

HEBER FERNANDES AMARAL

**ARQUITETURA PARA JOGOS COMO FACILITADORES NO PROCESSO DE
ENSINO-APRENDIZAGEM: UM ESTUDO DE CASO COM DISCIPLINA DE
CURSO DE GRADUAÇÃO**

Dissertação apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para obtenção do título de *Magister Scientiae*.

VIÇOSA
MINAS GERAIS – BRASIL
2014

**Ficha catalográfica preparada pela Biblioteca Central da Universidade
Federal de Viçosa - Câmpus Viçosa**

T

A485a
2014
Amaral, Heber Fernandes, 1980-
Arquitetura para jogos como facilitadores no processo de ensino-aprendizagem : um estudo de caso com disciplina de curso de graduação / Heber Fernandes Amaral. – Viçosa, MG, 2014.

x, 36 f. : il. (algumas color.) ; 29 cm.

Orientador: José Luis Braga.

Dissertação (mestrado) - Universidade Federal de Viçosa.

Referências bibliográficas: f. 34-36.

1. Arquitetura de software. 2. Software educacional - Jogos por computador. 3. Administração rural. 4. Métodos de simulação. I. Universidade Federal de Viçosa. Departamento de Informática. Programa de Pós-Graduação Mestrado em Ciência da Computação. II. Título.

CDD 22 ed. 005.1

“Não há saber mais ou saber menos: Há saberes diferentes.”

(Paulo Freire)

A minha esposa Carla por todo apoio, dedicação e cuidado. Ao meu filho João Lucas por toda inspiração emanada de seus sorrisos e abraços...

Agradecimentos

A Deus por todo cuidado, amor e por se fazer presente na minha vida, agradeço também pelas oportunidades das batalhas diárias que se fizeram necessárias para que eu estivesse aqui.

A minha esposa, Carla, por todo apoio e encorajamento sem o qual eu não estaria aqui. Ao meu filho João Lucas pelos sorrisos e abraços que renova minha forças e me inspiram.

Aos meus pais, Judas Tadeu e Margarida, pelos bons ensinamentos, pelo amor, carinho e principalmente por todo esforço na educação.

Aos meus irmãos, Vanessa e Marcos Vinícius, pelo apoio e por estarem presentes em minha vida.

Ao Professor José Luis, por ser um exemplo de pessoa e um grande orientador. Obrigado pela atenção, paciência, motivação e ajuda.

Ao Professor Aziz, pela oportunidade de desenvolver esse trabalho.

Ao Professor Vicente Lelis por todo aprendizado adquirido no trabalho realizado no Projeto Reuni.

Aos meus amigos, pela motivação, pelos bons momentos juntos e por estarem sempre presentes nas maiores dificuldades.

A todos que, diretamente ou indiretamente, deram sua contribuição para o desenvolvimento deste trabalho.

Sumário

Lista de Figuras	vii
Lista de Tabelas	viii
Resumo	ix
Abstract	x
1 Introdução	1
1.1 Motivação	1
1.2 Objetivo	3
1.3 Objetivos Específicos	3
1.4 Como o Texto está Organizado	3
2 Referencial Teórico	4
2.1 Arquitetura de Software	4
2.2 Jogos Educacionais	5
2.3 Trabalhos Relacionados	6
3 Arquitetura para jogos educacionais de simulação e estratégia	10
3.1 Levantamento e Análise de Requisitos	10
3.2 Projeto da Arquitetura de Software	12
4 Detalhes de Implementação	19
5 Adaptabilidade	24
5.1 Introdução	24
5.2 Trabalhos Correlatos	25

5.3 Jogos Educacionais	26
5.4 Modelagem do Sistema de Ajuda	27
5.5 Resultados e Conclusões	29
6 Conclusões e Trabalhos Futuros	32
Referências Bibliográficas	34

Lista de Figuras

2.1	Roteiro para o desenvolvimento de um jogo educacional Fonte: Levantado a partir do trabalho de Moreno-Ger et al. (Moreno-Ger et al., 2008)	7
3.1	Arquitetura para jogos educacionais. Fonte: Wenfeng Hu (Hu, 2010)	13
3.2	Input-Process-Outcome Game Model Fonte:GARRIS Garris et al. (2002)	14
3.3	Subsistema de conteúdo. Fonte: Desenvolvido pelo autor.	15
3.4	Subsistema de atividades. Fonte: Desenvolvido pelo autor.	16
3.5	Subsistema de feedback. Fonte: Desenvolvido pelo autor.	17
3.6	Fluxo do Jogo.	17
4.1	PROGRAMAÇÃO DINÂMICA (MULTIPERIÓDICA)	19
4.2	Tela do jogo de Administração Rural. Fonte: Desenvolvido pelo autor.	21
4.3	Tela do Jogo ao final da primeira rodada. Fonte: Desenvolvido pelo autor.	23
5.1	Tela do jogo de Administração Rural. Fonte: Desenvolvido pelo autor.	26
5.2	Algoritmo Desenvolvido.	28
5.3	Árvore minimax gerada. Fonte: Desenvolvido pelo autor.	30
5.4	Exemplo da função dica. Fonte: Desenvolvido pelo autor.	31

Lista de Tabelas

2.1	A relação entre técnicas de aprendizagem, atividades de aprendizagem e possível gênero de jogo Extraídos de Rapeepisarn et al. (Rapeepisarn et al., 2008)	8
2.2	Principais requisitos educacionais para jogos (Moreno-Ger et al., 2008)	9
2.3	Principais requisitos para Jogos de Simulação (Peixoto et al., 2012)	9
3.1	Principais requisitos para jogos educacionais (Hu, 2010)	11
3.2	Roteiro de uso da arquitetura	18
4.1	Recursos utilizados e receita produzida por cada cultura no jogo. Extraídos de Resende et. al. (REZENDE et al., 2005) e Embrapa (Embrapa, 2007)	22
4.2	Custo de cada ação. Extraídos do Portal do Governo do Estado de São Paulo. (do estado de SÃO Paulo, 2012) e Banco do Brasil (do Brasil, 2012)	22
5.1	Coefficientes climáticos das atividades. Extraídos de Resende et. al. Embrapa (Embrapa, 2007)	27
5.2	Climas usados na simulação	29

RESUMO

AMARAL, Heber Fernandes, M.Sc., Universidade Federal de Viçosa, janeiro de 2014. **Arquitetura para jogos como facilitadores no processo de ensino-aprendizagem: um estudo de caso com disciplina de curso de graduação.** Orientador: José Luis Braga. Coorientadores: Aziz Galvao da Silva Junior.

Jogos educativos são ferramentas importantes para auxiliar o processo de ensino e aprendizagem, pois permitem simular conceitos em um ambiente virtual e promover a colaboração e a comunicação entre os alunos. Porém, deve-se ter em mente que jogos educacionais eletrônicos são softwares e, por sua vez, seu desenvolvimento sofre os mesmos problemas e dificuldades já conhecidos na engenharia de software quando se trata de desenvolvimento de softwares tradicionais. Soma-se a esses problemas a necessidade de se atender aos requisitos educacionais e também aos requisitos específicos de um jogo, sendo que requisitos educacionais consistem em todo o conteúdo programático de determinada matéria ou curso e requisitos de jogos são todas as características que diferenciam um software de um jogo. O presente trabalho envolve a elaboração de uma arquitetura de software para auxiliar ou reduzir o retrabalho no desenvolvimento de jogos educacionais de simulação e estratégia. E para validar essa arquitetura foi desenvolvido um jogo para uma disciplina de administração rural como evidência da aplicabilidade da arquitetura proposta.

ABSTRACT

AMARAL, Heber Fernandes, M.Sc., Universidade Federal de Viçosa, January of 2014. **Game Architecture for teaching-learning process: an application on an undergraduate course.** Adviser: José Luis Braga. Co-advisers: Aziz Galvao da Silva Junior.

Educational games are important tools to assist the process of teaching and learning as they allow simulating concepts in a virtual environment and foster collaboration and communication among students. But, in order to develop an educational game, one should have in mind that electronic games are softwares and in turn their development suffers the same problems and difficulties known in software engineering when it comes to traditional software development. Added to these problems there is the need to meet the educational and also the specific requirements of a game: educational requirements consist of an entire syllabus of a particular subject or course requirements and the gaming factor is one of the features that differentiates a software from a game. This work proposes an educational game architecture aimed to help or reduce rework in the development of simulation and strategy educational games. We also describe an experience of developing a game about rural management using the proposed architecture as evidence of the applicability of the proposed architecture.

Capítulo 1

Introdução

Muitos problemas das diversas áreas da ciência como engenharia, saúde, negócios e outras áreas podem ser resolvidos de forma eficaz com o uso de computadores, mas encontrar uma solução exige conhecimentos de ciência da computação e conhecimento do domínio de aplicação particular. Para tal umas das especialidades da ciência da computação é a Engenharia de Software, que segundo a ACM (*Association for Computing Machinery*) (ACM, 2012) está preocupada com o desenvolvimento e manutenção de sistemas de software que se comportam de forma confiável, eficiente e acessível para desenvolver, manter e satisfazer todos os requisitos definidos pelo cliente visando atender suas demandas.

Assim o presente trabalho usa dos conhecimentos da engenharia de software para propor uma Arquitetura de Software para jogos educacionais usando como estudo de caso o desenvolvimento de um jogo para disciplinas de economia rural.

1.1 Motivação

Jogos educacionais têm o potencial de fornecer experiência motivadora de aprendizagem, além de ajudarem a dar sentido à experiência e organizar o conhecimento, despertando habilidades para resolução de problemas e aumentando a motivação. Permitem ainda que ocorra um processo de aprendizagem não intencional, por meio de uma experiência envolvente e atraente, em que quem aprende é visto como participante ativo na construção de seu próprio conhecimento (Hodhod et al., 2011).

Além das vantagens citadas acima, de acordo com Prensky (Prensky, 2003 apud POSSA, 2011), os "Nativos Digitais" jogaram muitos jogos eletrônicos na infância e sabem como lidar com uma grande quantidade de informação de forma rápida, usando caminhos alternativos para obter informações. Essa nova geração prefere fazer várias coisas simultaneamente usando vários caminhos para uma mesma meta, em vez de fazer uma coisa de cada

vez seguindo passos sequenciais. Eles preferem ser ativos, aprender através de tentativa e erro, e descobrir coisas por conta própria ao invés de ler e ouvir. Eles querem ser tratados como "criadores e fazedores", em vez de "receptáculos a serem preenchidos com conteúdo".

Dessa forma as instituições de ensino e também as organizações em todos os seus níveis, têm tirado proveito das potencialidades dos jogos educacionais. As empresas fazem uso de tais jogos para treinar funcionários aproveitando todo o seu potencial lúdico e imersivo. Como também poderão usar para simular situações diversas de risco material e até mesmo risco de vida, usando os jogos para treinar seu pessoal para lidar com situações que na vida real seriam impossíveis de serem simuladas sem um alto risco de perdas financeiras ou mesmo de vidas humanas. Um ambiente onde é possível errar sem que os resultados sejam catastróficos para o negócio ou para vida poderá ter grande utilidade no aprimoramento dos conhecimentos adquiridos teoricamente em um ambiente universitário, colegial ou técnico.

Tais vantagens têm levado muitas instituições de ensino nos diversos níveis e áreas a desenvolverem jogos educacionais. Contudo, um jogo de computador é um software que mantém um mundo virtual com o qual os jogadores interagem para atingir algumas metas. As características de tais jogos são um alto grau de interatividade, gráficos avançados, um universo virtual altamente dinâmico e até mesmo inteligente (Hu, 2010).

Assim, por serem software, seu desenvolvimento sofre os mesmos problemas e dificuldades já conhecidos na engenharia de software quando se trata de desenvolvimento de softwares tradicionais (Hu, 2010). Tais problemas são abordados na extensa literatura de engenharia de software. Soma-se a esses problemas a necessidade de se atender aos requisitos educacionais e também aos requisitos específicos de um jogo, sendo que, requisitos educacionais consistem do conteúdo programático de determinada matéria ou curso e requisitos de jogos são todas as características que diferenciam um software de um jogo como por exemplo imersão e jogabilidade que são, respectivamente, a capacidade de um sistema de trazer seus espectadores ou usuários para dentro da realidade (virtual) por ele construída e a forma específica em que os jogadores interagem com um jogo (Lindley, 2004). Diante de tal complexidade, de acordo com Wenfeng Hu (Hu, 2010), para obter-se um jogo educacional de alta qualidade com menor custo, tempo e alto padrão de qualidade, é necessário seguir os princípios da engenharia de software.

Segundo (Garlan & Shaw, 1993) e (Allen, 1997), quando a complexidade de um software cresce o problema vai além dos algoritmos e das estruturas de dados. À medida que a complexidade aumenta a estrutura geral do sistema torna-se o problema central do projeto. Para lidar com estas situações temos a Arquitetura de software que fornece um modelo das propriedades estruturais de forma abrangente. Tais propriedades incluem a decomposição e a interação entre as partes, bem como questões globais como: coordenação, sincronização e

desempenho do sistema.

Dessa forma é que se desenvolveu o presente trabalho, que envolve o desenvolvimento de uma arquitetura de software para auxiliar no desenvolvimento de jogos educacionais de simulação e estratégia. E para validar essa arquitetura foi desenvolvido um jogo para uma disciplina de administração rural como evidência da aplicabilidade da arquitetura proposta.

1.2 Objetivo

Obter uma especificação de uma arquitetura para jogos de simulação, para ser usada como ferramenta potencializadora no processo de ensino-aprendizagem.

1.3 Objetivos Específicos

- Estabelecer requisitos para uma arquitetura de jogos de simulação usados no processo de ensino e aprendizagem.
- Analisar tecnologias disponíveis para implementação de jogos e simulações aplicadas ao ensino.

1.4 Como o Texto está Organizado

Este trabalho foi dividido nos seguintes capítulos. No presente capítulo é feita a introdução e é levantada a problemática do trabalho aqui apresentado. No capítulo 2 define-se o embasamento teórico para nosso trabalho a partir de um estudo bibliográfico. O capítulo 3 mostra a implementação da arquitetura proposta para o caso de jogos de simulação e estratégia. No capítulo 4 é apresentado o uso da arquitetura proposta no capítulo 3 para a construção de um jogo educacional como ferramenta auxiliar no ensino de Administração Rural. No capítulo 5 é apresentado um trabalho publicado, derivado desta dissertação onde se demonstra a adaptabilidade da arquitetura implementando um sistema de ajuda ao jogador no jogo desenvolvido. Por fim, no capítulo 6 são apresentadas as conclusões.

Capítulo 2

Referencial Teórico

Neste capítulo, primeiramente, são apresentados conceitos de Arquitetura de Software e suas características. Na Seção 2.2 são apresentadas as características e definições de jogos educacionais e os diferentes gêneros de jogos, e, por fim, relatamos trabalhos mais diretamente relacionados ao assunto desta dissertação.

2.1 Arquitetura de Software

Vários pesquisadores têm proposto definições para o termo "arquitetura de software" ao longo dos últimos anos. Segundo Bass (Bass et al., 2003) a arquitetura de software de um programa ou de um sistema computacional é a estrutura ou as estruturas do sistema, que compreende elementos de software, as propriedades visíveis e externas desses elementos, e as relações entre eles. De acordo com KRAFZIG (Krafzig et al., 2005) uma arquitetura de software é um conjunto de declarações, que descreve os componentes de software e atribui funcionalidades de sistema para cada um deles. Ela descreve a estrutura técnica, limitações e características dos componentes, bem como as interfaces entre eles. A arquitetura é o esqueleto do sistema e, por isso, torna-se o plano de mais alto-nível da construção de cada novo sistema.

Como pode ser visto nas definições, uma arquitetura do software de um sistema é a estrutura geral do sistema, que compreende os componentes do software e as interações entre eles. A arquitetura de software é uma abstração de alto nível de um sistema que pode ser reutilizada no desenvolvimento de soluções semelhantes. Uma analogia fácil de ser entendida é feita por Pressman (Pressman, 2006) que diz que a arquitetura de software se assemelha ao projeto arquitetônico de um edifício, pois descreve toda a solução arquitetural do edifício suas características e a maneira como os vários componentes do edifício interagem, e tal solução pode ser implementada várias vezes pelos engenheiros em lugares diferentes,

desde que tenham as mesmas características. Assim uma arquitetura de software visa reduzir o retrabalho no projeto de um software propondo uma solução padrão para problemas iguais.

2.2 Jogos Educacionais

Metodologias tradicionais de ensino, centradas nos professores, não são motivadoras para os alunos. Vários métodos alternativos foram propostos para contornar esta situação. Dentre os métodos alternativos, a utilização de jogos educacionais computacionais é a mais atraente para a maioria dos alunos das novas gerações, uma vez que eles cresceram usando computadores, vídeo games, telefones celulares e outras ferramentas da era digital. Segundo Prensky (Prensky, 2003), os alunos de hoje, por terem este histórico, são chamados de "Nativos Digitais".

Segundo Prensky, (Prensky, 2003 apud POSSA, 2011) Os "Nativos Digitais" que jogaram muitos jogos eletrônicos na infância possuem habilidades como lidar com uma grande quantidade de informação de forma rápida, usando caminhos alternativos para obter informações. Esta nova geração prefere fazer várias coisas simultaneamente usando vários caminhos para uma mesma meta, em vez de fazer uma coisa de cada vez seguindo passos sequenciais. Eles preferem ser ativos, aprender através de tentativa e erro, e descobrir coisas por conta própria em vez de ler e ouvir. Eles querem ser tratados como "criadores e fazedores", em vez de "receptáculos a serem preenchidos com conteúdo".

Segundo Akilli (Akilli, 2011) o jogo é uma atividade competitiva, criativa e agradável em sua essência, que é limitado por certas regras e requer certas habilidades. Tais jogos possuem características que os diferenciam uns dos outros em gêneros. Os gêneros básicos de jogos eletrônicos, segundo levantamento bibliográfico (Arsenault, 2009; Myers et al., 1990; MobyGames, 2012) são: Ação, Aventura, Corrida, RPG, Simulação, Educacional, Esporte e Estratégia. Tais gêneros não são mutuamente exclusivos. Ao se classificar um jogo este poderá apresentar mais de um gênero em sua classificação (Arsenault, 2009; Myers et al., 1990; MobyGames, 2012). Nessa classificação encontramos o gênero Educacional que se refere a uma classificação quanto à finalidade do jogo, que consiste em ensinar algo (Arsenault, 2009; Myers et al., 1990; MobyGames, 2012). Porém busca-se uma classificação quanto à forma de se jogar e como as situações são apresentadas no jogo para nortear a construção do jogo educacional.

Portanto todos os gêneros de jogos podem também ser jogos educacionais e levar consigo todas as características do gênero específico. Daí temos o trabalho de Rapeepisarn et al. (Rapeepisarn et al., 2008), que fez uma correlação entre características do conteúdo a ser ensinado e os gêneros de jogos ideais aplicáveis a cada conteúdo como pode ser visto na

Tabela 2.1.

2.3 Trabalhos Relacionados

Nesta seção serão apresentados trabalhos relacionados ao desenvolvimento de jogos educacionais, especificamente, propondo um arquitetura de software e ou um framework, assuntos que estão mais diretamente relacionados com este trabalho.

Muitas pesquisas têm sido conduzidas com o objetivo de reduzir as dificuldades e o retrabalho de quem se propõe a desenvolver jogos educacionais. Um dos trabalhos é apresentado por Hu (2010) que propôs uma arquitetura para jogos educacionais e em seguida estendeu essa arquitetura para jogos no gênero Aventura. Há também os autores Moreno-Ger et al. (2008), que propõem um design de jogos educacionais para educação online. Outro trabalho é o de Peixoto et al. (2012) que desenvolveram um framework para o desenvolvimento de Jogos de Simulação para o ensino de Engenharia de Software.

Hu (2010) apresenta uma revisão bibliográfica e os principais requisitos encontrados referentes a jogos educacionais, apresentados parcialmente na Tabela 3.1. . No trabalho de Hu (2010) os requisitos foram classificados em Funcionais e Não Funcionais seguindo o padrão da área de Engenharia de Requisitos, organizados em quatro grupos: Representações do assunto educacional, Atividades educacionais, Tarefas e Feedback.

Após feito o levantamento e análise de requisitos temos a fase de Projeto na qual Wen-feng Hu (Hu, 2010), separou a arquitetura em quatro subsistemas, que são: Subsistema do conteúdo, Subsistema de atividades, Subsistema de tarefas, Subsistema de feedback. No Subsistema do conteúdo os requisitos educacionais, segundo o autor, devem ser decompostos em elementos do jogo, como NPCs (*non-person character*), animais, edifícios, adereços, interfaces gráficas, Animações, Diálogos e assim por diante. Portanto, o Subsistema do conteúdo é um grupo de metáforas para atender aos requisitos educacionais. O Subsistema de atividades contém ações do usuário para se comunicar com o jogo e executar ações no jogo. O Subsistema de tarefas trata das ações que o usuário deve executar durante o jogo de forma a cumprir objetivos intermediários como etapas para chegar ao objetivo final. O Subsistema de feedbacks é responsável por manter o jogador interessado no jogo, recompensando-o por ações corretas e penalizando-o por ações incorretas. A representação gráfica em UML dos subsistemas pode ser visto na Figura 3.1.

No trabalho de Moreno-Ger et al. (Moreno-Ger et al., 2008) o autor enumera jogos educacionais de sucesso e jogos comerciais com potencial educacional que foram bem sucedidos como SimCity e a série Civilization. E em seguida levanta o que seriam os requisitos pedagógicos para um jogo educacional, apresentados na Tabela 2.2. O autor também propõe

em seu trabalho um roteiro para o desenvolvimento de um jogo educacional de sucesso que pode ser visto na Figura 2.1. Os autores sugerem que primeiro seja definido o gênero do jogo e em seguida se faça o diagrama de estados do jogo a partir de seus requisitos iniciais e então adicione-se ao diagrama de estados os requisitos de Adaptação, Integração e Avaliação.

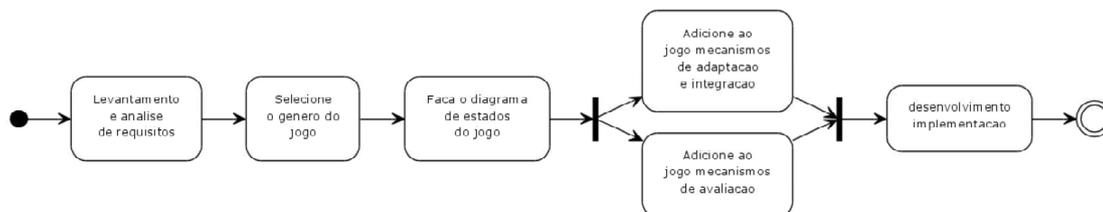


Figura 2.1. Roteiro para o desenvolvimento de um jogo educacional Fonte: Levantado a partir do trabalho de Moreno-Ger et al. (Moreno-Ger et al., 2008)

Já o trabalho de Peixoto et al.(Peixoto et al., 2012) apresenta os requisitos para desenvolvimento de um jogo de simulação para o ensino de Engenharia de Software que são extremamente úteis para o presente trabalho por se tratar também de um jogo de simulação. Os requisitos levantados por Peixoto et al.(Peixoto et al., 2012) podem ser encontrados na Tabela 2.3. O framework proposto no trabalho de Peixoto et al.(Peixoto et al., 2012) não acrescenta para solução do atual trabalho por se tratar de framework a nível de implementação impedindo alterações conceituais necessárias na adaptação para o problema aqui proposto.

Tabela 2.1. A relação entre técnicas de aprendizagem, atividades de aprendizagem e possível gênero de jogo Extraídos de Rapeepisarn et al. (Rapeepisarn et al., 2008)

Conteúdo de aprendizagem.	Atividades de aprendizagem.	Estilos de jogo possível.
Fatos: leis, políticas, produtos.	Perguntas, memorização, associação.	Game Show, Flashcard, Mnemônicos.
Habilidades: entrevista, ensino, administração.	Imitação, feedback, coaching, prática contínua.	Jogos de estado persistente, RPG, jogos de detetive.
Julgamento: gestão, decisões, cronograma, ética.	Revido casos, fazer perguntas, feedback, coaching.	Jogos de RPG, interação multiplayer, jogo de aventura, jogo de estratégia, jogo de detetive.
Comportamentos: supervisão, auto-controle, exemplo.	Imitação, feedback, coaching, prática.	Jogo de RPG.
Teorias: lógicas de marketing, como as pessoas aprendem.	Lógica, experimentação, questionamento.	Jogos de simulação, jogos de construção.
Raciocínio: o pensamento estratégico e tático, análise de qualidade.	Problemas, exemplos	Puzzles
Processo: Auditoria, criação de estratégia.	Sistema de análise e desconstrução, prática.	Jogos de estratégia, jogos de aventura.
Procedimento: montagem, caixa de banco, leis.	Imitação, prática, jogos.	Jogos Cronometrado, jogos de reflexos.
Criatividade: invenção, design de produto.	Jogar.	Puzzles, jogos de invenção
Idioma: siglas, língua estrangeira.	Imitação, a prática contínua, de imersão.	Jogos de RPG, jogos de reflexo, jogos de flashcard.
Sistemas: cuidados de saúde, mercados, refinarias.	Princípios de compreensão, tarefas graduadas.	Jogos de simulação
Observação: o humor, moral, problemas de ineficiências.	Observando, feedback.	Jogos de concentração, jogos de aventura.
Comunicação: linguagem apropriada, envolvimento.	Imitação, prática	Jogos de RPG, jogos de reflexos.

Tabela 2.2. Principais requisitos educacionais para jogos (Moreno-Ger et al., 2008)

Requisitos	Descrição
Integrado com o ensino on-line	Deve ser independente de acompanhamento (tutor ou monitor). Deve levar em conta diferenças culturais como violência, religião e questões sensíveis
Adaptação	O jogo deve ser adaptável para diversas etapas do aprendizado. Adaptar o comportamento de um jogo de aventura pode consistir em adição / remoção de quebra-cabeças, mudando os objetos que estão em cada quarto, adição / remoção de caracteres, pulando partes do jogo, etc
Avaliação	Um jogo educacional deve permitir a avaliação do aprendizado. Em particular, a avaliação da atividade do estudante dentro do jogo pode ser realizada como uma análise dos estados pelos quais o jogo passou durante a sessão de jogo. O motor de jogo deve acompanhar as transições, registrar eventos relevantes e gerar relatórios descrevendo-os.

Tabela 2.3. Principais requisitos para Jogos de Simulação (Peixoto et al., 2012)

Requisitos - Descrição
REQ1 - Um jogo de simulação deve dar suporte à transferência de aprendizagem.
REQ2 - Um jogo de simulação deve ser interativo.
REQ3 - Um jogo de simulação deve refletir a complexidade dos problemas encontrados no mundo real.
REQ4 - Um jogo de simulação deve facilitar a reflexão sobre a matéria da aprendida.
REQ5 - Um jogo de simulação devem apoiar trabalho em equipe desenvolvimento de competências.
REQ6 - Um jogo de simulação deve suportar diferentes ambientes de aprendizagem.

Capítulo 3

Arquitetura para jogos educacionais de simulação e estratégia

Diante do contexto apresentado na introdução, foi desenvolvida a arquitetura para jogos educacionais. Tendo como referência os trabalhos desenvolvidos por Wenfeng Hu (Hu, 2010), Peixoto et al. (Peixoto et al., 2012) e Moreno-Ger et al. (Moreno-Ger et al., 2008) dos quais a maior contribuição vem de Wenfeng Hu (Hu, 2010), que propõe uma arquitetura extensível para se adequar a problemas mais específicos. Assim evita-se que novos jogos educacionais precisem continuamente redescobrir e reinventar a estrutura comum dos jogos educacionais, aumentando, assim, a produtividade no seu desenvolvimento e sua efetividade educacional.

Para se obter uma arquitetura de software deve-se ter em mente o processo de desenvolvimento de software que é constituído de fases, comuns a projetos de Engenharia de Software: Levantamento e Análise de requisitos, Projeto, Implementação, Teste e Manutenção (Pressman, 2006). Porém a arquitetura aqui apresentada é uma abstração de alto nível e não descreve detalhes de implementação (por exemplo, algoritmos e estruturas de dados), o que significa que a arquitetura é independente de plataforma de desenvolvimento de jogos (game engine). Além disso, as funcionalidades dos jogos educacionais podem ser suportadas por quaisquer motores de jogos modernos. Portanto, neste trabalho as fases não contemplam Implementação, Teste e Manutenção.

3.1 Levantamento e Análise de Requisitos

Como apresentado na introdução ao se desenvolver um jogo deve-se levar em conta os requisitos necessários para desenvolvê-lo: enquanto software, enquanto jogo e enquanto ferramenta educacional. Para encontrar os requisitos do jogo enquanto software e jogo educacional temos o trabalho de Wenfeng Hu (Hu, 2010) onde encontramos, segundo o autor, os

principais requisitos não funcionais de jogos educacionais que são apresentados apresentados na Tabela 3.1.

Tabela 3.1. Principais requisitos para jogos educacionais (Hu, 2010)

Requisitos	Descrição
Fascinação	Gerar entusiasmo no jogador
Imersão	Envolver o jogador na trama do jogo.
Motivação	Querer jogar independente de ser um jogo educacional.
Desafio progressivo.	Não se tornar desmotivador por ser fácil demais e nem por ser difícil demais.
Controle	Possuir controles claros e objetivo de forma que o jogador não se torne um espectador da trama.
Competição	Competição com outros jogadores. Por exemplo, existência de um ranking.
Metáfora	Metáforas ou correlações com objetos do mundo real.
Feedback	O Feedback do jogo em forma de recompensas ou punições.

No trabalho de Hu (2010), os requisitos foram classificados em Funcionais e Não Funcionais seguindo o padrão da área de Engenharia de Requisitos, organizados em quatro grupos: Representações do assunto educacional, Atividades educacionais, Tarefas e Feedback.

Como apresentado na introdução, o atual trabalho usou como referência uma disciplina de economia rural e daí são extraídos os requisitos educacionais. Microempresas rurais estão inseridas em um ambiente com muitas variáveis dinâmicas, a maioria delas relacionadas com risco e sofrem impacto de adoção de novas tecnologias, intervenções políticas e as mudanças climáticas em sistemas agrícolas (Schreinemachers & Berger, 2011). No trabalho de Rapeepisarn et al. (Rapeepisarn et al., 2008), que fez uma correlação entre características do conteúdo a ser ensinado e os gêneros de jogos ideais aplicáveis a cada conteúdo, temos que os jogos de estratégia e simulação são ideais para ensinar planejamento, análise e relacionamento entre variáveis. Segundo Schreinemachers (Schreinemachers & Berger, 2011), planejamento, análise e relacionamento entre variáveis são a problemática da administração rural. Portanto, o ideal é usar jogos de estratégia e simulação para se ensinar conceitos de disciplinas nesta área de conhecimento.

O jogo de simulação é uma abordagem na qual o jogador assume um papel mais ativo, contribuindo de forma decisiva para o desenrolar da história por trás do jogo (POSSA, 2011). Jogos de simulação são criados com o objetivo de colocar o jogador no controle de uma determinada atividade durante a tentativa de torná-lo o mais realista possível. Dentre os

jogos de simulações temos as simulações gerenciais que colocam o jogador na posição de um gerente ou de um papel semelhante (MobyGames, 2012).

O jogo de estratégia enfatiza a resolução de problemas. Pensamento e planejamento são componentes necessários de estratégia, pois eles podem ser usados para fins tão diversos como a preparação e posicionamento de tropas para ganhar vantagem em uma guerra, ou descobrir o princípio de um quebra-cabeças. Assim, conclui-se neste trabalho que para desenvolver jogos educacionais para aplicação na administração rural, ele deve ser um jogo de simulação e estratégia.

Daí temos o trabalho de Peixoto et al.(Peixoto et al., 2012), como citado anteriormente, que apresenta os requisitos para jogos de simulação para o ensino Engenharia de Software na Tabelas 2.2. Por se tratar também de uma disciplina de planejamento, análise e relacionamento entre variáveis são extremamente úteis para o presente trabalho.

3.2 Projeto da Arquitetura de Software

Como apresentado no capítulo 2 Wenfeng Hu (Hu, 2010) propôs, em seu trabalho, uma arquitetura para jogos educacionais apresentada na Figura 3.1 e em seguida estendeu a arquitetura para jogos no gênero Aventura. Nessa seção pretende-se fazer o mesmo para os gêneros Simulação e Estratégia usando como base o trabalho de Wenfeng Hu (Hu, 2010), Peixoto et al.(Peixoto et al., 2012), Moreno-Ger et al. (Moreno-Ger et al., 2008) e GARRIS et al.(Garris et al., 2002).

A Figura 3.1 mostra a uma visão geral da arquitetura proposta por Wenfeng Hu (Hu, 2010). No canto superior esquerdo temos o Subsistema de Conteúdo, no canto superior direito temos o Subsistema de Atividades, no centro à esquerda temos o Subsistema de Tarefas, no centro à direita temos o Subsistema de Feedbacks, no canto inferior esquerdo temos a representação da colaboração entre os subsistemas e no canto inferior direito temos o Fluxo do Jogo.

Os itens levantados anteriormente servem como base para a arquitetura aqui proposta e são comuns a todos os jogos educacionais. Wenfeng Hu (Hu, 2010) ainda disse que seu trabalho deve ser adaptado para diferentes propósitos e que novos pesquisadores na área de jogos educacionais não devem mais ficar redescobrimo e reinventando essas estruturas. Partiu-se, neste trabalho, para a adaptação da arquitetura até aqui levantada, com o objetivo de atender ao propósito específico de jogos educacionais para o ensino da administração rural. De acordo com Wenfeng Hu Hu (2010), a arquitetura apresentada acima pode ser reutilizada para fazer a arquitetura de jogos de simulação e estratégia, mas primeiramente deve ser adaptada ou estendida. As regras para a adaptação da arquitetura são:

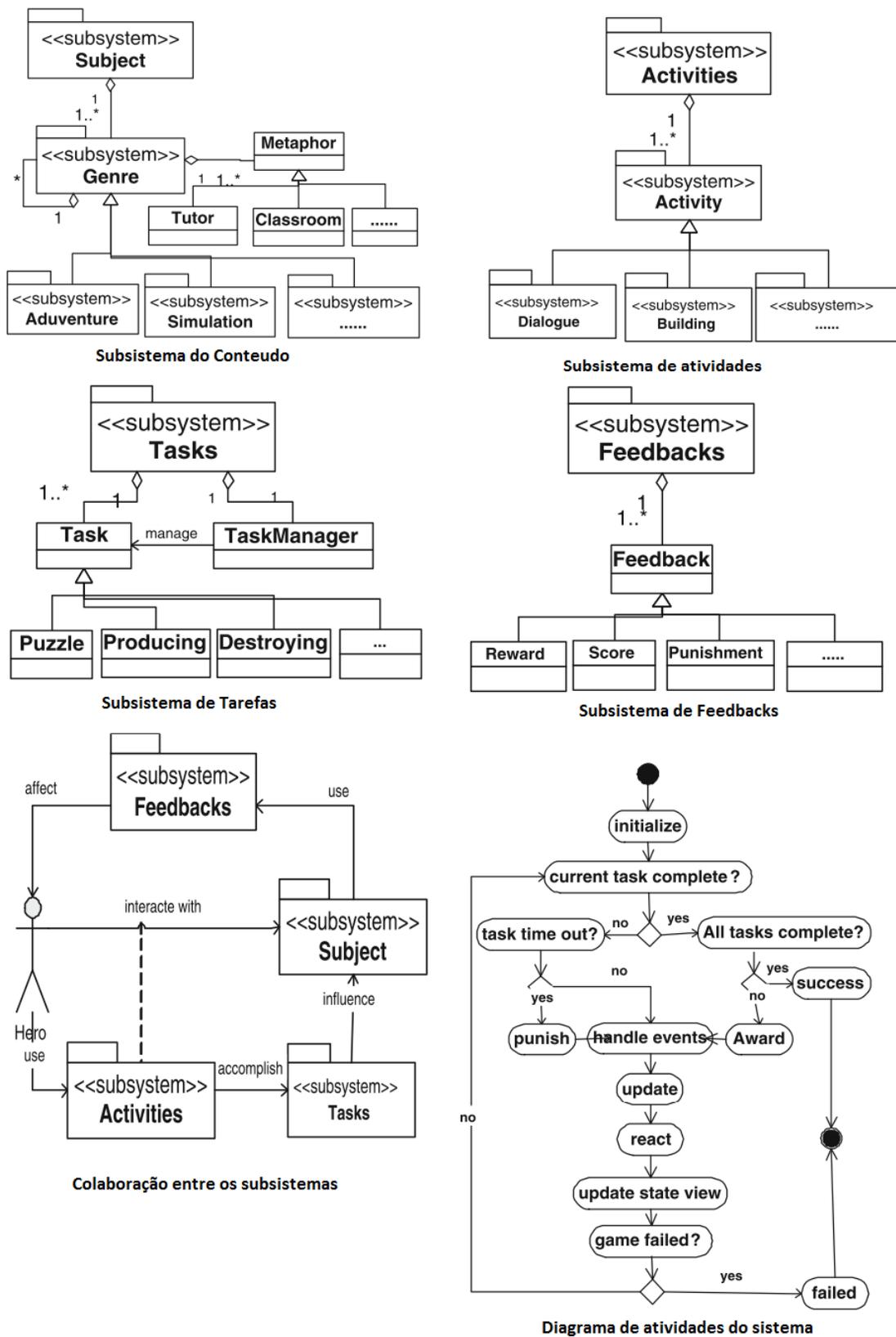


Figura 3.1. Arquitetura para jogos educacionais. Fonte: Wenfeng Hu (Hu, 2010)

- Não toque nas classes abstratas para manter toda a estrutura inalterada;
- Adicionar novas classes específicas para satisfazer novas exigências.

De posse desses subsistemas e sabendo que os jogos ideais para esses problemas são os jogos de simulação e estratégia, temos a partir dos trabalhos de Moreno-Ger et al. (Moreno-Ger et al., 2008) e Garris et al. (Garris et al., 2002), cinco aspectos dos jogos de simulação: Objetivo e regras claras, Feedback, Mundo virtual, Adaptação e Jogabilidade.

Como levantado na seção anterior é necessária a adaptação da arquitetura até aqui levantada, com o objetivo de atender ao propósito específico de jogos educacionais para o ensino da administração rural. Para isso partiu-se da arquitetura proposta por Wenfeng Hu (Hu, 2010), ilustrada na Figura 3.1.

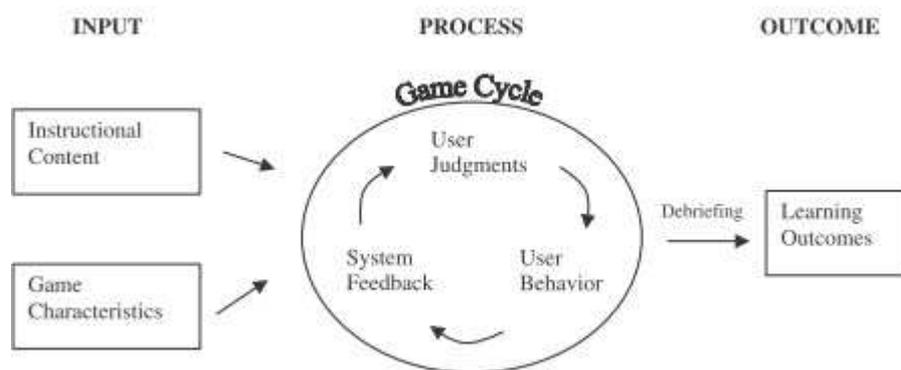


Figura 3.2. Input-Process-Outcome Game Model Fonte:GARRIS Garris et al. (2002)

Segundo GARRIS et al. Garris et al. (2002), jogar jogos educacionais de simulação deve seguir o fluxo apresentado na Figura 3.2, onde temos que o ciclo principal do jogo tem três estágios: Julgamento/Análise do usuário (*User Judgments*), Atitude do usuário (*User Behavior*) e Reação do Sistema (*System Feedback*). Isso pode ser exemplificado da seguinte forma: o usuário analisa a situação, executa as ações dentro da estratégia escolhida e por fim obtém o novo estado do sistema onde se inicia um novo ciclo de jogo. Em outros estilos de jogos o jogador tem várias tarefas a serem cumpridas em sequência, que podem ou não ser dependentes umas das outras. O jogo vai liberando essas tarefas seguindo uma árvore de dependência. Porém, como mostrado acima, em jogos de simulação e estratégia como o proposto, não há tarefas a serem cumpridas durante o jogo, nesse tipo de jogo o próprio jogador constrói sua estratégia tendo o jogador apenas uma tarefa que é o próprio objetivo do jogo: tomar a melhor decisão possível, utilizando as opções do jogo. Assim, para suportar jogos de simulação e estratégia, os subsistemas da arquitetura proposta por Wenfeng Hu (2010) devem ser modificados.

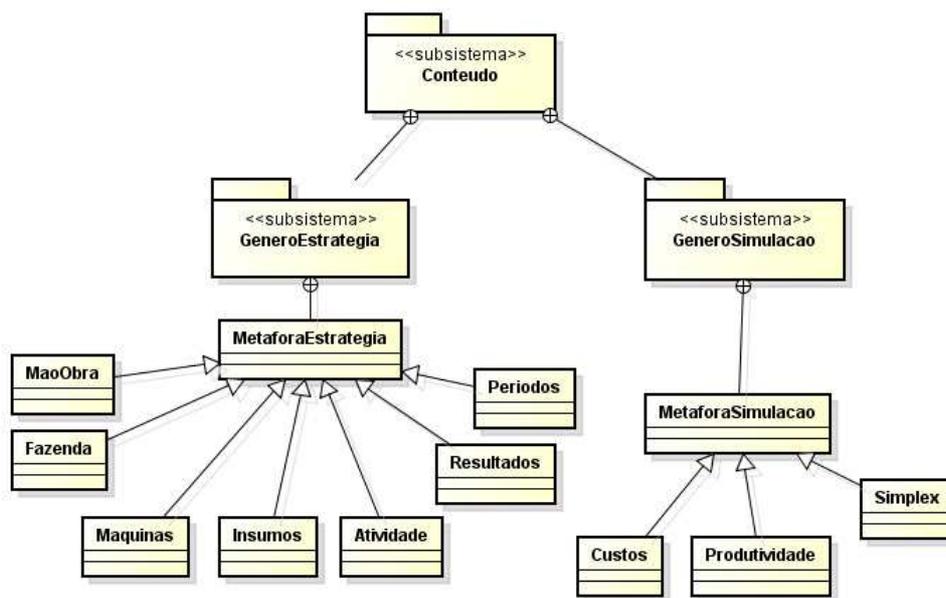


Figura 3.3. Subsistema de conteúdo. Fonte: Desenvolvido pelo autor.

A Figura 3.3 descreve a estrutura para um subsistema de conteúdo para jogos de simulação e estratégia que compreendem dois gêneros: simulação e estratégia. O subsistema de conteúdo trata das metáforas necessárias para se abordar o tema que se deseja ensinar. Como pretendemos ensinar administração rural, aqui estão representadas as metáforas necessárias:

- Metáforas da estratégia
 - Fazendas
 - Mão de obras
 - Máquinas
 - Insumos
 - Atividades
 - Resultados
 - Períodos
- Metáforas da simulação
 - Custos
 - Produtividade
 - Simplex

O subsistema de atividades para jogos de simulação e estratégia deve fornecer os meios adequados para que os jogadores interajam com o que eles estão aprendendo, nesse subsistema devem estar localizadas todas as ações de interação do jogador com o jogo como pode ser visto na Figura 3.4.

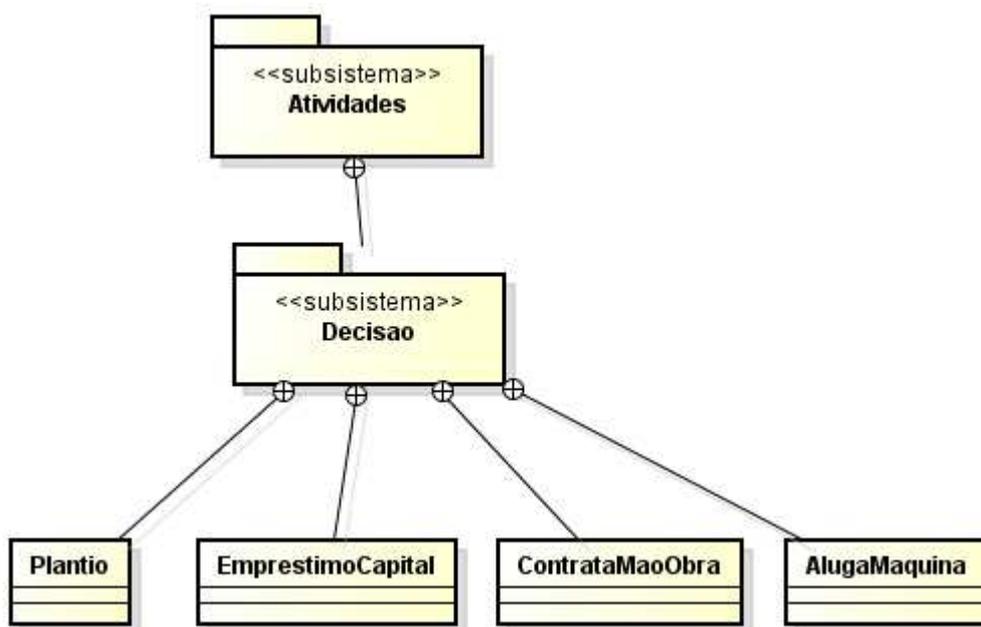


Figura 3.4. Subsistema de atividades. Fonte: Desenvolvido pelo autor.

Para o subsistema de tarefas para jogos de simulação e estratégia, como mencionado anteriormente ao analisar a Figura 3.2 proposta por GARRIS et al. Garris et al. (2002), a única tarefa deve ser tomar a melhor decisão possível a cada rodada. Nos outros estilos de jogos há muitas tarefas a serem cumpridas e muitas vezes há dependências entre elas, havendo a necessidade de um subsistema de tarefas. Porém, não havendo tais tarefas a serem cumpridas em jogos de estratégia e simulação não há necessidade de um subsistema de tarefas. Portanto a arquitetura apresentada por Hu (Hu, 2010) aqui não é aplicável.

O subsistema de feedback para jogos de simulação e estratégia fica como mostrado na Figura 3.5, tendo como feedback a alteração do capital e a alteração no valor de mercado da propriedade rural administrada. Sendo estes valores norteadores das decisões do usuário a cada rodada.

Foi necessário também alterar o fluxo do jogo proposto por Wenfeng Hu (Hu, 2010) para que atendesse ao fluxo proposto por GARRIS et al. (Garris et al., 2002), como mostrado na Figura 3.6.

Na Tabela 3.1 é apresentado uma sugestão passo-a-passo de como usar a arquitetura proposta.

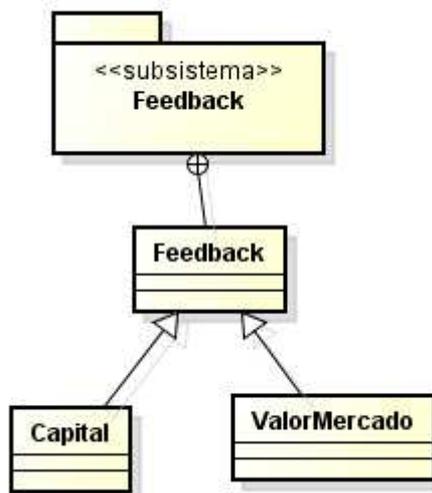


Figura 3.5. Subsistema de feedback. Fonte: Desenvolvido pelo autor.

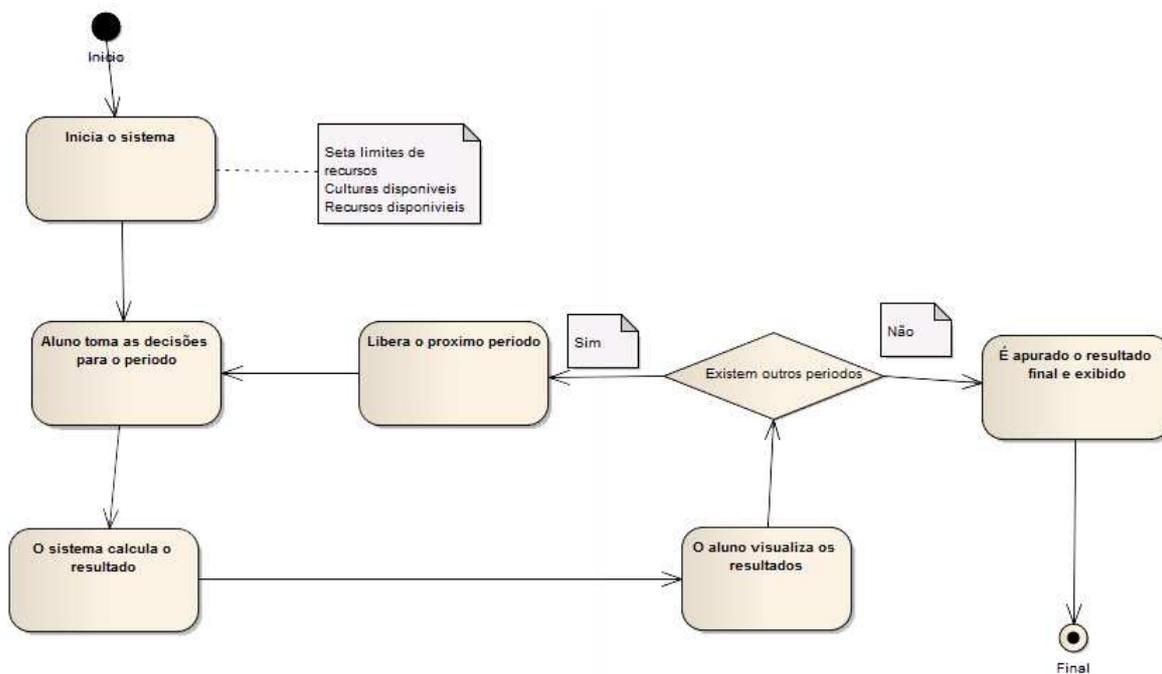


Figura 3.6. Fluxo do Jogo.

Tabela 3.2. Roteiro de uso da arquitetura

Passo	Descrição
Passo 1:	Levante todos os requisitos necessários para o jogo desejado.
Passo 2:	Busque na Tabela 2.1 qual é o melhor estilo de jogo para o que se deseja ensinar e adicione as características desse jogo aos requisitos.
Passo 3:	Enumere todas as metáforas necessárias para o que se deseja ensinar.
Passo 4:	Enumere todas as atividades necessárias para o que o jogo simule o que se deseja ensinar.
Passo 5:	Elabore formas para que o jogador tenha o feedback adequado de suas ações.
Passo 6:	Faça o diagrama de estados do jogo.
Passo 7:	A partir da figura Figura 3.6 monte o diagrama de fluxo do jogo.
Passo 8:	Verifique se atendeu aos requisitos básicos levantados nas Tabelas (Tabela 2.2, Tabela 2.3, Tabela 3.1). Caso não tenha atendido refaça os passos 3 a 8.
Passo 9:	Verifique se os subsistemas de conteúdo, atividade e feedback estão bem atendidos, caso contrário volte ao passo 3.
Passo 10:	Integre os subsistemas ao diagrama de estados.

Capítulo 4

Detalhes de Implementação

Esta seção apresenta um jogo de Simulação e Estratégia, mostrando como foi desenvolvido com base na arquitetura de Jogos de Simulação e Estratégia. Esse trabalho foi publicado no Games Innovation Conference (IGIC 2013) da IEEE International realizado em Vancouver, Canadá nos dias 23 a 25 de setembro de 2013 (Amaral et al., 2013).

A disciplina de administração rural oferecida aos alunos de graduação utiliza uma planilha conforme mostrado na Figura 4.1. A referida planilha é usada em sala de aula para simulação do ambiente de tomada de decisões.

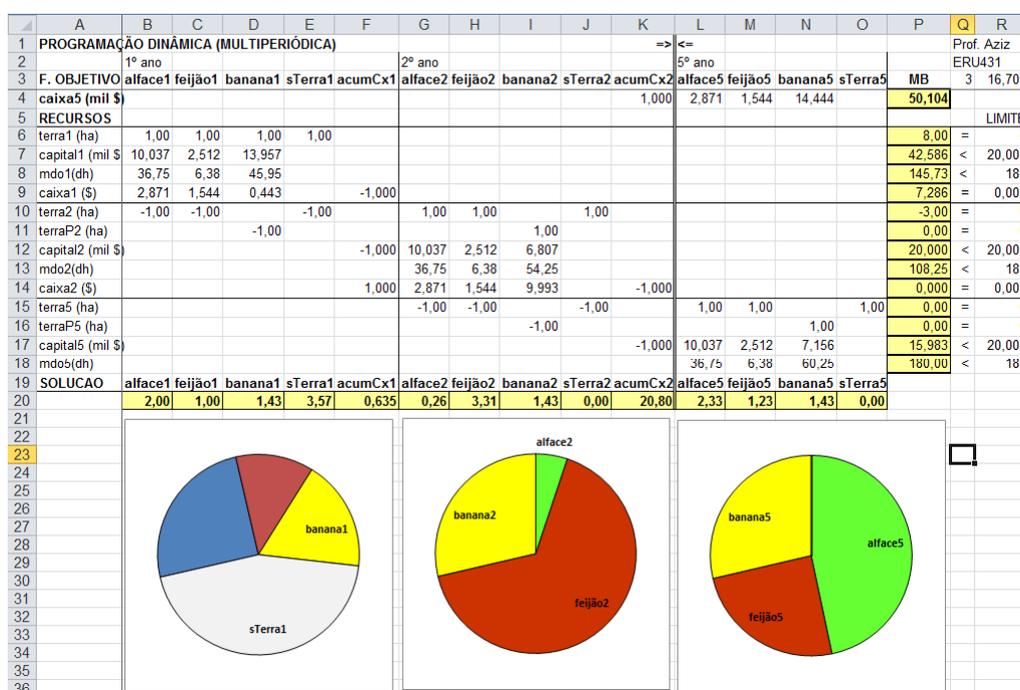


Figura 4.1. PROGRAMAÇÃO DINÂMICA (MULTIPERIÓDICA)

A planilha presente na Figura 4.1 oferece ao aluno a simulação de tomada de decisão acerca do que plantar. Como se pode observar ela oferece um período de até cinco anos, podendo este ser alterado pelos usuários. Para a tomada de decisão o usuário utilizará a linha 20 para preenchimento da quantidade de cada cultura a ser plantada em cada ano. Sendo que no exemplo dado na figura as células de B20 a F20 referem-se ao 1º ano de simulação; as células de G20 a K20 referem-se ao 2º ano de simulação; e as células L20 a O20 ao quinto ano de simulação. Para cada cultura a planilha oferece os coeficientes técnicos necessários para o plantio de uma unidade de área da cultura. Pode-se observar que no 1º ano os coeficientes para Alface, Feijão e Banana encontram-se nas células de B6 a D9, sendo que a linha 6 refere-se aos requisitos de terra, a linha 7 aos requisitos de capital, a linha 8 aos requisitos de mão de obra e a linha 9 aos requisitos de caixa. Sendo assim para cada hectare de terra a ser plantado deve-se levar em consideração os recursos consumidos. Para tal, encontra-se na coluna R o limite dos recursos disponíveis para a tomada de decisão. A partir desta planilha projetou-se o jogo de simulação e estratégia descrito neste capítulo.

O jogo é projetado para propagar o conhecimento de Administração Rural: o jogador deverá decidir quais atividades agrícolas implantar e quanto de área deve ser disponibilizada para cada atividade escolhida. Cada atividade possui características diferentes com relação à lucratividade e tempo de retorno e demandas de recursos, como benfeitorias, máquinas, mão de obra, capital e insumos para atender a cada unidade de área que o jogador decidir implantar. O jogador deverá desenvolver as atividades de forma a acumular o máximo de capital no final de todas as rodadas.

As seguintes três subseções mostram como os três subsistemas da arquitetura são derivados e implementados a partir da arquitetura. Já o subsistema de tarefas como mencionado anteriormente a única tarefa deve ser tomar a melhor decisão possível a cada rodada.

O Subsistema do Conteúdo

No mundo do jogo, uma fazenda com sua área disponível para plantio é apresentada para o jogador e um menu com as atividades disponíveis para serem implementadas. Na parte superior da tela são exibidos os recursos disponíveis para serem usados, de forma que o jogador se sinta como um verdadeiro fazendeiro diante de seus recursos e de suas possibilidades, tendo que resolver o que plantar, quando e quanto, como pode ser visto na Figura 4.2. Assim temos todas as metáforas necessárias para abordar o conteúdo pedagógico do jogo.

O Subsistema de Atividades

No Subsistema de atividade temos quatro atividades: Plantio, Contratação de Mão de Obra, Empréstimo de Capital e Aluguel de Máquinas, sendo a Contratação de Mão de Obra, Empréstimo de Capital e Aluguel de Máquinas selecionados no menu Ações, enquanto o plantio é selecionado no menu Atividades apresentados na Figura 4.2.



Figura 4.2. Tela do jogo de Administração Rural. Fonte: Desenvolvido pelo autor.

O Subsistema de Feedbacks

O principal retorno que o jogador tem é o capital obtido no final do ciclo de cada cultura plantada, sendo que estes ciclos podem durar uma ou mais rodadas do jogo. Como o jogador tem compromissos financeiros em todas as rodadas, é necessário equilibrar o plantio entre culturas com ciclos de poucas rodadas e culturas com ciclos de muitas rodadas. E esse valor ganho em cada ciclo de cada cultura pode ser usado em replantio, na compra de máquinas e contratação de mão de obra, aumentando, assim, os recursos da propriedade. O feedback ruim é quando o capital da propriedade acaba e ainda não houve tempo de as culturas plantadas darem retorno financeiro, o que obriga o fazendeiro a pegar empréstimo de capital ou até mesmo abandonar o jogo. A implementação também preocupou-se em atender a cinco aspectos principais dos jogos de simulação levantados na literatura (Moreno-Ger et al., 2008) e (Rapeepisarn et al., 2008) e (Garris et al., 2002)

Um exemplo de jogada

O jogo se inicia como mostra a Figura 4.2, indicando no canto superior esquerdo "Período Atual 1" o que significa primeiro ano de simulação, cada período equivale a um ano e as decisões são tomadas ano a ano. No canto superior direito temos os recursos disponíveis para as tomadas de decisão: 18dh (dezoito dias homens) de mão de obra, 50 hm (cinquenta horas máquina) disponíveis para uso de máquinas e R\$ 20.000,00 (vinte mil reais) disponíveis para o custeio do plantio e 25 ha (vinte e cinco hectares). Para se tomar as decisões para este período é preciso selecionar a atividade no menu de atividades no lado esquerdo da Figura 7 e digitar o valor que se deseja plantar dessa atividade e clicar em Registrar. Porém, antes de tomar essa decisão deve-se levar em conta os coeficientes técnicos de cada cultura, ou seja, o que cada cultura gasta de recursos para cada hectare plantado. Os valores usados pelo jogo nesse exemplo estão na Tabela 4.1, extraídos da literatura técnica da área.

Tabela 4.1. Recursos utilizados e receita produzida por cada cultura no jogo. Extraídos de Resende et. al. (REZENDE et al., 2005) e Embrapa (Embrapa, 2007)

Atividade	Custo/há(R\$)	Mão de Obra/há(dh)	Maquinas/há(hm)	Receita líquida / há(R\$)
Milho	700,00	1	6	2700,00
Feijão	2000,00	24	25	3000,00
Alface	4000,00	600	100	12000,00

Como podemos ver na Tabela 4.1 tendo apenas 50hm (cinquenta horas homem) disponíveis seria impossível plantar alface nessa primeira rodada, pois seriam necessários 100 hm (cem horas homem) para apenas um hectare de alface. Então uma decisão possível para essa rodada seria plantar dois hectares de feijão consumindo assim 48 hm e não restando recurso para plantar mais nenhuma cultura. Como o jogador terá dinheiro sobrando após essa decisão, ele poderá nesta rodada ainda contratar horas de máquinas extra, selecionado no menu "Ações" no lado direito da tela. E assim poder plantar mais alguns hectares de milho, feijão ou alface. Como dito anteriormente, no jogo é possível contratar mais mão de obra, adquirir empréstimo de capital e alugar máquinas. Tudo isso no menu Ações, basta selecionar a ação, digitar o valor desejado e clicar em registrar, o custo de cada ação dessas está na Tabela 4.2.

Tabela 4.2. Custo de cada ação. Extraídos do Portal do Governo do Estado de São Paulo. (do estado de São Paulo, 2012) e Banco do Brasil (do Brasil, 2012)

Ações	Custo
Contratação de Mão de Obra	R\$ 50,00 dh
Empréstimo de Capital	5,5% a.a
Aluguel de Máquinas	R\$ 100,00 hm

Olhando a Tabela 2 podemos ver que mesmo podendo contratar mais recursos com apenas R\$20.000 (vinte mil reais) sendo que, após ter plantado dois hectares de feijão nos sobra R\$ 16.000,00 (dezesseis mil reais) disponíveis nessa primeira rodada, seria impossível plantar alface, uma vez que, para contratarmos seiscentos dias homens e cem horas máquinas seriam necessários R\$ 40.000 (quarenta mil reais), mais que o dobro do dinheiro disponível. Então, tomando uma decisão mais conservadora contratamos apenas mais 10hm (dez horas maquina) para plantarmos mais dois hectares de milho. E sobrar algum dinheiro para o próximo período além do que será acrescentado pelos resultados das culturas atuais. E ao final dessa jogada a tela do jogo estará como mostra a Figura 4.3.

Ao finalizar esse período o jogo soma os resultados dos plantios atuais no capital disponível, caso uma atividade plantada demore mais do que um período para dar resultados,

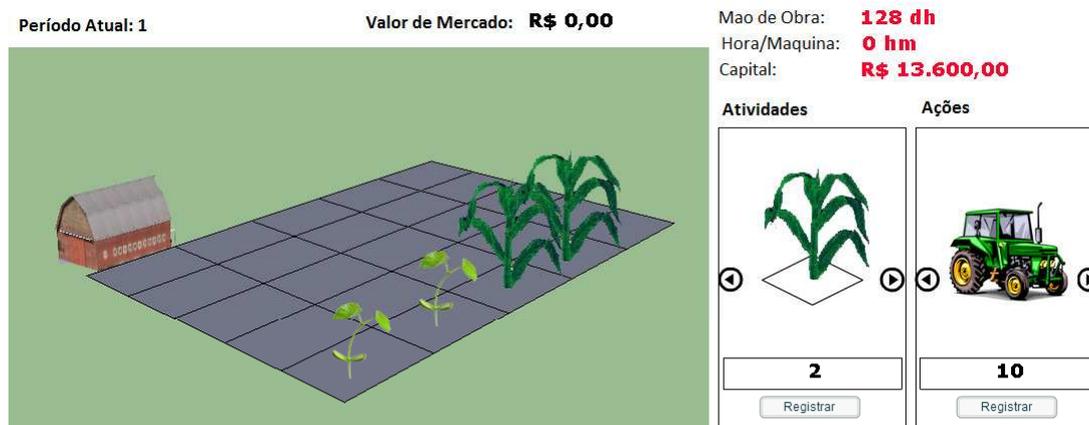


Figura 4.3. Tela do Jogo ao final da primeira rodada. Fonte: Desenvolvido pelo autor.

como por exemplo eucalipto ou frutas de pomar, somente ao final de todos os períodos necessários é que o valor é creditado no capital disponível. As decisões tomadas acima são de caráter apenas ilustrativo, uma vez que para tomar decisões ótimas, dado o número de variáveis envolvidas, seria necessário um modelo de otimização mais complexo para guiar o andamento do jogo. O campo valor de mercado na parte superior da tela só será preenchido ao final de todas as rodadas e mostrará o valor referente a soma de todos os recursos disponíveis na fazenda juntamente com as áreas plantadas, o jogador que tiver a fazenda com maior valor de mercado é o vencedor.

Capítulo 5

Adaptabilidade

O atual capítulo descreve o trabalho publicado na XVIII Conferência Internacional sobre Informática na Educação (TISE) realizado em 8 de Dezembro de 2013 (Amaral et al., 2013)¹. Tal trabalho descreve uma melhoria feita no jogo de estratégia descrito no Capítulo 4 demonstrando assim a adaptabilidade da arquitetura. Para melhorar o aprendizado dos jogadores foi implantado um sistema de ajuda de forma a auxiliar o jogador a toma a melhor decisão utilizando-se de algoritmos de inteligência artificial. Portanto, o objetivo desse artigo é propor um algoritmo de inteligência artificial para ajudar os jogadores em seu processo de aprendizado.

5.1 Introdução

Jogos educacionais têm o potencial de fornecer experiência motivadora de aprendizagem, além de ajudarem a dar sentido à experiência e organizar o conhecimento, despertando habilidades para resolução de problemas e aumentando a motivação. Permitem ainda que ocorra um processo de aprendizagem não intencional, por meio de uma experiência envolvente e atraente, em que quem aprende é visto como participante ativo na construção de seu próprio conhecimento (Hodhod et al., 2011).

Diante das características dos jogos educacionais e dos problemas encontrados nos métodos tradicionais de ensino, as instituições de ensino e também as organizações em todos os seus níveis, poderão tirar proveito das potencialidades dos jogos educacionais. Um tipo de empresa em que os jogos educacionais são úteis são as microempresas rurais, uma vez que elas estão inseridas em um ambiente com muitas variáveis (Schreinemachers & Berger, 2011) dinâmicas, a maioria delas relacionadas com risco. Outro motivo que vale ressaltar é

¹Foi sugerido a alteração do nome dessa publicação uma vez que nela não é implementado inteiramente o algoritmo Minimax

a percepção de que no meio rural são infinitas as formas de gestão por diversas razões, como por exemplo, visão estreita do negócio, aplicação de forma de gestão de grande para pequena empresa, entre outros. Tais diferenças ocasionam resultados muito diferentes se compararmos propriedades de mesmo porte e com recursos similares (Kaw, 1981). Os jogos educacionais podem auxiliar na simulação de um ambiente de uma microempresa e serem usados no treinamento de futuros administradores de fazendas, para que os riscos de decisões sejam minimizados, melhorando os resultados econômicos das empresas. As simulações tornaram-se também ferramentas importantes para a administração da empresa rural, podendo auxiliar na avaliação do impacto de adoção de novas tecnologias, intervenções políticas e as mudanças climáticas em sistemas agrícolas (Schreinemachers & Berger, 2011).

A partir do que foi citado acima se desenvolveu um jogo de estratégia para auxiliar no aprendizado de administração rural. Nesse jogo o jogador tem que administrar um limitado grupo de recursos tentando atingir uma maior lucratividade em sua fazenda. E para melhorar o aprendizado dos jogadores foi implantado um sistema de ajuda de forma a auxiliar o jogador a toma a melhor decisão utilizando-se de algoritmos de inteligência artificial.

Portanto, o objetivo desse artigo é propor o uso de um algoritmo de inteligência artificial para ajudar os jogadores em seu processo de aprendizado.

Este trabalho foi dividido em seis seções. Na presente seção é feita a introdução e é levantada a problemática do trabalho aqui apresentado. Na seção 2 é apresentado trabalhos correlatos. Na seção 3 é apresentada a arquitetura do jogo educacional utilizado. Na seção 4 é proposto a arquitetura do sistema de ajuda. Na seção 5 é apresentada aplicação do sistema proposto na seção 4. Por fim, na seção 6 são apresentadas as conclusões.

5.2 Trabalhos Correlatos

Cunha e Chaimowicz em *An Artificial Intelligence system to help the player of Real-Time Strategy games* (de Freitas Cunha & Chaimowicz, 2010) elaboraram um sistema de ajuda ao jogador usando inteligência artificial. O trabalho de Cunha e Chaimowicz difere do atual trabalho por terem trabalhado em um jogo de estratégia em tempo real não educativo. Outro trabalho correlato o de Mishra e Ram em *Case-Based Planning and Execution for Real-Time Strategy Games* (Ontañón et al., 2007) onde os autores desenvolveram um sistema IA de planejamento de jogada para um jogo de estratégia em tempo real. A diferença aqui é mais uma vez que se trata um jogo de estratégia em tempo real e não educativo. O atual trabalho trata de um jogo de estratégia por turnos e as dicas possuem um fundo educativo.

5.3 Jogos Educacionais

O jogo é projetado para propagar o conhecimento de Administração Rural: o jogador deverá decidir quais atividades agrícolas implantar e quanto de área deve ser disponibilizada para cada atividade escolhida. Cada atividade possui características diferentes com relação à lucratividade e tempo de retorno e demandas de recursos, como benfeitorias, máquinas, mão de obra, capital e insumos para atender a cada unidade de área que o jogador decidir implantar. O jogador deverá desenvolver as atividades de forma a acumular o máximo de capital.

No mundo do jogo, uma fazenda com sua área disponível para plantio é apresentada para o jogador e um menu com as atividades disponíveis para serem implementadas. Na parte superior da tela são exibidos os recursos disponíveis para serem usados, de forma que o jogador se sinta como um verdadeiro fazendeiro diante de seus recursos e de suas possibilidades, tendo que resolver o que plantar, quando e quanto, como pode ser visto na Figura 5.1. No canto superior direito encontramos o botão Dica que serve para auxiliar o jogador em suas decisões.

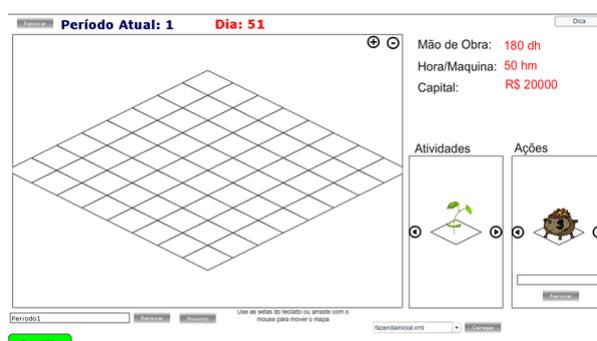


Figura 5.1. Tela do jogo de Administração Rural. Fonte: Desenvolvido pelo autor.

O jogo se inicia como mostra a Figura 5.1, indicando no canto superior esquerdo "Período Atual 1"o que significa primeiro ano de simulação, cada período equivale a um ano e as decisões são tomadas ano a ano. No canto superior direito temos os recursos disponíveis para as tomadas de decisão: 180dh (cento e oitenta dias homens) de mão de obra, 50 hm (cinquenta horas máquina) disponíveis para uso de máquinas e R\$ 20.000,00 (vinte mil reais) disponíveis para o custeio do plantio e 25 ha (vinte e cinco hectares).

Para se tomar as decisões para este período é preciso selecionar a atividade no menu de atividades no lado esquerdo da Figura 5.1 e digitar o valor que se deseja plantar dessa atividade e clicar em Registrar. Porém, antes de tomar essa decisão deve-se levar em conta os coeficientes técnicos de cada cultura, ou seja, o que cada cultura gasta de recursos para cada

hectare plantado. Os valores usados pelo jogo nesse exemplo são os mesmos da Tabela 4.1, extraídos da literatura técnica da área. Porém, nessa versão do jogo os valores mostrados na Tabela 4.1 é apenas uma estimativa, pois todos estes coeficientes podem variar de acordo com o modelo de clima programado no jogo como mostra a Tabela 5.1. O jogo simula o clima para cada dia. Sendo que cada atividade possui modelos que usam as variáveis de clima para calcular os coeficientes técnicos em tempo real. Dessa forma os coeficientes podem sofrer variações dependendo do clima.

Para simplificar a simulação o modelo de produtividade baseado no clima tem apenas duas variáveis aleatórias que são a temperatura e a quantidade de chuvas. Cada dia o programa simula uma quantidade de chuva e temperatura. Para que tudo ocorra bem é necessário que esses valores estejam de acordo com a Tabela 5.1.

Tabela 5.1. Coeficientes climáticos das atividades. Extraídos de Resende et. al. Embrapa (Embrapa, 2007)

Atividade	Temperatura	Pluviometria
Milho	10°C a 30°C	5 a 10 mm/dia
Feijão	15 a 21°	5 a 10 mm/dia
Alface	15 a 21°	indiferente

No jogo também é possível contratar mais mão de obra, adquirir empréstimo de capital e alugar maquinas. Tudo isso no menu Ações, basta selecionar a ação, digitar o valor desejado e clicar em registrar, o custo de cada ação dessas está na Tabela 4.2.

Ao finalizar cada período o jogo soma os resultados dos plantios atuais no capital disponível, caso uma atividade plantada demore mais do que um período para dar resultados, como por exemplo, eucalipto ou frutas de pomar, somente ao final de todo os períodos necessários é que o valor é creditado no capital disponível.

5.4 Modelagem do Sistema de Ajuda

Como pôde ser visto na seção anterior é muito difícil prever o que pode acontecer no jogo, devido à incerteza do resultado, pois o clima e ocorrência de pragas são aleatórios. Mas se encaramos as ocorrências de pragas e o clima como um adversário em um jogo de perde-ganha. Sendo que se a fazenda do jogador falir o sistema vence o jogo, porém, se o jogador jogar todas as rodadas e acumular dinheiro o jogador vence.

Sendo assim, segundo Russel (Russell et al., 1995) para os casos de jogos perde-ganha (soma-zero) existem as árvores de decisão Minimax que podem ser pensadas como a maximização do ganho mínimo. Ou seja encontra a melhor solução se o pior caso acontecer.

Poderemos aplicar nesse caso uma árvore de decisão Minimax para auxiliar as decisões do jogador dando a melhor solução caso haja ocorrência de praga e um clima ruim.

De posse dessas informações partiu-se para implementação da árvore de decisão Minimax. Que, nesse caso, se aplica especialmente na busca em árvores de jogo para determinar qual a melhor jogada. O algoritmo se baseia no princípio de que em cada jogada, o jogador irá escolher o melhor movimento possível. A árvore de decisão é composta de todas as jogadas possíveis para o jogador atual como nós filhos da raiz, e todas as jogadas disponíveis para o próximo jogador como filhas destes nós e assim por diante, até o nível que se desejar. Cada ramificação da árvore representa um movimento que o jogador pode fazer em tal momento do jogo. Uma busca mais profunda na árvore fornece mais informações sobre as possíveis vantagens ou armadilhas e portanto resulta em uma jogada melhor.

O Minimax faz uma busca que determina todas as possíveis continuações do jogo até o nível desejado, avaliando e atribuindo um valor a cada movimento possível. A busca então retorna na árvore de jogo alternando entre escolher o valor mais alto e o valor mais baixo entre os valores da jogadas em um nível. Na Figura 5.2 é temos o algoritmo desenvolvido.

```

1 BestDecision (GameBoard game, Ingerger endturn, best_decision) {
2   if (endturn<=0) {
3     return game.capital;
4   }
5   else {
6     best_capital <- 0;
7     decisions <- GenerateDecisions(game);
8     ForEach decisions {
9       GameBoard newgame = ApplyDecision(game, decision)
10      capital <- WorseWeather (newgame, endturn-1, best_decision);
11      capital <- capital * (1+ decision.numativ*0.05);
12      if (capital > best_capital) {
13        best_capital <- capital;
14        best_decision <- decision;
15      }
16    }
17  }
18 }
19 WorseWeather (GameBoard game, Ingerger endturn, best_decision) {
20   worse_capital <- 0;
21   weathers <- GenerateWeather();
22   ForEach weathers {
23     GameBoard newgame = ApplyWeather(game, weather)
24     capital <- BestDecision(newgame, endturn, best_decision);
25     if (capital < worse_capital) {
26       worse_capital <- capital;
27     }
28   }
29   return worse_capital;
30 }

```

Figura 5.2. Algoritmo Desenvolvido.

No caso atual consideraram-se apenas três tipos de clima, mostrados na Tabela 5.2, de

forma a simplificar a geração da árvore e facilitar a análise dos resultados para o presente trabalho.

O algoritmo presente na Figura 5.2 inicia chamando BestDecision que gera todas as jogadas possíveis e para cada jogada gerada chama WorseWeather que gera todos os climas possíveis e para cada clima gerado chama BestDecision e assim sucessivamente até chegar no último turno do jogo. No último turno BestDecision retorna para a instância de WorseWeather que o chamou o capital acumulado durante as jogadas. WorseWeather por sua vez retorna menor capital recebido de todas as chamadas que fez de BestDecision como forma de minimização da jogada do oponente para a instância de BestDecision que o chamou. BestDecision retorna o maior capital recebido de todas as instâncias WorseWeather que chamou e salva a decisão que gerou esse capital na variável best_decision. E assim sucessivamente até desempilhar todas as chamadas recursivas. Ficando no final com a decisão que gerou a maior quantidade de capital.

Tabela 5.2. Climas usados na simulação

Nome	Clima
Clima 0	Temperatura média 30°C Precipitação média diária 0 mm/dia
Clima 1	Temperatura média 20°C Precipitação média diária 3 mm/dia
Clima 2	Temperatura média 25°C Precipitação média diária 5 mm/dia

Na Figura 5.3 podemos ver uma árvore gerada pelo algoritmo e na Tabela 5.2 temos os climas usados na simulação.

A Figura 5.3 para o atual trabalho foi simplificada, pois o algoritmo minimax gera mais de 10 folhas por nível da árvore minimax.

5.5 Resultados e Conclusões

Ao clicar no botão Dica no canto superior da tela como mostra a Figura 5.4 podemos ver o sistema descrito na seção anterior funcionando. Onde a tela Dica mostra a solução encontrada pelo algoritmo minimax para aquela jogada.

Com tudo podemos dizer que um algoritmo baseado na árvore Minimax pode ser usado para gerar dicas em jogos de estratégia educacionais como facilitador do aprendizado aumentado assim o potencial que os Jogos educativos têm de fornecer experiência motivadora de aprendizagem para o aluno, além de ajudar a dar sentido à experiência e organizar o conhecimento, despertando habilidades para resolução de problemas.

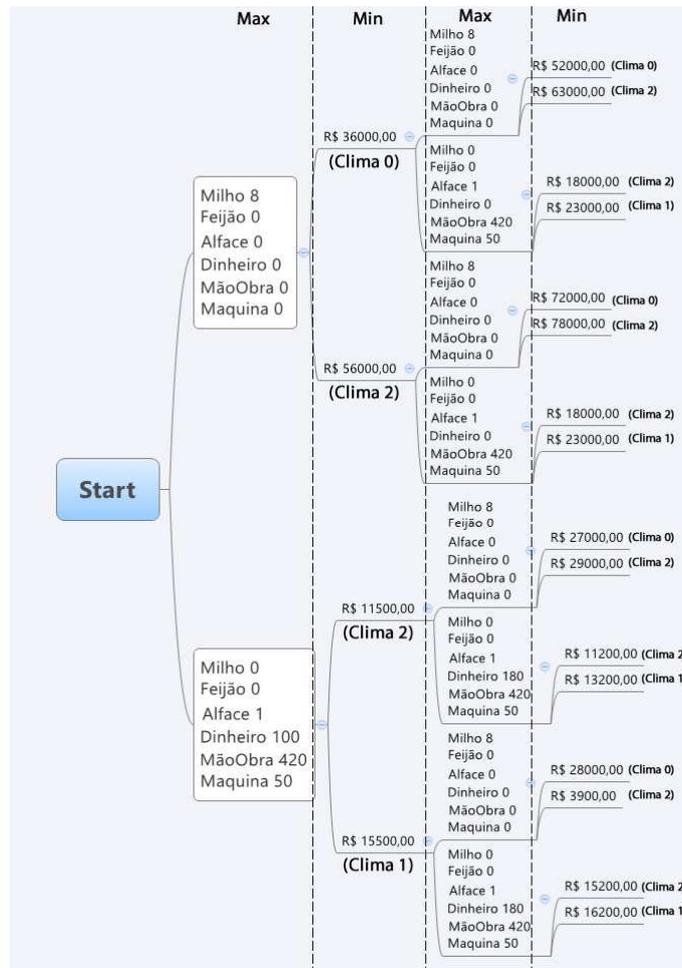


Figura 5.3. Árvore minimax gerada. Fonte: Desenvolvido pelo autor.

Como o algoritmo minimax gera mais de 10 folhas por nível da árvore, consumindo muito tempo de processamento, pretende-se em trabalhos futuros implementar alterações no algoritmo que permita não gerar folhas não promissoras na árvore de decisão. Ainda pretende-se usar o jogo com alunos de graduação para levantar dados que avaliem a eficácia do jogo e da função de ajuda discutida no presente trabalho.

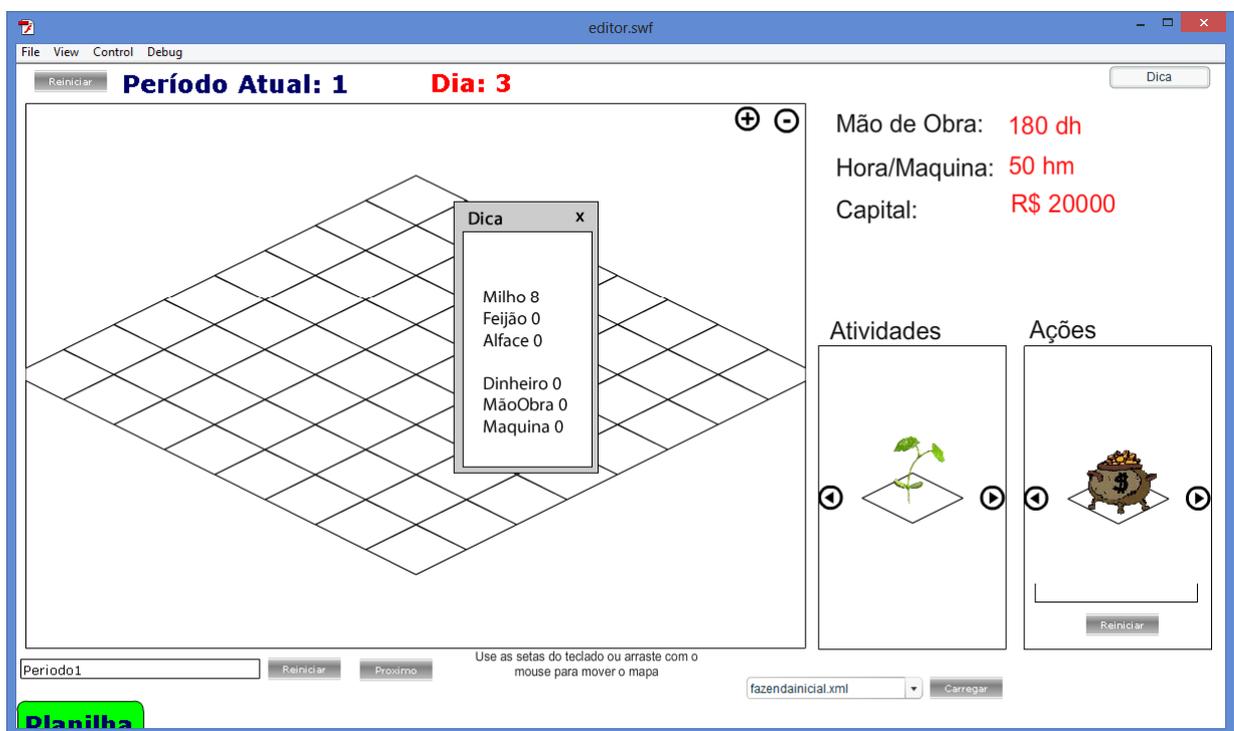


Figura 5.4. Exemplo da função dica. Fonte: Desenvolvido pelo autor.

Capítulo 6

Conclusões e Trabalhos Futuros

Jogos educativos têm o potencial de fornecer experiência motivadora de aprendizagem para o aluno, além de ajudar a dar sentido à experiência e organizar o conhecimento, despertando habilidades para resolução de problemas e aumentando a motivação. O jogo foi apresentado a alunos de Economia Rural da UFV e foi muito bem recebido, gerando de imediato interesse e motivação. Vale ressaltar, que pretende-se usar o jogo de forma sistemática com alunos de graduação para levantar dados que avaliem a eficácia do jogo e da arquitetura.

Pôde-se observar que a arquitetura facilita o desenvolvimento do jogo, principalmente no âmbito conceitual uma vez que manteve o foco do desenvolvimento nas funcionalidades que realmente importavam para tornar o jogo efetivo.

Como pôde ser observado no capítulo 3 pode-se inferir que o objetivo principal deste trabalho foi alcançado. Obteve-se uma especificação de uma arquitetura para jogos de simulação baseados em sistemas multiagentes, para ser usado como ferramenta potencializadora no processo de ensino-aprendizagem.

O estabelecimento de requisitos para uma arquitetura de jogos de simulação usados no processo de ensino e aprendizagem, foi um passo necessário para o levantamento da arquitetura apresentada neste trabalho. Tal levantamento encontra-se presente também no capítulo 3

No que se refere a análise de tecnologias disponíveis para implementação de jogos e simulações aplicadas ao ensino e elaboração de um estudo de caso específico para uso em disciplina de administração rural, estas foram concluídas a partir dos artigos apresentados nos capítulos 4 e 5.

Como trabalho futuro, no âmbito da arquitetura pretende-se detalhar mais a arquitetura a um nível mais próximo do desenvolvimento para evitar ainda mais o retrabalho e no âmbito do jogo pretende-se estender a simulação da fazenda para simular melhor a ocorrências de sinistros, estratégia de vendas e suporte ao uso de AHP (Alphonse, 1997) no processo de

tomada de decisão.

Referências Bibliográficas

- ACM (2012). Software engineering. Acessado: 24 Outubro de 2012.
- Akilli, G. K. (2011). Games and simulations: A new approach in education. *Gaming and Simulations: Concepts, Methodologies, Tools and Applications*, ed. Information Resources Management Association, USA, pp. 150--167.
- Allen, R. J. (1997). A formal approach to software architecture. Relatório técnico, DTIC Document.
- Alphonse, C. B. (1997). Application of the analytic hierarchy process in agriculture in developing countries. *Agricultural systems*, 53(1):97--112.
- Amaral, H.; Braga, J. & Galvao, A. (2013). Game architecture for teaching-learning process: An application on an undergraduate course. Em *Games Innovation Conference (IGIC), 2013 IEEE International*, pp. 1--6. ISSN 2166-6741.
- Amaral, Heber Fernandes ; Machado, Alex FV ; Braga, José Luis (2013). Um sistema de Apoio a Decisão Baseado em MiniMax para um Jogo de Estratégia em Administração Rural. Em In: *Proceedings TISE-2013 - XVIII Congreso Internacional de Informática Educativa.*, pp.
- Arsenault, D. (2009). Video game genre, evolution and innovation. *Eludamos. Journal for Computer Game Culture*, 3(2):149--176.
- Bass, L.; Clements, P. & Kazman, R. (2003). *Software architecture in practice*. Addison-Wesley.
- de Freitas Cunha, R. L. & Chaimowicz, L. (2010). An artificial intelligence system to help the player of real-time strategy games. Em *Games and Digital Entertainment (SBGAMES), 2010 Brazilian Symposium on*, pp. 71--81. IEEE.
- do Brasil, B. (2012). Credito rural. Acessado: 24 Outubro de 2012.
- Embrapa (2007). Embrapa Hortaliças Sistemas de Produção, 2 ISSN 1678-880x Versão Eletrônica <<http://sistemasdeproducao.cnptia.embrapa.br>> Acessado: 24 Outubro de 2012.
- do estado de São Paulo, G. (2012). Portal do governo do estado de São paulo. Acessado: 24 Outubro de 2012.

- Garlan, D. & Shaw, M. (1993). An introduction to software architecture. *Advances in software engineering and knowledge engineering*, 1:1--40.
- Garris, R.; Ahlers, R. & Driskell, J. E. (2002). Games, motivation, and learning: A research and practice model. *Simulation & gaming*, 33(4):441--467.
- Hodhod, R.; Cairns, P. & Kudenko, D. (2011). Innovative integrated architecture for educational games: challenges and merits. Em *Transactions on edutainment v*, pp. 1--34. Springer.
- Hu, W. (2010). A common software architecture for educational games. Em *Entertainment for Education. Digital Techniques and Systems*, pp. 405--416. Springer.
- Kaw, R. (1981). Farm management/planning, control and implementation.
- Krafzig, D.; Banke, K. & Slama, D. (2005). *Enterprise SOA: service-oriented architecture best practices*. Prentice Hall Professional.
- Lindley, C. A. (2004). Narrative, game play, and alternative time structures for virtual environments. Em *Technologies for Interactive Digital Storytelling and Entertainment*, pp. 183--194. Springer.
- MobyGames (2012). Mobygames. glossary.
- Moreno-Ger, P.; Burgos, D.; Martínez-Ortiz, I.; Sierra, J. L. & Fernández-Manjón, B. (2008). Educational game design for online education. *Computers in Human Behavior*, 24(6):2530--2540.
- Myers, D. et al. (1990). Computer games genres. *Journal for Computer Game Culture, North America issue 3*, 3(4):286--301.
- Ontañón, S.; Mishra, K.; Sugandh, N. & Ram, A. (2007). Case-based planning and execution for real-time strategy games. Em *Case-Based Reasoning Research and Development*, pp. 164--178. Springer.
- Peixoto, D. C.; Possas, R.; Resende, R. F. & Pádua, C. (2012). Faseng: A framework for development of software engineering simulation games. Em *Proceedings of the 42nd Frontiers in Education Conference*.
- POSSA, R. (2011). Um estudo sobre os requisitos de jogos de simulação usados no ensino de engenharia de software. Dissertação de mestrado, Universidade Federal de Minas Gerais.
- Prensky, M. (2003). Digital game-based learning. *Computers in Entertainment (CIE)*, 1(1):21--21.
- Pressman, R. S. (2006). *Engenharia de software*. Makron Books.
- Rapeepisarn, K.; Wong, K. W.; Fung, C. C. & Khine, M. S. (2008). The relationship between game genres, learning techniques and learning styles in educational computer games. Em *Technologies for E-Learning and Digital Entertainment*, pp. 497--508. Springer.

- REZENDE, B.; CECÍLIO FILHO, A.; MARTINS, M.; COSTA, C. & FELTRIM, A. (2005). Viabilidade econômica das culturas de pimentão, repolho, alface, rabanete e rúcula em cultivo consorciado. *Informações Econômicas*, 35(3).
- Russell, S. J.; Norvig, P.; Canny, J. F.; Malik, J. M. & Edwards, D. D. (1995). *Artificial intelligence: a modern approach*, volume 74. Prentice hall Englewood Cliffs.
- Schreinemachers, P. & Berger, T. (2011). An agent-based simulation model of human-environment interactions in agricultural systems. *Environmental Modelling Software*, 26(7):845 – 859. ISSN 1364-8152.