

Ensino guiado por estudo dirigido: uma experiência no contexto da Engenharia de *Software*

Ronnie E. S. Santos¹, Cleyton V. C. Magalhães¹

Jorge S. Correia-Neto², Guilherme Vilar²

¹ Centro de Informática – Universidade Federal de Pernambuco (CIn/ UFPE)
Av. Jornalista Anibal Fernandes – Cidade Universitária, 50670-901 – Recife-PE – Brasil

² Programa de Pós-Graduação Em Informática Aplicada (PPGIA) – Universidade
Federal Rural de Pernambuco (UFRPE)
Rua Dom Manoel de Medeiros, s/n, Dois Irmãos, 52171-900 – Recife-PE – Brasil

{ress, cvcm, eprs}@cin.ufpe.br, jorgecorreianeto@gmail.com,
guilherme_vilar@yahoo.com.br

Abstract. Teaching quality and the understanding of *Software* Engineering contents and subjects has direct impact in improving *software* development process, and it can support the search for solutions to issues from *software* industry. This research presents the results an experience report of teaching *software* engineering in a context in which the students already have practical and professional experience about the subjects addressed. Thus, a teaching technique guided by directed study used an analysis of case studies, experiments, fieldworks and experiences reports published in conferences and journals to discuss theoretical contents of the discipline by scientific research. It provide to the students to compare results obtained by studies analyzed with their personal daily experience.

Resumo. A qualidade do ensino e a compreensão de conteúdos e temas da engenharia de *software* tem impacto direto na melhoria do processo de desenvolvimento de *software*, podendo apoiar a busca por soluções de problemas e deficiências da indústria de *software*. Esta pesquisa apresenta o relato de experiência de ensino da disciplina de engenharia de *software* em um contexto no qual os alunos já possuem experiência prática e profissional sobre os temas abordados. Desta forma, uma técnica de ensino guiada por estudos dirigidos utilizou a análise de estudos de caso, experimentos, pesquisas de campo e relatos de experiências publicadas em congressos e revistas da área, para discutir o conteúdo teórico da disciplina através do contexto das pesquisas científicas, permitindo aos alunos comparar os resultados obtidos pelos estudos, com o que é vivenciado diariamente no seu ambiente profissional.

1. Introdução

A engenharia de *software*, de forma ampla, é a área do conhecimento que fornece toda a estrutura para o desenvolvimento e manutenção de *software* (IEEE, 2004). Esta área está preocupada principalmente com a aplicação de teoria, conhecimento e prática para o

desenvolvimento efetivo e eficiente de sistemas de *software*, visando a implementação de produtos de qualidade, econômicos, úteis e no prazo esperado, cujos resultados devem satisfazer os requisitos dos usuários (ACM/IEEE, 2008).

Segundo Gibbs (1994), a qualidade do ensino e a compreensão dos conteúdos e temas da engenharia de *software* têm grande impacto na melhoria do processo de desenvolvimento de *software* e, conseqüentemente, podem apoiar na busca por soluções de problemas e deficiências observadas na prática, no contexto da indústria de *software*. Por isso, por ser uma importante disciplina da área de Ciência da Computação, muitos defendem a aplicação prática das teorias estudadas de modo a enfatizar a aprendizagem de aspectos tecnológicos, tais como linguagens de especificação, métodos e ferramentas e também os aspectos sociais do desenvolvimento, como a cooperação e a comunicação entre os desenvolvedores (SILVA; LEITE; BREITMAN, 2004).

De acordo com dados de 2010 divulgados pelo Ministério da Educação (MEC), dos quase 5,5 milhões de estudantes do ensino superior brasileiro, 63% frequentam as aulas à noite e trabalham para cobrir as despesas com a graduação. Além disto, outra realidade apontada pela pesquisa é que, enquanto na rede pública apenas 37% dos universitários têm aulas à noite, esse número salta para 72% na rede privada (TERRA EDUCAÇÃO, 2012). É neste contexto que está situada a proposta desta pesquisa, considerando que muitos dos estudantes dos cursos noturnos de computação ocupam empregos na indústria de *software* ou áreas relacionadas. Por isso a abordagem de Schneider e Johnston (2003), que discute a necessidade de trabalhos práticos nas disciplinas de computação para que os alunos consigam assimilar o conhecimento transferido, pode não ser tão eficiente. Isso pode ser observado pelo fato de que os alunos dos cursos noturnos possuem experiências práticas diárias sobre o conteúdo que está sendo ensinado.

Neste sentido, esta pesquisa apresenta um relato de experiência de ensino de engenharia de *software* aplicando uma abordagem de ensino guiada por estudos dirigidos, na qual o conhecimento prático da disciplina é construído pelos alunos através da análise de pesquisas científicas publicadas anteriormente sobre um determinado tema. Conforme definido por Nérici (1992), o estudo dirigido é o primeiro passo para tornar o educando independente do professor, orientando-o para estudos futuros e participação na sociedade. Além disso, o estudo dirigido apresenta duas funções principais; a primeira é de consolidação dos conhecimentos por meio de uma combinação da explicação do professor com exercícios; a segunda, é a busca da solução dos problemas por meio de questões que os alunos possam resolver criativamente e de forma independente (LIBÂNEO, 1994). Desta forma, a análise de estudos de caso, experimentos, pesquisas de campo e relatos de experiências sobre diversos conteúdos da área permite aos alunos comparar os resultados obtidos em um determinado contexto, com aquilo que é vivenciado diariamente no ambiente profissional.

A construção do presente estudo segue organizado em quatro seções, a partir desta introdução. A segunda seção apresenta informações conceituais acerca do método científico adotado para a realização desta pesquisa, além de relatar experiências semelhantes no contexto da Informática na Educação (IE). Na terceira seção é apresentado o relato de experiência na aplicação dos estudos dirigidos na engenharia de *software* e, a quinta seção, apresenta as considerações finais sobre o tema.

2. Trabalhos Relacionados

Esta seção apresenta pesquisas anteriores sobre o mesmo tema estudado neste trabalho.

2.1. Uma experiência de ensino de Engenharia de *software* orientada a trabalhos práticos

Em sua pesquisa, Soares (2003) descreveu uma experiência de ensino da disciplina de Engenharia de *software* em um curso de graduação em Ciência da Computação em um processo orientado ao desenvolvimento de projetos práticos por grupos de alunos na sala de aula, visando aplicar conceitos teóricos da disciplina. Os alunos foram incentivados a serem participantes do processo de ensino de maneira ativa em um processo em que o professor se tornou um gestor de ensino, auxiliando e provendo suporte para o aprendizado dos alunos e não somente repassando a informação disponibilizada em livros.

O autor reforça a inclusão de trabalhos práticos, em equipe, como um artifício para desenvolver aspectos mais técnicos e práticos do futuro profissional da turma. Aspectos como saber trabalhar em equipe, sintetizar, escrever e apresentar oralmente resultados obtidos, dentre outros, vivenciados no desenvolvimento dos projetos são fundamentais na carreira de estudantes do curso de Ciência da Computação. A abordagem de ensino orientada a trabalhos práticos procurou balancear a teoria e a prática para que os conceitos pudessem ser compreendidos e não simplesmente memorizados.

Esta abordagem apresenta resultados satisfatórios, no entanto no contexto em que esta pesquisa está inserida, pode não desenvolver os mesmos benefícios, uma vez que os alunos já têm contato com os conceitos na prática. Como discutido anteriormente na introdução, todos os alunos da disciplina discutida nesta pesquisa já têm experiência prática na indústria de *software*. Desta forma, foi necessário identificar um outro tipo de abordagem para o ensino de engenharia de *software* neste contexto.

2.2. Ferramentas, métodos e experiências no ensino de Engenharia de *Software*: um mapeamento sistemático

Em uma pesquisa anterior (no prelo), foi realizado um mapeamento sistemático, que de acordo com KITCHENHAM *et al.* (2004), trata-se de um estudo secundário empregado para integrar os resultados oriundos de diversos estudos publicados anteriormente. Este mapeamento foi realizado com o intuito consolidar informações quantitativas e qualitativas acerca do ensino de engenharia de *software*, discutindo sobre ferramentas, métodos e experiências de ensino desta disciplina publicadas em cinco importantes canais de divulgação da pesquisa sobre IE no Brasil: a) Revista Brasileira de Informática na Educação (RBIE); b) o Simpósio Brasileiro de Informática na Educação (SBIE); c) o *Workshop* de Informática na Escola (WIE); d) o *Workshop* sobre Educação em Computação (WEI) e; e) a Revista Novas Tecnologias na Educação (RENOTE).

Nesta pesquisa foram identificadas 26 contribuições sobre este tema, em um universo de mais de 1200 publicações dos congressos e revistas da comunidade

brasileira de IE, como mostra a figura 1. Destas propostas, 48% das pesquisas estiveram voltadas para o desenvolvimento de ferramentas computacionais de suporte ao ensino de engenharia de *software* (*software* educativo, jogos, simuladores, dentre outros), 32% apresentaram propostas didáticas e métodos de ensino da disciplina sem necessariamente uso de *software* educativo (debates, projetos, dinâmicas e jogos reais) e os demais estudos analisados (20%) apresentaram o relato de experiências de professores ao lecionar esta disciplina em cursos de graduação ou especialização.

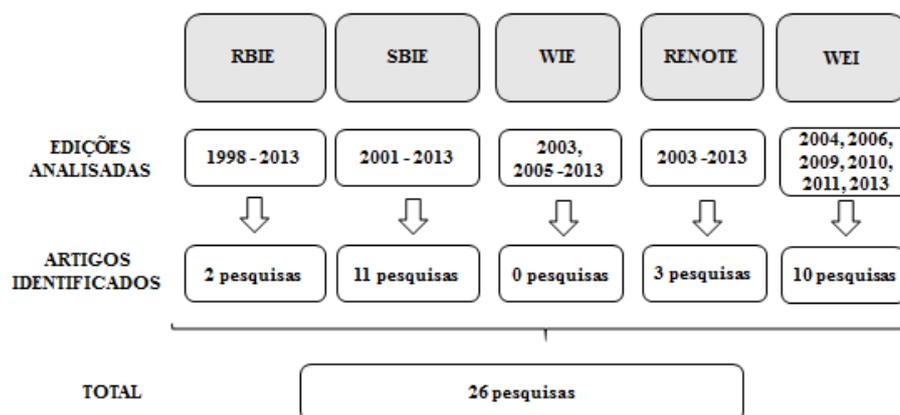


Figura 1. Experiência sobre o ensino de engenharia de *software*.

Ao analisar os tópicos da engenharia de *software* englobados pelas pesquisas, tomando como base as definições do SWEBOK¹, foi observado que 28% dos estudos identificados pelo mapeamento não tiveram foco específico em um único conteúdo ou tema, englobando os aspectos gerais e questões diversas sobre a área. Enquanto isso, 72% das pesquisas estiveram diretamente relacionadas com um tópico específico da área dentre os apresentados no SWEBOK. Entretanto, estas propostas discutiram questões sobre o ensino de apenas cinco dos quinze capítulos sobre engenharia de *software* englobados pelo guia: *Software Requirements*, *Software Construction*, *Software Testing*, *Software Engineering Management* e *Software Quality*.

O mapeamento demonstrou que vários tópicos de estudo da engenharia de *software* ainda não apresentam nenhum tipo de experiência de ensino publicada nos meios de divulgação de pesquisa da comunidade brasileira de engenharia da computação e indicou pontos em que novas pesquisas são necessárias para o desenvolvimento do ensino nesta área. Entretanto, nenhuma pesquisa identificada discute o ensino de engenharia de *software* para grupos de aluno que já possuem vivência prática na indústria de *software*.

3. Experiência de ensino de engenharia de *software* guiado por estudos dirigidos

Esta pesquisa pode ser caracterizada como um estudo do tipo descritivo-exploratório, pois investiga, observa, descreve e evidencia aspectos de uma experiência real no ensino

¹ *Guide to the Software Engineering Body of Knowledge*. Disponível em: <http://www.computer.org/portal/web/swebok> Acesso em: 12 jul. 2014.

de engenharia de *software* para alunos de graduação em Computação que possuem experiência prática na indústria de *software*. As observações foram realizadas em uma faculdade particular². A disciplina de engenharia de *software* foi ministrada no sexto período de Ciência da Computação, com dez alunos matriculados, sendo que todos estes alunos possuíam experiências atuais ou anteriores na indústria de *software*.

Para desenvolver as atividades, o tempo total de cada aula (2 horas) guiada por estudos dirigidos, foi dividido de maneira a permitir três atividades: apresentação teórica do conteúdo da aula (aula expositiva), distribuição de pesquisas científicas sobre o tema para análise por grupos de alunos, e por fim, socialização e discussão sobre os artigos analisados por cada grupo, envolvendo a participação de todos os alunos da disciplina. Os estudos dirigidos foram aplicados para discussões temáticas em oito conteúdos programáticos: requisitos de *software*, análise e projeto de *software*, interface com o usuário, gerência de *software*, gerência de configuração, teste de *software*, qualidade de *software* e métodos ágeis.

Para cada artigo analisado aula com estudo dirigido, cada grupo de alunos tinha a tarefa de apresentar e discutir com a turma o contexto no qual a pesquisa científica analisada estava inserida, o objetivo da pesquisa, o método de pesquisa aplicado e os resultados obtidos na pesquisa. Além disto, os alunos compararam as conclusões apresentadas pelos autores da pesquisa analisada com situações particulares e semelhantes vivenciadas por eles no seu ambiente de trabalho. Estas discussões foram guiadas por uma pergunta geral, porém, abrindo espaço para debates relevantes sobre o mesmo tema. A tabela 1 apresenta a correlação de pesquisas científicas analisadas e o tema da discussão da aula ministrada.

Tabela 1. Aplicação dos estudos dirigidos.

Conteúdo/Tema	Discussão da aula	Estudos aplicados
Requisitos de <i>software</i>	Quais as particularidades do processo de levantamento de requisitos de sistemas <i>software</i> educativos?	[EST001] [EST002] [EST003]
Análise e projeto de <i>software</i>	Quais as vantagens da aplicação do padrão MVC no contexto o qual a pesquisa está inserida?	[EST004] [EST005] [EST006]
Interface com o usuário	Que tipos de cuidados os pesquisadores apontaram para o tipo de interface desenvolvida na pesquisa? Quais desses cuidados se aplicam a sistemas comerciais?	[EST007] [EST008] [EST009]
Gerência de <i>software</i>	Como as abordagens de gerência de projetos de <i>software</i> discutidas na pesquisa poderiam ser aplicadas no seu ambiente profissional?	[EST010] [EST011] [EST012]
Gerência de configuração	Importância da gerência de configuração para o desenvolvimento de <i>software</i> .	[EST013] [EST014] [EST015]
Teste de <i>software</i>	Novos conhecimentos sobre teste de <i>software</i> adquiridos através da experiência relatada no estudo.	[EST016] [EST017] [EST018]
Qualidade de <i>software</i>	Como a gerência de configuração está	[EST013] [EST014]

² Dados omitidos via *blind review*.

	relacionada à qualidade do produto de <i>software</i> ?	[EST015]
Métodos ágeis	Percepções do SCRUM entre a teoria, a prática abordada no estudo analisado e a prática profissional do aluno.	[EST019] [EST020] [EST021]

Os resultados destas atividades foram encorajadores, pois permitiram um alto nível de interação entre os alunos. O estudo dirigido induz o aluno à criatividade, uma vez que a análise das pesquisas científicas leva ao pensamento reflexivo, de acordo com as circunstâncias do ambiente no qual a pesquisa foi desenvolvida e a vivência profissional do aluno. O nível de aprovação dos alunos na disciplina foi alto, não havendo nenhuma reprovação e com apenas um aluno atingindo a nota mínima necessária para aprovação, os demais com notas acima da média. Este fato pode dever-se também à avaliação constante dos alunos em sala de aula permitida pelas discussões geradas pelos estudos dirigidos. Além disso, os alunos mostram real interesse nas atividades quando percebem que estão discutindo questões práticas diretamente relacionadas com o seu ambiente profissional, porém em outro contexto ou outra realidade.

Entretanto este estudo é um relato de experiência, de modo que estes resultados não podem ser generalizados para todos os contextos. Futuramente a aplicação de um método empírico em uma nova pesquisa poderá permitir a generalização dos resultados para contextos diversos.

4. Conclusões e Trabalhos Futuros

A engenharia de *software* é uma disciplina de extrema importância no currículo dos cursos de computação, geralmente tratada no ensino formal no nível de graduação e/ou pós-graduação ou por meio de treinamentos profissionais de curta duração e, neste contexto, diversas teorias discutem a necessidade da união entre o conhecimento teórico e prático no desenvolvimento da disciplina. A proposta desta pesquisa foi relatar e discutir uma experiência no ensino de engenharia de *software* guiado por estudos dirigidos como abordagem para discussão sobre situações práticas, na indústria de *software*, em um contexto no qual os alunos participantes da disciplina já possuem vivência profissional dos conteúdos teóricos apresentados.

A principal contribuição deste estudo é a apresentação de uma abordagem de ensino no qual o conhecimento prático sobre os temas da engenharia de *software* é construído através da comparação e relação da vivência profissional dos estudantes com experiências em outros contextos publicados anteriormente em congressos e revistas científicas. Esta abordagem se tornou eficiente pelo fato de que a turma pode construir o conhecimento através de experiências diferentes das quais estão usualmente acostumados. Contudo, é importante salientar que os debates e discussões sobre os temas não foi a única abordagem usada para avaliar o conhecimento da turma, sendo aplicados também seminários sobre processos de *software* e métodos de desenvolvimento, dentre outros. Além disso, foram aplicadas avaliações escritas discursivas sobre casos comuns da indústria, como levantamento de requisitos de um sistema específico, criação de casos de uso, diagramas de classe, casos de teste e *design* de interfaces, por exemplo.

Como trabalho futuro, pretende-se aplicar o método de ensino guiado por estudo dirigido em novos semestres da disciplina e aplicar um método empírico para avaliação da abordagem, através de um estudo de caso formal utilizando instrumentos como notas de observações, questionários e entrevistas com os alunos participantes do processo. A ideia é permitir que esta proposta possa ser aplicada em contextos diferentes do vivenciado nesta pesquisa, como em cursos de Computação de universidades públicas e outros cursos diferentes da Computação que também demandam de atividades práticas no desenvolvimento das disciplinas.

Referências

- GIBBS, W. *Software's chronic crisis*. **Scientific American**. 271 3 (1994), pp. 86–95.
- IEEE COMPUTER SOCIETY. **SWEBOK**: guide to the *software* engineering body of knowledge, 2004.
- KITCHENHAM, B.; DYBÅ, T.; JØRGENSEN, M. **Evidence-based Software Engineering**. 26th International Conference on *Software Engineering*, (ICSE '04), Proceedings. IEEE, Washington DC, USA, p. 273 – 281, 2004.
- LIBÂNEO, J. C. **Didática**. 2ª ed. São Paulo: Cortez; 1994.
- NÉRICI, I. G. **Didática geral dinâmica**. 11ª ed. São Paulo: Atlas; 1992.
- SCHNEIDER, J., JOHNSTON L. **Extreme Programming at Universities – An Educational Perspective**. Proceedings of the 25th International Conference on *Software Engineering*, Portland, Oregon, May 03-10, 2003, p. 594-599.
- SILVA, L. F.; LEITE, J. C. S. P.; BREITMAN, K. K. **Ensino de Engenharia de software: Relato de Experiências**. Anais do XII Workshop de Educação em Informática (WEI). Salvador, 2004.
- SOARES, M. S. **Uma Experiência de Ensino de Engenharia de software Orientada a Trabalhos Práticos**. Anais do XII Workshop de Educação em Computação. Salvador, 2004.
- TERRA EDUCAÇÃO. **Trabalho leva universitários a procurar cursos noturnos na rede privada**. Disponível em: < <http://noticias.terra.com.br/educacao/trabalho-leva-universitarios-a-procurar-cursos-noturnos-na-rede-privada,f758febb0345b310VgnCLD200000bbcecb0aRCRD.html>>. Acesso em: julho de 2014.

Apêndice I – Pesquisas científicas analisadas

- [EST001] GOMES, A. S.; WANDERLEY, E. G. **Elicitando requisitos em projetos de Software Educativo**. Anais do IX Workshop de Informática na Escola (1), 119-130, 2003.
- [EST002] BENITTI, F. B. V.; SEARA, E. F. R.; SCHLINDWEIN, L. M. Processo de Desenvolvimento de *Software* Educacional: proposta e experimentação. **Revista RENOTE**. V. 3 Nº 1, Maio, 2005.
- [EST003] OLIVEIRA, C. C.; OLIVEIRA, D. C.; OLIVEIRA, C. F.; CATTELAN, R. G.; SOUZA, J. N. **Árvore de Características de Software Educativo: Uma Proposta para Elicitação de Requisitos pelo Usuário**. Anais do XXI Simpósio Brasileiro de Informática na Educação. João Pessoa, 2010.

- [EST004] BARROS, T.; SILVA, M.; ESPÍNOLA, E. **State MVC: Estendendo o padrão MVC para uso no desenvolvimento de aplicações para dispositivos móveis.** Disponível em: <http://www.cesar.org.br/site/files/file/SMVC_01_0104.pdf>. Acesso em: abril de 2014.
- [EST005] CLARO, F. M. D.; MARCHI K. R. C. **Desenvolvendo Aplicações Utilizando Javasever Faces e MVC.** Disponível em: <<http://ftp.unipar.br/~seinpar/2013/artigos/Felipe%20Moreira%20Decol%20Claro.pdf>>. Acesso em: abril de 2014.
- [EST006] Balthazar, G. R.; Guimarães, F. M. G.; de Paula, M. M. V.; LOVISI FILHO, E. **Uma Abordagem Prática sobre a Aplicação do padrão MVC com o Framework Struts.** Disponível em: <<http://re.granbery.edu.br/artigos/MjQy.pdf>>. Acesso em: abril de 2014.
- [EST007] NETTO, A. V.; MUNOZ, M. E.; SANTOS, R. G. **Desenvolvimento de interface gráfica 3D para sistema de tomada de decisão.** Proceedings of X Symposium on Virtual and Augmented Reality. João Pessoa, 2008.
- [EST008] SENA, C. P. P.; SANTOS, C. A. S. **Desenvolvimento de interfaces multimodais a partir da integração de comandos vocais à interface gráfica.** Anais do VII Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais. Natal, 2006.
- [EST009] Cunha, K. K. C.; Guimarães, M. P. **Desenvolvimento de uma Interface Gráfica para o Artoolkit com aplicação na área educacional.** Anais do III Workshop de Realidade Virtual e Aumentada. Itumbiara, 2007.
- [EST010] BARROS, M. O.; WERNER, C. M. L.; TRAVASSOS, G. H. **Um Estudo Experimental sobre a Utilização de Modelagem e Simulação no Apoio à Gerência de Projetos de Software.** Anais do XVI Simpósio Brasileiro de Engenharia de *software*. Gramado, 2002.
- [EST011] SILVA, D. V. S.; SANTOS, F. A. O.; SANTOS NETO, P. **Os benefícios do uso de Kanban na gerência de projetos de manutenção de software.** Anais do VIII Simpósio Brasileiro de Sistemas de Informação. São Paulo, 2012.
- [EST012] ROSITO, M. C.; CALLEGARI, D. A.; BASTOS, R. M. **Gerência de Projetos e Processos de Desenvolvimento de Software: uma proposta de integração.** Anais do IV Simpósio Brasileiro de Sistemas de Informação. Rio de Janeiro, 2008.
- [EST013] WERNER, C.; MURTA, L.; CORREA, C.; SANTOS, R.; PRUDENCIO, J. G.; FERNANDES, P.; SCHOTS, M.; CEPEDA, R.; LYRA, F. **Um Processo de Implantação de Gerencia de Configuração na Indústria.** Anais do VII Workshop de Manutenção de *Software* Moderna. Belém, 2010.
- [EST014] NUNES, V. B.; FALBO, R. A. **Uma Ferramenta de Gerência de Configuração Integrada a um Ambiente de Desenvolvimento de Software.** Anais do V Simpósio Brasileiro de Qualidade de *Software*. Vila Velha, 2006.
- [EST015] FIGUEIREDO, S.; SANTOS, G.; ROCHA, A. R. **Gerência de Configuração em Ambientes de Desenvolvimento de Software Orientados à Organização.** Anais do III Simpósio Brasileiro de Qualidade de *Software*. Brasília, 2004.

- [EST016] ISABEL, S. S.; TRAVASSOS, G. H. **Características de Técnicas de Teste de Software para Uso em Projetos WEB**. Anais do XIV Congresso Ibero-Americano em Engenharia de *software*. Rio de Janeiro, 2011.
- [EST017] SOUZA, E. P. R.; GUSMÃO, C. M. G. **RBTProcess - Modelo de Processo de Teste de Software baseado em Riscos**. Anais do VII Workshop de Teses e Dissertações em Engenharia de *software*. Campinas, 2008.
- [EST018] CRUZ, J.; L. JINO, M.; CRESPO, A. N. ARGOLLO, M. **Suporte automatizado à rastreabilidade em um processo de teste de software baseado em documentação**. Anais do V Simpósio Brasileiro de Qualidade de *Software*. Vila Velha, 2006.
- [EST019] BORGES JÚNIOR, A. N.; KAMEI, F. K.; ALVES, P. R. C.; ROCHA, R. G. C.; AZEVEDO, R. R.; MEIRA, S. L. **A Utilização de Práticas Scrum no Desenvolvimento de Software com Equipes Grandes e Distribuídas: um Relato de Experiência**. Anais do V Workshop de Desenvolvimento Distribuído de *Software*. São Paulo, 2011.
- [EST020] CARNEIRO, L. S.; QUEIROZ, M. A.; BARROS, R. M.; BRANCHER, J. D. **Implementação da ISO 9001 com Scrum: Um Estudo de Caso**. Anais do VIII Simpósio Brasileiro de Sistemas de Informação. São Paulo, 2012.
- [EST021] Mariz, L M. R. M.; França, A. C. C.; da Silva, F. Q. B. **Um Estudo Experimental sobre a Relação entre o Uso de Práticas Ágeis e o Sucesso de Projetos de Software que utilizam Scrum**. Anais do XXIV Simpósio Brasileiro de Engenharia de *software*. Salvador, 2010.