

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO – BACHARELADO**

**PROTÓTIPO DE UM VISUALIZADOR DE MODELOS 3D**  
**PARA DISPOSITIVOS MÓVEIS UTILIZANDO A**  
**PLATAFORMA .NET CF 2.0**

**MARCOS DELL' ANTONIO DE SOUZA**

**BLUMENAU**  
**2006**

**MARCOS DELL' ANTONIO DE SOUZA**

**PROTÓTIPO DE UM VISUALIZADOR DE MODELOS 3D  
PARA DISPOSITIVOS MÓVEIS UTILIZANDO A  
PLATAFORMA .NET CF 2.0**

Proposta de Trabalho de Conclusão de Curso submetida à Universidade Regional de Blumenau para a obtenção dos créditos na disciplina Trabalho de Conclusão de Curso I do curso de Ciências da Computação — Bacharelado.

Prof. Paulo César Rodacki Gomes – Orientador

**BLUMENAU  
2006**

## 1 INTRODUÇÃO

A representação digital de modelos tridimensionais (3D), definidos por Santee (2005, p. 150) como corpos formados por um conjunto de vértices, torna-se mais comum à medida que aumenta a capacidade de processamento dos computadores. Segundo Moore (2005), esta capacidade tende a dobrar a cada dezoito meses.

Essa evolução aplica-se também a dispositivos móveis, tais como celulares e *Personal Digital Assistants* (PDA). Segundo Pamplona (2005, p. 16), outras mudanças acontecem paralelamente, tais como a incorporação de uma tela sensível ao toque (*touch screen*) com resolução maior e colorida e o uso de sistemas operacionais juntamente com linguagens de programação para gerenciar e acessar os recursos dos dispositivos.

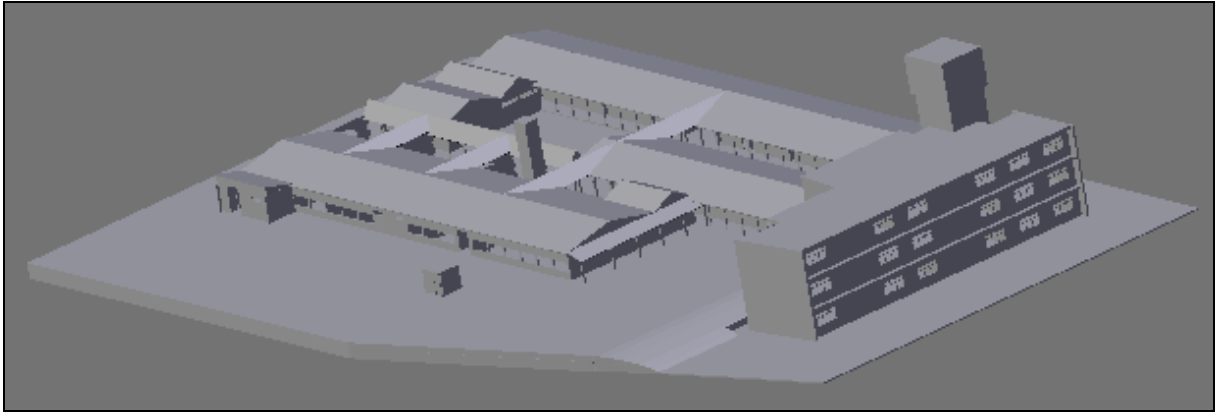
Quanto ao desenvolvimento, a necessidade de padronização e simplicidade molda um futuro onde toda e qualquer espécie de código será gerenciada por um *framework*<sup>1</sup>. Os benefícios que acompanham essa tendência são vários: portabilidade, segurança, padronização, reusabilidade, etc. O sistema operacional para dispositivos móveis *Windows Mobile 5* (MICROSOFT, 2005c), por exemplo, suporta o *.NET Compact Framework 2.0*<sup>2</sup> (.NET CF 2.0) (MICROSOFT, 2005a). Levando em consideração que alguns celulares e PDAs rodam este sistema operacional, uma aplicação escrita para um modelo de dispositivo rodará no outro.

Tendo em vista esses recursos oferecidos pelos dispositivos móveis em questão, surgiu a idéia de explorá-los através de uma aplicação para a visualização de modelos 3D. Em outras palavras, o desafio proposto neste trabalho é criar um protótipo de um visualizador de modelos 3D para celulares e PDAs compatíveis com o .NET CF 2.0. Será utilizado um modelo do Campus IV da Universidade Regional de Blumenau (FURB) (vide figura 1) como base para a construção da aplicação. Como este modelo é específico para computadores, o presente trabalho também compreende um estudo para torná-lo compatível com as dimensões dos dispositivos e com a biblioteca que será empregada no desenvolvimento. No que diz respeito à biblioteca, duas serão avaliadas: OpenGL for Embedded Systems (OpenGL ES) (KHRONOS GROUP, 2004b) e Direct3D Mobile (MD3DM) (MICROSOFT, 2005d).

---

<sup>1</sup> *Framework* é um conjunto de classes/rotinas pré-definidas que são reutilizáveis em vários projetos.

<sup>2</sup> O *.NET Compact Framework* é a versão compacta para dispositivos móveis do *Framework .NET 2.0* da Microsoft.



Fonte: Bruns (2003, p. 34)

Figura 1 - Modelo 3D do Campus IV

Parte da aplicação a ser desenvolvida compreende a navegação pelo ambiente do Campus utilizando um sistema de câmeras em primeira pessoa e girando conforme o usuário desejar. A aplicação também deverá detectar colisões no decorrer do trajeto para delimitar a trajetória do usuário.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver um protótipo de um visualizador de modelos 3D para dispositivos móveis (celulares e PDAs) que permita ao usuário navegar pelo ambiente visualizado.

Os objetivos específicos do trabalho são:

- a) verificar se os dispositivos móveis atuais possuem características suficientes para rodar aplicações gráficas em três dimensões;
- b) validar o uso de uma biblioteca gráfica (OpenGL ES ou MD3DM) para dispositivos móveis no que diz respeito ao desempenho utilizando a métrica de *frames* por segundo;
- c) desenvolver a aplicação utilizando o .NET CF 2.0.

## 1.2 RELEVÂNCIA DO TRABALHO

Segundo Pamplona (2005, p. 16), mobilidade é a palavra do futuro. Assim como os

outros equipamentos, celulares e PDAs estão conquistando o espaço no mercado e tornando-se indispensáveis ao homem.

Grandes empresas enxergam nesta área uma excelente oportunidade de negócios. A Microsoft, por exemplo, lançou no ano passado uma versão móvel e gerenciada da sua biblioteca para o desenvolvimento de aplicações gráficas em três dimensões, o MD3DM. Além desta, lançou recentemente um *framework* para o desenvolvimento de jogos chamado XNA (MICROSOFT, 2006). O seu objetivo é tornar o desenvolvimento de jogos totalmente portátil entre o sistema operacional Microsoft Windows e os *consoles*<sup>3</sup> Xbox 360 (MICROSOFT, 2005e). Estas informações, mais uma vez, confirmam a tendência de que as futuras aplicações serão escritas utilizando código gerenciado e *frameworks* reutilizáveis.

Tendo em vista que a área de desenvolvimento para dispositivos móveis é algo extremamente novo, muitas pesquisas e estudos ainda são necessários para que ela chegue à sua maturidade, sendo este um dos objetivos deste trabalho: pesquisar e validar alguns equipamentos, bibliotecas e demais variáveis presentes no desenvolvimento de aplicações móveis focadas no processamento gráfico tridimensional.

### 1.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- a) levantamento bibliográfico: pesquisar e consultar os documentos acessíveis sobre desenvolvimento 3D, bibliotecas gráficas, .NET CF 2.0 e editores gráficos 3D;
- b) escolha da biblioteca gráfica: através de testes práticos, escolher uma biblioteca gráfica (OpenGL ES ou MD3DM) que será empregada durante o desenvolvimento da aplicação;
- c) reavaliação de requisitos: analisar e reavaliar os requisitos propostos com base no levantamento bibliográfico e na biblioteca escolhida. Nesta etapa, de acordo com as informações encontradas, alguns requisitos serão adicionados ou removidos, se necessário;
- d) especificação: projetar e documentar a aplicação utilizando a *Unified Modeling Language* (UML). Para tal, será usada a ferramenta Enterprise Architect com o

---

<sup>3</sup> Console é o termo utilizado para designar aparelhos de videogame.

propósito de documentar os requisitos, casos de uso e diagrama de classes. Todo o projeto será realizado utilizando os conceitos de orientação a objetos;

- e) implementação: desenvolver a aplicação utilizando o Visual Studio 2005, a linguagem C# e uma biblioteca a ser definida. Ainda nesta fase, serão utilizados alguns emuladores específicos para a plataforma .NET CF 2.0;
- f) testes e validações: as etapas de testes e validações utilizando os emuladores do Visual Studio 2005 serão realizadas em paralelo, pois são intrínsecas e fornecerão o resultado final da avaliação de desempenho de tudo o que foi utilizado durante o desenvolvimento;
- g) redação da monografia: redigir a monografia.

As etapas serão realizadas nos períodos relacionados no Quadro 1.

etapas / quinzenas	2007									
	fev.		mar.		abr.		maio		jun.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico	■	■	■							
escolha da biblioteca gráfica		■	■							
reavaliação de requisitos			■	■						
especificação				■	■					
implementação					■	■	■	■	■	
testes e validações						■	■	■	■	
redação da monografia				■	■	■	■	■	■	

Quadro 1 - Cronograma

## 2 REVISÃO BIBLIOGRÁFICA

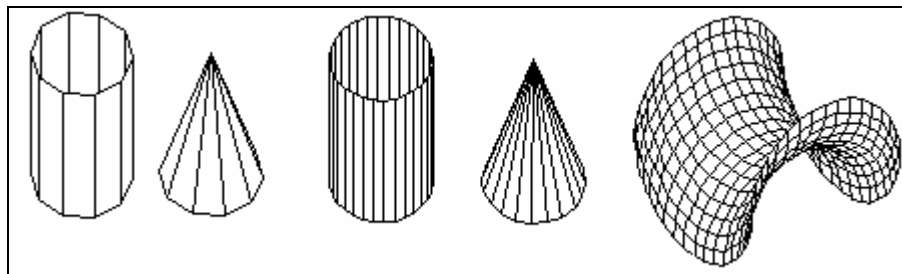
Neste capítulo são apresentados alguns conceitos teóricos relacionados ao trabalho. Na seção 2.1 é definido o que é um modelo 3D e apresentada uma descrição concisa do modelo do Campus IV da FURB. Na 2.2 é descrita a plataforma .NET CF 2.0 e sua arquitetura. Já na 2.3, são apresentadas duas bibliotecas gráficas. Finalmente, a seção 2.4 detalha alguns trabalhos correlatos.

### 2.1 MODELOS TRIDIMENSIONAIS E O MODELO DO CAMPUS IV DA FURB

Segundo Santee (2005, p. 150), um corpo formado por um conjunto de vértices, como uma bola ou o corpo humano, é chamado de modelo 3D. Em aplicações e jogos profissionais, estes modelos são criados utilizando editores gráficos, tais como: 3D Studio Max (AUTODESK, 2006b), Maya (AUTODESK, 2006c) e LightWave (NEWTEK, 2006). Todos estes editores permitem que o usuário represente o modelo em questão de várias formas, sendo a malha de polígonos a mais comum.

A representação por malha de polígonos é definida como sendo uma coleção de polígonos que juntos formam um objeto. Segundo Tenorio (2003), se for preciso modelar uma superfície plana, as malhas se adaptam perfeitamente ao modelo. Se o objeto possuir superfícies curvas, elas devem ser aproximadas por um conjunto de polígonos.

A figura 2 ilustra alguns objetos modelados utilizando malha de polígonos.



Fonte: Tenorio (2003).

Figura 2 - Exemplos de objetos modelados utilizando malha de polígonos

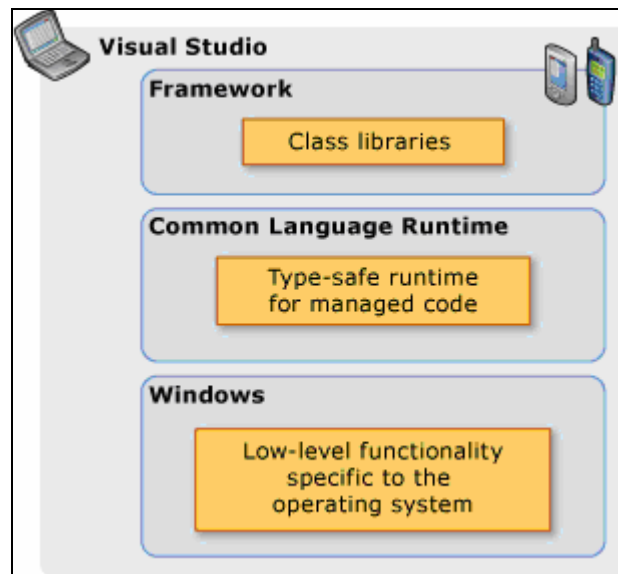
No que diz respeito ao modelo do Campus IV da FURB (vide figura 1), ele foi criado utilizando a ferramenta AutoCad (AUTODESK, 2006a) a partir das plantas baixas originais do Campus. Ele também é um exemplo de modelo 3D formado por uma coleção de

polígonos.

## 2.2 PLATAFORMA .NET CF 2.0

De acordo com Barnes (2003), o .NET CF 2.0 é uma plataforma de desenvolvimento para dispositivos móveis que traz ao desenvolvedor a possibilidade de criar aplicações gerenciadas e aproveitar um extenso conjunto de funcionalidades já implementadas e suficientemente testadas. Além disso, a experiência adquirida no desenvolvimento para computadores é reaproveitada para os dispositivos móveis.

A arquitetura do .NET CF 2.0 (figura 3), segundo a Microsoft (2005b), possui três componentes básicos: o *framework* em si, que é formado por uma biblioteca de classes reutilizáveis; a *Common Language Runtime* (CLR), que é responsável por executar as aplicações; e o sistema operacional Microsoft Windows para dispositivos móveis.



Fonte: Microsoft (2005b).

Figura 3 – Arquitetura do .NET CF 2.0

## 2.3 BIBLIOTECAS GRÁFICAS

Da mesma forma que existem muitas opções de bibliotecas gráficas para o desenvolvimento de aplicações em computadores, os dispositivos móveis também permitem

ao desenvolvedor a possibilidade de escolha. Dentre as existentes, duas serão avaliadas neste trabalho:

- a) OpenGL ES: é uma *Application Programming Interface* (API) que fornece suporte à programação em duas dimensões (2D) ou 3D em sistemas embarcados. Ela é um subconjunto da OpenGL (KHRONOS GROUP, 2004a). Além disso, é uma API multiplataforma e possui vasta documentação;
- b) MD3DM: é a proposta da Microsoft para dispositivos móveis. Ela implementa várias funcionalidades da API Direct3D para computadores e é suportada por equipamentos que rodam os sistemas operacionais da Microsoft versão móvel. Além disso, já vem incorporada ao .NET CF 2.0, dispensando qualquer trabalho de instalação ou configuração para tê-la funcionando.

Essas duas bibliotecas foram escolhidas, pois são compatíveis com o sistema operacional *Windows Mobile 5* e, conseqüentemente, podem ser acessadas pelo .NET CF 2.0. Além disso, são projetos consolidados entre os desenvolvedores, ou seja, são confiáveis e bem documentados.

## 2.4 TRABALHOS CORRELATOS

Existem vários trabalhos que estão relacionados com o desenvolvimento desta aplicação. Entretanto, somente dois foram estudados. Ambos são considerados motores de jogos, ou seja, ferramentas que tornam o desenvolvimento de jogos um processo automatizado, padronizado, rápido e permitem a exibição de modelos 3D. São eles: Mobile Graphics Engine (mOGE) (MACEDO JÚNIOR, 2005) e Mobile 3D Game Engine (m3ge) (PAMPLONA, 2005).

### 2.4.1 mOGE

Segundo Macedo Júnior (2005, p. 4), mOGE é um motor gráfico 3D voltado ao desenvolvimento de jogos para dispositivos móveis. Ele foi inspirado em outros motores gráficos existentes para computadores e celulares, portanto implementa diversas

funcionalidades conhecidas, tais como:

- a) gerenciador de entrada: identifica os eventos de entrada do dispositivo e encaminha para outro módulo executar o processamento;
- b) gerenciador de inteligência artificial: gerencia o comportamento de objetos controlados pela máquina;
- c) gerenciador de objetos: armazena em alguma estrutura de dados e controla o ciclo de vida dos objetos presentes no jogo;
- d) gerenciador de mundo: armazena o estado atual do jogo.

Este motor utiliza a biblioteca OpenGL ES e o sistema operacional Symbian. A figura abaixo apresenta alguns resultados obtidos neste projeto.



Fonte: Macedo Júnior (2005, p. 42).

Figura 4 – Resultados obtidos com a mOGE

#### 2.4.2 m3ge

De acordo com Pamplona (2005, p. 37), m3ge é um motor de jogos escrito em Java e baseado na *Mobile 3D Graphics API (M3G)* (NOKIA, 2003). Dentre outras implementações, esta biblioteca suporta detecção de colisão, Inteligência Artificial (IA), controla a entrada e saída de informações, etc. Além disso, segue uma especificação definida pela *Java Community Process (JCP)* (JCP, 2004a), a *Java Specification Request (JSR) 184* (JCP, 2004b).

A m3ge foi projetada para ser utilizada em conjunto com a M3G (figura 5). Ou seja, as duas bibliotecas interagem entre si. Isso proporciona ao desenvolvedor do jogo flexibilidade e velocidade quando for preciso (PAMPLONA, 2005, p. 39).



Fonte: Pamplona (2005, p. 39).

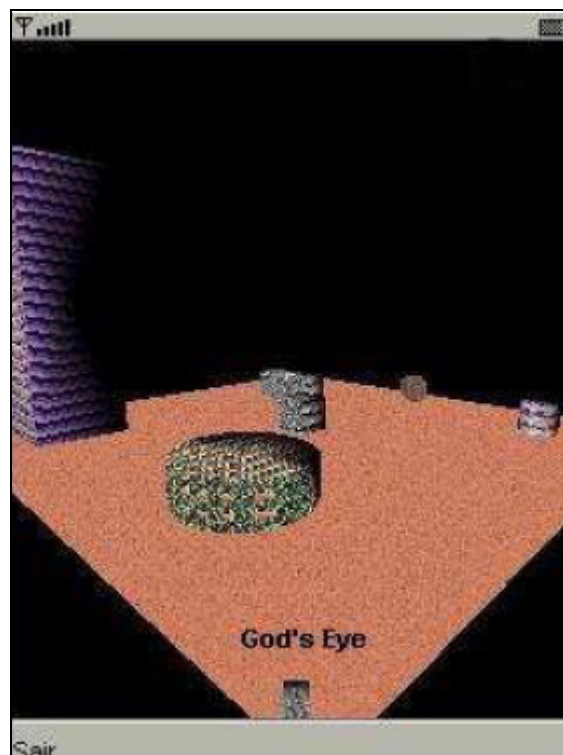
Figura 5 – Visão geral da m3ge

Ainda segundo Pamplona (2005, p. 39), este projeto está dividido em dois grandes componentes: o responsável pela leitura dos arquivos (modelos 3D) no formato Wavefront e o motor de jogos em si (*core*). É sobre o *core* que as implementações dos desenvolvedores devem ficar, ou seja, o enredo e a lógica do jogo são implementados sobre ele.

Quanto aos resultados obtidos com este trabalho, destacam-se as seguintes informações:

- a) foram realizados testes com celulares e emuladores, sendo que a diferença entre os resultados apresentados por ambos foi mínima;
- b) a velocidade do jogo ficou em torno de 4 a 12 *frames* por segundo para movimentação e 15 a 20 *frames* por segundo para rotação do personagem, onde não existe teste de colisão.

A imagem abaixo ilustra o resultado obtido com este trabalho:



Fonte: Pamplona (2005, p. 75)

Figura 6 – Resultado obtido com a m3ge

### 3 REQUISITOS DO SISTEMA A SER DESENVOLVIDO

A aplicação deverá contemplar os seguintes requisitos funcionais (RF) e não funcionais (RNF):

- a) carregar o modelo tridimensional do Campus IV da FURB (RF);
- b) permitir a navegação em primeira pessoa pelo cenário a partir de uma entrada de dados informada pelo usuário (RF);
- c) detectar colisões conforme o usuário caminha pelo ambiente (RF);
- d) ser compatível com o .NET CF 2.0 e com o sistema operacional Microsoft Windows Mobile 5 (RNF);
- e) ser desenvolvida utilizando a linguagem C# (RNF);
- f) executar nos emuladores integrados ao Visual Studio 2005 (RNF);
- g) ser portátil para todos os dispositivos compatíveis com o .NET CF 2.0, isto é, não deverá acessar recursos do hardware sem utilizar as classes do *framework* (RNF).

#### 4 CONSIDERAÇÕES FINAIS

O presente trabalho propõe a construção de um protótipo de uma aplicação gráfica para exibir modelos 3D em dispositivos móveis. Para isto, serão utilizados os emuladores disponíveis do Visual Studio 2005 para simular o comportamento da aplicação. Através deste projeto, será possível validar o desempenho e a capacidade de processamento 3D destes equipamentos, emuladores e bibliotecas utilizadas durante o desenvolvimento do trabalho.

Com as duas bibliotecas gráficas a serem testadas, serão feitos testes comparativos de desempenho para averiguar qual é a mais apropriada para o desenvolvimento da aplicação. A biblioteca escolhida deverá ser compatível com o sistema operacional *Windows Mobile 5*.

Os trabalhos correlatos apresentados neste documento serão usados como fonte de pesquisa para o desenvolvimento da aplicação. Os dois motores mencionados oferecem diversas técnicas que serão úteis no decorrer do projeto, tais como: implementação de câmeras, gerenciamento de eventos e objetos e detecção de colisão.

## REFERÊNCIAS BIBLIOGRÁFICAS

AUTODESK. **AutoCad**. [Montreal], 2006a. Disponível em: <<http://www.autodesk.com/autocad>>. Acesso em: 30 out. 2006.

\_\_\_\_\_. **Autodesk 3ds max**. [Montreal], 2006b. Disponível em: <<http://www.autodesk.com/3dsmax>>. Acesso em: 10 set. 2006.

\_\_\_\_\_. **Autodesk Maya**. [Montreal], 2006c. Disponível em: <<http://www.autodesk.com/maya>>. Acesso em: 10 set. 2006.

BARNES, D. **Fundamentals of Microsoft .NET compact framework development for the Microsoft .NET framework developer**. [Redmond], 2003. Disponível em: <[http://msdn.microsoft.com/netframework/programming/netcf/gettingstarted/default.aspx?pull=/library/en-us/dnnetcomp/html/net\\_vs\\_netcf.asp](http://msdn.microsoft.com/netframework/programming/netcf/gettingstarted/default.aspx?pull=/library/en-us/dnnetcomp/html/net_vs_netcf.asp)>. Acesso em: 10 set. 2006.

BRUNS, M. **Desenvolvimento de um ambiente virtual distribuído não-imersivo utilizando DIVE**. 2003. 43 f. Relatório final de Iniciação Científica (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

JCP. **The Java community process (SM) program: JCP procedures – JCP 2 process document**. [Palo Alto], 2004a. Disponível em: <<http://www.jcp.org/en/procedures/jcp2>>. Acesso em: 10 set. 2006.

\_\_\_\_\_. **The Java community process (SM) program: JSRs – Java specification requests – JSR overview**. [Palo Alto], 2004b. Disponível em: <<http://www.jcp.org/en/jsr/detail?id=184>>. Acesso em: 17 set. 2006.

KHRONOS GROUP. **OpenGL: overview**. [San Francisco], 2004a. Disponível em: <<http://www.opengl.org/about/overview/>>. Acesso em: 10 set. 2006.

\_\_\_\_\_. **OpenGL ES: overview**. [San Francisco], 2004b. Disponível em: <<http://www.opengl.org/opengles/index.html>>. Acesso em: 10 set. 2006.

MACEDO JÚNIOR, I. J. A. **mOGE – mobile graphics engine: o projeto de um motor gráfico 3D para a criação de jogos em dispositivos móveis**. 2005. 75 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife. Disponível em: <<http://www.cin.ufpe.br/~tg/2004-2/ijamj.pdf>>. Acesso em: 17 set. 2006.

MICROSOFT. **.NET compact framework**. [Redmond], 2005a. Disponível em: <<http://msdn.microsoft.com/mobility/netcf/>>. Acesso em: 10 set. 2006.

MICROSOFT. **.NET compact framework architecture**. [Redmond], 2005b. Disponível em: <<http://msdn2.microsoft.com/en-us/library/9s7k7ce5.aspx>>. Acesso em: 18 set. 2006.

\_\_\_\_\_. **Windows mobile 5**. [Redmond], 2005c. Disponível em: <<http://www.microsoft.com/windowsmobile/5/default.mspx>>. Acesso em: 10 set. 2006.

\_\_\_\_\_. **Windows mobile DirectX and Direct3D**. [Redmond], 2005d. Disponível em: <<http://msdn2.microsoft.com/en-us/library/ms172504.aspx>>. Acesso em: 10 set. 2006.

\_\_\_\_\_. **Xbox 360**. [Redmond], 2005e. Disponível em: <<http://www.microsoft.com/xbox/>>. Acesso em: 10 set. 2006.

\_\_\_\_\_. **XNA**. [Redmond], 2006. Disponível em: <<http://msdn.microsoft.com/directx/XNA/default.aspx>>. Acesso em: 10 set. 2006.

MOORE, G. **Moore's law 40th anniversary**. [S.l.], 2005. Disponível em: <[http://www.intel.com/pressroom/kits/events/moores\\_law\\_40th/index.htm](http://www.intel.com/pressroom/kits/events/moores_law_40th/index.htm)>. Acesso em: 10 set. 2006.

NEWTEK. **Lightwave 3D**. [Texas], 2006. Disponível em: <<http://www.newtek.com/lightwave/>>. Acesso em: 10 set. 2006.

NOKIA. **JSR-184 mobile 3D API for J2ME**. [P.O.Box], 2003. Disponível em: <[http://forum.nokia.com/info/sw.nokia.com/id/9e6bdb61-b739-44d7-a7ae-42a4f56ee119/jsr184-specification-1.0\\_Installer.zip.html](http://forum.nokia.com/info/sw.nokia.com/id/9e6bdb61-b739-44d7-a7ae-42a4f56ee119/jsr184-specification-1.0_Installer.zip.html)>. Acesso em: 10 set. 2006.

PAMPLONA, V. F. **Um protótipo de motor de jogos 3D para dispositivos móveis com suporte a especificação mobile 3D graphics API for J2ME**. 2005. 83 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

SANTEE, A. **Programação de jogos com C++ e DirectX**. São Paulo: Novatec, 2005.

TENORIO, G. S. **Modelos 3D: produção, visualização e desdobramentos**. Brasília, 2003. Disponível em: <<http://www.unb.br/fau/disciplinas/cg2/texto1.html>>. Acesso em: 18 set. 2006.