

# RPG BUILDER: UMA FERRAMENTA PARA O DESENVOLVIMENTO DE JOGOS <sup>1</sup>

Edirlei E. Soares de Lima<sup>2</sup>, Pedro Luiz de Paula Filho<sup>3</sup>

## Resumo

Este artigo apresenta o *RPG Builder*, uma ferramenta para criação de jogos de RPG (*Role Playing Game*) em 2D. O *RPG Builder* possibilita que qualquer pessoa possa desenvolver seus próprios jogos com relativa facilidade. Para demonstrar a aplicabilidade deste software foi criado um jogo sobre a Guerra do Contestado. No jogo os jogadores podem vivenciar e entender com mais clareza os fatos ocorridos na época. O trabalho também visa demonstrar que o jogo, quando aplicado de maneira correta, pode ser uma grande ferramenta educacional.

**Palavras-chave:** Game Engine, Jogos, Guerra do Contestado.

## Abstract

This paper presents the *RPG Builder*, a tool to the creation of 2D RPG games. The *RPG Builder* allows anyone to create their own games with relative easiness. To demonstrate the applicability of this software was created a game about the *Guerra do Contestado*. In the game the players can experience and understand more clearly the events that occurred in that time. The work also aims to demonstrate that the game, when applied correctly, can be a great educational tool.

**Keywords:** Game Engine, Games, Guerra do Contestado.

## 1 Introdução

Para a cultura infantil, os jogos representam o que há de mais moderno e inovador em matéria de diversão eletrônica. Antes mesmo de se constituir em entretenimento, o jogo tem para qualquer criança, um valor cultural construído a partir do mundo que a rodeia. Assim compreendidos, os jogos eletrônicos não podem ser considerados como um simples passatempo, mas sim uma atividade lúdica de aprendizado. De acordo com Rabsch e Baum (2004) "O lúdico aplicado à prática pedagógica não apenas contribui para a aprendizagem da criança, como possibilita ao educador tornar suas aulas mais dinâmicas e prazerosas".

As crianças e os adolescente de hoje vivem cercados de videogames, computador, TVs e DVDs. As últimas descobertas da ciência dizem que o uso desses recursos na medida certa, ao contrário do que se pensava, pode ajudar a melhorar a inteligência das crianças. Estas descobertas revelam que na década de 80 o QI das crianças e adolescente aumentou cerca de 10 % em comparação aos anos 50, e hoje em dia esse numero é ainda maior, chegando a um crescimento de 25% em comparação a mesma época (CORDEIRO, 2006).

Segundo Cabral (1997), os jogos eletrônicos auxiliam no desenvolvimento do raciocínio lógico e criatividade da criança, que ao mesmo tempo em que se diverte está absorvendo informações de uma forma mais prazerosa. E também oferecem àqueles que os manipulam a possibilidade de se inserirem em uma "realidade" mais rica de emoção do que aquela que poderia ser encontrada em suas próprias experiências, particularmente quando estas não correspondem às suas potencialidades criadoras. Enquanto a realidade é muitas vezes tomada como ilegítima, no jogo o que importa é a fantasia vivida eletronicamente. Assim sendo, os jogos, principalmente os que sugerem violência, possivelmente funcionam como uma espécie de "catarse" para a angústia, o sonho, e também para a inclinação que jovens e crianças têm pelo perigo e pelo desafio competitivo.

Atualmente, a área dos jogos educacionais ainda é pouco explorada no mercado da computação. O principal motivo disto é a dificuldade em se criar um jogo que seja interessante ao jovem. Com base nisto buscou-se uma maneira para facilitar este desenvolvimento, onde qualquer pessoa, mesmo que esta não possua nenhum conhecimento de linguagens de programação, possa criar seus jogos. A forma encontrada para realizar esta tarefa é o uso de uma *Game Engine*.

Uma *Game Engine* é um software com diversas funções que facilitam o desenvolvimento de jogos, possibilitando assim uma maior abstração da programação envolvida no processo. O termo surgiu durante a década de 90, juntamente com os primeiros jogos 3D (Doom, Quake) e a necessidade de criar-se jogos mais complexos, além disto, surgiram equipes de desenvolvimento, e para manter um certo padrão de programação entre os desenvolvedores, foram criadas as primeiras *Engines*. As técnicas de desenvolvimento evoluíram muito nas últimas décadas, hoje em dia é muito difícil encontrar jogos que não tenham sido desenvolvidos através de alguma ferramenta deste tipo, muitas vezes as produtoras criam as suas próprias *Engines*, assim podem utilizar a mesma para o desenvolvimento de outros jogos. Já outras optam por comprar licenças de produtos terceirizados, pois muitas empresas se dedicam exclusivamente na criação deste tipo de software.

Além do caráter educacional, os jogos, atualmente, movimentam o bilionário mundo das produtoras, e vem causando uma revolução no entretenimento. Em 2003 o mercado mundial de jogos eletrônicos movimentou cerca de US\$ 22,3 bilhões, somente com a comercialização dos jogos, se incluirmos também os gastos com hardware e acessório este valor chegaria a US\$ 55,5 bilhões, superando até mesmo a indústria cinematográfica que no mesmo ano obteve um lucro de US\$ 19 bilhões (ASSIS, 2003).

## 2 Metodologia

A *Game Engine* desenvolvida, chamada de “*RPG Builder*”, é destinada a criação de jogos 2D, estilo RPG (*Role Playing Game*), ela permite qualquer usuário elaborar as suas próprias aventuras. Assim, surge a possibilidade de se criar inúmeros jogos educacionais, pois qualquer educador pode desenvolver uma aventura relevante ao seu tema de ensino, passando assim para os seus alunos uma maneira lúdica de aprendizado. Uma outra maneira de utilização da ferramenta de modo educacional é o desafio aos próprios alunos a continuarem uma história, ou mesmo desenvolverem a sua própria aventura, estimulando assim a sua criatividade.

O *RPG Builder* é dividido em diversos módulos, o principal deles é o editor de cenários, onde são desenhados todos os mapas do jogo, estes são baseados em *tiles* (pequenas imagens posicionadas de maneira seqüencial), como pode ser visto na figura 1, cada pequeno quadrado é um *tile*, a união de vários deles forma um cenário. Para um maior grau de realismo o mapa é dividido em três camadas, duas de imagens e uma terceira de eventos, a primeira serve como fundo, possibilitando assim o uso de imagens transparentes pela segunda camada, na terceira camada ficam os eventos, ou seja, as ações que podem ocorrer durante o jogo.

A criação de personagens é feita através da interface de edição de personagens (figura 2), nela são definidas as características e animações dos personagens. Estas animações são baseadas em *frames* (exibição seqüencial de imagens em um curto espaço de tempo), cada estado de animação é formado por cinco imagens exibidas repetidamente, estes estados são referentes a determinadas situações que podem ocorrer durante o jogo, como por exemplo, um ataque em uma batalha.



Figura 1: edição de cenários.

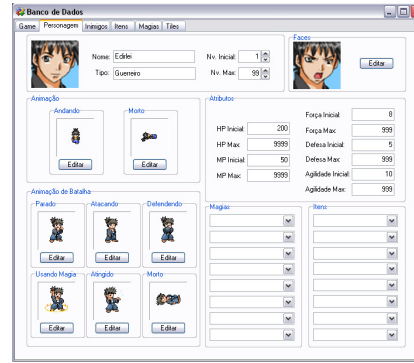


Figura 2: edição de personagens.

Os dados gerados na ferramenta são gravados em uma base de dados XML (*eXtensible Markup Language*). Este formato atende bem as necessidades para a criação de documentos com dados organizados de forma hierárquica, como é o caso dos mapas. Além da organização o XML também oferece uma grande portabilidade, pois não necessita da instalação de nenhum servidor de banco de dados específico para o funcionamento da aplicação. Na figura 3 é possível visualizar a estrutura de um mapa gerado pela ferramenta.

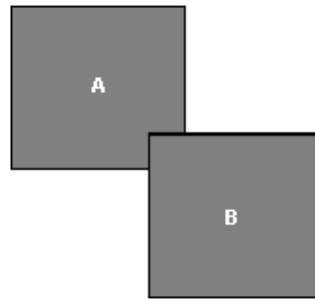
```
<FIELDS>
  <FIELD attrname="Terreno" fieldtype="String"/>
  <FIELD attrname="Fundo" fieldtype="String"/>
  <FIELD attrname="Solido" fieldtype="Boolean"/>
  <FIELD attrname="Evento" fieldtype="Text"/>
  <FIELD attrname="Transparente" fieldtype="Boolean"/>
</FIELDS>
```

Figura 3: Estrutura de um Mapa

Na criação de um mapa, o usuário pode definir se uma determinada área é sólida ou não, sendo sólida o jogador não poderá passar por ela, neste caso teríamos uma colisão. A detecção de colisões é uma parte essencial para o jogo, pois ela é a responsável por dar limites aos objetos, sem ela, um objeto poderia passar livremente sobre o outro. A implementação da detecção de colisão, normalmente é dividida em duas etapas, a primeira é verificar se a colisão ocorreu e a outra é qual a reação do objeto, como por exemplo, quando o jogador colide com uma parede, ele não poderá continuar se movimentando na sua direção, porém pode deslizar ao seu redor, um outro tipo de reação poderia ser a movimentação do objeto colidido, tudo depende do sistema de colisão utilizado. Colisões em ambientes 2D são facilmente detectáveis, basta verificar se dois objetos estão ocupando uma mesma área. Na figura 4 é possível perceber que o objeto A e o objeto B estão apenas próximos, ou seja, sem colisão, já na figura 5 é visível que o objeto B está sobre o objeto A, neste caso ocorre uma colisão. Computacionalmente para chegar este resultado basta manipular as coordenadas x e y do objeto, juntamente com a sua altura e tamanho (ARMSTRONG, 2003). Ao utilizar o *RPG Builder* não é necessário se preocupar com isto, pois é um processo automatizado para facilitar a criação dos jogos, deste modo, para o usuário basta definir se uma área é sólida ou não.



**Sem Colisão**  
*Figura 4*



**Com Colisão**  
*Figura 5*

No *RPG Builder* os eventos são baseados em *scripts*, e podem ocorrer a partir de duas situações: uma colisão com uma área que possua um evento de *OnPlayerTouch* ou com uma interação do jogador com determinada área de evento *OnActionPress*. Estes eventos são criados a partir de um editor que possui a inserção de *scripts* automáticos para facilitar o seu uso. Existem cerca de 40 funções para serem utilizadas nos eventos, como mudança de mapa, exibição de diálogos, movimentação de personagens secundários, efeitos de clima, entre outras. Na figura 6 é possível ver um exemplo de *script* para mudança de mapa, onde, ao toque do jogador sobre a área do evento ocorreria uma mudança de mapa para o “Mapa03.xml” (nome do arquivo do mapa criado na ferramenta), nele o jogador surgiria nas coordenada x 250 e y 160.

```
OnPlayerTouch>  
MudarMapa :Mapa03.xml | 250 | 160 ;
```

*Figura 6: Script para mudança de mapa*

Para comprovar a funcionalidade da ferramenta desenvolvida, foi criado um jogo que representa passagens da Guerra do Contestado, onde o jogador pode vivenciar e entender com mais clareza os fatos ocorridos na época. O jogo inicia-se em 1912 com a aparição do terceiro monge e vai até a batalha de Irani, onde ele morre. Durante a aventura o jogador viverá o papel de um caboclo que acompanha o monge, conhecerá a história dos outros monges, entenderá o porquê de tanta revolta da população da região e participará de todo o desenrolar dos fatos.

A história é toda contada através de diálogos entre os personagens como pode ser visto na figura 7, para poder avançar no jogo o usuário deve acompanhar e entender a história para saber os próximos passos a serem seguidos. Existem diversos desafios a serem vencidos pelo jogador, como, por exemplo, reunir alguns seguidores para o grupo do monge José Maria.

Os cenários da aventura são formados por pequenos vilarejos e florestas como mostra a figura 8, em determinados lugares o jogador deve enfrentar inimigos como animais que habitam a região, com isto, o personagem principal adquire experiência, ficando assim mais forte para as próximas batalhas.



Figura 7: Dialogo com o monge José Maria

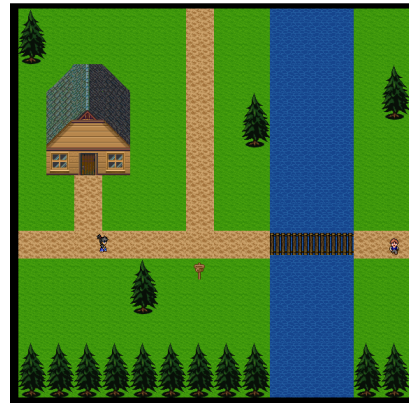


Figura 8: Cenário do jogo

Para demonstrar o caráter lúdico do jogo ele foi aplicado na APADAF (Associação dos Pais e Amigos do Deficiente Auditivo e da Fala) com alunos de idade variada, antes da aplicação, todos tiveram uma palestra sobre o tema com um professor especialista na história do contestado e posteriormente puderam vivenciar os fatos através da interação com o programa. Durante a aplicação, percebeu-se um grande entusiasmo dos alunos, pois além de estarem se divertindo, estavam ao mesmo tempo aprendendo sobre o passado do estado de Santa Catarina.

### 3 Conclusão

Em relação ao projeto, pode concluir-se que os objetivos foram atingidos. Foi demonstrado que o jogo, utilizado de maneira correta, pode ser uma ferramenta de auxílio pedagógico poderosa, pois, além de ser divertido possui caráter educativo, estimulando a ludicidade.

A *Game Engine* desenvolvida proporciona a criação de jogos de maneira rápida e fácil. E vai possibilitar a criação de novos projetos na área de educação, pois com ela novos pesquisadores poderão desenvolver outros jogos ou mesmo dar continuidade ao desenvolvido durante o projeto, estimulando assim o raciocínio lógico, a criatividade e organização, necessários para criação dos cenários, personagens, enredo, eventos, entre outros.

A ferramenta e o jogo foram disponibilizados para *download* em [www.pu.unc.br/games](http://www.pu.unc.br/games).

### Referências

CORDEIRO, Tiago. **Imersos na Tecnologia - E Mais Espertos**. Revista Veja, Edição de 11 jan. 2006.

CABRAL, Fatima. **Jogos eletrônicos: técnica ilusionista ou emancipadora?**. Revista USP. Edição de 10 jan. 1997.

RABSCH, Marisete; BAUM, Romilda. **Jogos Pedagógicos na Pré-escola no Desenvolvimento do Processo Ensino Aprendizagem**. Trabalho de Conclusão do Curso de Pedagogia. Universidade do Contestado. Porto União, 2004.

ASSIS, D., MATIAS, A.. **Game supera cinema como opção de entretenimento em 2003**. Artigo da Folha de São Paulo. Disponível em: [www1.folha.uol.com.br/folha/ilustrada/ult90u40114.shtml](http://www1.folha.uol.com.br/folha/ilustrada/ult90u40114.shtml). 31 dez. 2003.

ARMSTRONG, Chad Wesley. **Platform Independent Game Engine**. Dissertação em Ciência da Computação - Montana State University. 2003.

---

<sup>1</sup> Artigo Científico financiado pelo Fundo de Amparo à Pesquisa do Estado de Santa Catarina – FAPESC, apoiado pela Universidade do Contestado – UnC, Campus de Canoinhas/Porto União – SC

<sup>2</sup> Acadêmico da 6ª fase do curso de Ciência da Computação da Universidade do Contestado, Campus de Canoinhas/Porto União – SC

<sup>3</sup> Professor orientador, Mestre em Ciência da Computação