

Um Jogo para o Ensino de Engenharia de Software Centrado na Perspectiva de Evolução

**Eduardo Figueiredo,
Cidiane Lobato, Klessis
Dias, Julio Leite, Carlos
Lucena**

Carlos Eduardo Paulino Silva

**Reutilização de Software – 02/09/2013
DCC / ICEx / UFMG**

Introdução

- Utilização de jogos para estimular a **curiosidade** e prover **motivação** para o aprendizado [deLaet et al. 2005] [Oh e Hoek 2001] [Virvou et al. 2005]
 - Pouco explorado em Engenharia de Software
- **Evolução de Software** é de vital importância devido ao processo **incremental** de desenvolvimento de **sistemas complexos** composto de múltiplos ciclos de **retro-alimentação** [Lehman, 1996]

Introdução

- Utilização do jogo “Problems and Programmers” (PnP) [Baker et al. 2005] [Navaro et al. 2004]
 - Bem **conhecido** e aplicado com **sucesso** no ensino de ES [Baker et al. 2005]
 - Desenvolvido na Universidade da Califórnia
 - Técnicas de ensino **comuns a jogos educacionais** similares, como por exemplo, **simular** um projeto de software [deLaet et al. 2005]

Introdução

- Utilização do jogo “Problems and Programmers” (PnP) [Baker et al. 2005] [Navaro et al. 2004]
 - Contempla importantes **fases do processo de desenvolvimento de software** (documentação, implementação, inspeção e testes)
 - **Não considera aspectos modernos de software, como desenvolvimento iterativo e evolução**

Introdução

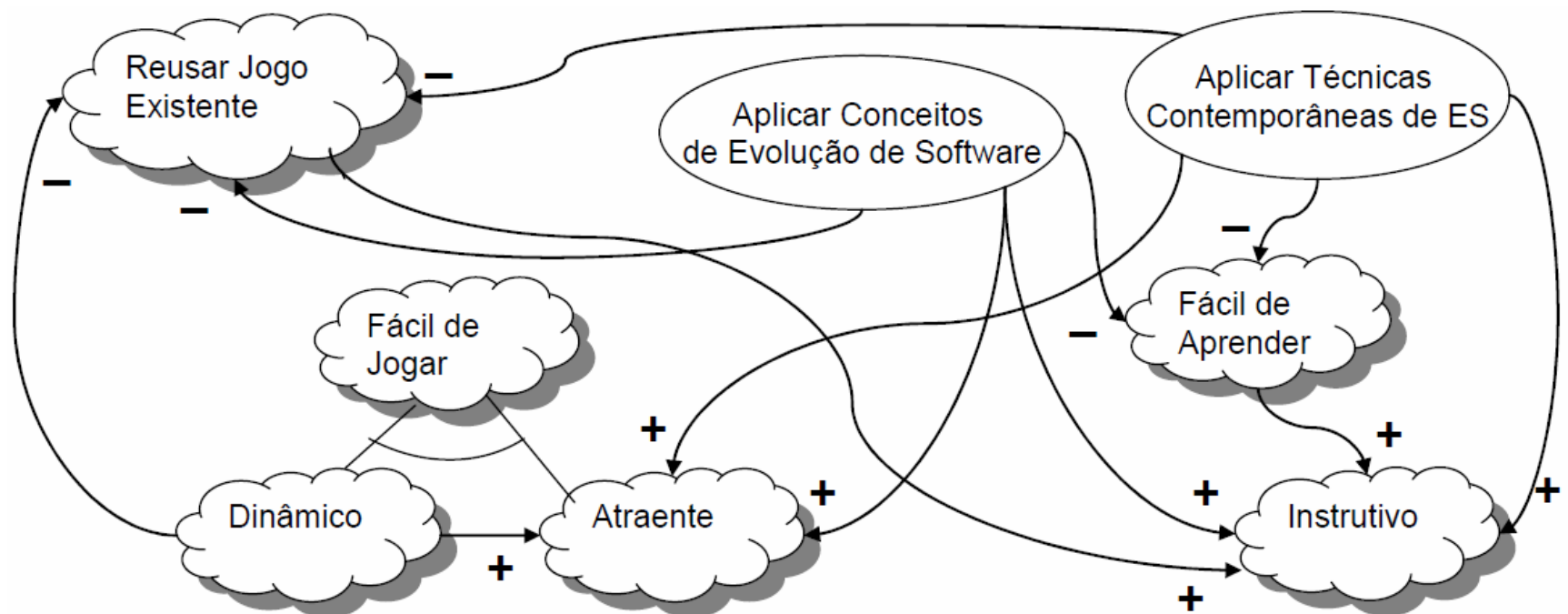
- O artigo apresenta em detalhes uma avaliação do jogo PnP enfatizando os critérios que afetam a **qualidade** do jogo para o ensino de ES com **ênfase em evolução**
 - Outros pontos de avaliação: **jogabilidade** e **dinamismo** do jogo

Introdução

- **SimuleS**: Simulador de Uso da Engenharia de Software
 - Alteração do **formato do jogo, regras e artefatos**
- Análise comparativa com o PnP
 - Utilização de ambos os jogos e observações referentes ao jogo original

Contexto do Estudo

● Principais objetivos do experimento



– Modelagem orientada a metas [Mylopoulos et al. 1999]

Contexto do Estudo

- **Avaliação do Jogo Original (PnP)**
 - Estudo dos **conceitos e regras**
 - Exercício prático de jogar várias vezes simulando **diferentes estratégias** e com **diversas quantidades de jogadores**

Contexto do Estudo

- **Pontos positivos (PnP)**
 - A simulação auxilia o estudante a **compreender** lições de **boas práticas de ES**
 - Ensinar a partir de **decisões** tomadas de forma **equivocada**

Contexto do Estudo

- **Limitações**

- Muito **amarrado** a um único **processo de software** (*Problema A*): Modelo cascata [Royce, 1970]
- Utilização de cartas muito **abstratas** e sem **informações adicionais** (*Problema B*): **Falta de clareza na interpretação** de sua mensagem e **não oferece ou aponta** uma fonte de **informação extra**

Contexto do Estudo

- **Limitações**

- Jogo **não desperta entusiasmo** dos jogadores nas primeiras rodadas (*Problema C*): Não permiti que cartas de problemas sejam jogadas antes dos jogadores possuírem cartas de código. **Jogadas iniciais se limitam a compra de cartas**
- Número de jogadores **não é limitado**, mas a **estrutura** do jogo dificulta que **mais de 4 pessoas participem** (*Problema D*): **Muitas cartas de problemas** são lançadas dificultando que qualquer participante **evolua e ganhe** o jogo

Contexto do Estudo

- **Limitações**

- Ausência de um tabuleiro para **organização da área de projeto** do jogador (*Problema E*): Organização das cartas na mesa, dificultando o **entendimento do jogo** (cartas idênticas)
- Ausência de um **mapeamento explícito** entre os **artefatos** do jogo e os **conceitos** de ES aplicados (*Problema F*): Falta de um **elo** que conecte determinado **conceito** ao conjunto de **artefatos** ou **cenários** do jogo

Solução Proposta: O Jogo SimulES

- Ensino de ES com **foco em evolução**
- Resolver as **limitações** do PnP listados anteriormente
- O **objetivo** é que jogadores (alunos) disputem para terminar um **projeto de software** e o **vencedor** será aquele que primeiro entregar ao cliente um produto com **qualidade adequada**.

Solução Proposta: O Jogo SimulES

- **Recursos do jogo** (totalmente reformulados em relação ao PnP)
 - Cartões de projeto, tabuleiro, cartas e um dado
 - O **dado** e o **tabuleiro** não existem no PnP. Foram incluídos para aumentar o **dinamismo** e **organizar a área do jogo** (*tratando os problemas C e E*)

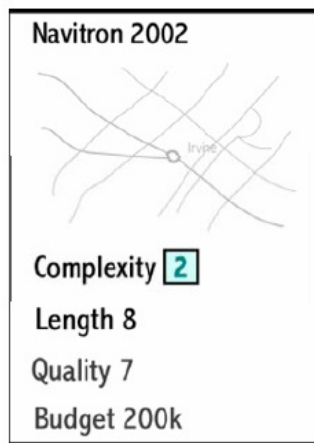
Solução Proposta: O Jogo SimulES

- **Recursos do jogo** (totalmente reformulados em relação ao PnP)
 - **Cartas atualizadas** para **conceitos** e **práticas** mais **modernas** de ES e estendidas com **informações adicionais** (*tratando problema B*)
 - Informações mais detalhadas sobre suas regras e artefatos disponíveis em [Figueiredo et al. 2006] [SimulES, 2007]

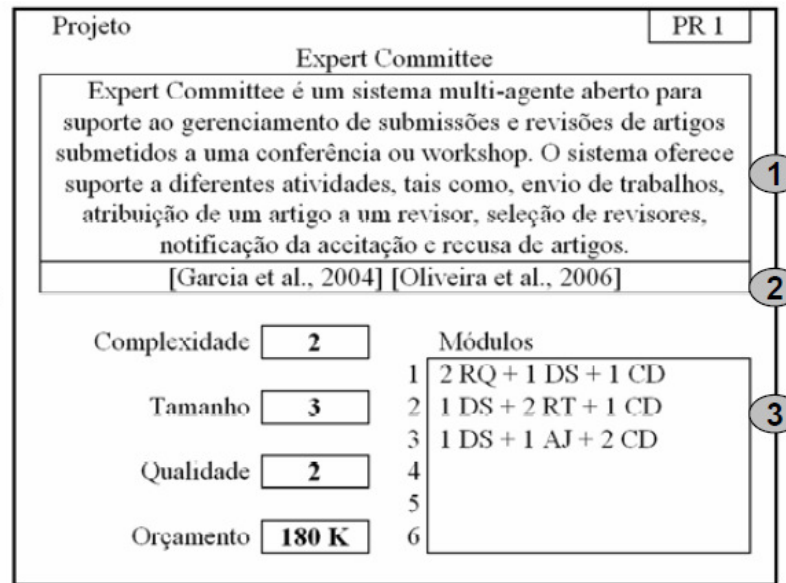
Solução Proposta: O Jogo SimulES

- **Cartões de Projetos**

- Muito simples em PnP o que motivou a criação de cartões de projetos mais **realísticos**



(a) PnP



(b) SimulES

Solução Proposta: O Jogo SimulES

- **Cartões de Projetos**

- (1) **Descrição**: Principais **características** do projeto, ajudando o jogador a compreender os principais **requisitos** do sistema (*tratando o problema B*)
- (2) **Referências**: Citações para trabalhos publicados relacionados ao **projeto** ou a seus principais **conceitos educacionais** (*tratando os problemas B e F*)

Solução Proposta: O Jogo SimulES

- **Cartões de Projetos**

- **(3) Módulos:** Para tornar o jogo mais **realista** foi introduzida a noção de que **diferentes artefatos** devem **compor os módulos** (ex.: RQ: requisitos, DS: desenho e CD: código)

Solução Proposta: O Jogo SimulES

- **Cartões de Projetos: atributos herdados do PnP**
 - **Complexidade:** pontos de tempo que um engenheiro de software precisa gastar para completar um **bom artefato**
 - **Tamanho:** indica quantos **módulos integrados** devem ser completados para terminar o projeto

Solução Proposta: O Jogo SimulES

- **Cartões de Projetos: atributos herdados do PnP**
 - **Qualidade:** o quão **livre de defeitos** deve ser o produto final
 - **Orçamento:** quantidade de **dinheiro** disponível para gastar com o projeto. **Restrição** para contratação de **engenheiros de software** e para o uso de **cartas conceitos**

Solução Proposta: O Jogo SimulES

- **O Tabuleiro** (*tratando o problema E*)
 - **Engenheiros de software** em colunas e os **artefatos** em linhas
 - Artefatos que **não existem** no PnP: **rastros e ajuda aos usuários**
 - **Rastros: interligar** artefatos e contribuir para a **gerência por requisitos** (conceitos mais atuais de ES)

Solução Proposta: O Jogo SimulES

- O Tabuleiro (*tratando o problema E*)

	Engenheiro ES1 Janaína	Engenheiro ES21 Carlos	Engenheiros de Software		
	Profissional veterano, mas com pouca habilidade no desenvolvimento. Salário: 40 K Habilidade 1 Maturidade 4	Experiência em eng. de software, mas não é amigável à equipe. Salário: 70 K Habilidade 5 Maturidade 1	Engenheiro 3	Engenheiro 4	Engenheiro 5
Requisitos					
Desenhos					
Códigos					
Rastros					
Ajudas					

Solução Proposta: O Jogo SimulES

- **As Cartas**

- **Papel principal**, assim como no PnP
- Dividas em quatro categorias: **(a) problemas**, **(b) conceitos**, **(c) engenheiros de software** (programadores em PnP) e **(d) artefatos**
 - Todas elas, exceto as cartas de artefatos, possuem **nome** e **código de identificação**. Possibilita o **controle de versão** e **gerência de configuração** do SimulES

Solução Proposta: O Jogo SimulES

• As Cartas

CD6

Entrelaçamento de Interesses

[Kiczales et al., 1997]

Jogador com desenho < 3 e código > 3

Problema Persistente:
Coloque esta carta na área do jogador. Ele irá tratar o projeto com adição de um em tamanho.

(a)

Conceito: Modelagem UML C49


[Booch et al., 1999]

Se você tem pelo menos 2 desenhos, você pode descartar esta carta para ignorar os efeitos de qualquer problema de desenho jogado para você.

(b)

Eng. de Software ES21

Carlos

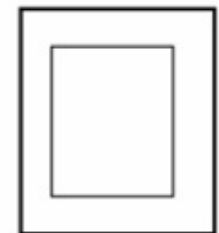
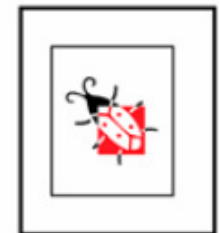


Experiência construída em dez anos de engenharia de software, mas não é amigável à equipe.

Salário: 70 K

Habilidade	5
Maturidade	1

(c)



(d)

Solução Proposta: O Jogo SimulES

- **As Cartas**

- **Grupo de cartas:** conceito criado no SimulES
(*tratando o problema F*)

- Cartas que tratam de um **mesmo tema** (ex.: RQ: requisitos, RH: recursos humanos)

Categorias	Subcategorias	Nº de Cartas
Gerenciamento	Gerência Técnica, Gerência de Versão e Rastros, Treinamento	11
Recursos Humanos	Personalidade, Recursos Financeiros, Educação	11
Requisitos	Contexto, Problemas ou Alteração, Requisitos Não-Funcionais	12
Desenho	Padrões, <i>Frameworks</i> , Qualidade de Desenho	12
Código	Testes e Assertivas, Refatoração	12
Comunicação	Interface com Usuário, Documentação de Ajuda	10

A Dinâmica do Jogo

- Similar a do PnP (diferenças destacadas abaixo)
- Jogo **limitado** entre **4 e 8 jogadores**
- **Liberdade** para escolha do **processo de desenvolvimento** (*tratando o problema A*)
- Um **cartão de projeto** é escolhido aleatoriamente (visível a todos os jogadores)
- Cada jogador monta seu tabuleiro e as cartas são separadas em quatro montes: **engenheiros de software, problemas e conceitos, artefatos brancos e artefatos cinzas**
- Com o dado é selecionado **quem começa** seguindo o **sentido horário**

A Dinâmica do Jogo

- A cada jogada, o jogador da vez lança o dado, e retira cartas no monte dependendo do valor obtido.
- As cartas de problemas, conceitos e engenheiros de software são guardadas na mão do jogador até ele achar oportuno jogá-las
- O engenheiro de software pode exercer uma série de tarefas (**construir**, **inspecionar**, **corrigir** e **integrar** artefatos) dependendo da sua habilidade (**pontos de tempo**)

A Dinâmica do Jogo

- Ao fim da sua jogada, o jogador está **apto** a receber as **cartas de problemas** dos seus adversários
 - Cada jogador recebe até **três cartas de problemas** que podem ser lançadas dos **três jogadores imediatamente anteriores a ele** (*tratando o problema D*)
- Alcançando o número de **módulos necessários** para o término do projeto, o jogador pode afirmar que **completou o sistema**

A Dinâmica do Jogo

- O sistema é **validado** pelo cliente (professor)
 - Alguns módulos são aleatoriamente conferidos e devem estar **livres de problemas**. O **número de módulos conferidos** é igual ao valor do atributo **Qualidade** do **cartão do projeto**
- O jogador **somente** é declarado vencedor se não for encontrado defeito em **nenhum** dos artefatos que compõem os módulos conferidos

Trabalhos Relacionados e Limitações do Estudo

- **Projeto SESAM: simulador** para treinamento em projetos de software [Drappa e Ludewig 2000]
 - Usuário assume o papel de **gerente** e utiliza modelos e regras complexas no desenvolvimento de um sistema
 - **Não possui características de jogos** (entretenimento e competitividade)
 - **Destinado a profissionais**

Trabalhos Relacionados e Limitações do Estudo

- **SimSE**: versão digital do PnP [Navaro et al. 2004]
 - Recai nos **mesmos problemas** do PnP
 - Perde muito do seu atrativo por **não permitir competição entre múltiplos jogadores**

Trabalhos Relacionados e Limitações do Estudo

- **Limitações do SimuleS**

- **Não foi suficiente exercitado em sala de aula:** *minimizada pelo fato do PnP, jogo em qual se baseia, ter sido muito praticado em sala de aula*
- **Baseado apenas em melhorias propostas para o PnP:** *justificável pelo fato do PnP apresentar características comuns a outros jogos educativos*

Conclusões e Trabalhos Futuros

- **Contribuições:**

- Uma **extensa avaliação do PnP** resultando na **solução** das **limitações** encontradas
- Um jogo que considera **práticas modernas** de ES e **evolução de software**, além de ser mais **dinâmico** e **atrativo** que o PnP
- **Integração** de **esforços** anteriores na **melhoria do ensino de ES**

Conclusões e Trabalhos Futuros

- **Trabalhos futuros:**
 - Avaliação **quantitativa** e/ou **qualitativa** do uso do jogo SimulES por alunos com diferentes graus de conhecimento em ES
 - Criação de uma **versão digital** do jogo utilizando recursos tecnológicos que permitam **alta interatividade** entre os jogadores

Referências

- Baker, A., Navarro, E. and Hoek A. (2005) "An Experimental Card Game for Teaching Software Engineering Processes". In: Journal of Systems and Software, v. 75, 1-2, pp. 3-16.
- deLaet, M., Kuffner, J., Slattery, M. and Sweedyk, E. (2005) "Computer Games and CS Education: Why and How". In: Symposium on Computer Science Education, USA.
- Drappa, A. and Ludewig, J. (2000) "Simulation in Software Engineering Training". In: International Conference on Software Engineering (ICSE), pp. 199-208, Limerick, Ireland.
- Figueiredo, E., Lobato, C., Dias, K., Leite, J. e Lucena, C. (2006) "SimulES: Um Jogo para o Ensino de Engenharia de Software". Relat. Técnico 34/06, Depto de Informática, PUC-Rio.
- Lehman, M. (1996) "Laws of Software Evolution Revisited". In: 5th European Workshop on Software Process Technology, LNCS, vol. 1149, pp. 108-124.
- Mylopoulos, J., Chung, L. and Yu, E. (1999) "From Object-oriented to Goal Oriented Requirements Analysis". In: Communications of the ACM, vol. 42, no. 1, pp. 31-37.

Referências

- Navarro, E., Baker, A. and Hoek, A. (2004) "Teaching Software Engineering Using Simulation Games". In: International Conference on Simulation in Education (ICSIE), California, USA.
- Oh, E. and Hoek, A. (2001) "Adapting Game Technology to Support Individual and Organizational Learning". In: Int'l Conf. on SE & Knowledge Eng. (SEKE), p. 347-354, AR.
- Royce, W. (1970) "Managing the Development of Large Software Systems", In: IEEE WESCON, IEEE Press, pp. 1-9, San Francisco.
- "SimulES: Simulador de Engenharia de Software" (2007). Disponível on-line em <http://www.teccomm.les.inf.puc-rio.br/emagno/simules/>. Acessado em Fevereiro, 2007.
- Virvou, M., Katsionis, G., Manos, K. (2005). "Combining Software Games with Education: Evaluation of its Educational Effectiveness". Educational Technology & Society, pp. 54-65.