

# ENGSOFT: Ferramenta para Simulação de Ambientes Reais para auxiliar o Aprendizado Baseado em Problemas (PBL) no Ensino de Engenharia de Software

Bruno Rodrigues Bessa<sup>1,3</sup>, Mônica Ximenes Carneiro da Cunha<sup>1</sup>, Felipe Furtado<sup>2,3</sup>

<sup>1</sup>IFAL – Instituto Federal de Alagoas

<sup>2</sup>Centro de Informática – Universidade Federal de Pernambuco

<sup>3</sup>C.E.S.A.R – Centro de Estudos e Sistemas Avançados do Recife

professorbessa@gmail.com, mxcc@hotmail.com, fsfs@cesar.org.br

***Abstract.** The use of educational tools that stimulate curiosity and foster motivation for learning is a topic widely researched and discussed. However, it is uncommon in the teaching of Software Engineering. This paper presents the EngSoft, a tool that simulates a context of a metropolis with software factories, educational institutions, companies and their participants, where each has a specific role and responsibilities in their environments. The tool enables the development of existing experiences in the real world, thus reducing the gap between theory and practice.*

***Resumo.** O uso de ferramentas educacionais que estimulam a curiosidade e promovem motivação para o aprendizado é um tema amplamente pesquisado e discutido. No entanto, é pouco comum no ensino da Engenharia de Software. Esse artigo apresenta a EngSoft, uma ferramenta que simula um contexto de uma metrópole com fábricas de softwares, instituições de ensino, empresas e respectivos participantes, onde cada um tem papel e atribuições específicas nos seus ambientes. A ferramenta permite o desenvolvimento de experiências existentes no mundo real, reduzindo assim as lacunas existentes entre teoria e prática.*

## 1. Introdução

O cenário brasileiro apresenta escassez de profissionais devidamente qualificados para atuarem em profissões que envolvem as etapas do processo de desenvolvimento de software, compreendidas pela Engenharia de Software (ABES, 2009).

O ensino da Engenharia de Software (ES) está passando por vários tipos de questionamentos, principalmente, por parte dos estudantes e da indústria (THOMPSON, 2007; HILBURN; TOWHIDNEJAD, 2007). Infelizmente, as disciplinas relacionadas à ES têm sido ministradas de forma teórica e metódica, tornando o ensino pouco atrativo aos estudantes. Dessa forma, as práticas relacionadas à indústria de software têm ficado prejudicadas.

Tomomitsu, Camargo e Novelli Filho (2006) registraram que o ensino da ES precisa de ferramentas que simulem ambientes de fábrica de software e respectivos clientes a fim de minimizarem as lacunas de formação tais como: a identificação de papéis dentro das fábricas, o relacionamento com o cliente e os problemas relacionados a escopo, prazo, custo e qualidade.

Os profissionais de engenharia de software que trabalham na indústria apresentam insatisfação quanto ao nível de preparação dos universitários recém-formados que entram no mercado de trabalho (THOMPSON, 2007; HILBURN; TOWHIDNEJAD, 2007). As empresas de software têm que complementar os conhecimentos dos recém-formados com treinamentos e prover habilidades nos aspectos técnicos e não técnicos relacionados ao desenvolvimento de sistemas de software. Embora a universidade não seja uma empresa, deve preparar seus estudantes para os reais desafios que encontrarão no mercado de trabalho. Geralmente, as instituições de ensino não dispensam a devida atenção aos aspectos não técnicos na formação do aluno, como, por exemplo, ensiná-lo a se comunicar de forma eficiente e ativa dentro dos ambientes de fábrica de software.

A raiz do problema pode estar na forma como a ES está sendo ministrada, ou seja, na metodologia adotada para o ensino de ES. Nos últimos anos, a academia tem investido muito esforço para mitigar esse problema através da elaboração de novas formas de ensinar ES. Algumas instituições chegaram a inovar com o desenvolvimento de algumas ferramentas que têm como objetivo mobilizar os alunos, bem como ensinar ou apoiar o ensino de ES (BAKER; NAVARRO; HOEK, 2005).

Nesse contexto, o objetivo deste trabalho é propor uma ferramenta que possibilite a redução da insatisfação quanto ao nível de preparação dos alunos de graduação dos cursos voltados à Engenharia de Software que entram no mercado de trabalho. Esta ferramenta fornece um conjunto de ambientes como: ambiente de comunicação síncrona e assíncrona, ambiente baseado em fábrica de software para auxiliar a condução de projetos de software, ambiente de ensino para a avaliação e acompanhamento dos alunos na execução de projetos de software e ambiente empresarial para contextualizar de forma bastante realista situações encontradas no mercado. Todos elaborados de forma a possibilitar o desenvolvimento de experiências existentes no mundo real, utilizando a metodologia de aprendizagem baseada em problemas (PBL).

Este artigo está organizado em seis seções. A seção 2 apresenta o referencial teórico envolvendo conceitos inerentes ao ensino da ES e ao uso do método PBL no ensino de ES. A seção 3 apresenta os trabalhos relacionados e as características mais relevantes da ferramenta proposta em relação às demais citadas. A seção 4 apresenta a ferramenta ENGSOFT e suas funcionalidades. A seção 5 apresenta os dados colhidos junto aos alunos, referentes à implantação e à validação da ferramenta no ambiente institucional. E, finalmente, a seção 6 apresenta as conclusões e considerações finais.

## **2. Referencial Teórico**

### **2.1 O Ensino da Engenharia de Software**

A ES é uma disciplina preocupada com a aplicação da teoria, do conhecimento e da prática para o desenvolvimento efetivo e eficiente de sistemas de software que satisfaçam os requisitos dos usuários. Geralmente, ES é ensinada no ensino formal no nível de graduação e/ou pós-graduação ou por meio de treinamentos profissionais de curta duração.

No ensino tradicional da ES, o professor apresenta diversos assuntos e aplica uma ou mais provas de avaliação individual no semestre e uma série de documentos relacionados a um pequeno projeto, que geralmente não apresenta resultados satisfatórios. Com essa proposta, não raro os alunos saem da academia sem terem participado de um processo de ES próximo ao que irão encontrar no mercado de trabalho, que envolve: sistemas grandes e complexos; a participação em uma equipe também grande e distribuída geograficamente, com mudanças de objetivos durante o projeto; problemas com os clientes; pressão com tempo de entrega do produto; uma maior demanda por qualidade de software; e outros fatores como gerenciamento, questões de espaço de trabalho e cultura corporativa.

### **2.2 O Aprendizado Baseado em Problemas Reais**

Desenvolvido na educação médica na década de 70, o método PBL<sup>1</sup> tem sido adaptado em um número crescente de áreas de atuação e em diferentes níveis educacionais (GÜZELIS, 2006). Sob uma ótica bem simples, PBL pode ser definido como um método instrucional que usa um problema para iniciar, direcionar e motivar o aprendizado. Como tal, PBL é consistente com os princípios da abordagem construtivista, que defende que o que as pessoas entendem é uma função do conteúdo, contexto, atividades e objetivos do aprendiz (SAVERY, 2006).

Peterson (1997) ressalta três importantes princípios que promovem um aprendizado mais eficaz com o uso de PBL: 1) O aprendizado acontece em um ambiente onde os estudantes estão imersos na prática, em atividades em que recebem *feedback* de seus colegas estudantes e professores; 2) Os estudantes recebem guias e suporte de seus pares, de maneira a promover um ensino multidirecional envolvendo outros estudantes, professores e monitores, diferentemente do ensino convencional, normalmente unidirecional (professor para estudante); 3) O aprendizado é funcional, a partir de problemas reais.

A adoção de PBL não é uma tarefa fácil, principalmente porque propõe a quebra do paradigma do ensino convencional já tão enraizado na cultura brasileira. Exige mudança de postura tanto do professor como do estudante, trabalhando uma forma de aprendizagem altamente investigativa e questionadora, além de certa maturidade profissional (MORAES; MEIRA; ALBUQUERQUE, 2003).

---

<sup>1</sup> Problem Based Learning (PBL) ou **Aprendizagem Baseada em Problemas**.

### 3. Trabalhos Relacionados

A utilização de ferramentas como forma de melhorar o aprendizado é um tema bem explorado na literatura. Na área de ensino da ES, propriamente dito, já existem iniciativas para utilização de jogos ou simuladores de ambientes educacionais, como as apresentadas no Quadro 1.

**Quadro 1: Resumo das principais ferramentas de ensino de Engenharia de Software.**

| Projeto    | Descrição   | Autor                                  |
|------------|---|--|
| Honey      | O Honey é um jogo que simula um ambiente no qual o usuário pode clicar em portas de salas de escritório onde conteúdos sobre XP (Extreme Programming) são apresentados. | Souza <i>et al.</i> (2008)             |
| Elicit@ção | O jogador realiza entrevistas com personagens virtuais para identificar requisitos.   | Bernardi, Fontoura e Cordenonsi (2008) |
| Planager   | O Planager apoia o ensino de conceitos de gerência de projetos, focado no grupo de processos de planejamento do PMBOK.  | Rosa e Kieling (2006)                  |
| Scrumming  | O Scrumming é um jogo de apoio ao ensino e prática de conceitos do SCRUM, permitindo a simulação da definição de um sprint de um projeto por vez.                       | Isotton (2008)                         |

Na maioria das ferramentas citadas acima, a forma de utilização é individual e cerca de 70% é executada localmente. As áreas de conhecimentos que são exploradas são processos de software, requisitos e, na grande maioria, gestão de projetos. É importante destacar, que todas as ferramentas têm seu contexto de uso focado em cursos de graduação em computação e sistemas de informação provendo resultados baseados em conhecimentos e habilidades.

Diferente das ferramentas supracitadas, a ferramenta apresentada neste trabalho propõe um aprendizado focado na coesão de grupo e na educação da coletividade. A aquisição do conhecimento está focada na absorção do mesmo através de problemas reais e aprendizagem prática caracterizada pelo envolvimento dos alunos em determinadas atividades, com resultados mais positivos, visto que os alunos tornam-se responsáveis pela definição de rumos de uma situação proposta.

### 4. EngSoft: Ferramenta Virtual de Aprendizagem

A ferramenta EngSoft contém uma interface intuitiva, que simula um contexto de parque tecnológico universitário, apresentado na Figura 1, composto por fábricas de projetos, fábricas de programas, fábrica de testes, instituições de ensino e ambientes empresariais.

Todos os ambientes possuem seus respectivos representantes ou atores. Ressalta-se que cada ator tem seu papel e atribuições no contexto do seu ambiente. A ferramenta foi desenvolvida utilizando JAVA JDK 6 e Flex como sistema na plataforma Web e banco de dados MySQL.



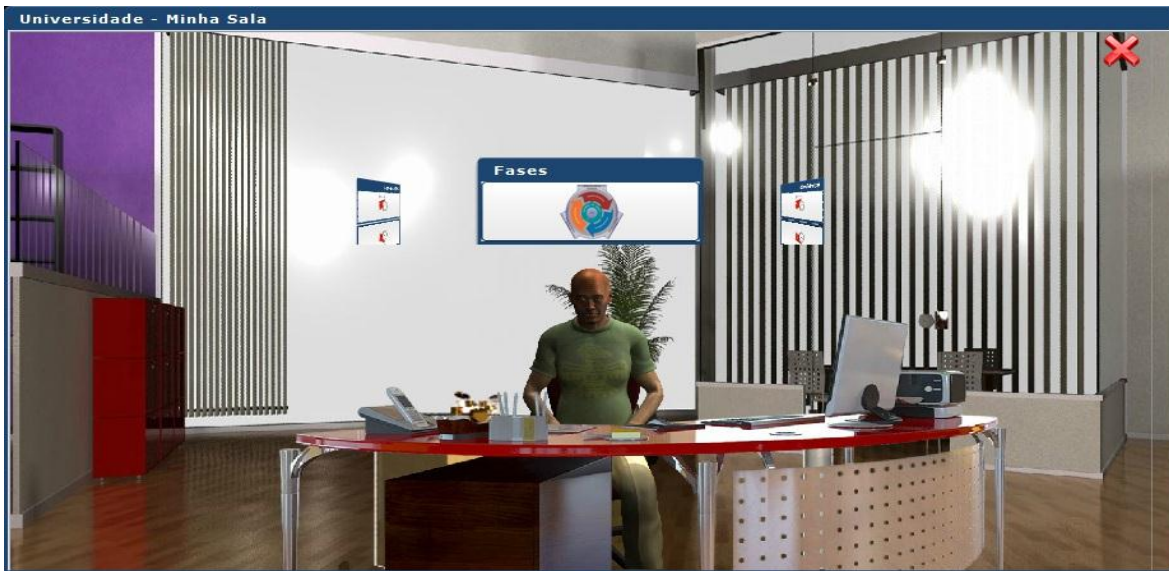
**Figura 1 – Parque Tecnológico Universitário.**

No ambiente de fábrica de software, apresentado na Figura 2, o gerente de projeto poderá cadastrar projetos, vincular participantes aos projetos, definir cronograma do projeto, cadastrar tarefas e anexar arquivos as mesmas e encaminhá-los aos respectivos participantes da fábrica de software. A comunicação com os participantes da fábrica ocorre de forma assíncrona e síncrona, possibilitando o agendamento das reuniões com as partes envolvidas no projeto (empresa cliente e participante do projeto).



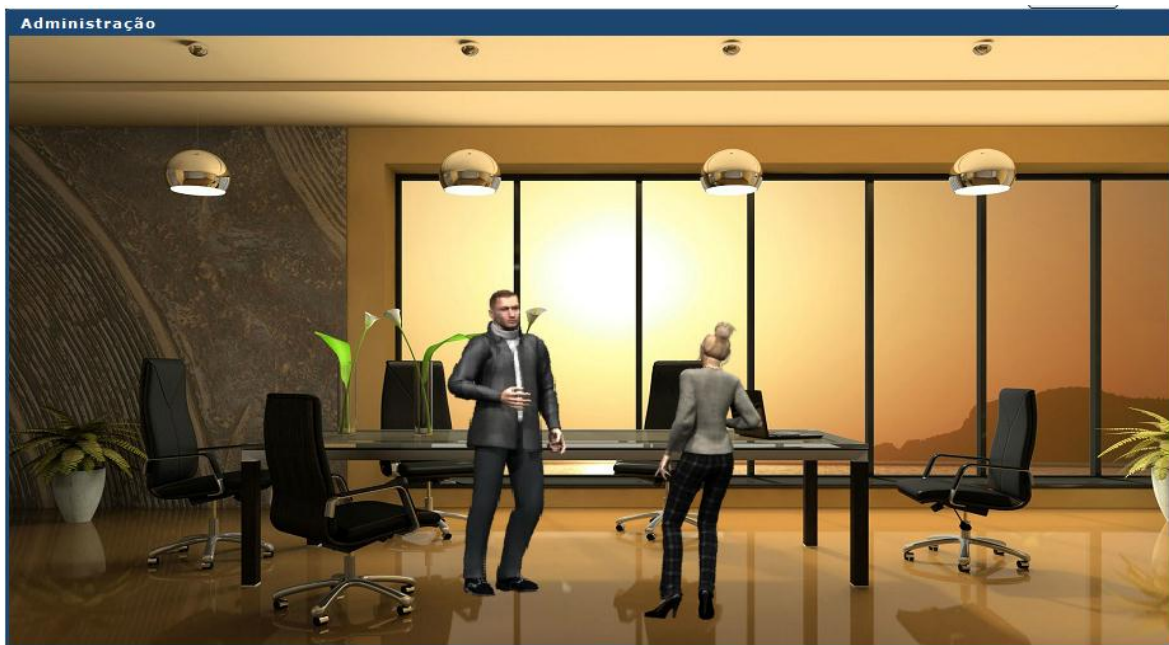
**Figura 2 – Ambiente Fábrica de Software.**

No ambiente instituição de ensino, apresentado na Figura 3, o participante com perfil de professor poderá atuar como consultor, orientador, facilitador e mediador entre as fábricas e clientes e realizar seminários.



**Figura 3 – Ambiente Instituição de Ensino.**

Já no ambiente empresarial, apresentado na Figura 4, o participante com perfil de empresário poderá acompanhar e participar ativamente de todos os seus projetos.



**Figura 4 – Ambiente Empresa.**

A metodologia identifica três grupos de usuários envolvidos no processo e criação das fábricas de software: os funcionários (analista de sistemas, analista de teste, gerente de qualidade etc) que compõem uma fábrica, responsáveis pela definição do negócio da fábrica e desenvolvimento de software; os professores, responsáveis por ministrar seminários via web, executando o papel de consultores especialistas em áreas de sua disciplina e facilitadores no processo de aprendizado; os clientes, responsáveis pela demanda do software, validação de requisitos e dos resultados acordados com as fábricas de software. O diagrama de caso de uso, que sinaliza as potencialidades de cada grupo de usuários supracitado, está ilustrado na Figura 5.

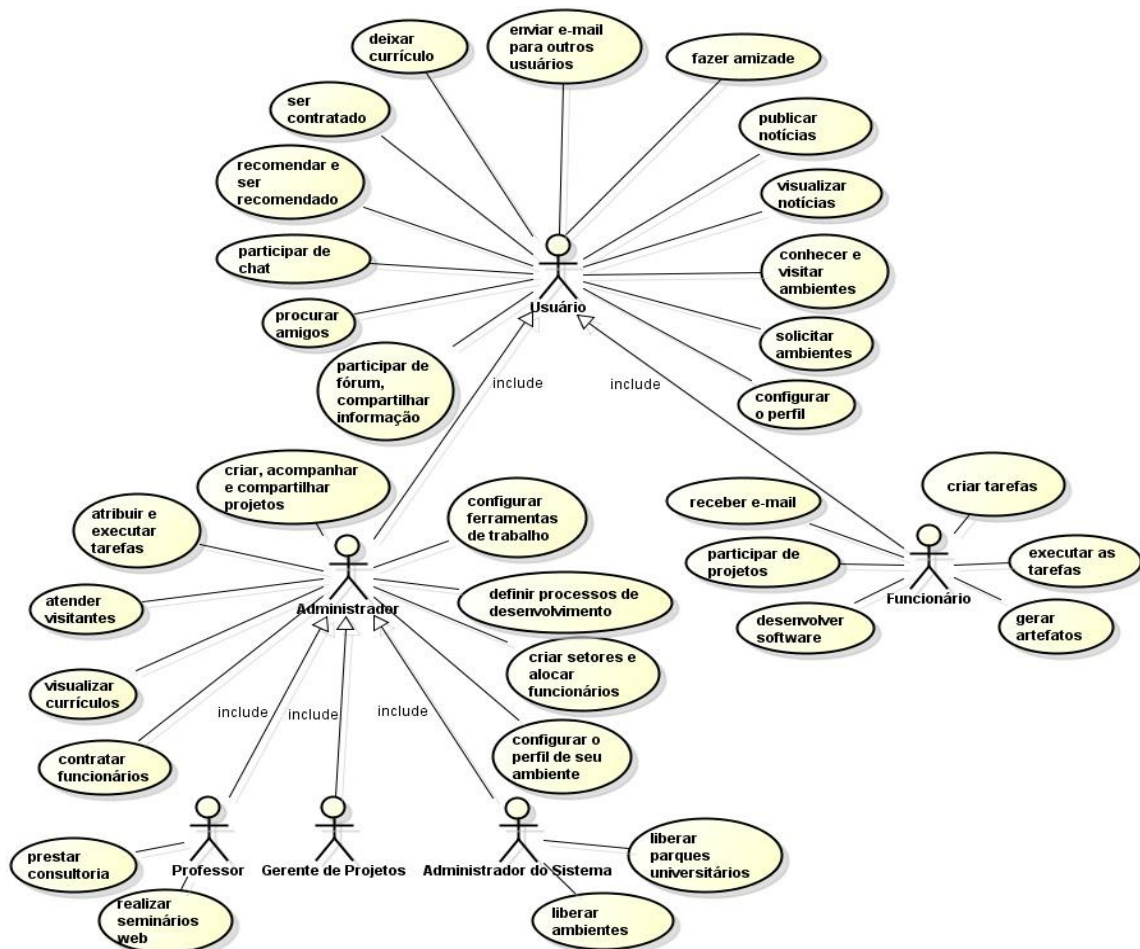


Figura 5 – Diagrama de Caso de Uso da EngSoft.

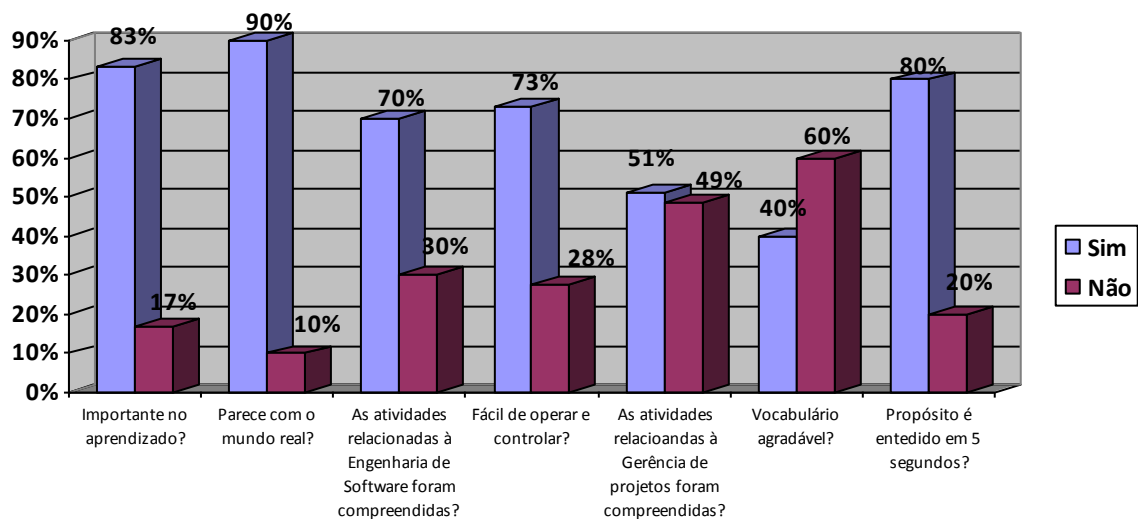
## 5. Implantação e Validação da Ferramenta em uma Instituição de Ensino

No período de 09/2011 a 06/2012 a ferramenta está sendo utilizada pelos 25 alunos da disciplina Projeto Integrador e pelos 02 professores da disciplina do curso de Análise e Desenvolvimento de Sistemas de uma instituição de ensino, a fim de contribuir para a melhoria do processo-aprendizagem da Engenharia de Software dos seus alunos.

Visando atender o princípio 1 de Peterson (1997) sobre a PBL, os professores coordenaram o processo de construção das fábricas de software dentro da ferramenta. Esse processo aconteceu através dos estudantes que foram orientados a compor grupos de 4 a 5 integrantes e a formalizarem suas respectivas fábricas de software. A divisão de equipes considerou afinidades, competências e habilidades de cada estudante para que se evitasse concentração e/ou escassez de competências e habilidades em uma mesma equipe. Foi necessário alocar os papéis entre os membros da fábrica, tais como: gerente de projeto, arquiteto de software, engenheiro de software, analista de negócio, engenheiro de testes.

Para atender o princípio 2 de Peterson (1997), os professores criaram um ambiente universitário para atender as fábricas de software executando o papel de consultores especialistas em áreas de sua disciplina, facilitando no processo de aprendizado e auxiliando na criação e análise dos produtos gerados, por meio de reuniões periódicas de acompanhamento, análise de artefatos gerados e avaliações individuais de desempenho. Por fim, em atenção ao princípio 3 de Peterson (1997), os mesmos criaram um ambiente empresarial para simular o cliente, e demandaram 1 projeto de software para cada fábrica de software. Simulando assim o envolvimento cliente-fornecedor com a demanda de projetos de software, validação de requisitos e dos resultados acordados com as fábricas de software, com o intuito de promover a familiaridade com problemas reais.

Uma avaliação da ferramenta EngSoft foi realizada como teste de aceitação, em que os professores, juntamente com um grupo de estudantes, utilizaram a ferramenta em conjunto. Ao final da análise os estudantes e professores responderam um questionário de avaliação do sistema, composto de sete perguntas fechadas (apenas com opções de resposta SIM ou NÃO). Os resultados obtidos são apresentados na Figura 6.



**Figura 6 – Opinião dos usuários sobre a ferramenta EngSoft.**

De maneira geral, os conceitos disseminados pelos professores foram assimilados e exercitados através das práticas e dinâmicas da ferramenta. Devido aos aspectos comportamentais das práticas, que englobam os diversos perfis da ES, muitos alunos tiveram reações semelhantes às encontradas em ambientes reais de desenvolvimento.



## 6. Conclusões e Trabalhos Futuros

Este artigo apresentou uma ferramenta que auxilia o ensino da ES, tomando por base os princípios gerais da metodologia PBL. A ferramenta disponibiliza ambientes de comunicação síncrona e assíncrona, com vistas a melhorar a comunicação entre os alunos e professores, ambiente baseado em fábrica de software para auxiliar a condução de projetos e aumentar a capacidade gerencial, bem como favorecer o trabalho em equipe, além de um ambiente de ensino para a avaliação e acompanhamento dos alunos, permitindo que os professores atuem como consultores, orientadores, facilitadores e mediadores entre as fábricas e clientes. E ainda ambientes empresariais para contextualizar de forma bastante realista situações encontradas no mercado, reduzindo assim a lacuna existente entre teoria e prática.

De acordo com os dados coletados após a implantação, foi possível perceber que os alunos têm demonstrado uma melhora significativa no aprendizado das práticas de ES. A metodologia aplicada permite um aprendizado mais consistente do modelo de processo de software com todas as etapas do desenvolvimento dos projetos. O sentimento positivo demonstrado pelos alunos em relação à usabilidade, aplicabilidade e flexibilidade da ferramenta foi fator decisivo para utilizá-la nos semestres seguintes para fins de melhoramento e refinamento. No futuro planeja-se desenvolver um ambiente que suporte o desenvolvimento distribuído de software e um de gerenciamento de reuniões. Também se pretende disponibilizar a ferramenta de forma mais ampla para a academia e para a indústria.

## Referências

- ABES. Associação Brasileira das Empresas de Software. Mercado Brasileiro de Software: panorama e tendências. São Paulo: ABES, 2009.
- BAKER, A.; NAVARRO, E. O; HOEK, A. (2005) An Experimental Card Game for Teaching Software Engineering Processes. **Journal of Systems and Software**. Volume 75. p. 3-16. New York.
- BERNARDI, G.; FONTOURA, L. M.; CORDENONSI, A. Z. Elicit@ção: Ferramenta de Apoio ao Ensino de Elicitação de Requisitos de Software baseada em Instituições Eletrônicas. In: II Workshop-Escola de Sistemas de Agentes para Ambientes Colaborativos. **Anais...** Santa Cruz do Sul – RS, 2008.
- GÜZELIS, C. An Experience on Problem Based Learning in an Engineering Faculty. **Turk J Elec Engin**, Vol.14, No., p. 67-76, 2006.
- HILBURN, T.B.; TOWHIDNEJAD, M. A Case For Software Engineering. In: Conference On Software Engineering Education & Training – CSEET '07, 20, 2007, Dublin. **Proceedings ...** Washington: IEEE Computer Society, 2007, pp. 107-114.
- ISOTTON, E. (2008) **Scrumming - Ferramenta Educacional para Ensino de Práticas de SCRUM**. Trabalho de Conclusão de Curso (Graduação em Bacharelado em Sistemas de Informação) - Pontifícia Universidade Católica do Rio Grande do Sul.

- MORAES, A. K.O.; MEIRA, S. R. L.; ALBUQUERQUE, J. O. Open Source Software Factory – Step by Step: A Case Report. In: FIRST INTERNATIONAL CONFERENCE ON OPEN SOURCE COLLABORATIVE DEVELOPMENT PLATFORMS (Héphaïstos), 2006.
- PETERSON, M. Skills To Enhance Problem-Based Learning. **Med Educ Online** [Serial Online] 2, 3, 1997.
- ROSA, R. E KIELING, E. (2006) **Planager - Um Jogo para Apoio ao Ensino de Gerência de Projetos de Software**. Trabalho de Conclusão de Curso (Graduação em Bacharelado em Sistemas de Informação) - Pontifícia Universidade Católica do Rio Grande do Sul.
- SAVERY, JOHN R. (2006) Overview of Problem-based Learning: Definitions and Distinctions, **Interdisciplinary Journal of Problem-based Learning**. Vol. 1, Iss. 1, Article 3. Disponível em: <http://docs.lib.purdue.edu/ijpbl/vol1/iss1/3>. Acesso em 05 de maio de 2012.
- SOUZA, D. A. C. M.; VASCONCELOS, C. R.; AZEVEDO, R.; FUJIOKA, R. C.; ALMEIDA, M. J. S. C.; FREITAS, F. (2008) Honey: Um Ambiente Virtual Baseado em Agentes para Apoiar o Ensino de Engenharia de Software. In: Simpósio Brasileiro de Informática na Educação, 19, 2008. **Anais...** v. 1. p. 55-64. Fortaleza-CE, 2008.
- THOMPSON, J. B. Improving Software Practice through Education: Challenges and Future Trends. In: FUTURE OF SOFTWARE ENGINEERING, INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING – ICSE '07, 29, 2007, Minneapolis. **Proceedings...** Washington: IEEE Computer Society, 2007, p. 12-28
- TOMOMITSU, C. K. A.; CAMARGO, V. L. S.; NOVELLI FILHO, A. Dimensões a considerar na análise dos problemas de ensino e aprendizagem de engenharia de software. In: Workshop de Pós-Graduação e Pesquisa, 1, 2006. **Anais...** São Paulo: FATEC, 2006.