

GAMES: Panorama Histórico, Processo e Equipe de Desenvolvimento e Linguagens de Programação

Alexandre Honorato Sabino

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

Brian Pelegrini Dantas

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

Isaque Junior Pereira dos Santos

Graduando em Engenharia da Computação,
Faculdades Integradas de Três Lagoas – FITL/AEMS

Alan Pinheiro de Souza

Mestre em Informática – UFRJ;
Docente das Faculdades Integradas de Três Lagoas – FITL/AEMS

Olavo Alves dos Santos Neto

Especialista;
Docente das Faculdades Integradas de Três Lagoas – FITL/AEMS

Thiago de Oliveira Correia

Docente das Faculdades Integradas de Três Lagoas – FITL/AEMS

RESUMO

Esse projeto realizou um levantamento biográfico sobre jogos digitais dado à importância desses produtos nos dias atuais. A pesquisa conduziu uma análise histórica no intuito de conhecer os primeiros *games* criados, passando ao entendimento de fases de desenvolvimento de um jogo, destacando como é o seu processo de produção e, por último, discutiu a linguagem Java de modo a identificar sua aplicabilidade no contexto sendo estudado.

PALAVRAS-CHAVE: *Games; Interfaces; História dos Jogos; Desenvolvimento de Games; Equipes de Desenvolvimento; Java.*

INTRODUÇÃO

O mercado de jogos pode ser considerado um dos maiores do mundo na área de entretenimento. A indústria dos *games* teve início nas décadas de 1960 e 1970, ainda como uma curiosidade, porém cresceu rapidamente contribuindo para o crescimento da computação (BARBOZA, 2008). Em razão desse destaque crescente do mercado de jogos digitais, esse projeto científico possui como objetivo apresentar um panorama da história dos *games* (seção 1), além de abordar passos necessários para construção de jogos digitais (seção 2). A pesquisa estudou

também a linguagem *Java*, em razão da sua aplicabilidade no contexto de jogos, especialmente, os recursos de computação gráfica do pacote *Java 2D* (seção 3).

2 PANAMORAMA HISTÓRICO

Esta seção apresenta características de jogos eletrônicos e caminhos tomados ao longo do seu processo de evolução. Para adequada assimilação do panorama de evolução histórica desses jogos, torna-se necessário recuperar os primeiros *games* desenvolvidos, embora fossem considerados inovadores para as suas épocas, esses jogos eram pobres de estrutura e elaborados sem abordagens sistemáticas, quando comparados aos produtos mais modernos do ramo (BATISTA, 2007).

2.1 Primeiros Jogos

As primeiras impressões dos jogos eletrônicos na história vieram com o cientista *Alexander Douglas*, considerado inovador nas áreas de computação e desenvolvimento de jogos. Em 1952, na Universidade de *Cambridge*, inventou um jogo eletrônico, *tic-tac-toe*, para sua tese de doutorado sobre a “*Interação Homem-Computador*”, desenvolvido no computador EDSAC (*Electronic Delay Storage Automatic Calculator*) (KUROSU, 2014). Na visão de Crowley (1993, p.533), “*tic-tac-toe* é um jogo simples para crianças em que dois jogadores revezam, desenhando fichas (X ou O) em uma malha 3x3, onde ganhar exige colocar três fichas em uma fileira, coluna ou diagonal”.

Em 1958, surgiu o *Tennis For Two*, jogo desenvolvido para simular uma partida de tênis, operava a partir de um computador analógico ligado a um osciloscópio, com um controle composto de um *dial* (botão giratório) e um botão de pressionar. Esse jogo funcionava da seguinte maneira: a bola era rebatida na parte inferior da tela e no centro existia uma linha vertical que representava a rede (BATISTA, 2007). De acordo com Leite (2006), um ponto piscando representava a bola e os jogadores controlavam seu movimento pelo *dial*.

2.2 Evolução dos Games

Para que ocorresse a evolução dos *games* era necessário que também houvesse progresso dos computadores. Na década de 60, foram desenvolvidos os

chips de silícios, permitindo aos computadores, que um dia ocuparam salas inteiras, ficarem menores, mais rápidos e potentes (LEITE, 2006). Esse avanço de arquitetura possibilitou a criação de jogos com programações mais robustas e complexas.

Em 1973, *Steve Colley*, aluno dos centros de pesquisas da NASA (*National Aeronautics And Space Administration*), inovou ao desenvolver o primeiro jogo com tecnologia tridimensional, além de ser pioneiro na criação dos jogos que funcionava em primeira pessoa (BEATY, 2010; STANTON, 2015). Esse jogo conhecido como *Mazewar* funcionava a partir de um jogador que se encontrava em um labirinto 3D. Originalmente foi desenvolvido para um *player*, porém, por sua falta de objetivos, interligaram os PDS-1 (*Programm able Display System*) pelas suas portas e séries de modo a proporcionar interação entre dois ou mais jogadores na mesma partida, cada *player* era representado por um globo ocular (LOWOOD, 2016).

Esse *game* funcionava de maneira similar aos jogos de tiros modernos, onde um jogador procura ao outro visando eliminá-lo da partida. Após sofrer por atualizações, *Mazewar* ficou conhecido por permitir a adoção de trapaça, pois por meios de códigos adicionais, os jogadores tinham acesso a um mapa do labirinto mostrando as posições dos oponentes (OLIVEIRA, 2010). O jogo ainda pode ser apreciado nos dias atuais após diversas modificações.

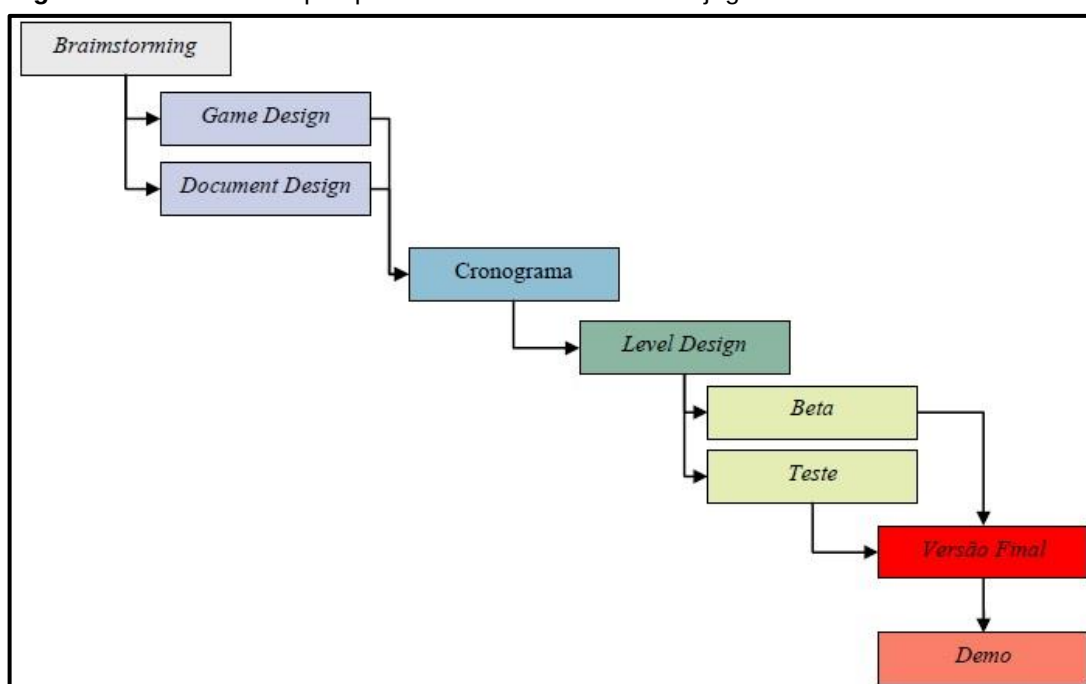
3 PLANEJAMENTO DE GAMES

Para que haja sucesso no desenvolvimento de um jogo é necessário haver planejamento das ações a serem executadas pela equipe especializada de profissionais. Nesta seção será discutida a documentação do projeto a ser elaborada mediante a realização de planejamento e a definição da equipe de desenvolvimento do jogo. A etapa de planejamento deve ser executada a partir do momento da definição do conceito do jogo a ser construído. Essa atividade é importante para garantir alcance de um resultado de qualidade. Antes do desenvolvimento do jogo deve-se pensar e esquematizar o projeto, pois qualquer modificação radical de rumo no decorrer da fase de produção do jogo pode ser crítica e comprometer os objetivos e as metas do projeto previamente definidas (PERUCIA et al., 2005).

Para Amazonas (2007), primeiramente deve ser feita uma reunião com todos os membros da equipe para a organização das ideias. Esta etapa definida como

Brainstorming consiste em “reunir toda a equipe para discutir originalidade, inovação e público-alvo, elaborar todo o detalhamento da ideia desde a apresentação até as especificações técnicas”. Após esse encontro e com as informações já coletadas, deverão iniciar as etapas de projeto e documentação do jogo, identificadas, respectivamente, como *game design* e *document design* (AMAZONAS, 2007; CLUA; BITTENCOURT, 2005). As etapas seguintes são, conforme apresentado na Figura 1: (i) Cronograma: definição da ordem de cumprimento das tarefas; (ii) *Level Design*: estabelecimento dos desafios e missões que o jogador deverá exercer para concluir cada fase, utilizado como referência para o desenvolvimento dos cenários; (iii) *Beta*: fase de lançamento de versões e realização de testes, são elaboradas várias configurações *alpha* até alcançar a versão *beta*, que possui todas as fases e interatividade do jogo; (iv) Testes: fase de correção de erros e aceitação de opiniões para melhoria do produto; (v) Versão Final: configuração pronta do jogo a ser comercializada; (vi) *Demo*: demonstração lançada da versão final.

Figura 1. Ordem das etapas para desenvolvimento de um jogo.



Fonte: Extraído de Amazonas, 2007.

Há também uma segunda visão de Fullerton (2008) para a criação de jogos, em que é executado um processo com sete passos, que difere apenas na ordem de execução de alguns itens: (i) Criação de ideias por intermédio de *Brainstorming*; (ii) Execução de um protótipo físico do jogo; (iii) Apresentação de *storyboards*¹ e

rascunhos mais completos do jogo; (iv) Protótipo digital, no caso de jogos eletrônicos; (v) Documento do projeto (*Design Document*); (vi) Produção e finalização do *game*; (vii) Certificação de qualidade, testes finais e correções.

3.1 Document Design

O *Design Document* ou *Design Bible* é um documento que contém todas as especificações do jogo e que guiará todas as partes envolvidas no seu processo de produção (LEMES, 2009). Essa etapa pode ser comparada ao roteiro de elaboração de um filme (CLUA; BITTENCOURT, 2005, p.1327).

Assim como não é possível criar um filme sem antes ter um roteiro bem elaborado, também é impossível desenvolver um jogo sem antes ter um documento com todas as suas especificações. Costuma chamar-se este documento de *Design Bible*, que pode ser visto como uma espécie de manual de instruções para os futuros desenvolvedores do jogo. De fato, tão importante é este documento, que o processo de desenvolvimento não pode começar sem que esse não esteja pronto (CLUA; BITTENCOURT, 2005, p.1327).

É neste documento que se encontram informações como nome do jogo, descrição resumida, público-alvo, estilo do *game*, história, especificações técnicas de *hardware* e sistema operacional, gráficos, dispositivos de entrada e sonorização, onde são definidas as músicas nos *menus* e nas fases do jogo (Perucia et al., 2005). Com relação ao *game design*, a seguinte afirmação encontra-se em Lemes (2009, p.55):

Este documento usa palavras, tabelas e diagramas para explicar o funcionamento de um *game* a partir da história do mundo ficcional do *game* até a organização de botões em uma *interface* e ao modo como um arqueiro luta contra um espadachim (LEMES, 2009, p.55).

De acordo com Clua e Bittencourt (2005), o *Design Bible* deve conter em sua estrutura roteiro, *game play*, *interface* gráfica e *game design*. O *game design*, na visão de Clua e Bittencourt (2005), é entendido como a conceituação artística do jogo. No entanto, este termo não pode ser explicado apenas por sua tradução literal (TOCCHIO, 2007).

3.2 Game Design

Este é um assunto muito importante e vasto no mundo dos *games*. Nessa seção foram reunidas algumas ideias a respeito dessa atividade para que haja breve

noção do que conteúdo. A ideia de *game design* não está somente associada ao conceito de *layout* do jogo. Segundo Mastrocola (2012, p.37), “se jogo fosse uma pessoa, *game design* seria sua personalidade”. O processo de *game design* envolve animação, trilha sonora, criação da mecânica, regras e objetivos do jogo:

A modelagem 3D, o desenho de um tabuleiro, o cenário de fundo e até mesmo uma trilha sonora de um jogo eletrônico fazem parte do processo de criação de um jogo, fazem parte do processo de *game design*, mas, em essência, o desafio do *game designer* é criar mecânicas, regras e objetivos de um jogo (MASTROCOLA, 2012, p.35).

Para Brathwaite e Schreiber (2009, p.2), *game design* é “o processo de criar as disputas e as regras de um jogo”. O *game design* envolve algo que chame a atenção do jogador, sendo considerado processo pelo qual o *designer* cria um contexto a ser encontrado por um participante, a partir do qual o significado emerge (SALEN; ZIMMERMAN, 2004). Segundo Brathwaite e Schreiber (2009), um bom *game design* é o processo de criar objetivos para que o jogador sinta-se motivado a alcançar e regras a serem seguidas como se estivesse tomando decisões significativas para atingir estes propósitos.

Segundo Rolling e Morris (2004), este documento é um manifesto que apresenta o seu conceito para o jogo e, acima de tudo, o sentimento de como seria jogá-lo. Este conceito inicial ajuda a solidificar as ideias do jogo. O conceito de *game design* também pode ser entendido do ponto de vista de jogabilidade, conforme Rouse III (2005, p.xxi):

O *game design* determina quais escolhas o jogador será capaz de fazer no mundo do jogo e que ramificações estas escolhas terão no restante do jogo. O *game design* determina qual o critério de ganho ou perda o jogo deverá incluir, como o usuário será capaz de controlar o jogo e que informações o jogo comunicará a ele, e isto estabelece o quão difícil o jogo será. Resumidamente, o *game design* determina todos os detalhes de como a jogabilidade funcionará (ROUSE III, 2005, p.xxi).

Para Perucia et al.(2005), é no processo de *game design* onde são descritas as características principais do jogo como jogabilidade, controles, *interfaces*, personagens, armas, golpes, inimigos e fases. Um bom *game design* não pode ser muito fácil ou muito difícil, pois não pode ser fácil para que o jogador se entedie nem difícil de modo a evitar que ele se desmotive, ou seja, o *game design* deve ser balanceado (TAVARES; NEVES, 2006).

É por intermédio do *game design* que todas as outras atividades relacionadas ao desenvolvimento do jogo tomam rumo. Essa tarefa garante que a equipe compreenda e execute o trabalho a ser esperado (MARIASCH, 2011). O *game design* deve conter, em sua essência, as características necessárias para fazer com que o jogador sinta-se unido ao contexto do jogo e sinta-se ao centro da história (SATO, 2010; BRATHWAITE; SCHREIBER, 2009). Segundo Mastrocola (2012, p. 35), “*game design* tem a ver com a mecânica e o desenrolar da experiência significativa de jogar”.

3.3. Equipe de Projeto

Para que o planejamento e o desenvolvimento do projeto sejam realizados de maneira adequadas é preciso uma equipe especializada e engajada na realização do mesmo. Suas funções podem ser resumidas em três segmentos principais: *design*, arte e programação (BALISTA, 2013). (i) *Design*: o responsável por essa função é o *game designer*. De acordo com Mariasch (2011, p.5), “este profissional pode ter como formação acadêmica diversas origens, devido à dimensão extensa de possibilidades de *design*”. É nessa etapa em que as formas para manter o jogador interessado são elaboradas. Neste segmento existe também a participação do *level designer* responsável pela dificuldade de cada nível do jogo (BALISTA, 2013). (ii) Arte: segmento em que são elaborados visual e vários outros recursos artísticos do jogo (BALISTA, 2013). Os artistas podem criar objetos, texturas, personagens e animações (AMAZONAS, 2007). (iii) Programação: fase em que é feita toda a programação do jogo. Os programadores são responsáveis por criar a interatividade entre usuário e jogo (BALISTA, 2013).

A Tabela 1 ajuda a compreender melhor a estrutura de uma equipe de desenvolvimento de jogos. Nesse esquema é possível identificar as etapas e os possíveis papéis necessários para compor cada segmento de maneira satisfatória visando elaborar um jogo com qualidade. Entre essas funções estão gerenciamento e *design*, programação, arte visual, música e controle de qualidade (TAVARES; NEVES, 2006; ROLLINGS; MORRIS, 2004). Essa análise detalha aquela apresentada por Balista (2013) e evidencia o caráter interdisciplinar de projetos nessa área, pois há uma extensa relação de papéis e responsabilidades, especialmente, oriundos de outros campos de conhecimento.

Tabela 1. Estrutura da equipe de desenvolvimento de jogos.

1.1 <i>Game Designer</i> 1.2 <i>Level Designer (Designer de Níveis)</i> 1.3 <i>Character Designer (Designer de Personagens)</i> 1.4 Gerente de Projeto 1.5 Gerente de <i>Software</i>
2.1 <i>Lead Programmer (Programador de Conceito)</i> 2.2 Programadores
3.1 <i>Lead Artist (Artista de Conceito)</i> 3.2 Artistas Visuais (Modeladores, Ilustradores, entre outros)
4.1 Músico 4.2 Efeitos Sonoros e Diálogos 4.3 Programadores de Áudio
5.1 <i>Quality Assurance Lead (Condutor de Controle de Qualidade)</i> 5.2 <i>Quality Assurance Technicians (Controladores de Qualidade)</i> 5.3 <i>Play testers (Jogadores Avaliadores)</i>
6.1 Especialistas em outras áreas (Educadores, Consultores, entre outros) 6.2 Técnicos em áreas diretamente relacionadas (Captura de Movimento, Roteiristas, entre outros)

Fonte: Extraído de Tavares e Neves, 2006 e Rollings e Morris, 2004.

Cada função é definida por Rollings e Morris (2004): (i) Gerente de Projetos: estabelece cronograma da carga de trabalho, efetiva interações entre membros da equipe e atua como *interface* entre a equipe do projeto e os departamentos de gestão e *marketing*; (ii) Gerente de *Software*: estipula um conjunto detalhado de requisitos técnicos e estima tempo e esforço necessários para sua conclusão; (iii) *Lead Programmer*: coordenar e monitora esforços da equipe de programação para garantir que o cronograma está sendo mantido; (iv) *Lead Artist*: garante que todos os artistas estão produzindo obras de arte alinhadas aos propósitos do projeto; (v) Músicos: produzem músicas e áudios do jogo. De acordo com Amazonas (2007, p.9), o som pode acomodar trilhas e efeitos sonoros, vozes para o *game*, entre outros; (vi) *Quality Assurance Lead*: supervisionam a equipe de qualidade e cooperam com o gerente de projeto e o *designer* de jogos para se certificar de haverá ampla cobertura dos testes; (v) *Quality Assurance Technicians*: garantem qualidade do código escrito pela equipe de programação; (vi) *Players testers*: realizam testes que avaliam o *game*.

É importante ressaltar que, além desses profissionais envolvidos diretamente no desenvolvimento do jogo, podem existir outras pessoas que auxiliam em sua produção como consultores, produtores, executivos, educadores, advogados, entre outros (SELING, 2009), o que reforça novamente as características interdisciplinares de projetos de jogos.

4 LINGUAGEM JAVA NA CRIAÇÃO GRÁFICA DE INTERFACES

A linguagem Java foi lançada em meados da década de 90, sendo considerada simples, pois permite o desenvolvimento de projetos em diferentes sistemas operacionais e arquiteturas de *hardware*, sem que o programador tenha preocupação com detalhes de infraestrutura, o que gera um desempenho mais produtivo e eficiente (WARBURTON, 2014; MENDES, 2009). O Java segue o paradigma de orientação a objetos, possibilitando a utilização de conceitos de herança, polimorfismo e encapsulamento (MENDES, 2009). Para Junior (2014), Java pode ser caracterizado da seguinte maneira:

O Java é uma linguagem de programação de propósito geral, concorrente, baseada em classes e orientada a objetos. Projetada para ser simples o bastante para que a maioria dos programadores se torne fluente na linguagem. Java tem relação com C e C++, porém, é organizada de forma diferente, com vários aspectos de C e C++ omitidos e algumas ideias de outras linguagens incluídas (JUNIOR, 2014, p.17).

A linguagem Java possui a singular característica de compilar e ao mesmo tempo ser interpretada, isso é possível, pois o compilador Java transforma um programa-fonte em *bytecodes*², interpretados pela JVM (*Java Virtual Machine*) (SERSON, 2007), que funciona como uma máquina imaginária, isso possibilita a execução desses arquivos em qualquer arquitetura (*Windows, Linux, Mac e Unix*) que tenha uma máquina virtual Java instalada (MENDES, 2009).

Na linguagem Java inicialmente cria-se uma classe para abrigar um ou mais métodos, quantos forem necessários para o desenvolvimento do *software* (DEITEL; DEITEL, 2010). Uma classe define o modelo dos objetos (MENDES, 2009). Já os métodos são compostos por códigos que fazem executar as eventuais tarefas propostas pelo programador (MENDES, 2009; SERSON, 2007).

Para programação visual, o Java possui pacotes de desenvolvimento gráfico, os quais fornecem estruturas de programação gráficas chamadas API

(*Application Programming Interface*) (DEITEL; DEITEL, 2010). No entendimento de (NETO, 2011), as *interfaces* gráficas introduziram uma nova lógica no relacionamento do homem com o computador, mediando às relações de suas atividades, tendo como objetivo a diversão e a satisfação dos jogadores. Ainda segundo Neto (2011, p.133):

A importância da *interface* não está somente em um *layout* bonito e criativo, mas, principalmente, no planejamento das interações possíveis para o jogador. É a partir da interação humano-computador que outras experiências serão desenvolvidas, como por exemplo, a imersão, a diversão, o engajamento, o envolvimento, o fluxo, a jogabilidade, a aprendizagem, entre outras.

No desenvolvimento gráfico, o Java se baseia em duas bibliotecas: a AWT (*Abstract Window Toolkit*) e a *Swing*. A janela é a parte mais importante no desenvolvimento de *interfaces*, pois é nela que os demais elementos serão inseridos. No pacote *Swing* se encontra a *JFrame*, classe que fornece todas as propriedades, métodos e eventos para construir janelas “padrão *Windows*” (SERSON, 2007).

Para a criação de textos nas janelas é utilizada a classe *JLabel* (DEITEL; DEITEL, 2010), já para a criação de botão na janela é utiliza-se a classe *JButton*, podendo ainda adicionar textos e imagens (SERSON, 2007). Para que sejam acionados eventos correspondentes a ações do usuário, é utilizada a *interface Action Listener* que possibilita a identificação das ações do usuário (DEITEL; DEITEL, 2010).

Para programação visual, no Java ainda existe uma API chamada Java2D, que fornece estruturas gráficas bidimensionais avançadas que requerem manipulações gráficas complexas e detalhadas. Esta inclui recursos para processar artes de traços, textos e imagens nos pacotes *java.awt*, *java.awt.image*, *java.awt.color*, *java.awt.font*, *java.awt.geom*, *java.awt.print* e *java.awt.image*. Através dessas classes é possível criar desenhos coloridos, estabelecer várias formas em uma janela, como retângulos, elipses, polígonos, linhas e arcos (DEITEL; DEITEL, 2010).

5 CONSIDERAÇÕES FINAIS

Esse artigo científico apresentou uma introdução ao conhecimento dos jogos eletrônicos, buscando a ampliação e a compreensão para o assunto exposto. O

artigo teve início com um panorama histórico, com destaque para alguns dos primeiros *games* existentes. O trabalho discutiu também as etapas a serem seguidas ao longo do desenvolvimento de jogos. A pesquisa forneceu breve caracterização da linguagem Java, geralmente adotada na construção de *interfaces*. Como trabalhos futuros podem ser aprofundados estudos sobre desenvolvimento de *games*, detalhando cada uma das atividades técnicas necessárias, assim como aplicar a linguagem de programação Java e ferramentas relacionadas no processo de desenvolvimento de algum jogo, de modo que suas características, facilidades e dificuldades sejam melhor identificadas para esse contexto. Pode-se verificar a aplicabilidade de outras linguagens de programação, assim como *engines* disponíveis no mercado.

NOTAS

¹*Storyboards*: Sequência narrativa feita por meio de desenhos referente ao jogo (PERUCIA et al., 2005).

²*Bytecodes*: um arquivo intermediário, nem texto nem executável (MENDES, 2009).

REFERÊNCIAS

AMAZONAS, D. Desenvolvimento de Jogos 3D e Java com a Utilização do Motor Gráfico Irrlicht. Monografia (Bacharelado em Ciência da computação) – Programa de Bacharelado em Ciências da Computação, Faculdade Lourenço Filho de Fortaleza, 2007.

BALISTA, V. Desenvolvimento de Jogos Eletrônicos: Um Estudo Relacional entre Gerenciamento de Escopo e Custos de Desenvolvimento. In: XII SIMPÓSIO BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL, São Paulo, Brasil, 2013.

BATISTA, M. P. L. Um Estudo Sobre a História dos Jogos Eletrônicos. Faculdade Metodista Granbery, Juiz de Fora, 2007.

BEATY, B. How Canadians Communicate. 3. ed.; AU Press Athabasca University, Edmonton, 2010.

BRATHWAITE, B.; SCHREIBER, I. Challenges For Game Designers: Non-digital Exercises For Vídeo Game Designers. Boston: Cengage Learning, 2009.

CLUA, E.; BITTENCOURT, J. Desenvolvimento de Jogos 3D: Concepção, Design e Programação. In: ANAIS DA XXIV JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA DO CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, São Leopoldo, Brasil, 2005.

CROWLEY, K. Flexible Strategy Use In Young Children's Tic-Tat-Toe. Cognitive Science, 17, p.531-561, 1993.

DEITEL, H. M.; DEITEL, P. J. Java: Como Programar. 8. ed., São Paulo: Prentice-Hall, 2010.

FULLERTON, T. Game Design Workshop: A Playcentric Approach to Creating Innovative Games. 2. ed., Burlington: Elsevier, 2008.

JUNIOR, P. J. Java: Guia do Programador. 2. ed., São Paulo: Novatec, 2014.

KUROSU, M. Human-Computer Interaction: Applications And Services.16. International Conference, Heraklion, Crete, Greece, 2014.

LEITE, L. C. Jogos Eletrônicos Multi-Plataforma.Dissertação (Mestrado em Artes) – Programa de Pós-Graduação em Artes, Pontifícia Universidade Católica do Rio de Janeiro, 2006.

LEMES, D. Games Independentes: Fundamentos Metodológicos para Criação, Planejamento e Desenvolvimentos de Jogos Digitais. Dissertação (mestrado em Tecnologias da Inteligência e Design Digital) – Programa de Pós-Graduação em Tecnologias da Inteligência e Design Digital, Pontifícia Universidade Católica de São Paulo, 2009.

LOWOOD, H. Debugging Game History: A Critical Lexicon. Cambridge: MIT Press, 2016.

MARIASCH, S. Análises de Modelos de Documentos de Game Design e Proposta de Padrão Unificado. In: VII JORNADA DE INICIAÇÃO CIENTÍFICA, Universidade Presbiteriana Mackenzie, Brasil, 2011.

MASTROCOLA, V. Ludificador: Um Guia de Referências para o Game Design Brasileiro. São Paulo: Independente, 2012.

MENDES, D. R. Programação Java com Ênfase em Orientação a Objeto. São Paulo: Novatec, 2009

NETO, F. S. Interfaces de Usuário e Jogos Digitais: Possibilidades de Aprendizagem. Dissertação (mestrado em Modelagem Computacional Tecnologia Industrial) – Programa de Pós-Graduação em Modelagem Computacional

Tecnologia Industrial, Pontifícia Faculdade de Tecnologia SENAI/CIMATEC Salvador, 2011.

OLIVEIRA, M., STEED, A. Networked Graphics: Building Networked Games And Virtual Environments. Burlington: Morgan Kaufmann, 2010.

PERUCIA, A., BERTHÊM, A.; BERTSCHINGER, B.; CASTRO, R. Desenvolvimento de Jogos Eletrônicos: Teoria e Prática. São Paulo: Novatec, 2005.

ROLLINGS, A.; MORRIS, D. Game Architecture And Design: A New Edition. Indianápolis: New Riders, 2004.

ROUSE III, R. Game Design: Theory & Practice. 2ª Ed., Plano: Wordware Publishing Inc., 2005.

SALEN, K.; ZIMMERMAN, E. Rules Of Play: Game Design Fundamentals. Massachusetts: The MIT Press Cambridge, 2004.

SATO, A. Game Design e Prototipagem: Conceitos e Aplicações ao Longo do Processo Projetual. In: IX SIMPÓSIO BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL, Florianópolis, Brasil, 2010.

SELING, J. Desenvolvimento de um Jogo 3D. Monografia (Bacharelado em Informática) – Programa de Bacharelado em Informática, Universidade da Região da Campanha de Bagé, 2009.

SERSON, R. R. Programação Orientada a Objetos com Java 6. Rio de Janeiro: Brasport, 2007.

STANTON, R. A Brief History Of Video Games: From Atari To Xbox One. London: Robinson, 2015.

TAVARES, R.; NEVES, F. Introdução ao Game Design Através da Modificação de Um real 2004. In: SIMPÓSIO BRASILEIRO DE JOGOS PARA COMPUTADOR E ENTRETENIMENTO DIGITAL, Recife, Brasil, 2006.

TOCCHIO, L. Vídeo Game Design: Uma Análise da Estética Conceitual do Entretenimento Digital. Trabalho de Conclusão de Curso (Especialização em Mídias Interativas)– Programa de Especialização em Mídias Interativas, Centro Universitário SENAC de São Paulo, 2007.

WARBURTON, R. Java 8 Lambdas: Functional Programming For The Masses. Sebastopol: O'Reilly Media, 2014.