

Sistema Especialista para Apoio à Produção e à Gestão de Projetos de Jogos Móveis

Rafael Oliveira Martins

Universidade Presbiteriana Mackenzie
Programa de Pós-Graduação em Engenharia Elétrica
São Paulo, Brasil
E-mail: ra.martins13@gmail.com

Resumo—Atualmente, a definição de uma plataforma para desenvolvimento de jogos digitais e suas respectivas ferramentas de apoio é uma tarefa complexa. Essa dissertação propõe um modelo de sistema especialista capaz de estruturar o projeto de um jogo digital baseado na entrada de um usuário (acadêmico ou profissional), com enfoque em três áreas principais: plataformas e motores de desenvolvimento, programação e gestão de projetos. O objetivo do sistema será validado a partir de uma base de conhecimento, incluindo plataformas atuais existentes e regras de negócio baseadas em experiências de pesquisas e do mercado atual.

Palavras-Chave—sistemas especialistas; inteligência artificial; plataformas móveis; jogos digitais.

I. INTRODUÇÃO

Com o crescimento da indústria de fabricantes de tecnologia móvel, do investimento de empresas na mobilidade e na atual necessidade do mercado no consumo de aplicativos e jogos, muitas plataformas e dispositivos surgem para competir e atrair usuários e desenvolvedores. A área de jogos digitais é uma das mais beneficiadas com esse crescimento, pois com a alta performance dos dispositivos atuais, jogos que antes só eram possíveis de serem jogados em computadores também são desenvolvidos para telas menores. Além disso, essas plataformas oferecem atualmente jogos casuais usados por milhões de pessoas ao mesmo tempo, tornando isso um objeto de estudo para outros domínios do conhecimento [1].

O processo de desenvolvimento de jogos digitais envolve coordenar tarefas multidisciplinares e também um número amplo de habilidades, transformando um conjunto de ideias em um sistema complexo de tarefas coordenadas e comportamentos diversos [2]. No ambiente acadêmico, estruturar uma grade curricular para um curso de jogos é uma tarefa complexa em consequência do restrito número de profissionais no mercado com formação acadêmica em jogos. Além disso, há informações retidas em ambientes corporativos que não são compartilhadas diretamente com instituições de ensino. Já em empresas de pequeno porte, são comuns problemas de organização em projetos e definição de escopo, devido ao elevado número de fabricantes e plataformas de desenvolvimento, portanto, vivenciam alguns problemas em comum com a área acadêmica.

Pode-se, então, definir alguns tipos de problemas encontrados nos dois ambientes citados acima:

- Falta de informação, profissionais, tecnologia ou apoio na produção de jogos acadêmicos ou de pequenas empresas;
- Elevado número de plataformas, ambientes, ferramentas e fabricantes de dispositivos;
- Riscos de investimento em jogos ou projetos que envolvem tarefas complexas e multidisciplinares.

Baseado em pesquisas científicas e experiência no desenvolvimento de jogos em ambientes diversos, o autor propõe um modelo de sistema especialista para tentar gerar conhecimento, guiando o usuário na construção de um projeto de jogo em ambientes móveis e plataformas emergentes, priorizando as áreas de plataformas de desenvolvimento, programação e gestão de projetos.

II. CONTEXTUALIZAÇÃO DOS TEMAS RELACIONADOS

A. Plataformas de Desenvolvimento

Atualmente existem diversas plataformas de jogos digitais, que oferecem suporte desde jogos para computadores pessoais e Internet até celulares e novos dispositivos, que são provenientes de pesquisa ou investimento de um determinado fundo, como é o caso do *Ouya* (disponível em <http://www.ouya.tv>). As plataformas que mais têm recebido atenção são as móveis, que incluem como principais sistemas iOS (Apple, 2007), Android (Google, 2008) e Windows Phone (Microsoft, 2010).

Com o crescimento da indústria de fabricantes de tecnologia móvel, do investimento de empresas na mobilidade e na atual necessidade do mercado no consumo em aplicativos e jogos, muitas plataformas surgiram para competir e atrair usuários e desenvolvedores. Contudo, cabe destacar que os aplicativos móveis não são uma novidade. Muitas plataformas estiveram presentes por algum tempo na história. Em particular, a Tabela 1 define uma linha do tempo entre as tecnologias e principais características das plataformas móveis [3].

Eras de Desenvolvimento de Aplicações Móveis		
Era	Linha do Tempo	Características
Candy Bar Phones	1988-1998	Rede limitada para prover serviços
Feature Phones	1998-2002	WML, MMS/SMS, telas pequenas
SmartPhones	2002-Presente	Java-ME, Windows Mobile, Palm OS, RIM
TouchScreen	2009-Presente	iPhone, Android, Windows Phone

^a Tabela com as eras de desenvolvimento de aplicações móveis.

Desde a chegada das plataformas dominantes (citadas acima), o uso do celular e o desenvolvimento de aplicações móveis obteve um crescimento exponencial. O desenvolvimento dessas aplicações se tornou mais amigável, mas ainda possui complexidade e alguns problemas persistentes, como a fragmentação dos fabricantes [7].

Do ponto de vista de consumo, o número crescente dessas plataformas atrai clientes e consumidores, que por sua vez, têm podem escolher conforme suas conveniências. Do ponto de vista tecnológico, o número elevado de plataformas pode oferecer sempre uma nova opção de distribuição e revenda, porém, isso está diretamente ligado com o tempo requerido pelas empresas e profissionais para aprender e absorver conhecimentos em uma nova tecnologia.

B. Gestão de Projetos

Uma das etapas mais complexas no processo de desenvolvimento de um jogo é o seu planejamento e gerenciamento. Existem várias técnicas de gerenciamento de projetos, também chamadas de metodologias, que auxiliam na organização de equipes de produção. Uma das metodologias mais utilizadas é a *Agile*, que funciona como uma série de iterações de desenvolvimento. Essas iterações tem um tempo reduzido, geralmente duas ou quatro semanas, durante a construção do projeto [9].

Essas iterações, também são chamadas de *sprints*, são ciclos de produção envolvendo times com o objetivo de completar uma determinada tarefa. Esses objetivos não podem ser alterados durante a execução da *sprint* [9]. Dentro de cada ciclo pode-se destacar as seguintes métricas:

Lista de Funcionalidades: contém detalhes e requirements do projeto e devem ser implementadas de acordo com a prioridade de cada *sprint* do projeto.

Organização de tarefas: realizada por divisão de tarefas. Cada tarefa faz parte de uma disciplina específica, como, por exemplo, programação, som ou arte. Cada líder de equipe define como essas tarefas serão realizadas.

Tarefas associadas: O líder de cada grupo associa os membros às tarefas específicas e deve trabalhar para que estas sejam cumpridas no prazo estipulado.

Dependências: O gerente de projeto é responsável por avaliar a distribuição de tarefas do projeto feitas pelo líder de cada equipe e garantir que não haja dependências entre tarefas correlacionadas.

Agendamento/Escalonamento: Como todo projeto, o escalonamento de tarefas é a etapa que envolve maiores riscos para o projeto. Nos meios tradicionais de gerência de projeto, o modelo cascata [9] é utilizado para gerenciamento do tempo. A metodologia *Agile* propõe uma forma mais extensa de avaliação do tempo, porém se atenta a detalhes como férias de funcionários, possíveis ausências por doença ou atrasos comuns em algumas tarefas, o que auxilia no processo de detecção de riscos e otimização do calendário.

C. Sistemas Especialistas

Sistemas Especialistas foram criados para dar suporte à resolução de problemas em domínios específicos utilizando conhecimento humano. Para o desenvolvimento de um sistema especialista são necessários profissionais que atuem com programação, engenharia de conhecimento, gerentes e usuários do domínio definido, no caso, o objetivo do sistema. Os sistemas especialistas são uma das técnicas mais conhecidas de Inteligência Artificial e estão sendo estudados desde a década de 80.

Os sistemas especialistas são definidos como incorporadores de informações de utilidade e têm capacidades adicionais em comparação com sistemas de inferência puros, pois eles podem utilizar o valor da informação para determinar se devem adquiri-la ou não [2]. De forma complementar, pode-se dizer que um sistema especialista é um sistema computacional capaz de emular a maneira como os especialistas humanos resolvem problemas. A principal tarefa de desenvolvimento de um sistema especialista é, portanto, adquirir informação de um especialista humano e modelar computacionalmente a realização de tarefas cognitivas de forma eficiente [5].

A mecânica de um sistema especialista é composta pelos seguintes elementos: uma base de conhecimento, geralmente alimentada por especialistas em um determinado domínio; uma interface, para que o usuário entre com os dados que serão lidos para avaliação e um motor de inferência, para resolução de conflitos e execução do domínio de regras. Em termos mais gerais:

Bases de conhecimento: consistem na codificação dos domínios de especialidade. Essa base pode estar na forma de redes semânticas, representações procedurais, regras de produção ou quadros de conhecimento [3].

Interface: É a intermediadora entre o usuário e o programa que contém a base de conhecimento e as regras de produção. Por isso, as interfaces construídas para os sistemas inteligentes devem ser intuitivas e amigáveis para a utilização do usuário ou especialista. Como o processo de interação com o sistema pode variar de acordo com a entrada do usuário, as diversas regras dentro do domínio de conhecimento e a diferente contextualização de alguns problemas são importantes para projetos de interface, fazendo com que essa etapa seja essencial [4].

Motor de inferência: É o objeto computacional responsável por localizar a informação na base de conhecimento e inferir regras e novos conhecimentos, se utilizando de processamento lógico e esquemas para solução de problemas em um domínio.

As regras que serão examinadas pelo motor de inferência devem satisfazer pelo menos uma condição dentro do conjunto de regras de produção. Para trabalhar com as regras de produção, existem algumas técnicas conhecidas utilizadas em sistemas inteligentes como a *Backward Chaining*, *Forward Chaining*, entre outras.

O método de *Backward Chaining* é utilizado quando se tem o conhecimento de todos os fatos iniciais, mas não há o que concluir desses fatos. Esse método pode construir novas hipóteses para um dado existente, adicionando essas conclusões para a memória de trabalho [7]. Ampliando a definição do método de *Backward Chaining*, pode-se dizer que é uma técnica desenhada para fazer a conversão entre o usuário e o especialista humano. O método habilita os sistemas para saber quais questões responder e quando [8].

Quando os fatos no sistema estão representados na memória de trabalho e são continuamente atualizados, utiliza-se o método de *Forward Chaining*. As regras representam no sistema possíveis ações que podem ser tomadas quando uma determinada condição é aceita. Essas condições são padrões que devem se igualar aos itens na memória do sistema [7].

III. METODOLOGIA

Para construção do sistema especialista *MackGameExpert*, a solução técnica escolhida é o *Jess* (disponível em <http://www.jessrules.com>) pois oferece um ambiente para customização baseada em regras de produção. O *Jess* foi escrito na linguagem Java (Oracle) a partir da versão de seu antecessor *CLIPS* (<http://clipsrules.sourceforge.net>). Para geração de gráficos de informação, diferenciação por cores e outras funções extras visuais, o plug-in utilizado é o *GEF* (<http://www.eclipse.org/gef>).

Para prototipação do modelo computacional, são necessários conhecimentos de especialistas na área de jogos e inteligência artificial, programadores (para o modelo e para a aplicação em jogos) e de gestores de projeto, além de pesquisas acadêmicas e de mercado. O projeto é dividido em seis fases utilizando o modelo de Turban (1993), identificado como boas práticas para a construção de sistemas especialistas [6].

A. Fase 1: Planejamento

A definição do problema é o ponto chave da fase inicial. Identificou-se como problema principal a elevada quantidade de plataformas de desenvolvimento de jogos e o déficit acadêmico nacional e de pequenas empresas, mas alguns outros pontos podem ser destacados: 1) Falta de profissionais qualificados; 2) Falta de investimento no mercado nacional; 3) Problemas comuns em projetos complexos como programação e gestão de projetos; 4) Jogos são atividades multidisciplinares e complexas, logo requer uma atenção grande em sua construção.

Outros fatores como necessidade da ferramenta e avaliação sobre possíveis soluções alternativas também é levada em consideração, porém, como os sistemas especialistas na área de jogos em sua maioria resolvem problemas relacionados à mecânica, a proposta é válida como ferramenta de pré-produção.

O gerenciamento de riscos pode ser feito por meio do valor do retorno da informação ao usuário. Como o sistema apresentará uma elevada quantidade de nós de informações, a etapa de aquisição do conhecimento e de implementação poderão ajustar de forma mais eficiente os possíveis problemas e riscos ao projeto.

B. Fase 2: Análise do Sistema

Essa fase destaca o modelo conceitual como descritor das funções principais do sistema e seu objetivo, assim como os recursos necessários para sua construção.

No sistema proposto, as funcionalidades são: Leitura de dados do teclado do usuário; Avaliação da entrada e novas telas com possíveis escolhas; Aquisição de conhecimento, em caso de não haver certo dado de entrada; Consulta à base de conhecimento e disparo das regras de produção para validar a informação; Avaliação final, compilação de relatório e nós de informação ao usuário via interface visual.

C. Fase 3: Prototipação

Um protótipo é necessário para validar questões técnicas, de custo e operacionais antes do crescimento potencial do projeto. Para isso, um modelo simples foi desenvolvido e testado utilizando *Jess*, *Eclipse* e o *Excel*, a fim de obter informações sobre configuração de ambiente, aplicações de regras básicas (soma, subtração) e resultados visuais como gráficos e uso de cores específicas.

D. Fase 4: Desenvolvimento do Sistema

A primeira etapa dessa fase é a construção da base de conhecimento. A base de conhecimento é separada das regras de produção, pois precisam ser atualizadas conforme a demanda de conhecimento aumenta.

No caso do *Jess*, a base de conhecimento é definida como 'fatos' e são declaradas como classes Java. As regras de produção (Fig. 1), como descritas na contextualização dos tópicos abordados, são ativadas pelo motor de inferência e seguem o padrão condição-conclusão (*if* condição *then* conclusão).

```

if platform == ANDROID
then manufacturer = GOOGLE &
languageCore = JAVA &
languageNative = C++ &
render3D = OPEN_GL
mainDeviceManufacturer = SAMSUNG
...

if platform == WINDOWS_PHONE
then manufacturer = MICROSOFT &
languageCore = C# &
languageNative = C++ &
render3D = DIRECTX
mainDeviceManufacturer = HTC
...

```

Figura 1. Trecho de regra de produção

O fluxo a seguir representa a comunicação da interface com os outros mecanismos do sistema. Dado a complexidade do sistema e suas diversas bases de conhecimento, a Fig. 2 exemplifica algumas bases de conhecimento e o fluxo de entrada, decisão, saída e avaliação.

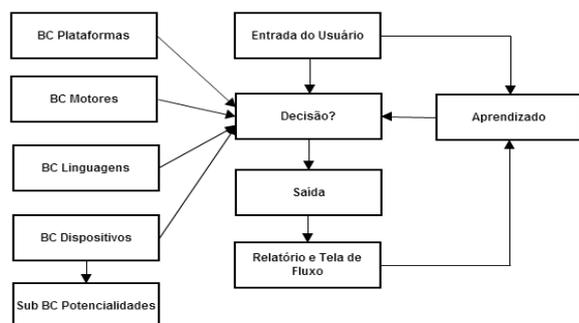


Figura 2. Fluxo reduzido de funcionamento do *MackGameExpert*

E. Fase 5: Implementação

A implementação do sistema deve ocorrer na fase de apresentação da dissertação e sua versão alfa na qualificação do trabalho. Existem requisitos funcionais necessários para que a implementação ocorra de forma eficiente: Aceitação de usuários em testar o sistema; Instalação e viabilização do sistema; Treinamento e segurança; Documentação.

F. Fase 6: Correções e Otimizações

Essa fase também pode ser chamada de manutenção do sistema. No caso do sistema proposto, as atualizações podem ser constantes, dado o grande número de ferramentas, tecnologias e plataformas de desenvolvimento. Além disso, as técnicas de programação também evoluem com o passar do tempo, portanto, são passíveis de atualização. Por fim, as regras para gerência de projeto podem ser atualizadas em um ritmo mais lento, pois as metodologias propostas são relativamente novas e cumprem bem seu papel durante o projeto.

IV. RESULTADOS ESPERADOS

Os resultados esperados são pertinentes ao desenvolvimento de um sistema (*MackGameExpert*) que permita ao usuário gerar as informações necessárias para a construção de um jogo em uma plataforma móvel (ou digital) resolvendo alguns dos problemas mais convencionais de pesquisa e mercado. O sistema deve plotar informações construtivas para que o usuário inicie um projeto de forma autônoma.

Para isso, o fluxo de funcionamento do sistema deve seguir os padrões dos principais sistemas especialistas. O primeiro passo é a entrada do usuário pela interface do sistema. Nessa etapa, o usuário deverá preencher os campos de plataforma, tipo de jogo, linguagem de programação e público alvo. Em seguida, as regras de produção são acionadas para comparação da entrada com as informações contidas nas bases de conhecimento. Essas bases são divididas em bases únicas de informação ou bases compostas (informações com dependência, como, por exemplo, dispositivos e suas funcionalidades). A saída do sistema, então, gera uma tela de relatório e uma interface gráfica com nós de informação para que o usuário se guie no desenvolvimento de um projeto de jogo, seguindo os requisitos mínimos.

O resultado visual esperado é uma interface onde o usuário possa visualizar toda fase de desenvolvimento de um jogo, dividido em etapas utilizando o conceito de metodologia Ágil. Sua vantagem reside na possibilidade de incluir o conhecimento de gestão de projetos de forma prática, direcionando o usuário a um planejamento eficiente (Fig. 3).

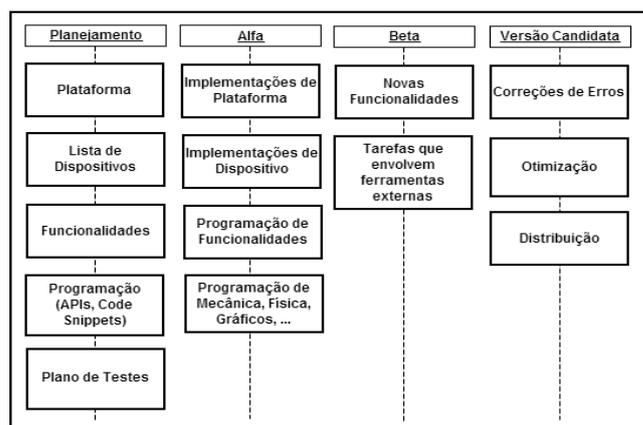


Figura 3. Resultado esperado para interface de informação

Espera-se propor uma interface capaz de auxiliar: acadêmicos/profissionais iniciantes na construção de projetos mais eficientes; profissionais técnicos a buscarem soluções mais rápidas para determinadas plataformas; gerentes de projetos na busca de melhores metodologias para um problema específico. Também pode ser útil tanto no desenvolvimento de um projeto de jogo ou como demonstração da aplicação de técnicas de inteligência artificial em um sistema mais complexo. Em médio prazo, o sistema pode incorporar novos módulos como game design, estratégias de marketing e vendas, etc. a fim de expandir sua base de conhecimento.

REFERÊNCIAS

- [1] Baniqued, P. et al. "Selling points: What cognitive abilities are tapped by casual video games?" *Acta Psychologica*, v. 142, 74-86, Jan 2013.
- [2] Norvig P., Russell S, *Artificial Intelligence: A Modern Approach*, 2nd. Edition, Elsevier, Editora Campus, 2004.
- [3] Griffin N. L., Lewis F. D. "A Rule-Based Inference Engine which is Optimal and VLSI Implementable". Department of Computer Science. University of Kentucky, Lexington, Kentucky.
- [4] Preece J., Rogers Y., Sharp H., Benyon D., Holland S., Carey T. *Human Computer Interaction*, 1st. Edition, Addison Wesley, 1994.
- [5] Hull L. G., Kay P. "Expert System Development Methodology and Management" National Aeronautics and Space Administration, GSFC, Greenbelt, Maryland, 1991.
- [6] Turban E. *Decisions Support and Expert Systems: Management Support Systems*. Macmillan, 3rd Edition, New York and Toronto, 1993.
- [7] Anwar Z. "Forward and Backward Chaining". Systems Research Group. University of Illinois. Illinois, IL., Lecture 12, 7-12. 2008.
- [8] Huntington D. "Back to Basics - Backward Chaining: Expert System Fundamentals". PCAI 16.
- [9] Keith C. *Agile Game Development with Scrum*. Boston, MA. :Addison Wesley / Pearson Education. 2010.
- [10] Universidade Federal de Santa Catarina. *Inteligência Artificial*. Jess. Disponível em: <http://www.das.ufsc.br/~gb/pg-ia/Jess/JessJava.txt>.
- [11] Tracy W. K. "Mobile Application Development Experiences on Apple's iOS and Android OS". *IEEE Potentials*, v. 31, issue 4, 30-34, 2012.

Dados

Pesquisador: Rafael Oliveira Martins

e-mail: ra.martins13@gmail.com

CV Lattes: <http://lattes.cnpq.br/3896697336415839>

Título da pesquisa: Sistema Especialista para Apoio à
Produção e à Gestão de Jogos Digitais Móveis

Orientadora: Profa. Dra. Pollyana Nostargiacomo Mustaro

e-mail: pollyana.mustaro@mackenzie.br

CV Lattes: <http://lattes.cnpq.br/5131975026612008>

Universidade Presbiteriana Mackenzie

Programa de Pós Graduação em Engenharia Elétrica

http://www.mackenzie.br/engenharia_eletrica0.html