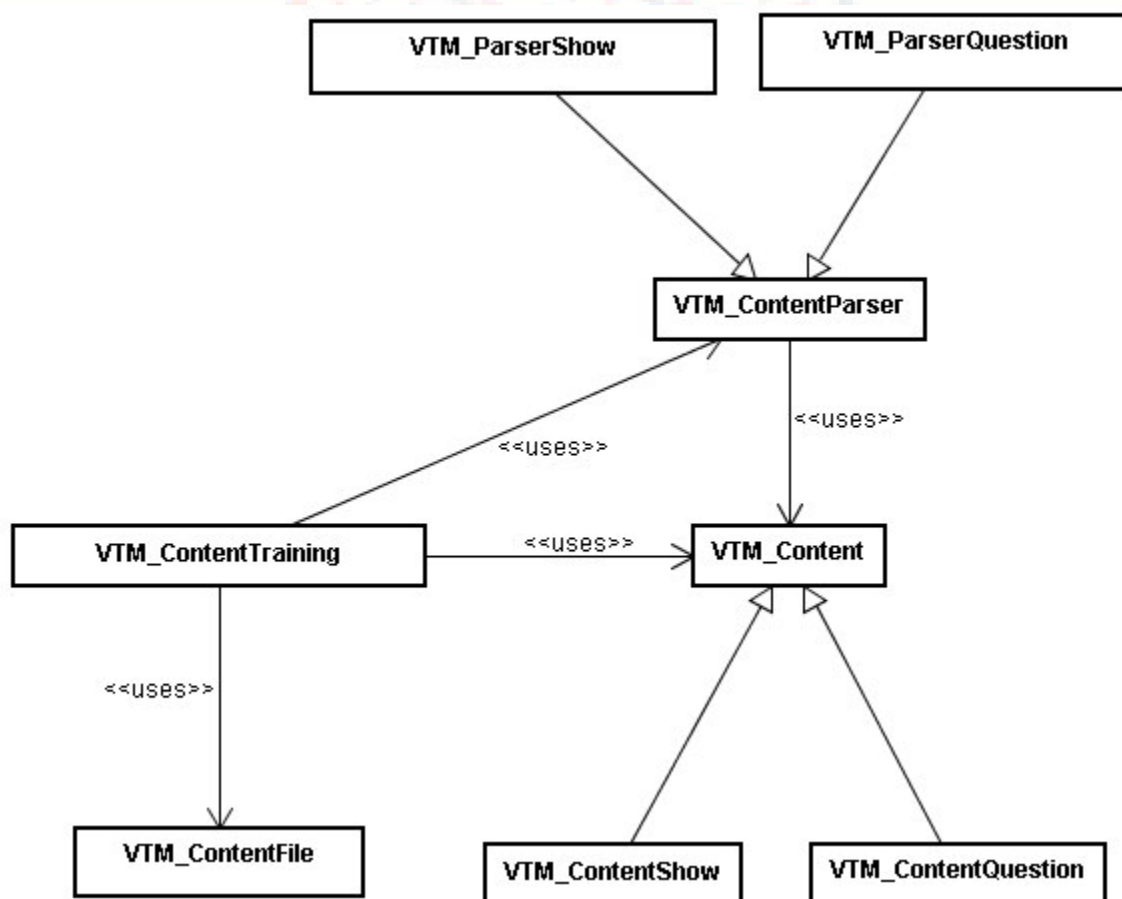


Figura 4-11: Diagrama de Classes do Pacote *Content*.

4.4.7. Pacote 3D

O Pacote *3D* contém as classes responsáveis pela apresentação gráfica do ambiente tridimensional. A principal função desse pacote é fornecer ao desenvolvedor uma camada para abstrair as funções gráficas da API 3D do dispositivo, oferecendo um conjunto de

classes mais específico e voltado ao treinamento. Apesar de constituir um pacote do VirTraM, o Pacote 3D não depende de nenhum outro pacote, podendo ser utilizado individualmente para o desenvolvimento de quaisquer aplicações 3D.

Para trabalhar com os objetos tridimensionais, este pacote possibilita a importação de diferentes tipos de modelos 3D prontos (.OBJ e .MD2) construídos em ferramentas de modelagem 3D. Permite também o controle da iluminação, câmera e fundo (*background*). Nos dois primeiros casos o desenvolvedor pode instanciar quantos componentes achar necessário. Com relação à câmera, é possível utilizar uma câmera móvel que simula a visão do usuário em primeira pessoa, podendo-se detectar a colisão da câmera com os objetos do cenário 3D. Para possibilitar a simulação de experimentos e o funcionamento de equipamentos, entre outros exemplos, o pacote possibilita a utilização e o controle das animações dos objetos no ambiente.

O pacote também fornece classes para possibilitar interações eficientes no ambiente 3D, possibilitando a seleção de objetos e a realização de operações gráficas (rotação e translação) no objeto selecionado. Assim o desenvolvedor poderá fornecer ao usuário a possibilidade de interagir com os objetos 3D com alto grau de liberdade. O pacote também disponibiliza recursos para avaliação do desempenho gráfico do dispositivo, medindo a quantidade de quadros por segundo durante a navegação no ambiente. Esta funcionalidade mostra-se bastante útil durante o desenvolvimento do treinamento, para a realização de testes em diferentes equipamentos, e conseqüente determinação dos dispositivos compatíveis e daqueles com apresentação gráfica satisfatória.

Os casos de uso desse pacote são: controlar iluminação; controlar câmera; controlar *background*; carregar modelo 3D; controlar objeto 3D; controlar animação; detectar colisão da câmera; controlar apontador; e registrar desempenho gráfico. Este último caso de uso tem importância fundamental durante a construção do treinamento, ao permitir que o desenvolvedor descubra os dispositivos que tem uma melhor execução da aplicação e aqueles que apresentam uma execução insatisfatória.

O VirTraM foi implementado utilizando J2ME. Para a implementação dos recursos tridimensionais utilizou-se uma API para J2ME, a M3G (Mobile 3D Graphics API) (descrita na seção 4.5.1). A M3G possibilita o desenvolvimento de aplicações otimizadas,

que levam em consideração as limitações de processamento e memória dos dispositivos móveis. Uma importante característica da M3G é a definição de um formato de arquivo padrão para uso nas cenas, o .m3g. O uso desse formato de arquivo reduz o tamanho e a complexidade das aplicações e aumenta a portabilidade, através da conversão de formatos conhecidos para o .m3g.

Através das primitivas gráficas disponibilizadas pela M3G, foram implementadas no VirTraM classes que permitem a utilização dos formatos .OBJ e MD2, modelos comumente utilizados em aplicações tridimensionais. O diagrama de atividades da Figura 4-12 ilustra como acontece o processo de leitura e construção do objeto 3D.

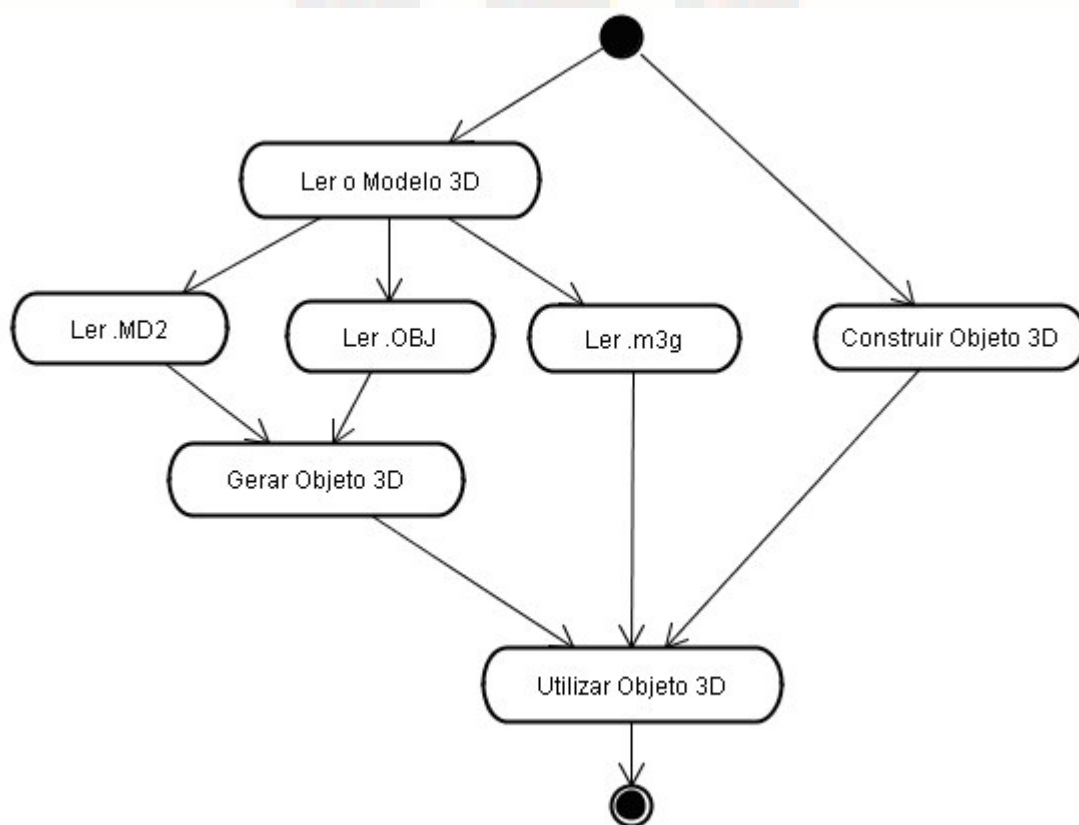


Figura 4-12: Diagrama de atividade com o processo de manipulação do objeto 3D.

Desta forma, o Pacote 3D permite ao desenvolvedor a possibilidade de trabalhar com três formatos de modelo 3D. Pela característica de expansibilidade do VirTraM, é possível que o desenvolvedor implemente novas classes que suportem outros padrões. Caso queira construir seu próprio objeto 3D, o desenvolvedor deverá utilizar diretamente as classes da M3G.

O Pacote 3D contém uma classe principal responsável pela montagem da cena tridimensional, a *VTM_3DScene*. O treinamento deve possuir uma única instância dessa classe e pode usá-la em qualquer parte do código. Para isso implementa-se o padrão *Singleton*. O pacote também possui uma classe responsável pela iluminação, a *VTM_Ligth*, e quatro especializações dela: *VTM_LAmbient*, *VTM_LOmni*, *VTM_LDIRECTIONAL* e *VTM_LSpot*. As classes responsáveis pelo ângulo de visão da cena são a *VTM_FixedCamera* e a *VTM_MobileCamera*, que herdam de *VTM_Camera*. Para controle do apontador são necessárias duas classes: a *VTM_Pointer* e *VTM_VisualPointer*, sendo a primeira responsável pela funcionalidades do apontador e a segunda responsável pela apresentação gráfica. As classes *VTM_Background* e *VTM_FPSCounter* são responsáveis pela apresentação do fundo da cena e por registrar o desempenho da renderização no dispositivo móvel, respectivamente. Uma das classes principais desse pacote, responsável pelo modelo 3D, é a *VTM_3DModel*. Esta classe possui três subclasses, uma para cada tipo de formato 3D implementado: *VTM_3DModelOBJ*, *VTM_3DModelMD2* e *VTM_3DModelM3G*. A classe *VTM_3DModel* pode conter uma ou mais classes do tipo *VTM_Node* e *VTM_Animation*, que tratam das partes do modelo 3D e de suas animações, respectivamente. A Figura 4-13 apresenta o Diagrama de Classes do Pacote 3D.

4.4.8. Pacote *HiLevel*

Este é o principal pacote do VirTraM, responsável por integrar os pacotes anteriores e executar o treinamento. Ao integrar os recursos dos pacotes do VirTraM, o Pacote HiLevel fornece ao desenvolvedor as funcionalidades necessárias para a construção do treinamento, utilizando poucas linhas de código. Para possibilitar essa integração, é necessário que os componentes do treinamento já estejam criados, a seguir: cenário 3D, conteúdo, telas de apresentação, controle de interação e *log*. O pacote disponibiliza funções para a adição e integração dos componentes do treinamento.

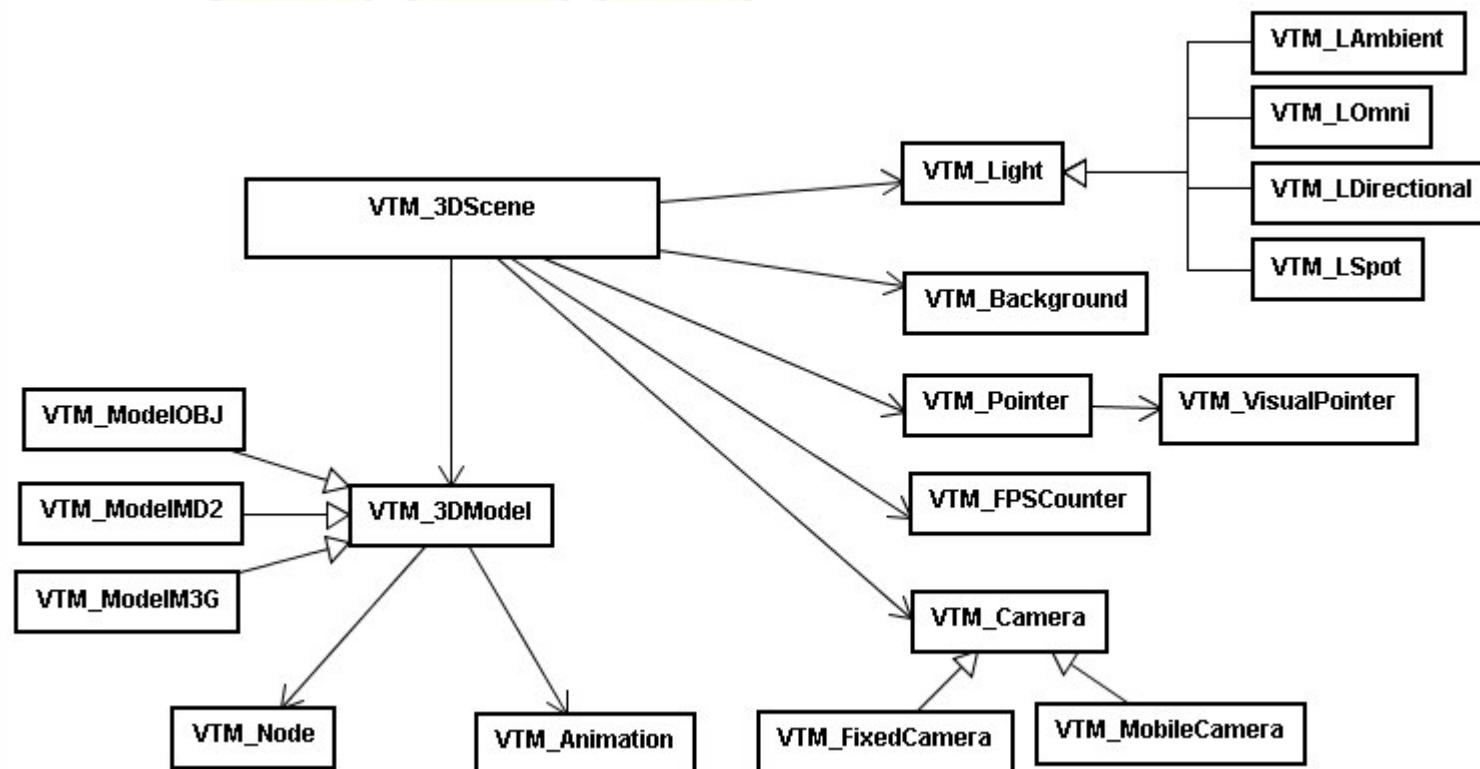


Figura 4-13: Diagrama de Classes do Pacote 3D.

O desenvolvedor, através deste pacote, inicia e gerencia a apresentação do treinamento, alternando entre o ambiente tridimensional e o conteúdo. É possível controlar as instruções que serão apresentadas no cenário 3D, determinando uma sequência de apresentação e permitindo uma apresentação passo-a-passo. Outra funcionalidade é a possibilidade de alternar entre diferentes câmeras no ambiente 3D, modificando o ângulo de visão do usuário. O pacote *HiLevel* também possibilita o retorno à posição inicial ou à anterior, dentro do ambiente 3D. Desta forma, a qualquer momento o aluno pode reiniciar o treinamento ou desfazer a última operação realizada. A Figura 4-14 apresenta o Diagrama de Classes deste pacote.

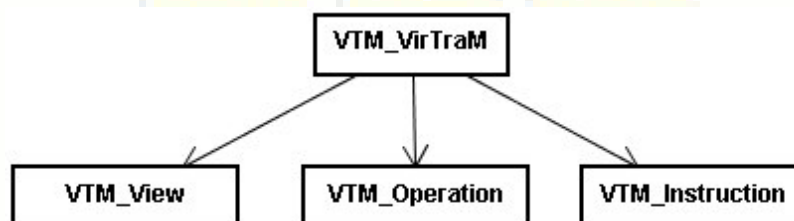


Figura 4-14: Diagrama de Classes do Pacote *HiLevel*.

Conforme mencionado anteriormente, para utilização deste pacote, as classes equivalentes aos componentes do treinamento já devem ter sido instanciadas, são elas: *VTM_DisplayIntroduction*, *VTM_DisplayDescription*, *VTM_3DScene*, *VTM_Interaction* e *VTM_ContentTraining*. Caso queira armazenar um histórico das ações do usuário, pode-se adicionar ao treinamento a classe *VTM_Log*. A classe principal deste pacote, base do treinamento, é a *VTM_VirTraM*. Ela possui métodos para adicionar os componentes e iniciar o treinamento. Este deve possuir uma única instância desta classe e pode utilizá-la em qualquer parte do código, para isso foi implementado o padrão **Singleton**. As principais funcionalidades do Pacote *HiLevel* estão na classe *VTM_VirTraM*, tendo as outras classes funções auxiliares: *VTM_Instructions*, responsável pela apresentação das animações tridimensionais; *VTM_Operation*, responsável por armazenar e desfazer a última operação no ambiente 3D; e *VTM_View*, responsável pelas mudanças de visão no cenário tridimensional.

4.4.9. Resumo dos Pacotes do VirTraM

A Tabela 4.1 apresenta um resumo dos pacotes do VirTraM, juntamente com suas descrições e suas classes, a fim de facilitar a compreensão do *framework*.

Tabela 4-1: Resumo dos pacotes do VirTraM.

Pacote	Função e Classes
<i>Interaction</i>	Fornecer classes para o desenvolvedor gerenciar a interação do usuário com o dispositivo móvel. Este pacote é composto pelas seguintes classes: <i>VTM_Interaction</i> , <i>VTM_KeyControls</i> , <i>VTM_VisualPanel</i> , <i>VTM_PanelPalm</i> , <i>VTM_PanelPocket</i> e <i>VTM_PanelCell</i> .
<i>Log</i>	Fornecer classes para gravação e consulta das ações do usuário. Este pacote é composto pelas seguintes classes: <i>VTM_Log</i> e <i>VTM_ObjectLog</i> .
<i>Optional</i>	Fornecer classes com funcionalidades extras e relacionadas ao dispositivo móvel, como monitoramento dos recursos, som e vibração. Este pacote é composto pelas seguintes classes: <i>VTM_Monitor</i> , <i>VTM_Sound</i> , <i>VTM_CollectionSound</i> e <i>VTM_Vibration</i> .
<i>Visual</i>	Fornecer classes que montam as interfaces de conteúdo para o usuário. Este pacote é composto pelas seguintes classes: <i>VTM_DisplayContent</i> , <i>VTM_DisplayIntroduction</i> , <i>VTM_DisplayDescription</i> , <i>VTM_VisualItem</i> , <i>VTM_Image</i> e <i>VTM_Text</i> .
<i>Persistence</i>	Fornecer classes para persistência dos dados no dispositivo móvel. Este pacote é composto pelas seguintes classes: <i>VTM_RecordBase</i> , <i>VTM_BaseContent</i> , <i>VTM_BaseLog</i> , <i>VTM_BaseConfig</i> e <i>VTM_ObjectPersist</i> .
<i>Content</i>	Fornecer classes para o tratamento do conteúdo do treinamento. Este pacote é composto pelas seguintes classes: <i>VTM_ContentTraining</i> , <i>VTM_Content</i> , <i>VTM_ContentShow</i> , <i>VTM_ContentQuestion</i> , <i>VTM_ContentParser</i> , <i>VTM_ContentFile</i> , <i>VTM_ParserShow</i> e <i>VTM_ParserQuestion</i> .
<i>3D</i>	Fornecer classes para a construção do ambiente tridimensional. Este pacote é composto pelas seguintes classes: <i>VTM_3DScene</i> , <i>VTM_3DModel</i> , <i>VTM_ModelOBJ</i> , <i>VTM_ModelMD2</i> , <i>VTM_ModelM3G</i> , <i>VTM_Node</i> , <i>VTM_Animation</i> , <i>VTM_Light</i> , <i>VTM_LAmbient</i> , <i>VTM_LOmni</i> , <i>VTM_LDIRECTIONAL</i> , <i>VTM_LSpot</i> , <i>VTM_Background</i> , <i>VTM_Pointer</i> , <i>VTM_VisualPointer</i> , <i>VTM_FPSCounter</i> , <i>VTM_Camera</i> , <i>VTM_FixedCamera</i> e <i>VTM_MobileCamera</i> .
<i>HiLevel</i>	Fornecer as classes de alto nível para construção do treinamento, integrando os recursos dos outros pacotes. Este pacote é composto pelas seguintes classes: <i>VTM_VirTraM</i> , <i>VTM_View</i> , <i>VTM_Operation</i> e <i>VTM_Instruction</i> .

4.5. Aspectos da Implementação

O VirTraM foi implementado utilizando-se a plataforma Java 2 para dispositivos móveis (J2ME - *Java 2 Micro Edition*), com o perfil MIDP (*Mobile Information Device Profile*) versão 2.0 e a configuração CLDC (*Connected Limited Device Configuration*) versão 1.1 (Sun, 2004). É importante destacar que esta escolha foi necessária para a implementação dos estudos de caso. Porém, a arquitetura conceitual do VirTraM não está limitada a J2ME, podendo ser implementada em outras plataformas. Os principais motivos para essa escolha foram: a grande portabilidade dessa plataforma, disponível em uma grande variedade de tipos e fabricantes de dispositivos móveis, e por ser baseada na tecnologia Java, uma das linguagens de programação com maior número de desenvolvedores. Dois outros fatores também foram importantes para esta escolha: a existência de uma API 3D para J2ME, a Mobile 3D Graphics API (M3G); e a possibilidade de o usuário instalar, remotamente e sob demanda, as aplicações no seu dispositivo utilizando o OTA (*Over-The-Air provisioning*). A API M3G e o mecanismo OTA são discutidos a seguir.

4.5.1. Mobile 3D Graphics API - M3G

A M3G é de um pacote opcional do J2ME, definido na JSR-184 (Java, 2004), que permite a construção de aplicações interativas com recursos gráficos tridimensionais. A especificação da M3G foi lançada em dezembro de 2003 com objetivo principal de fornecer um eficiente pacote gráfico 3D adequado a dispositivos com poder de processamento limitado, e nenhum suporte de *hardware* para gráficos 3D e ponto flutuante. No entanto, a API também é escalável para dispositivos com maior poder computacional, incluindo até *hardware* gráfico 3D especializado. A especificação M3G abrange jogos, animações e visualizações 3D.

Uma importante característica da M3G é a definição de um formato de arquivo padrão para uso nas cenas, o .m3g. O uso desse formato de arquivo reduz o tamanho e a complexidade das aplicações e aumenta a portabilidade, através da conversão de formatos conhecidos para o .m3g. A API é composta por cerca de 250 métodos em 30 classes, que fornecem as primitivas gráficas 3D.

As classes da Mobile 3D Graphics API estão definidas no pacote `javax.microedition.m3g`, e fornecem uma grande variedade de recursos 3D, tais como: câmera, luz, *background*, malha, textura, animação e transformações.

As aplicações foram testadas e são demonstradas no próximo capítulo utilizando-se o Wireless Toolkit (WTK) da SUN para J2ME (SUN, 2004).

4.5.2. Over-The-Air provisioning (OTA)

O termo OTA representa uma especificação que define um mecanismo de distribuição de aplicações aos usuários de dispositivos móveis em uma rede *wireless*. A transmissão acontece utilizando o paradigma cliente-servidor sobre o protocolo HTTP. A Figura 4-15 (OTA, 2004) mostra como é o processo de instalação remota da aplicação no dispositivo móvel.

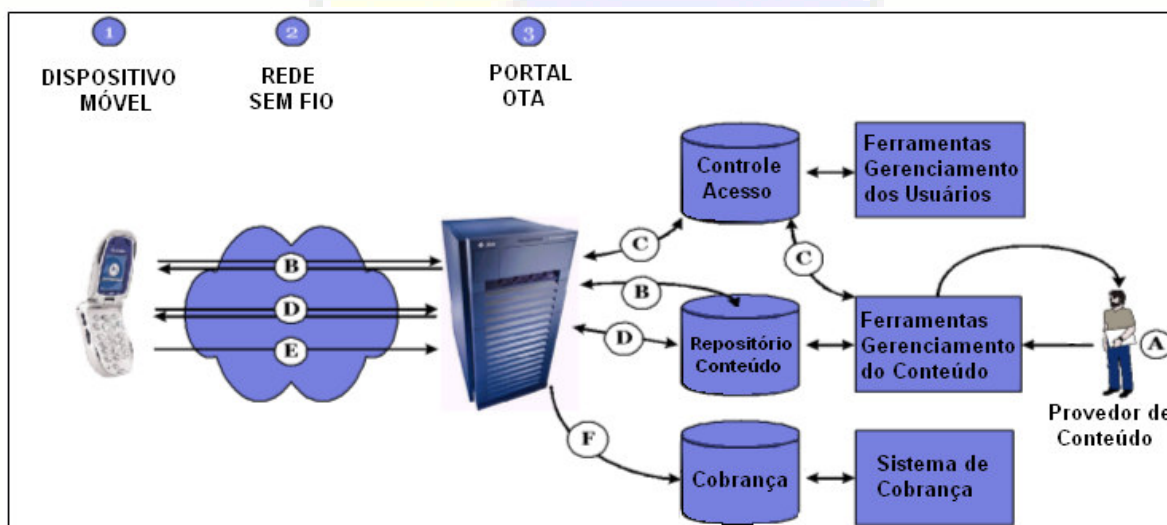


Figura 4-15: Diagrama de instalação de aplicações com OTA (OTA, 2004).

De uma forma sucinta, o usuário solicita uma aplicação em um determinado servidor que procura pela aplicação e a retorna para o usuário. A seguir, os detalhes envolvidos na distribuição da aplicação via OTA são descritos.

A. Gerenciamento do Conteúdo – Do lado do servidor, os desenvolvedores disponibilizam suas aplicações.

B. Descoberta do Conteúdo – O servidor procura, no repositório das aplicações, a aplicação solicitada pelo usuário. Este também pode acessar o repositório no servidor e este deverá mostrar um menu com as aplicações disponíveis.

C. Autenticação – Se o servidor de distribuição suporta um módulo de autenticação, o usuário será autenticado antes de obter acesso ao repositório.

D. Localização e Instalação da Aplicação – Uma vez que o processo de autenticação é completado e a aplicação selecionada, o *download* da aplicação é iniciado. O sistema de gerenciamento de aplicação no dispositivo móvel (AMS - application management system) é a aplicação nativa do dispositivo que gerencia o *download*, instalação, execução e remoção das aplicações no dispositivo móvel. Se o arquivo de descrição da aplicação (arquivo JAD) existe no servidor, o AMS faz o *download* dele. Baseado nas informações contidas no arquivo JAD, o AMS automaticamente faz o *download* da aplicação (arquivo JAR). Se a aplicação foi recuperada com sucesso, a instalação é realizada automaticamente.

E. Confirmação – O AMS envia ao servidor uma confirmação de sucesso ou falha na instalação.

F. Cobrança – A Informação de confirmação da instalação pode ser usada para identificar os usuários que instalaram as aplicações e possibilitar uma cobrança posterior.

O Provedor de Conteúdo é a entidade responsável pela elaboração e disponibilização do conteúdo.

4.6. Considerações Finais

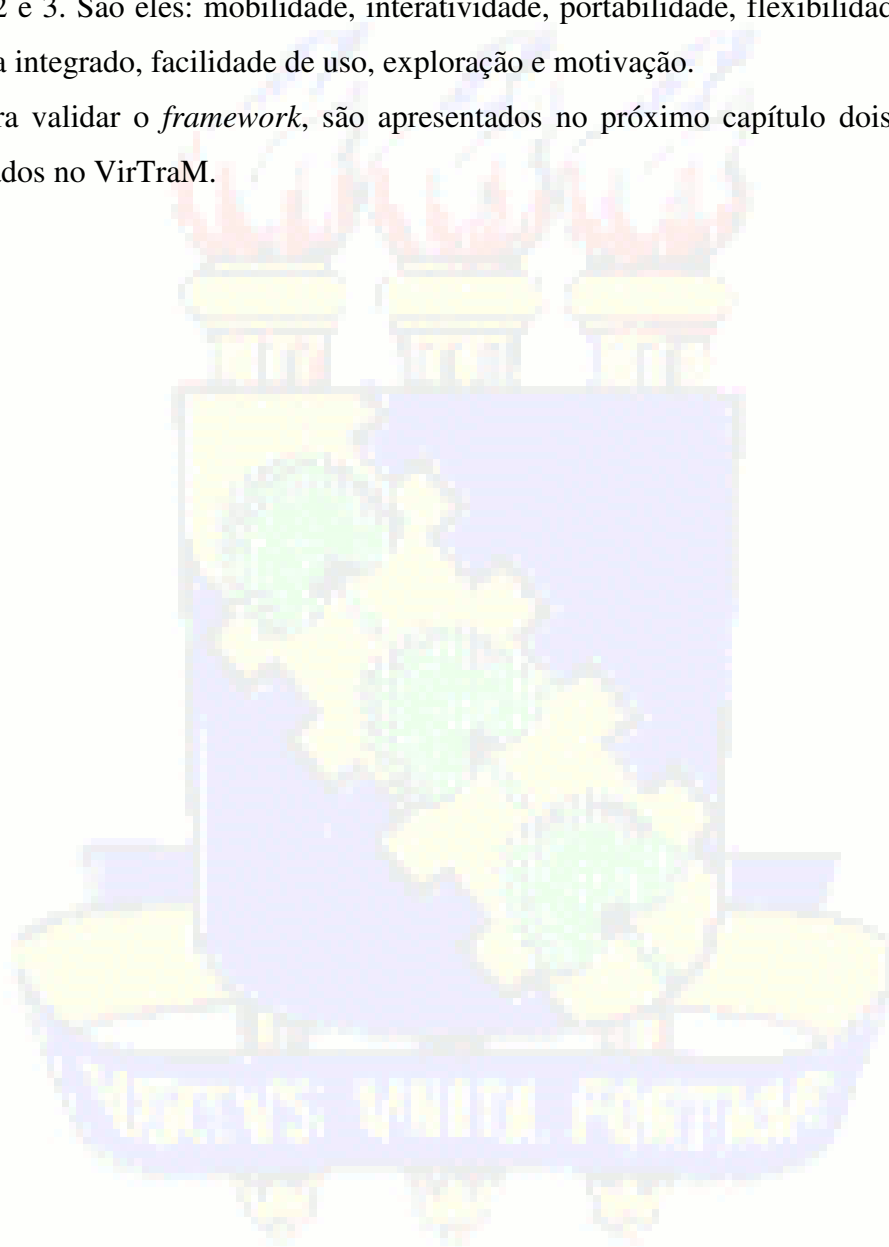
Um *framework* pode ser descrito como uma solução para o reuso de componentes de arquiteturas orientadas a objetos ou ainda um projeto abstrato para um domínio específico de aplicações. As principais vantagens da utilização de *frameworks* são a melhora na qualidade e redução do custo e tempo de desenvolvimento de software, através da reutilização tanto na implementação quanto no projeto.

Este capítulo apresentou um *framework* orientado a objetos para a construção de aplicações educacionais em dispositivos móveis com recursos de realidade virtual não imersiva, o VirTraM. O objetivo principal é fornecer ao desenvolvedor uma arquitetura extensível e reutilizável para implementação de programas em dispositivos móveis com

foco na aprendizagem. Este capítulo também apresentou a metodologia adotada para a construção do VirTraM e descreveu os detalhes da implementação, incluindo todas as classes e padrões utilizados.

O VirTraM busca atender um conjunto de requisitos, dentre os apresentados nos capítulos 2 e 3. São eles: mobilidade, interatividade, portabilidade, flexibilidade, conteúdo multimídia integrado, facilidade de uso, exploração e motivação.

Para validar o *framework*, são apresentados no próximo capítulo dois estudos de caso baseados no VirTraM.



CAPÍTULO 5

Estudos de Caso

Este capítulo apresenta o desenvolvimento de duas aplicações construídas a partir do VirTraM: o *museumM*, um museu virtual, e a *mPrinter*, uma impressora virtual. Estes estudos de caso foram escolhidos por que representam aplicações bem distintas, possuindo públicos alvos diferentes, possibilitando uma boa representação tridimensional, e permitindo validar os principais requisitos do VirTraM, que serão mostrados ao longo deste capítulo. Primeiramente, na Seção 5.1 é apresentado o *museumM*, uma aplicação de apoio à aprendizagem durante visitas a museus. Na Seção 5.2 é apresentada a *mPrinter*, uma aplicação para treinamento no funcionamento de impressoras. Finalmente, a Seção 5.3 apresenta as considerações finais deste capítulo.

5.1. O *museumM*

O primeiro estudo de caso é um cenário tridimensional representando um museu virtual. Esta aplicação tem como principal objetivo auxiliar a aprendizagem sobre as obras de arte expostas em museus.

Inúmeros trabalhos já foram desenvolvidos no meio acadêmico com o objetivo de retratar museus em computadores pessoais (Johnson & Rickel, 98, Wojciechowski et al., 2004; Corcoran et al., 2002; Charitos et al., 2001; Wazlawick, 2001). Na área da computação móvel, algumas iniciativas também já estão sendo desenvolvidas no sentido de se utilizarem os dispositivos móveis em museus (Tellis & Proctor, 2003; Sauer & Gobel, 2003; Manning & Glenda, 2004; Hsi, 2002).

Algumas das vantagens dos museus virtuais são: motivação para uma maior exploração do acervo do museu, através da possibilidade de aproximação e manipulação das obras; visualização e simulação de experimentos nos museus de ciências, para que os alunos possam fixar os conceitos teóricos aprendidos; e maior poder de ilustração, tendo em vista o fato de a realidade virtual, possibilitando a visão em três dimensões, permitir o emprego de múltiplas camadas de referência (profundidade, tamanho e posição) no

aprendizado. É importante salientar que este trabalho utiliza o termo museu virtual para os ambientes tridimensionais interativos que possibilitam navegação, simulação e sensação de imersão, mesmo que parcial (no caso da realidade virtual semi-imersiva).

Os computadores de mão criam uma possibilidade única ao fornecer aos visitantes uma ferramenta robusta para o acesso ao conteúdo das obras do museu. Textos, imagens, vídeos, fotografias, conversas via *chat* com os curadores e artistas e acesso à Internet podem ser disponibilizados. Em Manning & Glenda (2004), os autores apresentam um estudo comparativo entre dois grupos de visitantes ao Museu de Arte Blanton nos Estados Unidos: visitantes que utilizaram *handhelds* e visitantes que não utilizaram. A Tabela 5-1 apresenta os resultados dessa pesquisa com relação ao aprendizado adquirido sobre determinadas obras. Estes resultados demonstram uma perceptível melhora na fixação da informação das obras por parte dos visitantes que utilizaram os dispositivos móveis.

Tabela 5-1: Tabela comparativa do grau de fixação da informação dos usuários.

Obras-de-arte (pinturas)	Grau de Fixação	Com Handheld	Sem Handheld
1. Synecdoche	Bom	58.2%	47.1%
	Razoável	36.1%	32.1%
	Pouco	5.2%	15.0%
	Nenhum	.4%	5.7%
2. Untitled (Hands/Stranger in the Village)	Bom	56.9%	27.6%
	Razoável	31.9%	40.3%
	Pouco	11.3%	26.9%
	Nenhum	.0%	5.2%
3. I Pray and then I Play in the Collective Landscape	Bom	50.0%	37.6%
	Razoável	35.8%	35.5%
	Pouco	12.2%	22.0%
	Nenhum	2.0%	5.0%
4. By the River	Bom	58.7%	52.6%
	Razoável	29.1%	32.1%
	Pouco	10.5%	12.4%
	Nenhum	1.6%	2.9%

Em Tellis & Proctor (2003), são apresentados trechos de uma entrevista do curador do museu Tate Modern na Inglaterra. Ele acredita que nos próximos anos as tecnologias de computação móvel substituirão as atuais ferramentas utilizadas em museus. Ele destaca como principal vantagem a possibilidade de o visitante poder consultar conteúdos multimídia no momento em que está diante da obra.

Os trabalhos citados anteriormente apresentam exemplos da utilização de dispositivos móveis em museus, destacando suas vantagens e desvantagens. Não foi encontrado, porém, nenhum experimento utilizando ambientes tridimensionais, objeto desta dissertação.

5.1.1. Descrição

O museumM (Virtual Museum for Mobile Device) consiste de um protótipo desenvolvido a partir do VirTraM, que utiliza um cenário 3D para simular um museu real. Como as aplicações anteriormente citadas, ele também funciona como um guia virtual, porém apresenta características próprias que o destacam dos outros. Uma das principais vantagens do museumM é a utilização de cenários tridimensionais. O museumM integra objetos 3D, texto e imagem para possibilitar a melhor compreensão da obra ao visitante. Conforme Oliveira (2004), a utilização de ambientes tridimensionais tem um papel importante na representação de museus digitais, ao melhorar o processo de fruição e apreensão do saber sobre uma obra e ao fornecer uma informação qualitativamente eficaz.

Outro importante diferencial do museumM é a sua portabilidade. Embora desenvolvido inicialmente para telefones celulares, também pode funcionar em palmtops (handhelds). Isto se torna realmente importante no contexto dos museus, tendo em vista que os visitantes poderão utilizar seus próprios telefones para instalarem o guia virtual, sem que seja necessário que o museu mantenha um estoque de palmtops para disponibilizar aos seus visitantes.

O museumM utiliza recursos de realidade virtual não imersiva para simular um museu fictício, propiciando excursões virtuais interativas e aumentando a motivação para a aprendizagem. O ambiente tridimensional é composto por duas salas, uma com quadros e uma outra com uma escultura. O visitante pode navegar pelas salas, olhando para os lados,

para cima e para baixo. Toda a interação é feita com a utilização das teclas do telefone celular, de uma forma simples para facilitar a navegação do visitante. O usuário pode ora movimentar a câmera do sistema, ora movimentar o apontador, ou ainda movimentar um objeto 3D. Ao selecionar um objeto 3D, um menu aparece e o usuário pode escolher entre consultar a descrição daquela obra ou executar uma animação associada, caso exista. Ao escolher a descrição, uma nova tela é aberta trazendo o conteúdo relacionado àquela obra. As figuras 5-1 e 5-2 apresentam algumas telas do museuM, capturadas a partir do simulador do Wireless Toolkit (WTK) (SUN, 2004).

5.1.2. Detalhes de Projeto e Implementação

A implementação do museuM envolveu três etapas principais (modelagem 3D, conteúdo e desenvolvimento da aplicação), em diferentes áreas de conhecimento, que abrangem os requisitos funcionais do VirTraM, descritos na Seção 4.3.3.

5.1.2.1. Modelagem 3D

Etapla de construção do ambiente tridimensional do museu. Esta etapa foi desenvolvida por um designer gráfico com experiência em modelagem tridimensional. O modelo foi construído no software 3D Studio Max (Max, 2004). Foram utilizadas técnicas que diminuem a quantidade de polígonos dos objetos 3D (Steed, 1998), para obter um melhor desempenho do ambiente no celular. O designer também trabalhou na simplificação das texturas para não sobrecarregar o modelo, porém tentando não perder a sensação de realismo. O ambiente, após ter sido modelado no 3D Studio Max, foi exportado para o formato .m3g.

Conforme visto na Seção 4.4.7, o grande benefício da utilização do VirTraM para esta etapa é a capacidade de se poder utilizar diferentes formatos de modelos 3D (.obj, .md2 e .m3g). O formato padrão da API M3G é o .m3g, porém ainda são poucas as ferramentas de modelagem compatíveis com esse formato. Desta forma, as classes *VTM_3DModelOBJ* e *VTM_3DModelMD2* do Pacote 3D, para importação dos formatos .obj e .md2, fornecem ao designer certa liberdade para trabalhar com um número maior de ferramentas. A modelagem do museuM foi realizada em paralelo ao desenvolvimento da aplicação.



Figura 5-1 a) Uma visão ampla de uma das salas do museu virtual. b) Tela de introdução do museuM. c) Uma visão aproximada de um quadro e o menu associado.

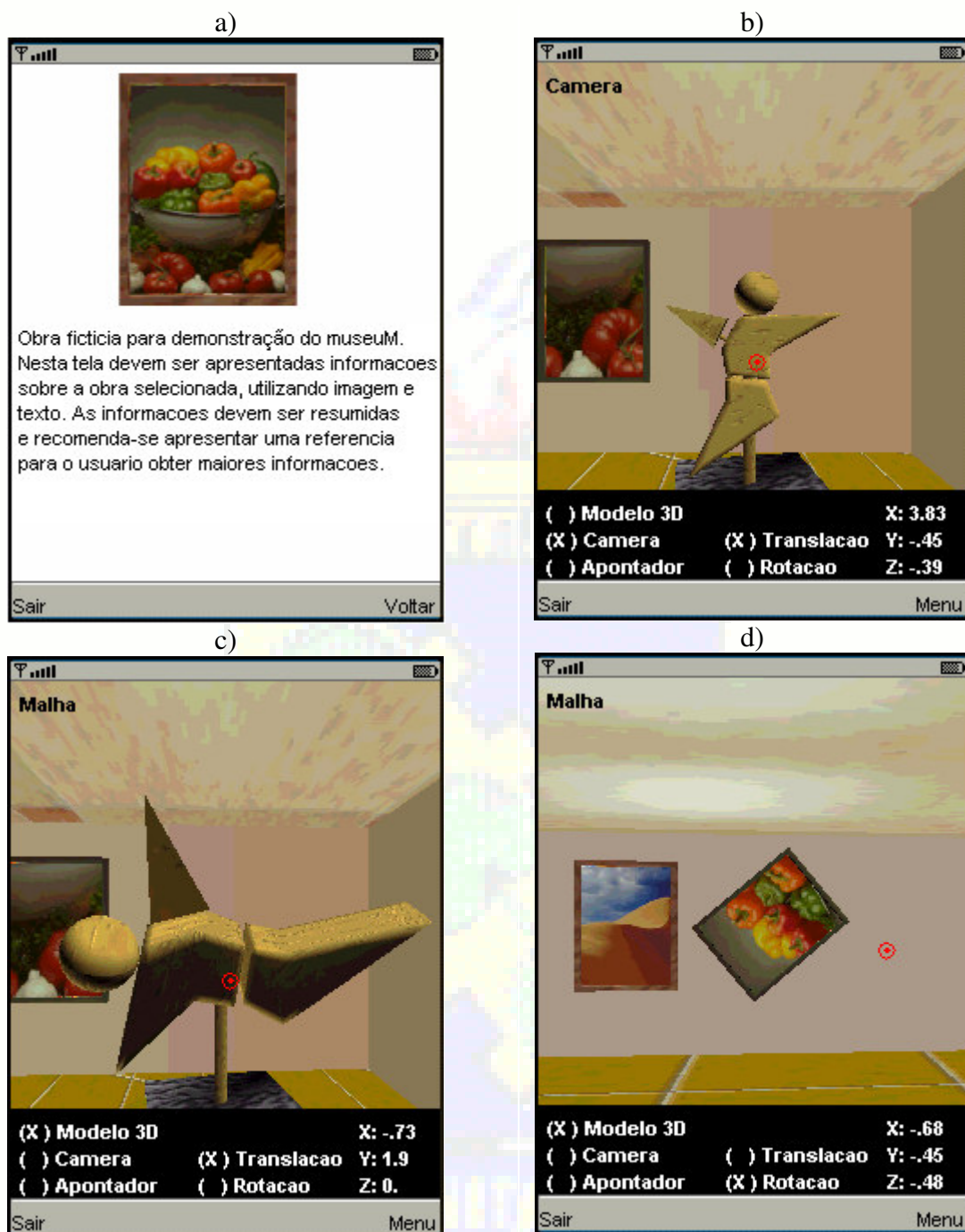


Figura 5-2 a) Uma tela com o conteúdo da obra selecionada. b) Uma visão aproximada da escultura virtual. c) Uma visão aproximada da escultura virtual rotacionada. d) Uma visão de dois quadros com um rotacionado e transladado.

5.1.2.2. Conteúdo

A etapa de preparação do material do treinamento consistiu da elaboração do arquivo XML, citado no Capítulo 4, que contém as informações das obras do museu: identificador, título, nome do arquivo com a imagem e a descrição.

Este documento também contém as informações da tela de introdução do museuM, descritas no documento no grupo com o identificador igual a zero (ver Figura 5-3). Para executar esta etapa, o conteudista deve ter um conhecimento de todos os objetos 3D do ambiente virtual que correspondem às obras, aos quadros e à escultura. A Figura 5-3 apresenta um trecho do documento com o conteúdo do museuM, incluindo as informações da introdução do treinamento e de uma obra.

```
- <Model3D>
- <parts>
- <part>
  <id>0</id>
  <title>museuM</title>
  <image>museum.png</image>
  <description>Bem vindo ao museu virtual movel. Aqui voce fara um
    passeio virtual em um museu utilizando seu telefone celular.
    Voce podera selecionar as obras e obter mais informacoes sobre
    elas. Bom passeio!</description>
</part>
- <part>
  <id>65344</id>
  <title>A Praia</title>
  <image>apraia.png</image>
  <description>Obra ficticia para demonstracao do museuM. Nesta
    tela devem ser apresentadas informacoes sobre a obra
    selecionada, utilizando imagem e texto. As informacoes devem
    ser resumidas e recomenda-se apresentar uma referencia para o
    usuario obter maiores informacoes.</description>
</part>
...
```

Figura 5-3: Trecho do arquivo de conteúdo do museuM.

Um dos benefícios da utilização do VirTraM na preparação do material do treinamento é a própria estrutura para o tratamento do conteúdo, descrita na Seção 4.4.6. Esta estrutura possibilita ao conteudista trabalhar nas informações do treinamento de forma independente, abstraindo-se da implementação e concentrando-se na associação do

conteúdo com o ambiente virtual. A utilização do arquivo XML também fornece flexibilidade ao treinamento, ao possibilitar que o conteúdo não dependa de um banco de dados específico, podendo assim ser utilizado em inúmeros dispositivos. O arquivo de conteúdo ainda estabelece os objetos 3D com os quais o usuário poderá interagir durante o treinamento.

5.1.2.3. Desenvolvimento da Aplicação

Principal etapa no desenvolvimento do treinamento, consiste na implementação do museuM incluindo a integração dos arquivos .m3g e .xml, gerados nas etapas anteriores.

A Figura 5-4, de uma forma resumida, ilustra o diagrama de classes utilizado na implementação do museuM.

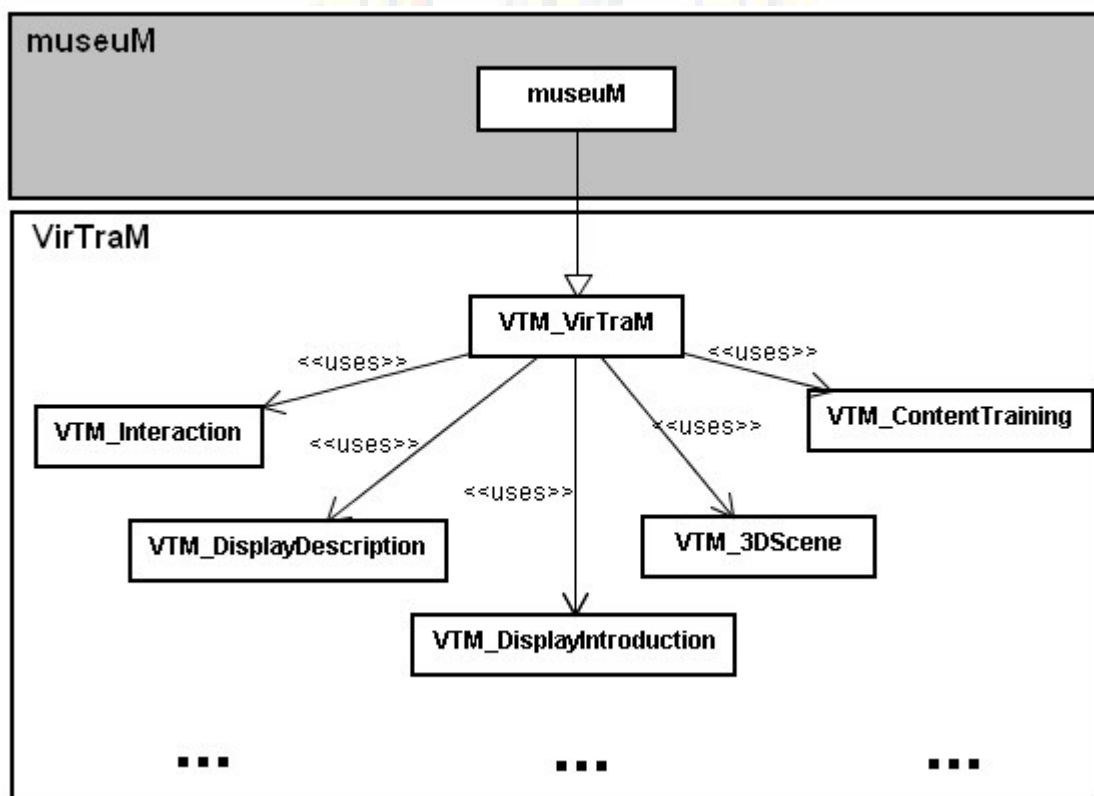


Figura 5-4: Representação resumida do Diagrama de Classes do museuM.

Com relação aos recursos tridimensionais, o VirTraM fornece recursos que abstraem os detalhes da programação 3D, possibilitando que o desenvolvedor centralize seus esforços nos aspectos funcionais do ambiente virtual. Conforme apresentado na Figura

5-4, utilizou a classe *VTM_3DScene* para integrar os componentes (iluminação, câmeras e objetos 3D, por exemplo) e montar a cena 3D. Foram utilizadas para compor o cenário tridimensional do museuM as seguintes classes: *VTM_MobileCamera*, para representar a visão do usuário; *VTM_Pointer* e *VTM_VisualPointer*, para apresentar o apontador; *VTM_LAmbiente*, responsável pela iluminação; e *VTM_ModelM3G*, responsável pela apresentação do modelo 3D do museuM. A *VTM_MobileCamera* simula um usuário, em primeira pessoa, andando pelo museu virtual. Esta classe implementa a função de colisão com o ambiente tridimensional, através da direção do ângulo de visão e do raio de interseção com os objetos 3D. Ainda, durante o desenvolvimento do ambiente tridimensional, foi utilizada a classe *VTM_Monitor*, para comparar o tamanho do modelo 3D com a memória disponível e avaliar a compatibilidade com o dispositivo móvel.

A classe *VTM_ContentTraining* é a classe principal utilizada na programação da associação do conteúdo com a aplicação. Ela é auxiliada por outras classes para realizar suas funções. São elas: a *VTM_ContentFile* é responsável pela leitura do arquivo XML; a *VTM_ParserShow* é a responsável por dividir as informações do arquivo de conteúdo; e a *VTM_ContentShow* é responsável por fornecer os dados tratados para serem armazenados. Assim, o desenvolvedor não precisa se preocupar com as funcionalidades de leitura de arquivos ou de persistência dos dados no dispositivo móvel, concentrando-se em aspectos mais relacionados ao treinamento em si.

Outro benefício obtido através da utilização do VirTraM está relacionado à interação com o usuário. Através da utilização das classes *VTM_Interaction*, *VTM_KeyControls* e *VTM_PanelCell*, do Pacote *Interaction*, o desenvolvedor não precisa se preocupar com a programação da função de cada tecla do dispositivo móvel, reduzindo o tempo de desenvolvimento. Além disso, através destas classes, o *framework* fornece uma configuração padrão de interação que permite a execução de doze ações utilizando-se nove teclas, sem complicar a usabilidade da aplicação. A Figura 5-5 e a Tabela 5-2 apresentam esta configuração padrão.



Figura 5-5: Teclado destacando as teclas padrões do VirTraM.

Tecla	Ação
1	Incrementa X
2	Decrementa X
4	Incrementa Y
5	Decrementa Y
7	Incrementa Z
8	Decrementa Z

Tabela 5-2: Descrição das funções das teclas padrões do VirTraM.

As classes *VTM_DisplayIntroduction* e *VTM_DisplayDescription*, do Pacote *Visual*, fornecem telas padrões para introdução e apresentação das obras do museuM. Desta forma, obtém-se agilidade no desenvolvimento e padronização na aplicação.

Entre os objetivos do VirTraM estão a reutilização de código e o aumento de produtividade para os desenvolvedores. Para isso, o *framework* tem uma classe principal, a *VTM_VirTraM*, do Pacote *HiLevel*, que fornece propriedades específicas e métodos necessários ao desenvolvimento do treinamento. Através dos seus métodos e utilizando classes auxiliares, o desenvolvimento do museuM é feito com poucas linhas de código, conforme trecho de código apresentado na Figura 5-6.

O código apresentado na Figura 5-6 demonstra o ganho de produtividade obtido com o uso da classe *VTM_VirTraM* que, ao reunir as funcionalidades necessárias para a construção do museuM, diminui a quantidade de linhas de código e o tempo de desenvolvimento.

Após a modelagem do museu e da elaboração do arquivo de conteúdo, o tempo necessário para a implementação do museuM foi de aproximadamente uma semana, com uma única pessoa trabalhando. O desenvolvimento nesse curto espaço de tempo só foi possível graças à utilização do *framework* VirTraM. Caso contrário, seria difícil

dimensionar o tempo necessário para o desenvolvimento, tendo em vista a implementação de todos os detalhes da aplicação.

Durante o desenvolvimento do *museumM*, foi comprovado que a modularidade fornecida pelo *framework* possibilita ainda um trabalho em equipe, com profissionais especialistas em diferentes áreas: programação, modelagem e conteúdo didático.

```
...
VTM_DisplayIntroduction tela_introducao;
VTM_DisplayDescription tela_descricao;
VTM_3DScene Cena3D;
VTM_Interaction interacao;
VTM_ContentTraining conteudo;
VTM_VirTram museum;

// Inicializacao dos componentes do VirTram: tela de introducao,
// de descricao, conteudo, controle de interacao e Cena3D
...
// Inicializacao do museum
museum = new VTM_VirTram(this);
//
museum.setIntroduction(tela_introducao);
museum.setDisplayDesc(tela_descricao);
museum.set3DScene(Cena3D);
museum.setControlInteraction(interacao);
museum.setContent(conteudo);
//Iniciar o museum
museum.start();
...
```

Figura 5-6: Trecho de código do *museumM*.

5.2. mPrinter

Chiavenato (1999) e Marra (2000) conceituam treinamento como um processo que envolve a transmissão de conhecimentos específicos, que visam facilitar a aprendizagem, em curto prazo, de conhecimentos e habilidades e de interesse imediato.

Conforme apresentado no Capítulo 3, a realidade virtual apresenta inúmeras aplicações para treinar indivíduos. Em Alberio & Oliveira (2004), os autores destacam que os ambientes tridimensionais podem ser usados como ferramenta de treinamento e educação, desde que essa interface seja de fácil entendimento, fácil uso e principalmente atrativa para o usuário. Oliveira et al. (2000) destaca a capacidade dos usuários poderem manipular e interagir com os objetos no ambiente tridimensional, ganhando experiência e prática, antes de utilizarem o equipamento real.

Na área da educação corporativa, os dispositivos móveis são usados para incrementar a eficiência e a produtividade dos empregados e melhorar o serviço de atendimento aos clientes, especialmente através do treinamento de pessoal técnico em campo e da aquisição remota de informações (Srujan, 2002). Yang (2002) destaca a capacidade de o empregado poder observar o desempenho prático do equipamento e comparar com a execução teórica como sendo um dos principais benefícios da utilização dos dispositivos móveis para treinamentos.

Neste contexto, o segundo estudo de caso consiste de uma aplicação para treinamento técnico do funcionamento de mecanismos de uma impressora. Esta aplicação tem como público alvo tanto usuários finais das impressoras, como técnicos de manutenção. Nas subseções seguintes maiores detalhes são apresentados.

5.2.1. Descrição

A mPrinter (Virtual Printer for Mobile Device) consiste de um protótipo desenvolvido a partir do VirTraM, que utiliza um ambiente tridimensional para simular uma impressora e seu funcionamento. De uma forma geral, é mostrado um treinamento passo a passo para a troca de cartuchos, além da apresentação da descrição de componentes da impressora.

A mPrinter caracteriza-se então como uma ferramenta móvel de apoio à aprendizagem técnica, com características diferenciadas, merecendo destaque a utilização do ambiente tridimensional para simulação de uma impressora real. Através da representação virtual, o usuário terá uma solução educacional com alta interatividade, podendo visualizar a impressora de diferentes ângulos, simular os passos para a troca de cartuchos e obter detalhes das peças. Conforme o primeiro estudo de caso apresentado na seção anterior, toda a interação é feita também utilizando somente o teclado do telefone celular, sendo possível também manipular tanto a câmera quanto o apontador e o modelo 3D. Utilizando-se os menus, o usuário poderá visualizar os passos da simulação da troca de cartuchos, recolocar a impressora na posição inicial e obter descrições de componentes. As figuras 5-7 e 5-8 apresentam imagens da mPrinter, capturadas a partir do simulador do Wireless Toolkit (WTK).



Figura 5-7 a) Uma visão aproximada da impressora virtual. b) Tela de introdução da mPrinter. c) Visão da parte traseira da impressora virtual.

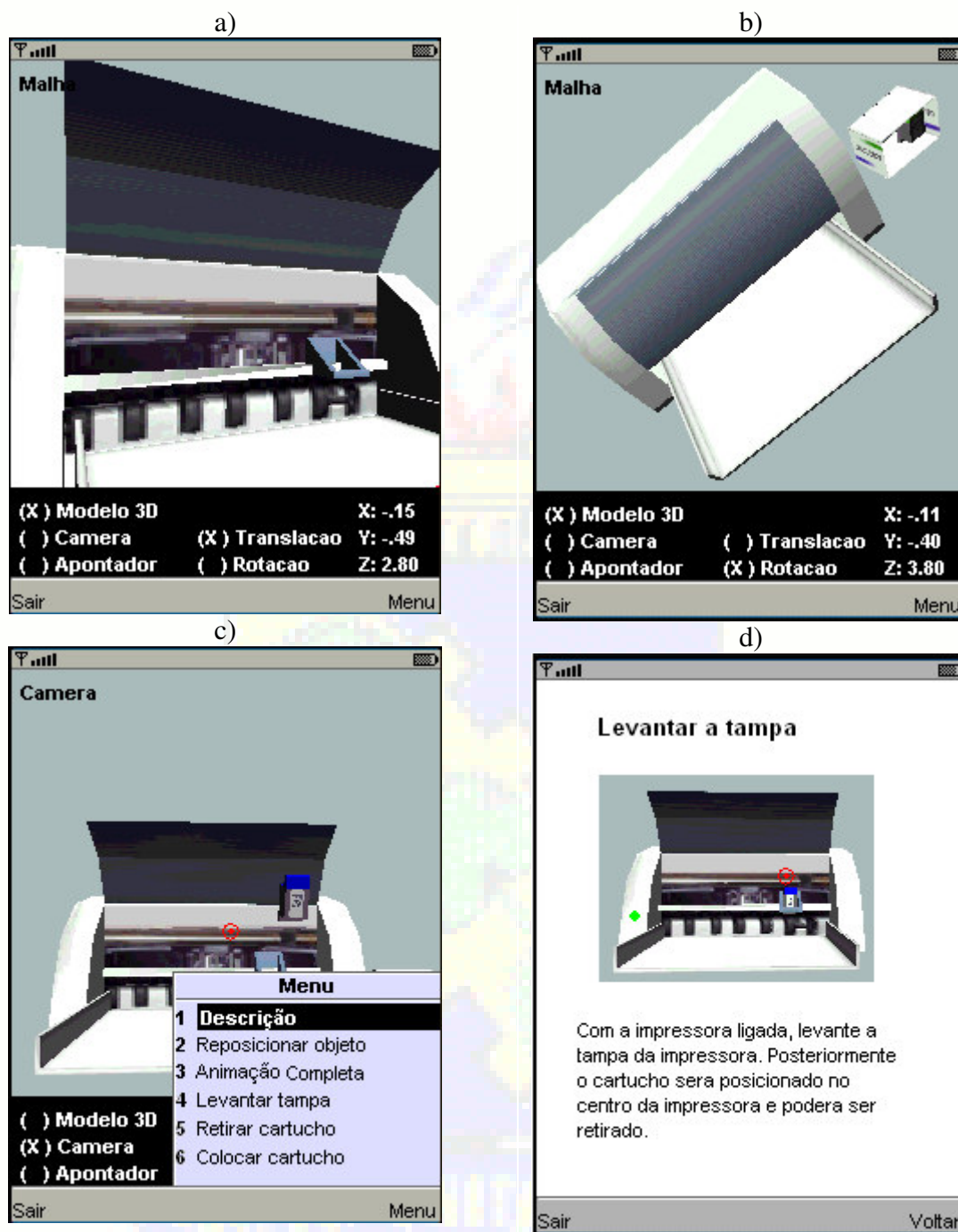


Figura 5-8 a) Uma visão aproximada da impressora com a tampa aberta. b) Uma visão com a impressora aproximada e rotacionada. c) Uma visão da impressora mostrando o menu com as opções. d) A tela de conteúdo antes da animação para levantar a tampa.

5.2.2. Detalhes de Projeto e Implementação

Semelhante ao primeiro estudo de caso, este segundo protótipo também foi desenvolvido em três etapas principais (modelagem 3D, conteúdo e desenvolvimento da aplicação), baseadas na arquitetura do VirTraM, porém com algumas diferenças que serão descritas nas subseções seguintes.

5.2.2.1. Modelagem 3D

A construção do modelo tridimensional da impressora, como no museuM, foi realizada por um designer gráfico. O modelo foi construído no software 3D Studio Max, utilizando-se técnicas para diminuição da quantidade de polígonos dos objetos 3D. As texturas também foram simplificadas para não sobrecarregar o modelo, porém tentando mais uma vez não perder a sensação de realismo. O ambiente, após ter sido modelado no 3D Studio Max, foi exportado para o formato .m3g.

A grande diferença desta etapa para a modelagem do museuM é a utilização da animação. Uma animação completa da impressora foi construída para simular a troca de cartuchos. Para isso, foram desenvolvidos modelos 3D de cartuchos de diferentes cores para demonstrar a troca. No ambiente virtual existe apenas uma única grande animação, porém, como o treinamento é apresentado em passos e implementado através do controle dos quadros da animação (*frames*) na etapa de desenvolvimento, passa-se ao usuário a idéia das seguintes pequenas animações:

- 1ª Levantamento da tampa da impressora;
- 2ª Retirada do cartucho vazio;
- 3ª Colocação do cartucho cheio;

A capacidade de se utilizar modelos tridimensionais de diferentes formatos permitiu que esta etapa não ficasse limitada a um único tipo de formato de arquivo 3D. Desta forma, foi utilizada uma ferramenta de modelagem adequada para se alcançar a melhor relação entre o realismo e as limitações do dispositivo, considerando a execução da animação.

A modelagem da mPrinter foi executada em paralelo ao desenvolvimento da aplicação.

5.2.2.2. Conteúdo

Na etapa de preparação do material, foi elaborado o arquivo de conteúdo, que contém as informações da impressora, de seus componentes e instruções relacionadas aos passos do treinamento. A principal diferença do arquivo XML da mPrinter em relação àquele empregado no museuM reside na divisão em animações (passos). Elas procuram orientar o usuário em cada um dos passos do treinamento. A Figura 5-9 apresenta trechos do documento XML com o conteúdo da mPrinter.

```
- <Model3D>
- <parts>
- <part>
  <id>0</id>
  <title>mPrinter</title>
  <image>mprinter.png</image>
  <description>Bem vindo ao treinamento mPrinter. Sera
    apresentado como e feita a troca de cartuchos e outros
    detalhes da impressora. Bom aprendizado!</description>
</part>
+ <part>
...
...
</parts>
- <animations>
- <animation>
  <id>1</id>
  <part>24234</part>
  <title>Levantar a tampa</title>
  <description>Com a impressora ligada, levante a tampa da
    impressora. Posteriormente o cartucho sera posicionado no
    centro da impressora e podera ser retirado.</description>
  <frame_start>1</frame_start>
  <frame_end>47</frame_end>
</animation>
...
...
```

Figura 5-9: Trechos do arquivo XML da mPrinter.

A partir da Figura 5-9 percebe-se a semelhança com o arquivo XML do museuM, distinguindo-se o conteúdo em si e as *tags* relacionadas com as animações citadas na seção anterior. Como no primeiro estudo de caso, o conteudista pode trabalhar nas informações

do treinamento de forma independente, abstraindo-se da implementação e concentrando-se na associação do conteúdo com a impressora virtual. A elaboração do conteúdo da mPrinter também se beneficiou da estrutura do VirTraM ao fornecer flexibilidade e independência de banco de dados. As *tags* relacionadas às animações permitem a segmentação do treinamento em passos e à associação com o modelo tridimensional.

5.2.2.3. Desenvolvimento da Aplicação

A implementação da mPrinter, incluindo a integração dos arquivos .m3g e .xml, foi muito semelhante à do museuM, com poucas diferenças que serão apresentadas nesta seção.

A Figura 5-10, de uma forma resumida, ilustra o diagrama de classes utilizado na implementação da mPrinter.

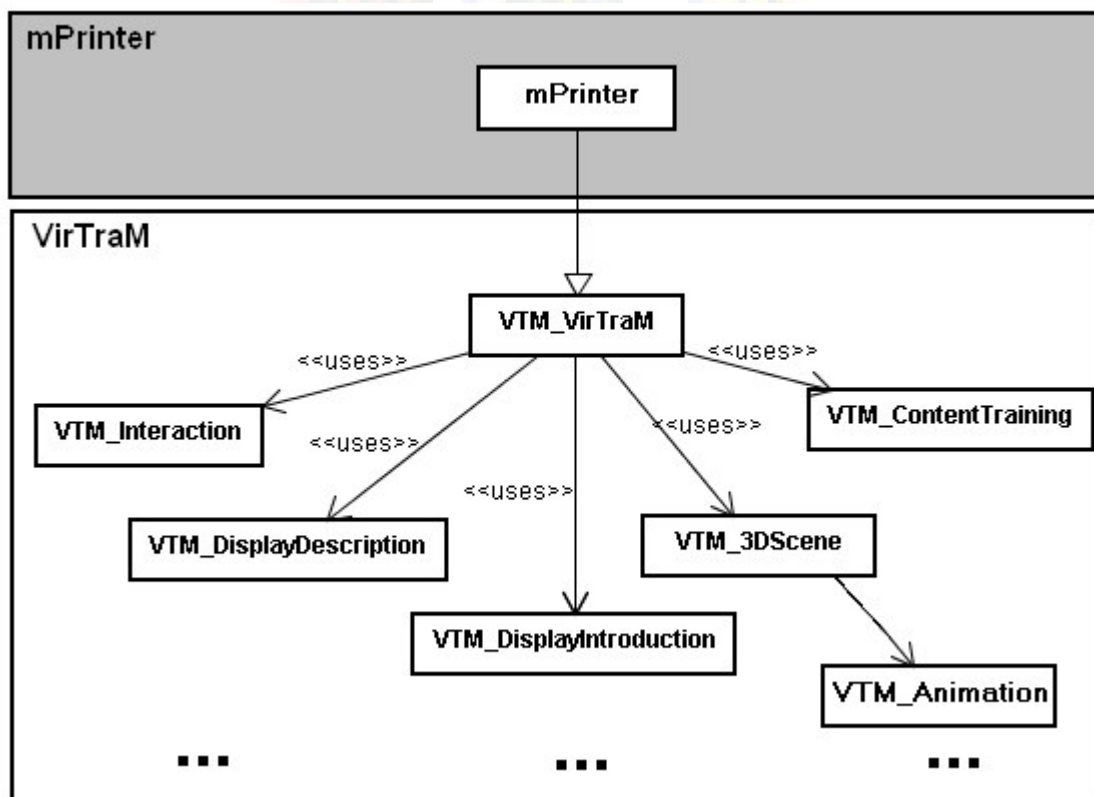


Figura 5-10: Representação resumida do Diagrama de Classes da mPrinter.

Para implementação do ambiente virtual a partir do VirTraM, conforme apresentado na Figura 5-10, utilizou-se a classe *VTM_3DScene*, responsável pela montagem da cena 3D. Para compor o cenário 3D da mPrinter, também foram utilizadas as seguintes classes:

VTM_MobileCamera, para representar a visão do usuário; *VTM_Pointer* e *VTM_VisualPointer*, para apresentar o apontador; *VTM_LAmbiente*, responsável pela iluminação; e *VTM_ModelM3G*, responsável pela apresentação do modelo 3D da mPrinter. A classe *VTM_Monitor* também foi utilizada para controlar o tamanho do modelo tridimensional. Além destas classes, o ambiente virtual deste estudo de caso se diferencia do primeiro por utilizar a classe *VTM_Animation*, para controlar a execução das animações, que correspondem aos passos do treinamento.

Com relação à montagem do conteúdo na aplicação, utilizou-se a classe *VTM_ContentTraining*. Para auxiliar a implementação das funções relacionadas ao conteúdo, outras classes também foram usadas: a *VTM_ContentFile* é responsável pela leitura do arquivo XML; a *VTM_ParserShow* é responsável por dividir as informações do arquivo de conteúdo; e a *VTM_ContentShow* é responsável por fornecer os dados tratados para serem persistidos. Apesar do arquivo de conteúdo da mPrinter possuir *tags* não existentes no arquivo do museuM, as classes utilizadas para o tratamento do conteúdo são as mesmas, necessitando apenas reimplementar o método de tratamento dos dados.

A mPrinter também utiliza os benefícios das classes *VTM_Interaction*, *VTM_KeyControls* e *VTM_PanelCell*, do Pacote *Interaction*, para o desenvolvimento das funções de interação. Não é necessária a utilização de nenhuma tecla a mais para a execução das animações, a configuração das teclas é a mesma do museuM.

As classes *VTM_DisplayIntroduction* e *VTM_DisplayDescription*, do Pacote *Visual*, fornecem telas padrões para introdução, apresentação das descrições dos componentes da impressora e para descrição das instruções dos passos do treinamento. Desta forma, o desenvolvedor não precisa desperdiçar tempo montando inúmeras telas e ainda obtém uma padronização na interface da aplicação.

Semelhante ao museuM, a mPrinter também tem como classe principal da sua implementação a *VTM_VirTraM*. Ela possibilita um desenvolvimento mais rápido ao integrar os recursos de realidade virtual, de conteúdo e de interação da aplicação em uma única classe. Desta forma, foi possível desenvolver a mPrinter também em um curto intervalo de tempo. Após a modelagem da impressora e da elaboração do arquivo de conteúdo, o tempo necessário para implementação da mPrinter foi de aproximadamente

uma semana, com uma única pessoa trabalhando. Mais uma vez, esse tempo só foi possível graças ao emprego do *framework* VirTraM. A Figura 5-11 apresenta um trecho do código-fonte da aplicação da mPrinter, destacando a classe *VTM_VirTraM*.

```
...
VTM_DisplayIntroduction tela_introducao;
VTM_DisplayDescription tela_descricao;
VTM_3DScene Cena3D;
VTM_Interaction interacao;
VTM_ContentTraining conteudo;
VTM_VirTraM mPrinter;

// Inicializacao dos componentes da mPrinter: tela de introducao,
// de descricao, conteudo, controle de interacao e Cena3D
...
// Inicializacao da mPrinter
mPrinter = new VTM_VirTraM(this);
//
mPrinter.setIntroduction(tela_introducao);
mPrinter.setDisplayDesc(tela_descricao);
mPrinter.set3DScene(Cena3D);
mPrinter.setControlInteraction(interacao);
mPrinter.setContent(conteudo);
//Iniciar a mPrinter
mPrinter.start();
...
```

Figura 5-11: Trecho de código da mPrinter.

5.3. Considerações Finais

De acordo com a metodologia escolhida para o desenvolvimento do VirTraM, para se avaliar um *framework* é necessária a realização de testes através do desenvolvimento de protótipos. Este capítulo apresentou duas aplicações construídas a partir do VirTraM, que validam os requisitos previstos no Capítulo 4, conforme resumido a seguir.

No primeiro estudo de caso, o *museumM*, é apresentada uma aplicação para dispositivos móveis voltada para aprendizagem em museus. O *museumM* se caracteriza como uma aplicação com recursos de realidade virtual, com cenário motivador, interativa, de fácil uso e que pode ser executada em diferentes equipamentos. O *museumM* permitiu a integração, de forma segmentada, entre o ambiente tridimensional e o conteúdo educativo, sem diminuir a usabilidade da aplicação e considerando sempre as restrições do visor do dispositivo. O *museumM* também apresentou formas alternativas de interação, que

permitiram ao usuário ora navegar ora manipular os objetos, interagindo apenas com as teclas do telefone celular.

A aplicação mPrinter, nosso segundo estudo de caso, foi mais voltada para técnicos em equipamentos. A aplicação se caracteriza como uma ferramenta de auxílio à aprendizagem do funcionamento de uma impressora para usuários em campo. A mPrinter pode ser utilizada em diferentes dispositivos móveis, simulando o comportamento de uma impressora real e fornecendo uma boa interatividade ao usuário. A mPrinter também apresenta os mesmos benefícios encontrados no museuM.

O tempo para se implementar as duas aplicações foi em torno de quinze dias. Esta produtividade só foi possível através da reutilização de código proporcionada pelo uso do VirTraM. Os estudos de caso demonstraram não apenas o atendimento aos requisitos previstos, mas também a abrangência do emprego do framework e sua extensibilidade, ao servir como base para a construção de aplicações com características e públicos diferentes.

Através da comparação dos diagramas de classes e dos trechos de códigos de ambos os estudos de caso, percebe-se o nível de reutilização do VirTraM: aplicações bem diferentes com praticamente a mesma estrutura de classes. Percebe-se ainda, o reuso, a padronização do código e a extensibilidade do VirTraM.

Uma dificuldade encontrada na construção das aplicações, em ambos os estudos de caso, foi a exportação do modelo 3D. Mais de uma vez, foi necessário reduzir o tamanho do modelo tridimensional, em polígonos e texturas, para compatibilizá-lo com os dispositivos móveis. As animações da mPrinter também tiveram que ser refeitas algumas vezes para se encontrar o melhor ponto entre o realismo e o tamanho do arquivo gerado. O tratamento da colisão da câmera também foi outro problema encontrado, que implicou reescrita de métodos do VirTraM.

O próximo capítulo apresenta os resultados alcançados durante a elaboração desta dissertação, as conclusões obtidas e as possibilidades para a sua continuação em trabalhos futuros.

CAPÍTULO 6

Conclusões

Neste capítulo são apresentadas as principais conclusões obtidas durante a elaboração desta dissertação. Na Seção 6.1 são mostradas as principais contribuições do trabalho e a Seção 6.2 apresenta algumas possibilidades de trabalhos futuros.

6.1. Resultados Alcançados

Os dispositivos móveis estão se transformando em importantes ferramentas para educação e treinamento. Estes dispositivos proporcionam o surgimento de novas aplicações, que advêm sobretudo da sinergia criada pela combinação da mobilidade com novas tecnologias de armazenamento digital, de transmissão de dados e do seu processamento. Esta combinação vem permitir a consulta e a atualização da informação praticamente em qualquer lugar e qualquer momento.

Esta dissertação envolveu o desenvolvimento de aplicações para dispositivos móveis voltadas para o treinamento utilizando ambientes tridimensionais interativos, oriundos da realidade virtual não imersiva. Inicialmente, o trabalho apresentou os benefícios e os requisitos de aplicações educacionais em dispositivos móveis. A partir dos trabalhos relacionados, foi feito um levantamento de características comuns às soluções móveis voltadas à aprendizagem. Da mesma forma, também foram relacionados requisitos importantes presentes em aplicações de realidade virtual voltadas para a educação.

Em seguida, este trabalho também apresentou benefícios da utilização de recursos de realidade virtual em soluções móveis voltadas à aprendizagem. Levando-se em consideração as características identificadas nos Capítulos 2 e 3, esta dissertação definiu um conjunto de requisitos necessários a aplicações educacionais para dispositivos móveis que utilizam ambientes tridimensionais.

Posteriormente, foi definido um *framework* orientado a objetos para auxiliar os desenvolvedores: o VirTraM. Entre outras vantagens, o *framework* atende os requisitos estabelecidos neste trabalho, padroniza as aplicações construídas a partir dele e proporciona

um aumento na produtividade do desenvolvimento. A idéia principal é permitir que um conjunto de recursos comuns seja reaproveitado para cada novo software criado. O VirTraM apresenta uma arquitetura modular dividida em oito pacotes que atendem às características previstas e facilita o desenvolvimento do treinamento, ao abstrair detalhes específicos de implementação. Esta modularidade possibilita ainda um trabalho em equipe, com profissionais especialistas em diferentes áreas: programação, modelagem e conteúdo didático. O *framework* também incorporou padrões de projetos para solucionar problemas encontrados durante a codificação.

Outro benefício da utilização do VirTraM está na preparação do material do treinamento através da estrutura para o tratamento do conteúdo, descrita na Seção 4.4.6. Esta estrutura possibilita ao conteudista concentrar-se na associação do conteúdo com o ambiente virtual, abstraindo-se dos detalhes da implementação e trabalhando nas informações do treinamento de forma independente. O uso de arquivo XML possibilita que o conteúdo não dependa de um banco de dados específico, permitindo, assim, que o treinamento possa ser utilizado em diversos dispositivos.

O fato de as aplicações construídas a partir do VirTraM poderem ser executadas em telefones celulares também merece destaque, principalmente por estes dispositivos já serem considerados utensílios de uso geral. Isto caracteriza as aplicações como realmente móveis.

Finalmente, para avaliar critérios como facilidade de uso, ganho de produtividade, extensibilidade e verificar se as aplicações desenvolvidas atendiam aos requisitos estabelecidos, dois estudos de caso foram construídos nesta dissertação: o *museum* e a *mPrinter*. Através da comparação de suas implementações, observou-se a reutilização de código proporcionada, o que implicou uma redução do tempo de desenvolvimento. A capacidade de extensibilidade do VirTraM, também pode ser notada, o que permitiu a adaptação do *framework* para aplicações diferentes com o mínimo de re-implementação.

O tempo total para a implementação das duas aplicações foi de aproximadamente quinze dias, com uma única pessoa trabalhando apenas quatro horas por dia. Essa produtividade só foi possível graças à utilização do VirTraM.

6.2. Trabalhos Futuros

A partir da limitação do VirTraM com relação à transmissão de dados, surge a possibilidade de se aprimorar o *framework* através da implementação de classes que facilitem a troca de informações entre as aplicações. Desta forma, planeja-se proporcionar uma aprendizagem em grupo, através da colaboração ou competição entre os usuários e utilizando-se tecnologias de comunicação sem fio. Vários trabalhos demonstram os benefícios da utilização de ambientes virtuais colaborativos na educação (Cadorin & Oliveira, 2004; Benford et al., 2003; Leite Júnior, 2000).

Outro trabalho futuro é a utilização de localização automática através de tecnologias como GPS (Global Positioning System) ou Wi-Fi, semelhante ao projeto desenvolvido por Göth et al. (2004), que utilizam essas técnicas para auxiliar um passeio em um campus universitário. Assim, a navegação do usuário dentro do ambiente virtual dar-se-ia de forma automatizada, à medida que ele se desloca no ambiente real.

Uma outra possibilidade de trabalho futuro é a construção de uma ferramenta de autoria para a elaboração dos treinamentos. Esta incorporaria recursos para manipulação do conteúdo do treinamento, do ambiente 3D e da codificação da aplicação. Com esta ferramenta, o desenvolvimento dos treinamentos dar-se-ia de forma mais rápida, fácil e integrada.

Uma vantagem do VirTraM é a possibilidade de o designer gráfico poder trabalhar com mais de um formato de arquivo 3D. Porém, atualmente só são suportados três formatos. Uma possibilidade para tornar o *framework* mais completo é desenvolver suporte para outros formatos de arquivo, como o VRML (VRML, 2004), por exemplo.

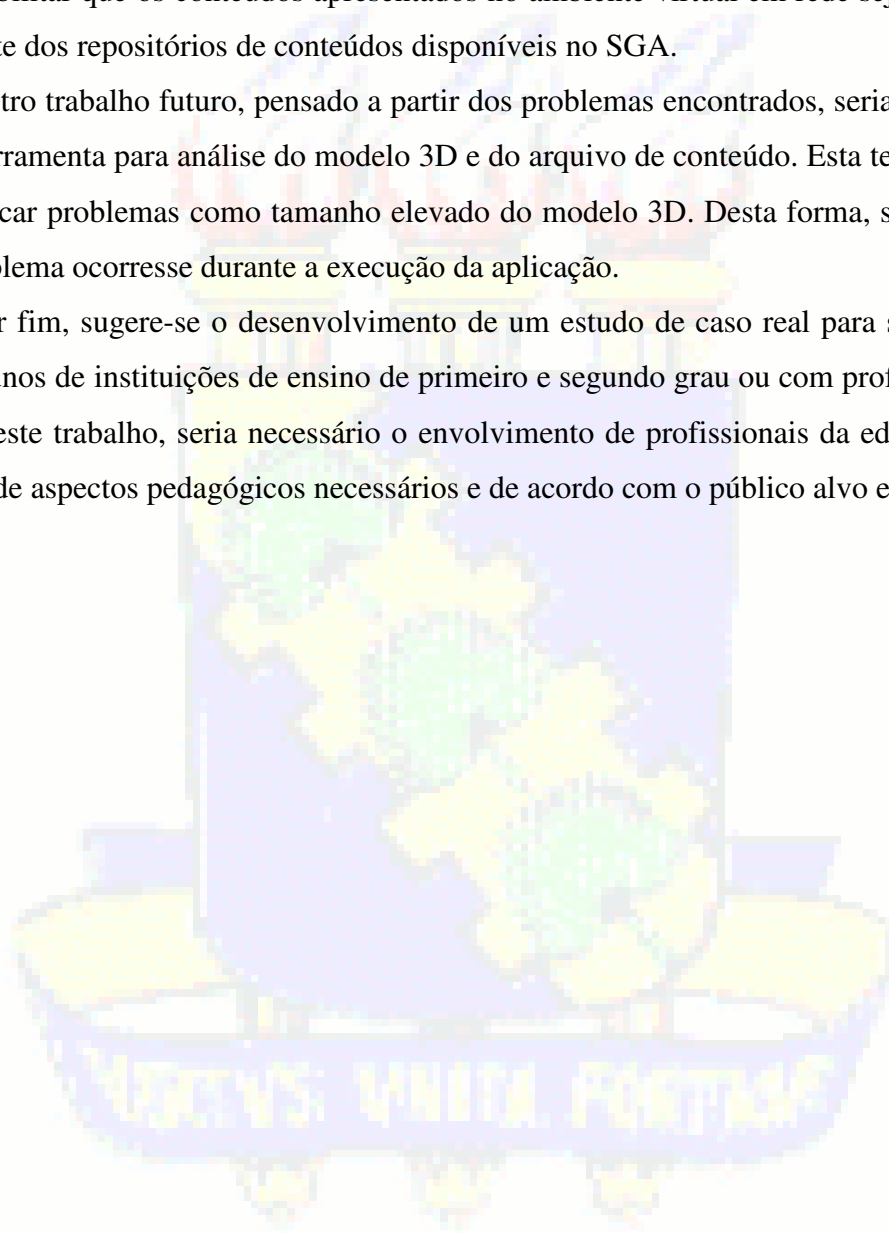
Outro possível trabalho futuro seria a implementação do VirTraM em outras plataformas. A sua arquitetura seria mantida, porém a sua codificação seria feita em outras linguagens para aumentar a quantidade de dispositivos compatíveis. Um exemplo seria converter o framework para o ambiente BREW (BREW, 2004) utilizando C/C++.

As aplicações desenvolvidas a partir do VirTraM são independentes e funcionam isoladamente. Porém, ao se imaginar a aprendizagem como um projeto mais amplo, um possível trabalho futuro seria possibilitar a integração do VirTraM a sistemas de

gerenciamento de aprendizado (SGAs). Estes sistemas conjugam funcionalidades de comunicação mediada por computador e formas de distribuição on-line de materiais didáticos, visando prover alunos e professores de ferramentas que facilitem o ensino e a aprendizagem através da Internet (Almendra, 2003). Assim, é preciso investigar modelos para possibilitar que os conteúdos apresentados no ambiente virtual em rede sejam trazidos diretamente dos repositórios de conteúdos disponíveis no SGA.

Outro trabalho futuro, pensado a partir dos problemas encontrados, seria a definição de uma ferramenta para análise do modelo 3D e do arquivo de conteúdo. Esta teria a função de identificar problemas como tamanho elevado do modelo 3D. Desta forma, seria evitado que o problema ocorresse durante a execução da aplicação.

Por fim, sugere-se o desenvolvimento de um estudo de caso real para ser validado junto a alunos de instituições de ensino de primeiro e segundo grau ou com profissionais de campo. Neste trabalho, seria necessário o envolvimento de profissionais da educação para definição de aspectos pedagógicos necessários e de acordo com o público alvo escolhido.



Referências Bibliográficas

- Adams, L. (1994) “Visualização e realidade virtual”, Ed. Makron Books, pp. 255-259, São Paulo.
- Alberio M. V. e Oliveira J. C. (2004) “ACOnTECe - Ambiente Colaborativo para Treinamento Cirúrgico”, Workshop de Teses e Dissertações do Simpósio Brasileiro de Realidade Virtual (SVR2004), São Paulo, SP, outubro.
- Ahonen, M.; Joyce, B.; Leino, M. e Turunen, H. (2003) “Mobile Learning – A Different Viewpoint”, In Kynäslähti, H e Seppälä, P. (Eds.) Professional Mobile Learning. It Press. Helsinki, p. 91-94.
- Almendra, C. C. (2003) “Administração Remota de Ambientes Virtuais em Rede para Integração com Sistemas de Gerenciamento de Aprendizado”. Dissertação de Mestrado, Mestrado em Ciência da Computação, UFC, Brasil.
- Avellis, G.; Scaramuzzi, A. e Finkelstein, A. (2003) "Evaluating non-functional requirements in mobile learning contents and multimedia educational software", European Conference on Mobile Learning – MLEARN, book of papers, p.13-20.
- Badler N. I. (1997) “Virtual Humans for Animation, Ergonomics, and Simulation”, IEEE Workshop on Non-Rigid and Articulated Motion, Porto Rico, 1997.
- Benford, S.; Anastasi, Rob.; Flintham, M.; Drozd, A.; Crabtree A.; Greenhalgh, C.; Tandavanitj, N.; Adams M. e Row-Farr, J. (2003) “Coping with Uncertainty in a Location – Based Game” IEEE Pervasive Computing journal.
- Bluetooth (2004) “The Official Bluetooth Wireless Info Site”. Disponível em <http://www.bluetooth.com>. Visitado em dezembro.
- Borges, V. M. C. (2001) “Proposta Metodológica para o Ensino de Línguas Estrangeiras no Projeto AVAL - O Emprego da ‘Abordagem com Base em Tarefas’”, In Proceedings of the VII Workshop de Informática na Escola, Fortaleza, CE.
- Bosh, J.; Molin, P.; Mattson, M.; Bengtsson, P. e Fayad, M. (1999) “Framework Problems and Experience. In Fayad, M.E.; Johnson, R.E; Schmidt, D.C. Building Application Frameworks: Object-Oriented Foundations of Framework Design”, John Wiley & Sons, p.55-82.

- Bowman, D.; North, C.; Chen, J.; Polys N.; Pyla P. e Yilmaz U. (2003) "Information-Rich Virtual Environments: Theory, Tools, and Research Agenda", ACM Symposium on Virtual Reality Software and Technology.
- BREW (2004). QUALCOMM BREW. Disponível em http://brew.qualcomm.com/brew/en/about/about_brew.html. Visitado em dezembro de 2004.
- Bricken, M. (1993) "Summer Students in Virtual Reality", Wexelblat, A. (Ed.) Virtual Reality: Applications and Explorations. New York: Academic Press Professional.
- Burdea, G. e Coiffet, P. (1994) "Virtual Reality Technology", John Wiley & Sons, New York, NY.
- Byrne, C. M.; Furness, T. e Winn, W. D. (1995) "The Use of Virtual Reality for Teaching Atomic/Molecular Structure", Annual Meeting of the American Educational Research Association, San Francisco.
- Cadorin, E.A. e Oliveira J.C. (2004) "ACAmPE: um Ambiente Virtual Colaborativo para a Área de Petróleo", Workshop de Teses e Dissertações do Simpósio Brasileiro de Realidade Virtual (SVR2004), São Paulo, SP, Brasil.
- Charitos, D.; Lepouras, G. e Vassilakis, C. (2001) "Designing a virtual museum within a museum", Proceedings of the 2001 Conference on Virtual Reality, Archeology, and Cultural Heritage, p.284-285.
- Chiavenato, I. (1999) "Treinamento e desenvolvimento de recursos humanos: como incrementar talentos na empresa", 4a. edição, Atlas.
- Cooties (2004) "Cooties", . Disponível em <http://www.goknow.com/Products/Cooties.html>. Visitado em dezembro de 2004.
- Corcoran, F.; Demaine, J.; Dicaire, L.; Picard, M. e Taylor J. (2002) "Inuit 3D: An Interactive Virtual 3D Web Exhibition", Museums and the Web Conference.
- Crawford, V. e Vahey, P. (2002) "Palm Education Pioneers Program", Evaluation Report. SRI International, Estados Unidos.
- Dachselt R. (2000) "The challenge to build flexible user interface components for non-immersive 3D environments", Proceedings of the 8th International Conference on Human- Computer Interaction, Lawrence Erlbaum Associates. Mahwah, NJ, USA, pp.1055-1059.

- Dainese, C.; Garbin, T. e Kirner C. (2003) “Sistema de Realidade Aumentada para Desenvolvimento Cognitivo da Criança Surda”, VI Symposium on Virtual Reality, São Paulo, Brasil.
- Danesh A.; Inkpen K.; Lau F.; Shu K. e Booth K. (2001) “Geney: designing a collaborative activity for the Palm handheld computer”, Proceedings of the SIGCHI conference on Human factors in computing systems, p. 388-395.
- Dede, C.; Loftin, R.B. e Regian, J. W. (1994) “The Design of Artificial Realities to Improve Learning Newtonian Mechanics”, Proceeding of the 1994 East-West International Conference on Multimedia, Hipermedia and Virtual Reality. Moscou, 14 a 16 de setembro.
- Ellis, S.R. (1991) “Nature and origin of virtual environments: a bibliographic essay”, Computing System in Engineering, 2(4).
- Erickson, T. (1993) “Artificial Realities as Data Visualization Environments”, Wexelblat, A. (Ed.) Virtual Reality: Applications and Explorations. New York: Academic Press Professional, p.1-22.
- Exploratorium (2004) “Exploratorium: the museum of science, art and human perception”. Disponível em <http://www.exploratorium.edu/index.html>. Visitado em dezembro.
- Fayad, M.; Schmidt, C. (1999) “Building Application Frameworks: Object Oriented Foundations of Framework Design”, Wiley Computer Publishing.
- Feiner, S.; Macintyre B.; Hollerer T. e Webster A. (1997). A Touring Machine: Prototyping 3d Mobile And Augmented Reality Systems For Exploring The Urban Environment. In Proceedings Of The First International Symposium On Wearable Computers – ISWC 1997.
- Ferreira L.; Tarouco L. e Becker F. (2004) “Fazer e compreender na Realidade Virtual: em busca de alternativas para o sujeito da aprendizagem”, RENOTE - Revista Novas Tecnologias na Educação, Porto Alegre.
- Ferro, E. e Paiva A. (2003) “Modelagem e Implementação de um Sistema para Treinamento de Técnicos em Radiologia Baseado em Realidade Virtual não Imersiva”, Symposium on Virtual Reality, São Paulo, Brasil.
- Fraser, M.; Glover, T.; Vaghi, I.; Benford, S.; Greenhalgh, C.; Hindmarsh, J. e Heath C. (2000) “Revealing the Realities of Collaborative Virtual Reality. Proceedings of

- the Third International Conference on Collaborative Virtual Environments”, Nova York: ACM Press.
- Gamma, E. R.; Johnson, R. e Vlissides J. (1995) “Design Patterns: Elements of Reusable Object-Oriented Software. Reading”, MA: Addison Wesley.
- Geller, M. (2004) “Educação a Distância e Estilos Cognitivos: construindo um novo olhar sobre os ambientes virtuais” Porto Alegre: UFRGS/PGIE, Tese de doutorado.
- Góes, V.; Zuffo, M. e Lopes R. (2003) “FADAS: Um Sistema Para Criação e Manipulação de Mundos Virtuais em Ambiente Imersivo Tridimensional”, Symposium on Virtual Reality, São Paulo, Brasil.
- Goh T. e Kinshuk, M. (2004). “Getting Ready For Mobile Learning”, In L. Cantoni & C. McLoughlin (Eds.) Proceedings of ED-MEDIA 2004 - World Conference on Educational Multimedia, Hypermedia & Telecommunications.
- Göth, C.; Häss,U.-P. e Schwabe, G. (2004) "Requirements for mobile learning games shown on a mobile game prototype", Proceedings MLearn 2004 conference, Rome, Italy .
- Hancock, D. (1995) “Viewpoint: virtual reality in search of middle ground”, IEEE Spectrum, 32.
- Hannemann J. e Kiczales G. (2002) “Design pattern implementation in Java and aspectJ”, Proceedings of the 17th ACM SIGPLAN Conference on Object-oriented Programming, Systems, languages and applications, Vol. 37 Issue 11.
- Hamit, F. (1993) “Realidade Virtual e a Exploração do Espaço Cibernético”, Berkeley Brasil Editora, Original Sams Publishing.
- Hsi, S. (2002) “The Electronic Guidebook: A Study of User Experiences using Mobile Web Content in a Museum Setting”, IEEE International Workshop On Wireless And Mobile Technologies In Education (WMTE).
- Hull, C. (1943) “Principles of Behavior”, Appleton-Century-Crofts.
- Java (2004) “Java Technology”. Disponível em <http://www.java.sun.com/>. Visitado em dezembro.
- Java 3D (2004) “Java 3D API”. Disponível em <http://java.sun.com/products/java-media/3D/>. Visitado em dezembro.
- Jacobson, L. (1991) “Virtual Reality: A Status Report”, AI Expert, pp. 26-33.

- Johnson, E. (1993) “How to Design Framework. Notes for Proceedings of the Conference on object-oriented programming system, languages, and applications”, OOPSLA93, University of Illinois.
- Johnson, W. e Rickel, J. (1998) “Steve: An animated pedagogical agent for procedural training in virtual environments”, SIGART Bulletin 8: 16-21.
- Kirner, C. (2004) “Sistemas de Realidade Virtual”. Disponível em <http://www.dcc.ufscar.br/~grv>. Visitado em dezembro.
- Krueger, M.W. (1991) “Artificial Reality II”, Addison-Wesley, Reading, MA.
- Kynaslahti, H.; Sariola, J. e Seppala, P.(2002) “Mobile learning in personnel training of university teachers Wireless and Mobile Technologies in Education”, *Proceedings IEEE International Workshop*.
- Lehner, F. e Nösekabel, H. (2002) “The Role Of Mobile Devices In E-Learning — First Experiences With A Wireless E-Learning Environment”, IEEE International Workshop On Wireless And Mobile Technologies In Education (WMTE), p.103-106.
- Leite Júnior, A.J.M. (2000) “Ataxia: uma arquitetura para a viabilização de NVE's voltados para a educação a distância através da Internet”. Dissertação de Mestrado, Mestrado em Ciência da Computação, UFC, Brasil.
- Leston, J. (1996) “Virtual reality: the it perspective”, Computer Bulletin, pp. 12-13 .
- Lipson, H., Shpitalni, M., Kimura, F. e Goncharenko, I. (1998) "On-line Product Maintenance by Web-Based Augmented Reality", New Tools and Workflows for Product Development, pp. 131-143, Berlin.
- Luchini, K.; Quintana C. e Soloway E. (2004) “Design Guidelines For Learner-Centered Handheld Tools”, Proceedings Of The 2004 Conference On Human Factors In Computing Systems, p.135-142.
- Magalhães, K.; Viana, Windson; Lemos, F.; Castro, J. e Andrade, R. (2004) “Um Framework de Persistência de Objetos em Aplicações para Dispositivos Móveis”. Simpósio Brasileiro de Banco de Dados - SBBD 2004. Brasília-DF, Brasil, Outubro.
- Maia, G.; Cavalcante J. N. e Vidal C. (2003) “CRAbGE: Um Motor Gráfico Customizável, Expansível e Portável Para Aplicações de Realidade Virtual”, VI Symposium on Virtual Reality, São Paulo, Brasil.

- Mandryk, L.; Inkpen, K.; Bilezikjian, M.; Klemmer, S e Landay, J. (2001). “Supporting children's collaboration across handheld computers”, Conference on Human Factors in Computing Systems (CHI). April 2001, Seattle, USA.
- Manning, A. e Glenda, S. (2004) “The Blanton iTour - An Interactive Handheld Museum Guide Experiment”, Conference Museums and the Web.
- Markiewicz, M e Lucena, C. (2000) “Understanding Object-Oriented Framework Engineering”, PUC-Rio Inf.MCC38/00.
- Marra, J. P. (2000) "Administração de recursos humanos: do operacional ao estratégico" 3a edição, Futura.
- Marson, F.; Jung, C. e Musse, S. (2003) "Modelagem Procedural de Cidades Virtuais", Symposium on Virtual Reality, p.103-116.
- Mattsson, M. (1996) "Object-oriented Frameworks: A Survey of Methodological Issues", Licentiate Thesis. Department of Computer Science, Lund University. Sweden.
- Max (2004) “Autodesk 3ds Max”. Disponível em <http://usa.autodesk.com/adsk/servlet/index?id=5659302&siteID=123112>. Visitado em dezembro.
- McCarty, W. D.; Sheasby, S.; Amburn, P.; Stytz M. e Switzer, C. (1994) “A virtual cockpit for a distributed interactive simulation”, IEEE Computer Graphics and Application, pp. 49-54.
- Meirelles, L.; Tarouco, L. e Alves C. (2004) “Telemática Aplicada à Aprendizagem com Mobilidade”, RENOTE - Revista Novas Tecnologias na Educação, Porto Alegre.
- Montavista Linux (2004) “MontaVista Software - Platform to Innovate”. Disponível em <http://www.mvista.com/>. Visitado em dezembro de 2004.
- Myers, B. e Beigl, M. (2003) “Handheld Computing”, IEEE Computer, 36, p.27-29.
- MySQL (2004) “MySQL AB : The World's Most Popular Open Source Database”. Disponível em <http://www.mysql.com/>. Visitado em dezembro.
- Najork, M. e Brown, M. (1995) “Obliq-3D: A High-Level, Fast-Turnaround 3D Animation System”, IEEE Transactions on Visualization and Computer Graphics, p.p 175-193.

- Nedel, L.; Freitas, C.; Schyn, A; Navarre D. e Palanque, P. (2003) Usando Modelagem Formal para Especificar Interação em Ambientes Virtuais: Por quê? Symposium on Virtual Reality, São Paulo, Brasil.
- Nyiri, K. (2002) "Towards a philosophy of m-Learning". Proceedings of WMTE Conference, p.121-124.
- Oliveira, J. A. (2004) "Do tradicional ao virtual: o museu diante da teoria sistêmica de Niklas Luhmann. O problema da informação do objeto.", II IBÉRICO.
- Oliveira J. C.; Hosseini M.; Shirmohammadi S.; Cordea M.; Petriu E.; Petriu D. e Georganas N. D. (2000) "VIRTUAL THEATER for Industrial Training: A Collaborative Virtual Environment", Proc. 4th WORLD MULTICONFERENCE on Circuits, Systems, Communications & Computers (CSCC 2000), ISBN: 960-8052-18-1, 294-299, Vouliagmeni, Greece, July 2000.
- Oosterholt, R.; Kusano, M. e De Vries G. (1996) "Interaction design and human factors support in the development of a personal communicator for children", Proceedings of the SIGCHI conference on Human factors in computing systems, p. 450-457.
- OTA (2004) "Over The Air Initiated Provisioning Recommended Praticce". Disponível em <http://java.sun.com/products/midp/OTAProvisioning-1.0.pdf>. Visitado em dezembro.
- Owen, J. V. (1995). "Simulation: art and science, Manufacturing Engineering", pp. 61-63.
- Palm (2004). "Palm Software and Palm OS". Disponível em <http://www.palmsource.com>". Visitado em dezembro.
- Palmgear (2004) "PalmGear.com - The source for your Palm Powered world!". Disponível em <http://www.palmgear.com/>". Visitado em dezembro.
- Piaget, J. (1975) "L'equilibration des structures cognitives", Paris:P-U-F.
- Pree, W. (1999) "Hot-spot Driven Development. In Fayad, M.E.; Johnson,R.E; Schmidt, D.C. Building Application Frameworks: Object-Oriented Foundations of Framework Design", John Wiley & Sons. p. 379-393.
- Quintana, C.; Krajcik, J. e Soloway, E. (2003). "Issues and Approaches for Developing Learner-Centered Technology". Advances in Computers, 57.

- Ratto, M.; Shapiro R. B.; Truong T. M. e Griswold W. G. (2003) “The ActiveClass Project: Experiments in Encouraging Classroom Participation”, Computer Support for Collaborative Learning 2003, Kluwer, June.
- Reiners, D.; Stricker S.; Klinker G. e Müller S.. (1999). “Augmented Reality for Construction Tasks: Doorlock Assembly Placing Artifical Objects in Real Scenes”.
- Ressler, S. (1997). “Virtual reality for manufacturing - case studies”, National Institute of Standards and Technology.
- Robertson, G.; Card, S. e Mackinlay, J.(1993). “Nonimmersive virtual reality”, IEEE Computer, Feb. 1993.
- Rose, H. e Billinghamurst, M. (1995) “Zengo Sayu: An Immersive Educational Environment for Learning Japanese”, HITL Technical Report, Seattle, WA.
- Roussou, M. (2004) "Learning by doing and learning through play: an exploration of interactivity in virtual environments for children", Computers in Entertainment (CIE), Volume 2 Issue 1, p.10.
- Rumbaugh, J.; Jacobson, I. e Booch, G. (1998) “The Unified Modeling Language Reference Manual”. Addison-Wesley.
- Sandberg, D. W. (1988) “Smalltalk and exploratory programming”, ACM SIGPLAN Notices, Volume 23 Issue 10.
- Sauer, S. e Gobel, S. (2003) “Focus your young visitors: Kids Innovation Fundamental changes in digital edutainment”, Conference Museums and the Web.
- Schmid, H. A. (1999) “Framework Design by Systematic Generalization. In Fayad, M.E.; Johnson,R.E; Schmidt, D.C. Building Application Frameworks: Object-Oriented Foundations of Framework Design”, John Wiley & Sons.
- Schrock, S. (1996) “A Brief History of Instructional Development From Anglin”, G. (Ed.) (1995) Instructional Technology: Past Present and Future. Englewood, CO: Libraries Unlimited. Chap 2 Pp 11 - 19.
- Sharples, M.; Colett, D. e Westmancott, O. (2002) “The Design and Implementation of a Mobile Learning Resource”, ACM Personal and Ubiquitous Computing , Volume 6, Issue 4, pp. 220-234, May.
- Shaw, C. (1993) “Decoupled Simulation in Virtual Reality with MR Toolkit”, ACM Transaction on Information Systems, 11(3);287-317.

- Shields, J. e Poftak, A. (2002). “A Report Card on Handheld Computing”. Disponível em http://www.techlearning.com/db_area/archives/TL/2002/02/handheld.html . Visitado em dezembro de 2004.
- Skinner, F. (1968). “The Technology of Teaching”, Appleton-Century-Crofts.
- Srujan, R. P. (2002) “Developing And Implementing Wireless Personal Digital Assistant (PDA) Applications For Training And Education”, Master's Thesis, University of Oklahoma, EUA.
- State, A.; Livingston M.; Garrett W.; Hirota G.; Whitton M.; Pisano E. e Fuchs E. (1996). Technologies For Augmented Reality Systems: Realizing Ultrasound-Guided Needle Biopsies. In Proc. Siggraph'96, Páginas 439–446.
- Steed P. (1998) “The Art of Low Polygon Modeling”, *Game Developer*, Junho.
- Stytz, M. (1996) “Distributed Virtual Environments”, IEEE Computer Graphics and Applications, Vol. 16, No. 33, pp. 19-31.
- Sukimoto, M. (2003) “How Sensing and Mobile Technologies can enhance collaborative learning in classrooms and museums”, Keynote speech at CSCL2003 Bergen, Noruega.
- Sun (2004) “Java 2 Platform, Micro Edition (J2ME)”. Disponível em <http://java.sun.com/j2me/index.jsp>,. Visitado em dezembro.
- Sutherland, I. (1968). A Head-Mounted Three Dimensional Display. In Proc. Fjcc 1968, Pages 757–764, Washington, DC. Thompson Books.
- Symbian (2004) “Symbian OS - the mobile operating system”. Disponível em <http://www.symbian.com/>. Visitado em dezembro.
- Syvänen, A.; Ahonen, M.; Jäppinen, A.; Pehkonen, M. e Vainio, T. (2003) “Accessibility And Mobile Learning”, Proceedings Of Ifip Etrain Conference In Pori, Finland.
- Tellis, C. e Proctor, N. (2003) “The State of the Art in Museum Handhelds in 2003”, Conference Museums and the Web.
- Trifonova A. e Ronchetti M. (2004) “A General Architecture to Support Mobility in Learning”, Proc. of the 4th IEEE International Conference on Advanced Learning Technologies (ICALT 2004 - "Crafting Learning within Context"), August 30 - September 1, Joensuu, Finland.

- Vidal, C.; Santos, E.; Melo, A.; Almendra, C. e Borges, V. (2003) “Collaborative Virtual Environments for Language Learning”, VI Symposium on Virtual Reality, São Paulo, Brasil.
- Vidal, C.; Gomes, G.; Mendonça, G.; Gomes, H. e Cavalcante, J. (2004) “Uma Ferramenta de Autoria de Ambientes Virtuais Adaptável a Diferentes Motores Gráficos”, VII Symposium on Virtual Reality, São Paulo, Brasil.
- VRML (2004) “VRML Virtual Reality Modeling Language”. Disponível em <http://www.w3.org/MarkUp/VRML/>. Visitado em dezembro.
- Wang, H. e Chee, Y. S. (2001). Supporting workspace awareness in distance learning environments: Issues and experiences in the development of a collaborative learning system. Proceedings of ICCE/SchoolNet 2001-Ninth International Conference on Computers in Education, pp. 1109-1116. Seoul, South Korea.
- WAP Forum (2004), "WAP Forum" . Disponível em <http://www.wapforum.org>. Visitado em dezembro.
- Wazlawick, S. (2001) “Collaborative Learning In A Virtual Museum Authoring Tool”, Symposium on Virtual Reality.
- Windows Mobile (2004) “Mobile Solutions, Mobile Applications, and Handheld Devices from Microsoft Windows Mobile”. Disponível em <http://www.microsoft.com/windowsmobile>. Visitado em dezembro.
- Wojciechowski, R.; Walczak K.; White M. e Cellary W. (2004) “Production: Building Virtual And Augmented Reality Museum Exhibitions”, Proceedings Of The Ninth International Conference On 3D Web Technology, p.135-144.
- Yang, J. S. (2002). “Lightweight Thin Clients Let Workers Roam”, PC Magazine, v21, novembro.

ANEXO A

Diagramas de Classes Completos dos Pacotes do VirTraM

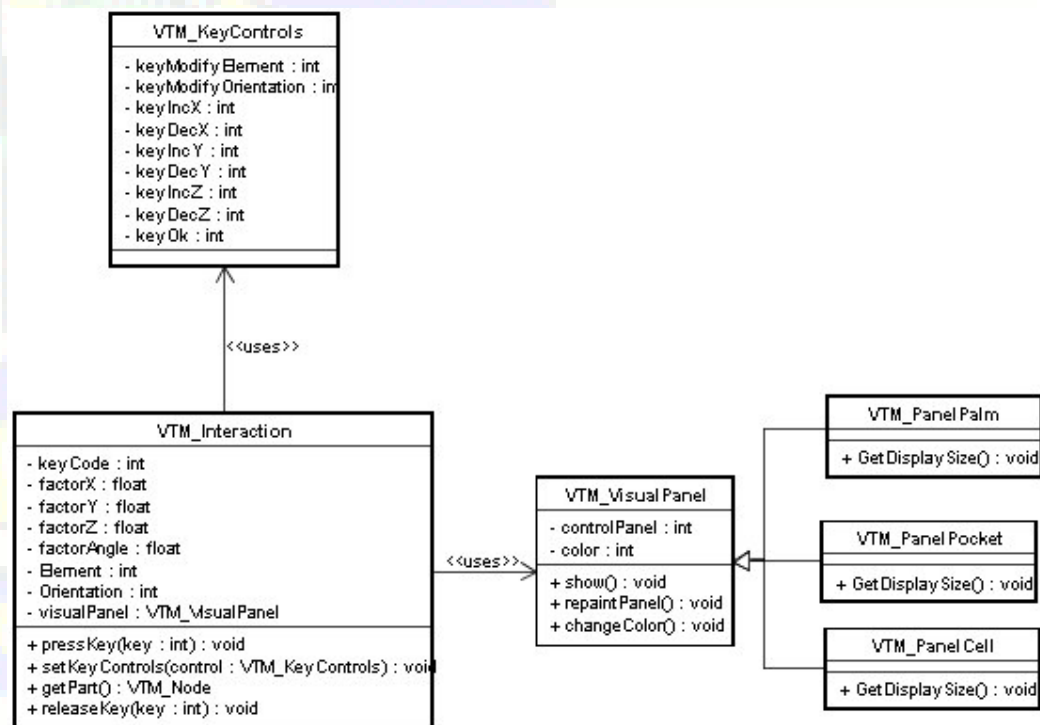


Figura A-1: Diagrama de Classes – Pacote *Interaction*.

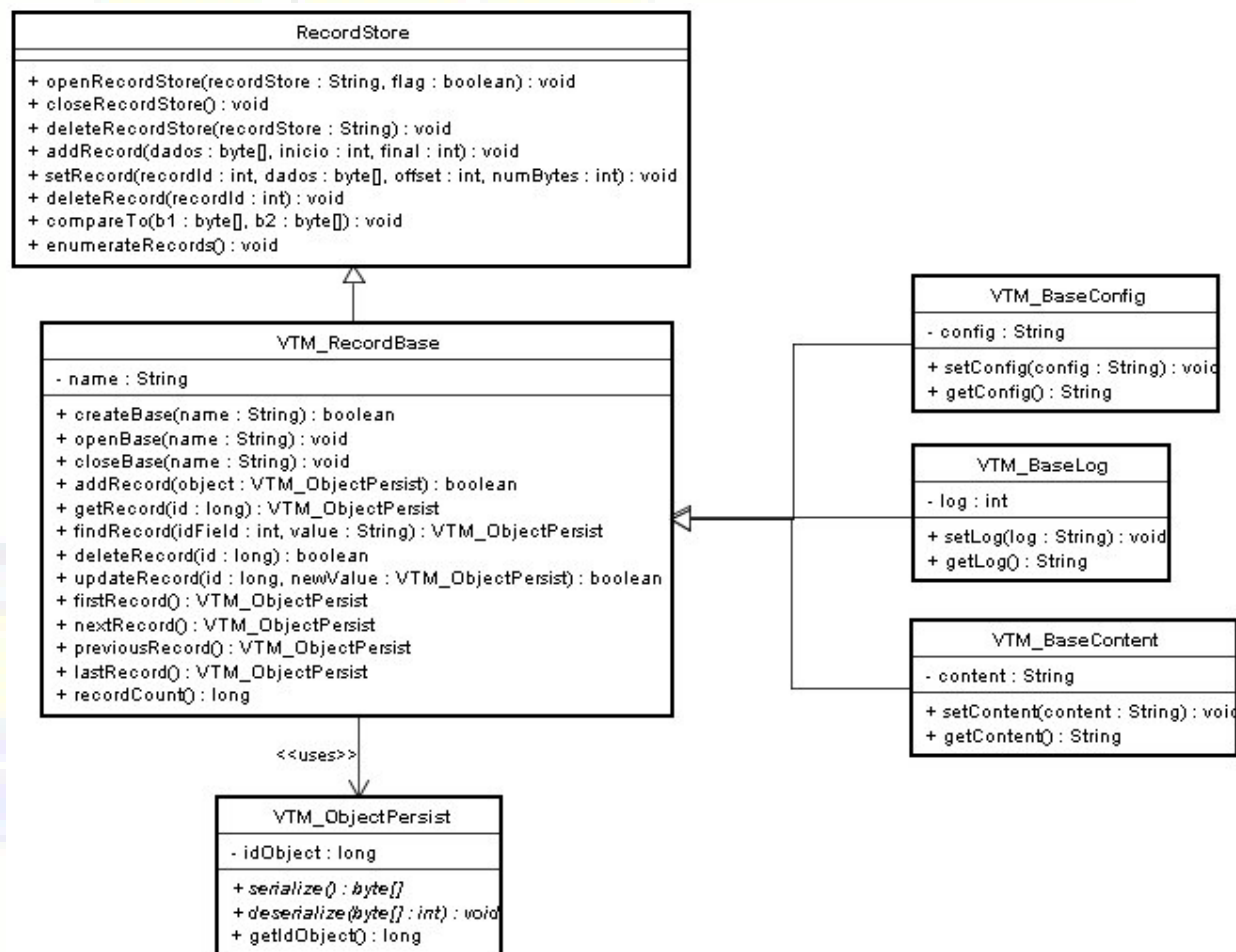


Figura A-2: Diagrama de Classes – Pacote *Persistence*.

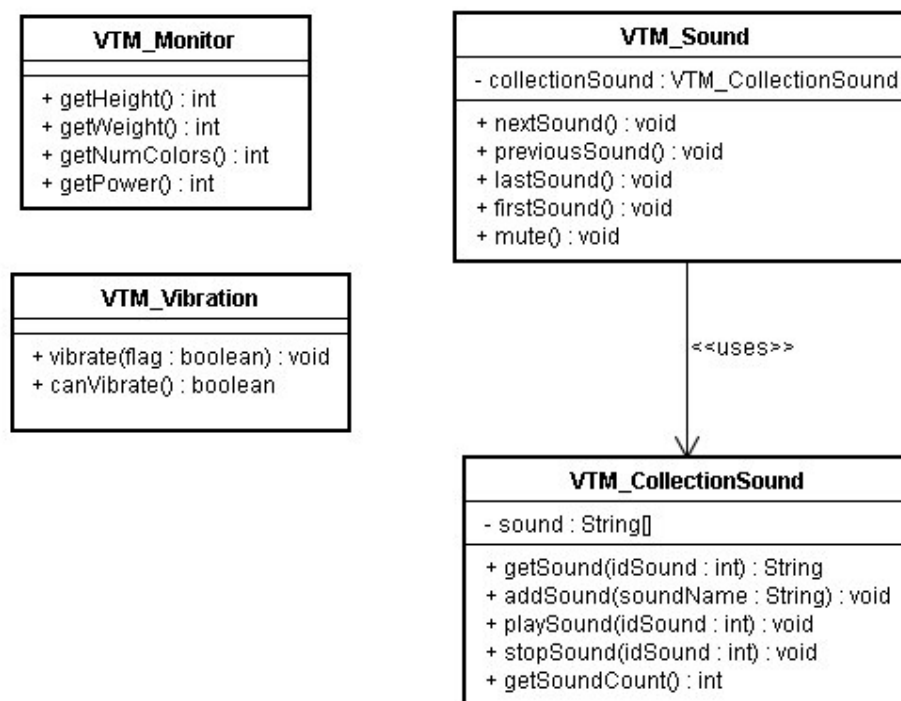


Figura A-3: Diagrama de Classes – Pacote *Optional*.

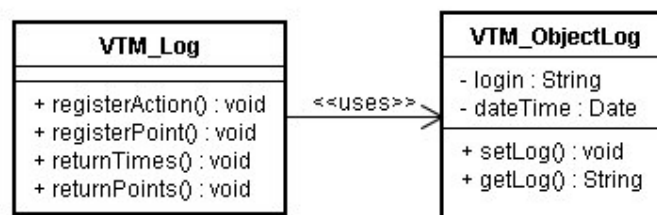


Figura A-4: Diagrama de Classes – Pacote *Log*.

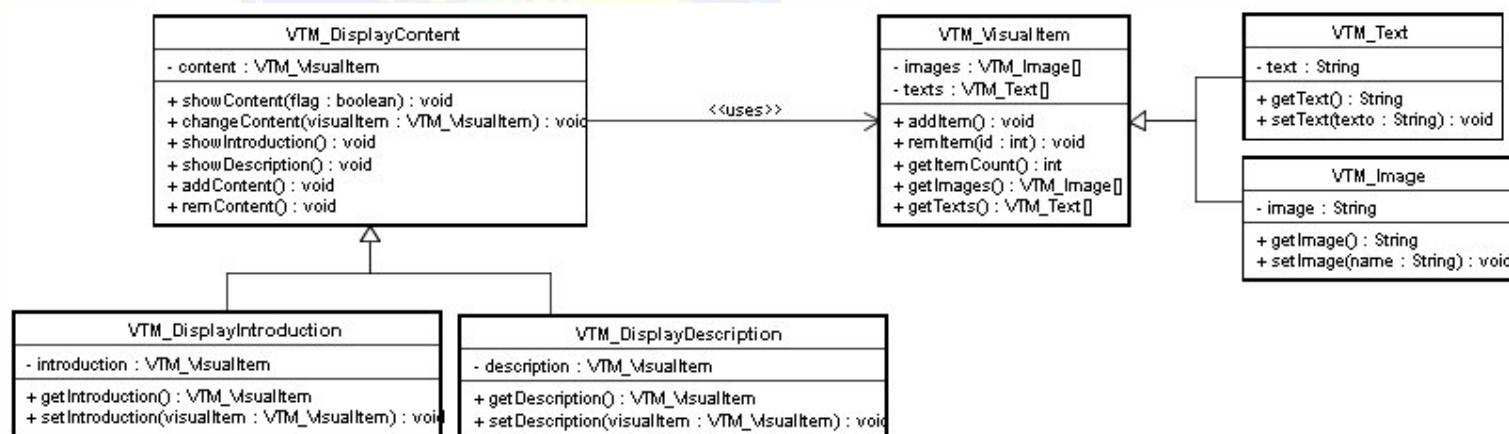


Figura A-5: Diagrama de Classes – Pacote *Visual*.

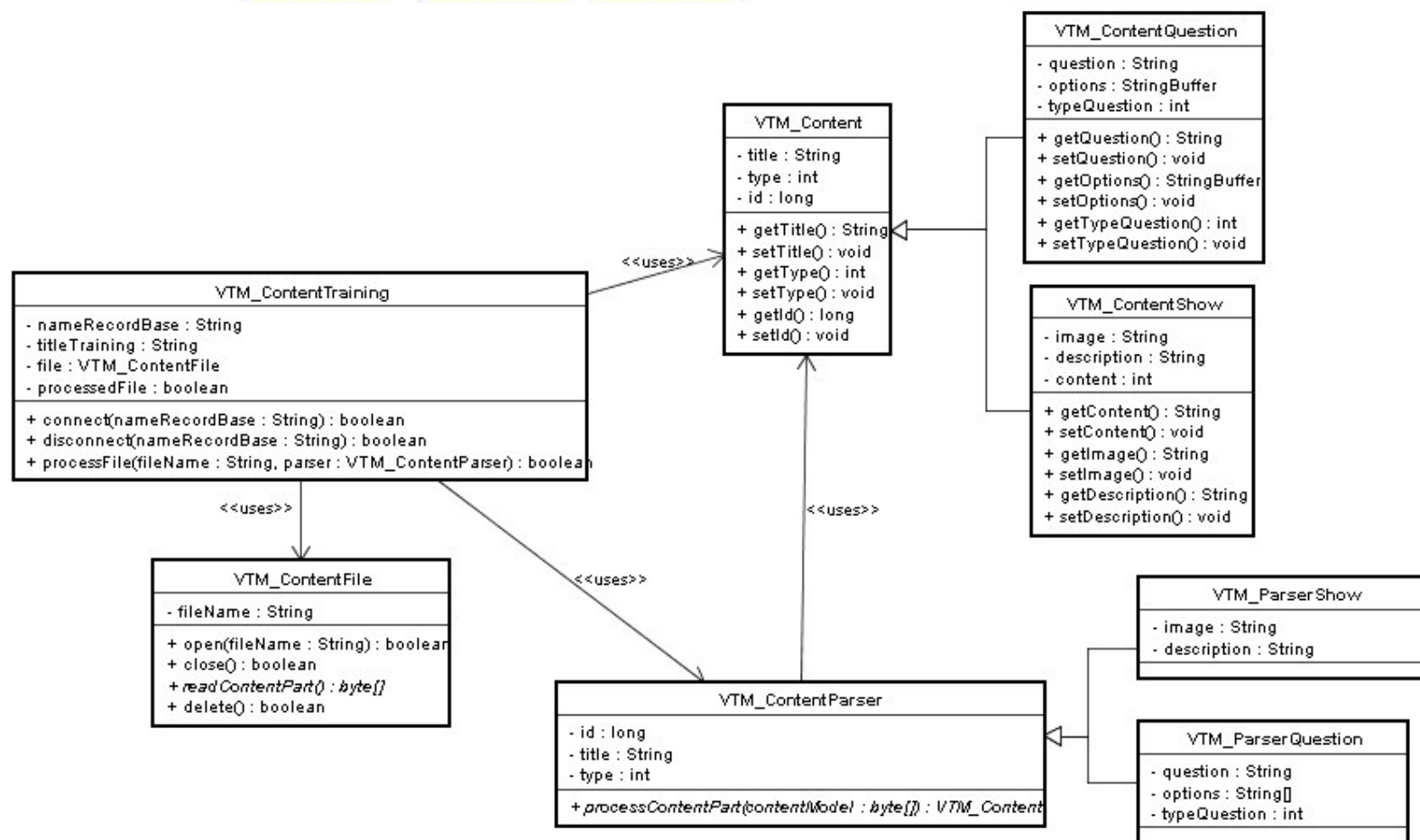


Figura A-6: Diagrama de Classes – Pacote *Content*.

Anexo A – Diagramas de Classes Completos dos Pacotes do VirTraM

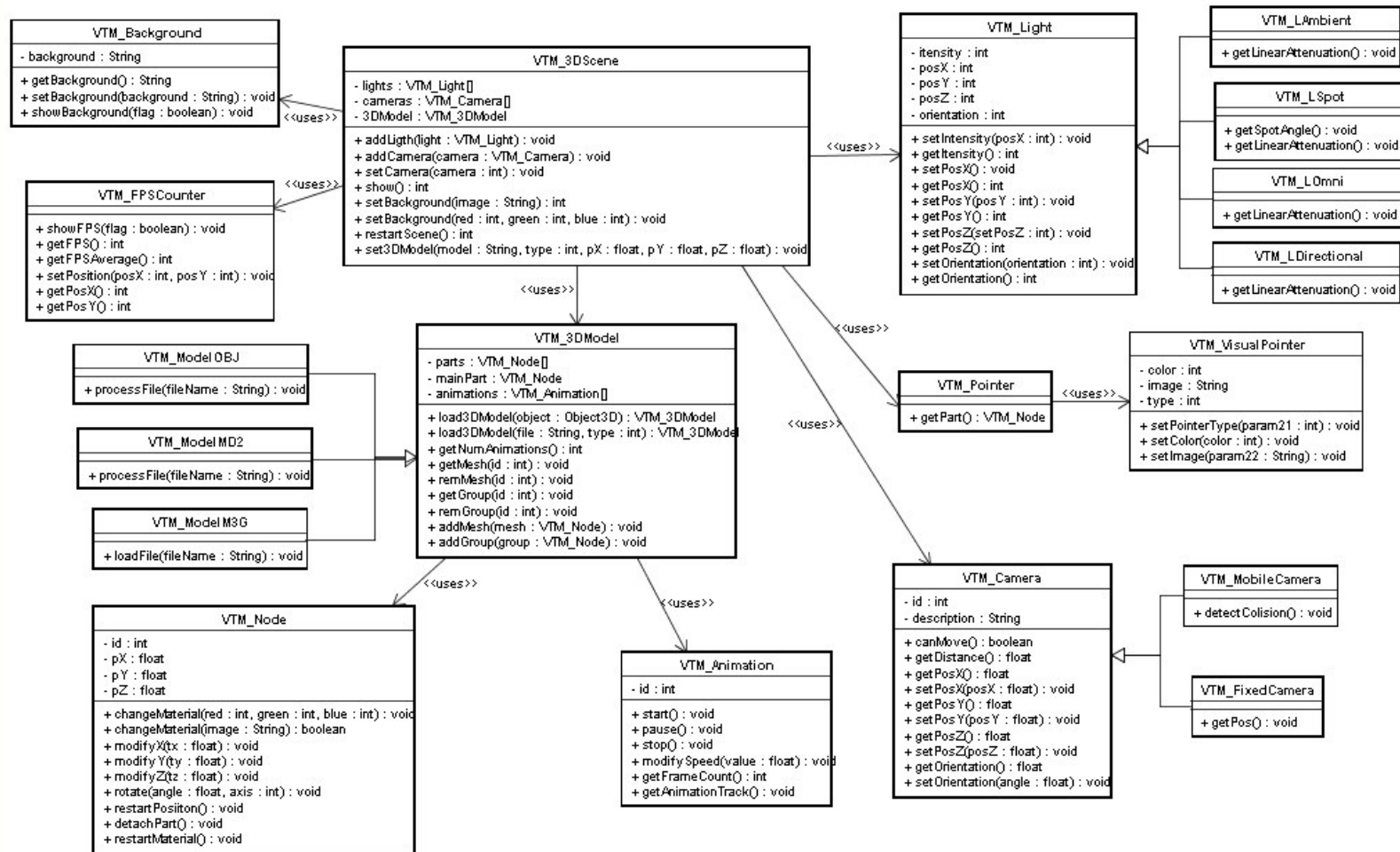


Figura A-7: Diagrama de Classes – Pacote 3D.

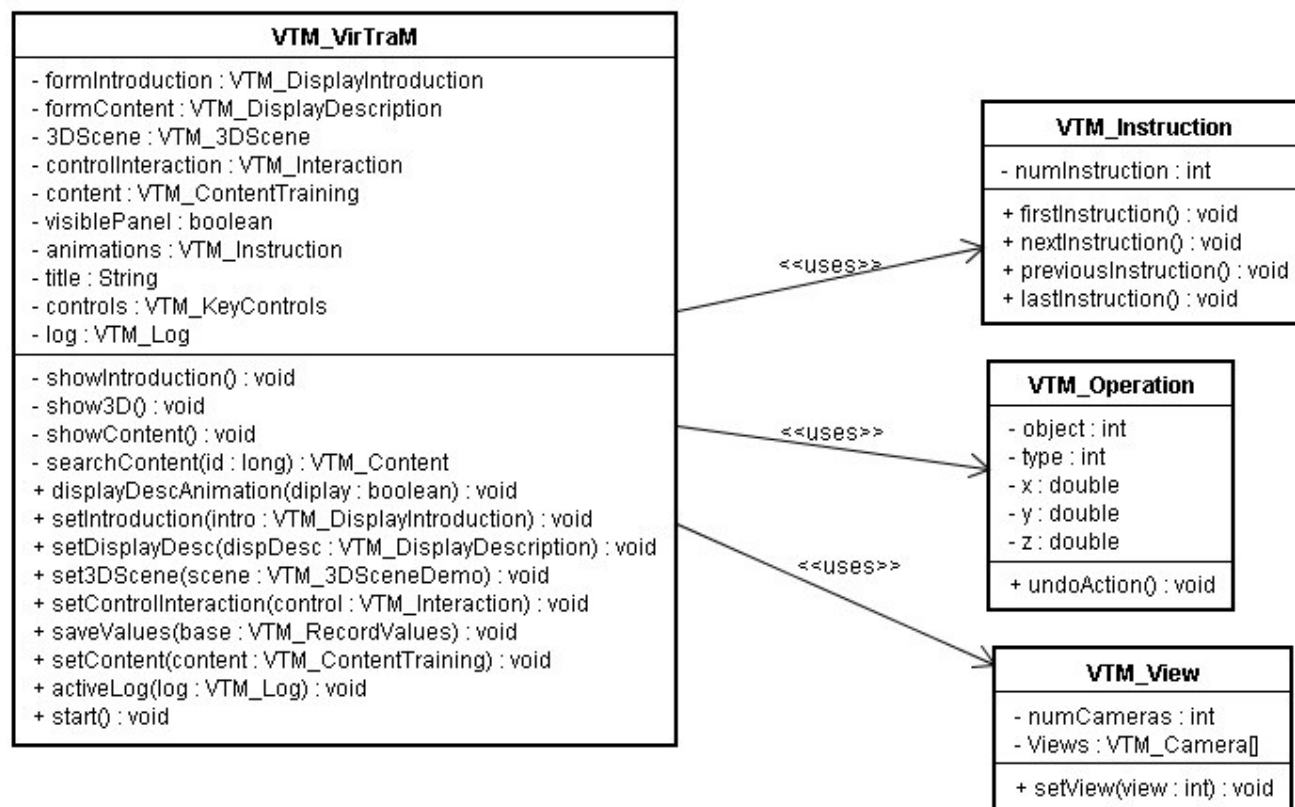


Figura A-8: Diagrama de Classes – Pacote *HiLevel*.