

# AprendES: um jogo educacional para auxiliar o processo de ensino-aprendizagem da Engenharia de Software

Aldefran C. Feitosa<sup>1</sup>, Glaucia M. M. Campos<sup>2</sup>

Universidade do Estado do Rio Grande do Norte (UERN)  
Av. Airton Senna, 4241 – 59.088-100 – Natal – RN – Brazil

Departamento de Computação – Campus de Natal  
aldefrancarvalho@gmail.com<sup>1</sup>, glauciamelissa@uern.br<sup>2</sup>

***Abstract.** This paper presents a proposal for an educational game, not electronic, based for charts, with the goal of helping the process of teaching and learning of the Engineering Software course, considering its the theoretical and the need for practical experience of its concepts, even through simulation, before the insertion of students in work market's.*

***Resumo.** Este trabalho apresenta a proposta de um jogo educacional, não eletrônico e baseado em cartas, com o objetivo de auxiliar o processo de ensino-aprendizagem da disciplina de Engenharia de Software, considerando o seu caráter teórico e a necessidade da vivência prática dos seus conceitos, mesmo que seja através de simulação, antes da inserção dos estudantes no mercado de trabalho.*

## 1. Introdução

De acordo com Kahal (2007), a utilização de práticas lúdicas contribui para o aprendizado do aluno. Por esta razão, diversos tipos de jogos educativos, independente do nível de ensino, vêm sendo utilizados para aprimorar a comunicação entre alunos e professores (Alves, 2006).

Os jogos educativos propiciam aos professores estimular os alunos quanto à aprendizagem. A sua utilização tem demonstrado isso quando estes entram em contato com o objeto de estudo, facilitando assim o trabalho do professor, pois a dinâmica dos jogos permite ao aluno desenvolver habilidades a partir da compreensão de suas regras, bem como apreender os conceitos do conteúdo alvo do jogo e desenvolver aspectos afetivo-social entre os alunos.

Nos cursos da área de Computação têm-se comprovado que a utilização de jogos auxilia o estudante a absorver melhor os conceitos estudados e a compreender as conseqüências das decisões tomadas, simulando durante o jogo a realidade que enfrentará no dia-a-dia quando estiver efetivamente trabalhando em projetos (Rossiou e Papadakis, 2007). Estudos já observaram que disciplinas como Engenharia de Software, que são estudadas de forma predominantemente teórica, tem os seus resultados, com relação à compreensão dos conceitos e métodos estudados, se mostrado insuficientes para preparar adequadamente o profissional para o mercado de trabalho, se este não tiver tido previamente alguma vivencia prática, por mais simples que seja (Reif e Mitri, 2005).

Considerando este cenário, a proposta deste trabalho consiste em desenvolver um jogo de cartas, não eletrônico, com caráter educativo, que seja aplicado na sala de aula, para auxiliar o processo de ensino-aprendizagem da disciplina de Engenharia de Software, dos cursos da área de Computação. A opção por um jogo não eletrônico vem da necessidade do trabalho em equipe, onde haja um contato real entre as pessoas propício às discussões a respeito do desenvolvimento do projeto. Além disso, um ambiente colaborativo possibilita aos indivíduos trabalharem com a regularidade, o limite, o respeito e a disciplina, por meio de ações necessariamente subordinadas às regras. Todos esses aspectos se fazem importantes para a vida do indivíduo em sociedade. Para complementar, os jogos de cartas podem ser jogados a qualquer hora e lugar, enquanto os

jogos eletrônicos dependem de condições que fogem ao controle dos seus usuários, como estrutura, tempo e lugar.

Este trabalho está organizado em 4 seções, sendo a Introdução a primeira delas. Na seção 2, são apresentados os trabalhos relacionados. A seção 3 descreve o jogo de cartas que está em fase de desenvolvimento, e faz parte de um Trabalho de Conclusão do Curso de Ciência da Computação. Os resultados parciais e as etapas finais do desenvolvimento do trabalho estão descritos na seção 4.

## 2. Trabalhos Relacionados

Alguns jogos foram desenvolvidos com o propósito de auxiliar no processo de ensino-aprendizagem da disciplina de Engenharia de Software, como: SESAM - *Software Engineering Simulation by Animated Models* (Monsalve, 2010); *Planager* (Kieling, 2006); *Scrumming* (Isotton, 2008) e SimuLES (Figueiredo, 2006). No entanto, a maioria são jogos eletrônicos ou não permitem a colaboração entre os jogadores, objetivos principais deste trabalho.

## 3. O jogo AprendES

O AprendES é um jogo de cartas que está sendo desenvolvido como Trabalho de Conclusão do curso de Ciência da Computação e está inserido na categoria dos jogos cooperativos onde todos os jogadores trabalham em equipe para vencer o tabuleiro. O AprendES teve como base o Simules, onde todos os conceitos de Engenharia de Software compreendidos pelo jogo são pré-determinados em cartões de projetos, o que não permite ao aluno vivenciar todas as etapas do processo de montagem do tabuleiro e apreender os fundamentos teóricos da disciplina. O jogo AprendES também permite a colaboração entre os membros da equipe, que são na verdade Engenheiros de Software, com o objetivo de finalizar o projeto com sucesso. Por este motivo, novas regras foram definidas para este jogo, assim como a introdução e/ou mudanças nos componentes do jogo Simules.

### 3.1 Componentes do Jogo

O jogo é composto por um tabuleiro, onde são organizados os componentes de acordo com as regras estabelecidas. A Tabela 1 apresenta estes componentes, assim como descreve as funções dos mesmos.

**Tabela 1. Componentes do jogo AprendES e suas funções**

Componentes	Funções
Cartas problemas	Apresentam os problemas a serem resolvidos e as penalidades associadas a estes problemas. Exemplo: a carta doença pode penalizar o engenheiro em questão que perde o direito a realizar qualquer tarefa nas rodadas seguintes.
Cartas conceitos	Apresentam soluções para as cartas problemas ou ajudam a resolver os desafios do jogo. Exemplo: carta conceito que anula qualquer carta problema.
Botões	Representam os artefatos que devem ser construídos. Os botões podem ser verdes, que correspondem aos artefatos de alta qualidade, definidos na proporção de 4 sem defeito, para um com defeito. Também podem ser rosas, que correspondem aos artefatos de baixa qualidade, definidos na proporção de 1 sem defeito para 1 com defeito.
Cartas qualidade	Cartas com valores que variam de 1 a 4 para determinar a qualidade do sistema, sendo este número a quantidade de módulos com defeito que pode ter o sistema ao final do seu desenvolvimento.
Cartas tamanho	Cartas com valores que variam de 3 a 4 e que correspondem ao tamanho de módulos a ser completados durante a partida.
Cartas complexidade	Cartas com valores que variam de 1 a 2 e contribuem para definir o orçamento do projeto. Definem o número de ações dos jogadores por rodada (1 tem direito a duas ações e 2 a três ações).
Moedas	Com valores que variam de 5 a 1000 unidades monetárias, são utilizadas para pagar o orçamento do projeto, que é calculado utilizando o número de artefatos necessários para o projeto multiplicando por um fator pré-estabelecido de 1,5, se a complexidade do sistema for 1, ou por 2,0 se a complexidade do sistema for 2.
Módulos	Contêm os tipos de artefatos que compõem o sistema: requisitos, desenho, código, rastro e ajuda.
Cartas status	Cartas que definem os níveis de produtividade dos engenheiros, de acordo com o sucesso na construção dos artefatos necessários em cada módulo, baseado em sua especialidade.
Cartas nível de experiência	Representam o somatório das habilidades e maturidade de um engenheiro.
Módulo do jogador	Corresponde à mesa de trabalho do engenheiro.

## 3.2 Regras do Jogo

Inicialmente, organize-se o tabuleiro como na Figura 1, distribuindo todos os componentes descritos na seção 3.1. Participam do jogo quatro jogadores, que são os engenheiros de software. Cada engenheiro assume uma especialidade, de acordo com discussões iniciais. Há também no jogo a figura do engenheiro geral, que assume o papel do gerente de projeto, cujas ações são determinadas pelos jogadores.

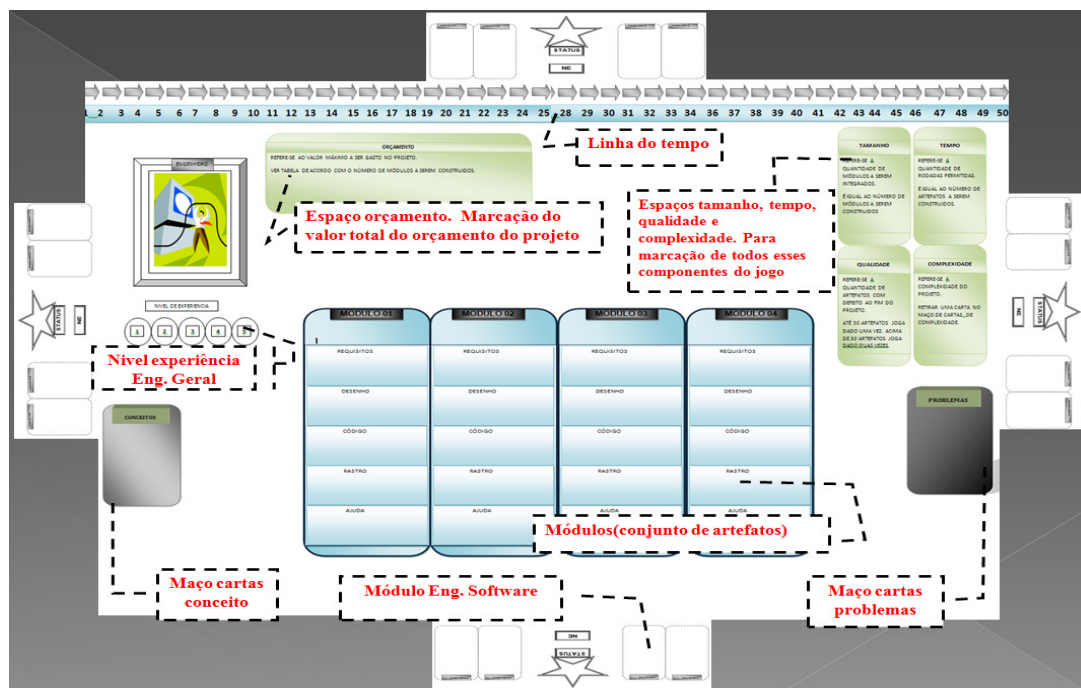


Figura 1. Tabuleiro do Jogo AprendES

Seguem os passos necessários para o jogo:

- Inicialmente, são definidas as especialidades dos engenheiros de software;
- Cada engenheiro define o seu nível de experiência utilizando as cartas 'nível de experiência'; o nível de experiência do gerente de projeto é definido como sendo maior (ou igual) do que os níveis dos quatro jogadores;
- Posteriormente, define-se o número de módulos do projeto utilizando as cartas 'tamanho';
- Define-se o número de artefatos a serem construídos em cada módulo, para cada uma das especialidades (requisitos, desenho, código, rastro e ajuda);
- O tempo é definido de acordo com a quantidade total dos artefatos de todos os módulos e especialidades. Este tempo determina a quantidade de rodadas permitidas por partida no jogo.
- Define-se a complexidade do projeto utilizando as cartas 'complexidade' e a qualidade do mesmo através do lançamento de dados;
- O orçamento é definido utilizando o valor que será pago a cada engenheiro de software, mais o valor a ser pago ao engenheiro geral, multiplicado pela quantidade de artefatos a ser construída na partida, sendo o resultado desta multiplicação multiplicado pelo fator de complexidade.

Depois de executados os passos acima descritos, têm-se início as rodadas do jogo onde os engenheiros de software podem executar as seguintes tarefas:

- Retirar uma carta ‘problema’ do conjunto destas cartas e dependendo do grau de dificuldade, executa a mesma ou solicita uma reunião no fim da rodada ao engenheiro geral, ficando impedido de realizar quaisquer ações nesta rodada;
- Retirar uma carta ‘conceito’ do conjunto destas cartas e executá-la imediatamente ou não. No entanto, um jogador não pode acumular mais do que duas de dessas cartas;
- Executar a quantidade de ações permitida, que podem ser construir, inspecionar ou corrigir artefatos.

Os engenheiros perdem o jogo pelos seguintes motivos:

- As unidades de tempo, que correspondem ao número de rodadas, foram finalizadas e o projeto não foi concluído;
- O orçamento foi extrapolado;
- Ao final do jogo, o número de módulos com defeito é maior do que a qualidade exigida para o projeto.

Considera-se ganho o jogo quando todos os artefatos definidos pelo projeto foram construídos dentro do tempo e orçamento estipulados e a quantidade destes artefatos com defeito é menor ou igual à qualidade exigida.

#### **4. Resultados parciais e projetos de melhoria**

Os resultados deste trabalho ainda são considerados parciais, pois apesar de terem sido definidos os componentes e as regras do jogo, passos fundamentais para a sua execução, ao colocarmos em prática observamos a necessidade de melhorar as cartas ‘problemas’ e ‘conceitos’ com a finalidade de tornar mais dinâmico o jogo. Além das cartas, também há a necessidade de redefinir alguns parâmetros, como o número de artefatos e de módulos. Para atingir este objetivo, diferentes grupos de estudantes da disciplina de Engenharia de Software estão se reunindo para testar o jogo e apresentar propostas de melhoria. Sendo assim, podemos afirmar que o jogo ainda encontra-se em fase de teste e validação.

#### **Referências**

- Alves, V. K. C. A importância dos jogos e brincadeiras no cotidiano escolar. Monografia apresentada na Pós-Graduação “Lato Sensu” – Instituto a Vez do Mestre, Universidade Candido Mendes. Rio de Janeiro, 2006.
- Figueiredo, E. M. L. et al. SimulES: Um Jogo para o Ensino de Engenharia de Software. Monografia apresentada no curso de Ciência da Computação. PUC-Rio, 2006.
- Isotton, E., *Scrumming* – Ferramenta Educacional para Apoio ao Ensino de Práticas de SCRUM. Monografia de Trabalho de Conclusão de Curso de Graduação, Sistemas de Informação, FACIN, PUCRS. Porto Alegre, 2008.
- Kahl, K., Lima, M. E de O., Gomes, I. Alfabetização: Construindo alternativas com jogos pedagógicos. *Extensio: Revista Eletrônica de Extensão*, v. 4, n. 5. Dezembro de 2007.
- Kieling, E., Rosa, R., *Planager* - Um Jogo para Apoio ao Ensino de Conceitos de Gerência de Projetos de Software. Trabalho de Conclusão de Curso de Ciência da Computação, FACIN, PUCRS, Porto Alegre, 2006.
- Monsalve, E. S., *Construindo Um Jogo Educacional com Modelagem Intencional Apoiados em princípios de Transparência*. Tese de Mestrado. Puc-Rio, Rio de Janeiro, 2010.
- Reif, H. L., Mitri, M. How University Professors Teach Project Management for Information Systems. *Communications of the ACM*, Vol. 48, N. 8, Ago/2005.
- Rossiou, E. Papadakis, S. Educational Games in Higher Education: a case study in teaching recursive algorithms. University of Macedonia and The Hellenic Open University, 2007. Disponível em: [http://www.ece.salford.ac.uk/proceedings/papers/17\\_07.pdf](http://www.ece.salford.ac.uk/proceedings/papers/17_07.pdf). Acesso em: junho de 2010.