

ESTADO DA ARTE SOBRE O USO DO SCRUM EM AMBIENTES DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE

STATE OF THE ART ON THE USE OF SCRUM IN DISTRIBUTED DEVELOPMENT SOFTWARE

Eneida Oliveira¹, Rosangela Lima²

¹ EMBASA - Empresa Baiana de Águas e Saneamento S.A.

² CPM Braxis Outsourcing S.A.

rosangela.santos@embasa.ba.gov.br, enaidas@gmail.com

Resumo

Métodos ágeis e desenvolvimento distribuído de software são duas realidades presentes no mundo altamente competitivo e globalizado vivenciado atualmente nas empresas. Existem vários trabalhos que ressaltam os resultados positivos alcançados com o uso de Scrum em projetos de desenvolvimento de software com equipes presenciais. No entanto, gerenciar um projeto de software com uma equipe que se encontra em diversos pontos geográficos implica realizar certas adaptações à implantação do Scrum neste tipo de projetos. Este artigo apresenta o estado da arte do uso de Scrum em equipes distribuídas geograficamente, ressaltando os problemas, desafios e alternativas de solução no contexto de gerenciamento ágil de projetos em desenvolvimento distribuído de software.

Palavras-chave: scrum distribuído; software distribuído; gerenciamento de projeto de software.

Abstract

Agile and software distributed development are two realities present in the highly competitive and globalized world experienced in business today. There are several studies that highlight the positive results achieved with the use of Scrum projects in software development teams face. However, managing a software project with a team that is in various geographic locations involves performing certain adaptations to the implementation of Scrum in such projects. This paper presents the state of the art of using Scrum in geographically distributed teams, highlighting the problems, challenges and possible solutions in the context of project agile management in distributed software development.

Keywords: distributed scrum; distributed software; software project management.

1 INTRODUÇÃO

As empresas necessitam cada vez mais desenvolver projetos de software que sejam flexíveis a fim de obter uma maior agilidade na entrega de produtos que atendam a seus clientes. Assim, algumas empresas adotam a filosofia ágil no gerenciamento de projetos, para aumentar a capacidade de oferecer novos produtos ou serviços, em períodos curtos de tempo, com participação constante do cliente.

Um exemplo dessa filosofia ágil é o *Scrum*. O *Scrum* é um *framework* de gerenciamento de projetos que promove um desenvolvimento iterativo e incremental, enfatizando a comunicação presencial e frequente de todos os envolvidos no projeto. Porém, as empresas têm expandido seus processos de desenvolvimento em lugares distintos devido, principalmente, à globalização, com isso, suas equipes acabam trabalhando de forma distribuída (1), não sendo possível realizar reuniões presenciais frequentes. Assim, surge o desafio de como aplicar o *framework Scrum* nestas equipes distribuídas para obter uma redução nos custos de locomoção e uma flexibilidade de alocação e desalocação das equipes de trabalho, entre outros.

A motivação desse trabalho surgiu da necessidade em saber a atual realidade dos estudos e pesquisas sobre a adaptação do *Scrum* a projetos que possuem equipes geograficamente distribuídas, a fim de identificar os principais problemas que estão sendo enfrentados, assim como, as possíveis soluções adotadas. Existe uma carência de materiais sobre o assunto que auxiliem as instituições que pensam em adotar o desenvolvimento distribuído juntamente com as técnicas ágeis. Assim, este será mais um trabalho que contribuirá para a exploração e entendimento sobre este tema.

Este trabalho tem como objetivo apresentar o estado da arte do uso de *Scrum* em equipes distribuídas geograficamente, ressaltando os problemas, desafios e alternativas de solução no contexto de gerenciamento ágil de projetos em desenvolvimento distribuído de software.

O artigo está organizado da seguinte forma: Na seção 2 são apresentados os fundamentos e conceitos sobre o *framework Scrum*. Na seção 3 é feita uma análise entre o Scrum Presencial e o Scrum Distribuído. Na seção 4 é realizada uma revisão da

literatura dos últimos 5 anos sobre *Scrum* Distribuído, com a finalidade de identificar problemas e desafios enfrentados, assim como as soluções propostas nos trabalhos publicados em diversas conferências sobre o assunto. Na seção 5 são descritas as considerações finais e sugestões para trabalhos futuros.

2 SCRUM

O *Scrum* é um framework de gerenciamento ágil de projetos, cujas características principais são: (i) entregas flexíveis conforme as necessidades do mercado ou do cliente; (ii) flexibilidade dos prazos para antes ou após do planejado inicialmente; (iii) pequenos times, de preferência, com seis membros; (iv) revisões frequentes do progresso do time; (v) colaboração entre os membros; (vi) orientação com objetos de comportamento bem definidos (2).

Segundo Schwaber (3), três pilares sustentam esse *framework*:

- a) Transparência: garante que todos os aspectos relevantes ao sucesso do processo se mantenham visíveis e conhecidos de modo a garantir que o resultado obtido seja o equivalente à definição de pronto utilizada;
- b) Inspeção: é feita com finalidade de se detectar qualquer não conformidade que possa vir a prejudicar os resultados da equipe;
- c) Adaptação: a partir da identificação da irregularidade são feitas adaptações no processo ou no material em processo, reduzindo a probabilidade de um resultado insatisfatório.

O *Scrum* propõe a adoção de alguns papéis e responsabilidades a fim de obter o sucesso do projeto. São eles:

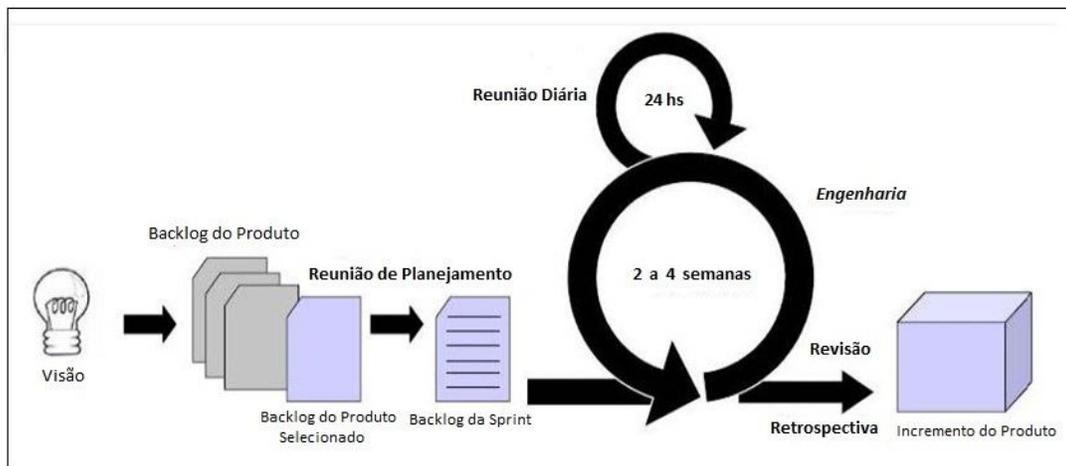
- *Product Owner* (PO): É o representante do cliente na equipe e tem como responsabilidades: acompanhar o ROI (*Return Of Investment*), conhecer as necessidades do cliente/usuário, priorizar o *Product Backlog*, aceitar ou rejeitar os resultados do time.
- *ScrumMaster* (SM): É quem garante que o *Scrum* está sendo seguido em todos os seus fundamentos e tem como responsabilidades: remover os impedimentos,

melhorar a produtividade do Time, promover a comunicação e proteger o time de interferências externas.

- Time: É a equipe de desenvolvimento que tem como responsabilidades: realizar todos os itens do *Sprint Backlog* com qualidade, auto-gerenciamento, aplicar boas práticas de engenharia, apresentar o resultado na reunião de revisão da Sprint.

A figura 1 exemplifica a visão geral do processo do *Scrum*.

Figura 1 - Visão geral do processo do Scrum (Adaptado de (4)).



De acordo com a Figura 1, o processo do *Scrum* inicia-se pelo *Backlog* do Produto, que é uma lista, criada pelo PO, onde contém todas as funcionalidades definidas com o cliente/usuário, melhorias e correções de defeitos a serem desenvolvidas no projeto. A partir dele elabora-se cada *Sprint* a ser executada com duração de uma a quatro semanas. No início da *Sprint*, o PO, o SM e o Time participam da reunião de planejamento, que é composta por duas partes. Na primeira parte participam o PO, o SM e o Time com a finalidade de selecionar as tarefas (*Backlog* do Produto Selecionado) que farão parte do *Backlog* da *Sprint* conforme priorização do PO. Na segunda parte, participam o SM e o Time para subdividir as tarefas com o intuito de melhor entendê-las para facilitar a implementação de cada uma delas.

Durante o *time-boxing* da *Sprint*, o SM e o Time realizam reuniões diárias sempre no mesmo horário e local para acompanharem o andamento do projeto.

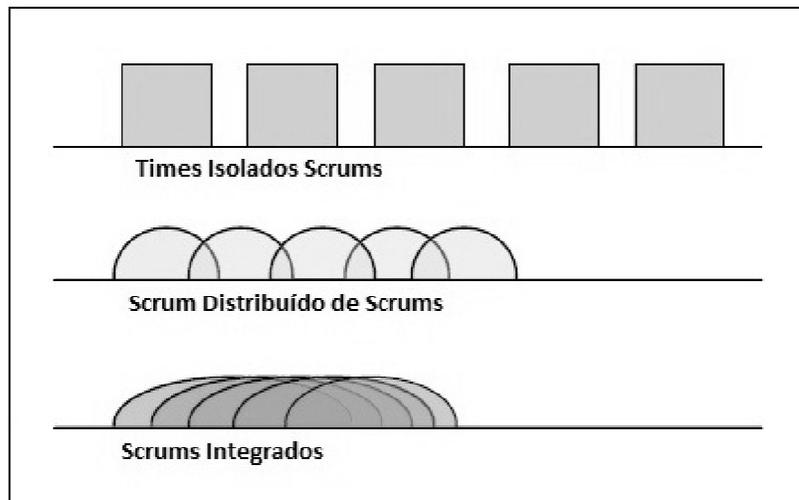
Ao término de execução da *Sprint*, o Time realiza a apresentação do incremento do produto desenvolvido durante a *Sprint* para o PO (Revisão). Por sua vez, o PO valida se todas as tarefas atendem à definição de pronto pré-estabelecida. Logo após, o SM conduz o Time na reunião de Retrospectiva para avaliação das lições aprendidas. Cada membro do Time avalia o que pode ser melhorado, o que deve ser mantido e o que não deve ser mantido para a próxima *Sprint*. Assim, a *Sprint* é finalizada iniciando-se um novo ciclo até concluir o produto final.

3 SCRUM PRESENCIAL VERSUS SCRUM DISTRIBUÍDO

O *Scrum* presencial enfoca um conjunto de pessoas no mesmo local físico facilitando a comunicação diária e, com isso, a interação do time. Porém, utilizar *Scrum* em ambientes de equipes geograficamente distintas implica em adaptações como a utilização de ferramentas de comunicação e conferência que facilitem as reuniões do *Scrum*.

A figura abaixo exemplifica os tipos de equipes distribuídas com *Scrum*.

Figura 2 – Tipos de equipes distribuídas com Scrum (Adaptado de (5)).



A Figura 2 apresenta três tipos de equipes distribuídas com *Scrum*, detalhadas a seguir:

- Times Isolados Scrum:*** Neste tipo, as equipes são independentes sendo que, ocasionalmente, alguns integrantes podem trabalhar de forma distribuída e cada equipe possui um PO;
- Scrum Distribuído de Scrum:*** Neste tipo, cada local possui uma equipe *Scrum* multidisciplinar com reuniões de *Scrum of Scrum*s periódicas para dispor esforços entre as equipes. O PO coordena o trabalho de múltiplas equipes;
- Scrum Integrados:*** Neste tipo, cada equipe possui integrantes em diversos locais, assim, as reuniões diárias ajudam a romper as barreiras culturais e as disparidades de estilos de trabalho.

Existem várias razões para a crescente popularização da distribuição do desenvolvimento de software: redução dos custos de produção, ganhos de escala, acesso aos recursos especializados, redução do tempo de colocação no mercado, melhoria na qualidade e acesso a novos mercados (6). Outra grande vantagem é que times distribuídos podem aumentar o aprendizado e a criatividade da equipe, já que os membros têm a chance de interagir com uma grande variedade de culturas, experiências e pontos de vista diversificados. No entanto, lidar com equipes

constituídas por pessoas de diferentes localidades, culturas e expectativas para trabalhar em projetos distribuídos pode, muitas vezes, gerar dificuldades (7).

Quando o *Scrum* é utilizado para o gerenciamento dos projetos com tais características, então, alguns problemas tornam-se ainda mais desafiadores, principalmente no tocante ao aspecto comunicacional das equipes envolvidas. Herbsleb em (6) considera que, para um apoio efetivo às atividades distribuídas e descentralizadas, deve existir uma comunicação frequente para colaboração e compartilhamento de informações.

A seguir serão discutidos vários trabalhos que apresentam características, tipos e resultados adquiridos com a utilização do *Scrum* Distribuído.

4 EXPERIÊNCIAS DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE USANDO SCRUM

Foi realizado um levantamento bibliográfico dos últimos cinco anos com a finalidade de mostrar o estado da arte em *Scrum* distribuído. Dos 25 trabalhos pesquisados, 6 foram descartados por não atenderem aos critérios de pesquisa (aliar *Scrum* com desenvolvimento distribuído). Dos 19 trabalhos considerados, 3 foram publicados no ano de 2007, 3 em 2008, 8 em 2009, 4 em 2010 e 1 em 2011.

A tabela abaixo relaciona os trabalhos selecionados por ano que foram utilizados para o desenvolvimento deste estado da arte.

Tabela 1 - Relação dos trabalhos selecionados por ano.

Ano	Trabalhos Publicados	Quantidade
2007	(5) ,(14), (20)	3
2008	(7), (13), (16)	3
2009	(9), (10), (11), (15), (17), (18), (19), (23)	8
2010	(1), (6) ,(8), (12)	4
2011	(22)	1
Total		19

Alguns trabalhos pesquisados tratam de revisões sistemáticas sobre o assunto ((1),(18)), vários citam estudos de casos envolvendo *Scrum* e desenvolvimento distribuído ((8),(14),(16),(17),(23)), alguns descrevem o uso de uma ferramenta que auxilia o gerenciamento ou a comunicação entre equipes distribuídas ((7),(9), (13) e (20)) e outros ainda, descrevem uma espécie de manual, com recomendações sobre o que fazer para obter sucesso em um projeto que não dispõe de todos os seus integrantes num mesmo local ((11), (15), (19) e (22)). O principal objetivo desse levantamento bibliográfico é identificar os problemas e desafios mais comumente citados em trabalhos e estudos de caso que envolvem o uso do *Scrum* juntamente com o desenvolvimento distribuído de software e quais as soluções adotadas para vencer esses desafios.

Os principais problemas são mostrados na tabela 2, mas existiram outros que foram levantados em menor escala, por alguns autores e trabalhos pesquisados, que não foram considerados neste trabalho por não ser um problema comum a mais de um estudo de caso. Os problemas identificados são: largura de banda insuficiente para videoconferências (infraestrutura) e times muito grandes (18), falta de visibilidade do status do projeto, demora nos *feedbacks* (11), perda de contexto técnico e de negócios e confiança reduzida dos times (21).

A tabela abaixo explicita os problemas enfrentados e as soluções propostas para adaptação do *Scrum* no desenvolvimento distribuído software.

Tabela 2 - Estado da Arte do Desenvolvimento Distribuído: problemas e soluções.

SCRUM EM DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE		
PROBLEMAS	SOLUÇÕES PROPOSTAS	FERRAMENTAS
Diferenças Culturais e Linguísticas	<ul style="list-style-type: none"> Linguagem simples e clara para evitar desentendimentos (11), (12), (15) Desenvolver relacionamentos mais pessoais (16) 	
Dificuldade de Gerenciamento e Planejamento	<ul style="list-style-type: none"> Coalocação de membros remotos (13), (15), (17) Definir um local base para equipes (15) Uso efetivo de softwares de gestão (1),(6), (8), (12) 	<ul style="list-style-type: none"> FireScrum (1)(6)(8) Pivotal Tracker (17) WEBAPSEE (7) DiSen (13), (20) XPlanner2002 (14)
Diferença de fuso horário	<ul style="list-style-type: none"> Comunicação escrita (7), (14), (16) 	<ul style="list-style-type: none"> Email (7) Fóruns e Listas de Discussão (14), (16)
Dificuldade de comunicação	<ul style="list-style-type: none"> Uso de ferramentas que centralizem a discussão de problemas (15), (17) Videoconferência (8), (13), (14), (16) Mensagens instantâneas (11), (12), (14) Documentação escrita (14) Linguagem simples e clara para evitar desentendimentos (11), (12), (15) 	<ul style="list-style-type: none"> Skype (8), (14), (16) Email (14) Wiki (16),(17) Fóruns e Listas de Discussão (14), (16) Tracking systems (15) Smartboard (16)
Dificuldade de Colaboração	<ul style="list-style-type: none"> Promover a cultura do compartilhamento (11), (13) Usar um repositório único de produtos e processos (11), (13), (16) 	<ul style="list-style-type: none"> DiSen (13), (20)

Para efeito de estudo, esta pesquisa focará nos problemas mais comuns que foram apresentados pela grande maioria dos trabalhos coletados juntamente com as soluções propostas: diferenças culturais e linguísticas, dificuldade de gerenciamento e planejamento, diferença de fuso horário, dificuldade de comunicação e dificuldade de colaboração.

- Diferenças Culturais e Linguísticas: Culturas diferentes envolvem línguas, tradições, crenças, normas e princípios diferentes. O líder de equipes distribuídas deve possuir um profundo conhecimento das diferenças culturais do time e saber lidar com elas. Uma das técnicas mais utilizadas nos trabalhos pesquisados ((11), (12) e (15)) foi o uso de uma linguagem simples e clara, sem regionalismos, gírias locais ou expressões que podem deixar o discurso confuso. Outro método que se mostrou eficaz no trabalho (16) foi o desenvolvimento de relações mais pessoais entre os membros das equipes, o que ajudou a derrubar as barreiras naturais na comunicação.
- Dificuldade de Gerenciamento e Planejamento: Uma dificuldade também muito comum apresentada nos trabalhos pesquisados se refere à dificuldade de gerenciar e planejar o andamento dos projetos que envolvem equipes geograficamente distribuídas. Para solucioná-las, alguns autores utilizaram a técnica da colocação de membros temporariamente ((13), (15) e (17)) para facilitar a comunicação e para sincronizar as atividades do projeto. Outra abordagem usada como estratégia de solução foi escolher um local físico onde serão definidos os documentos, os processos a serem adotados e onde será dado o pontapé inicial do projeto (15). Finalmente, a solução mais presente nos trabalhos relacionados ((1), (6), (8), (12)) foi o uso de melhores práticas de gerenciamento de projetos associado a um software de gestão de projetos. Os mais utilizados foram o *FireScrum* ((1), (6), (8)) e o *DiSen* ((13), (20)), seguidos por *WEBAPSEE* (7), *XPlanner2002*(14) e *Pivotal Tracker* (17).
- Diferença de Fuso Horário: Em equipes completamente distribuídas, ou seja, que não possuem horários de trabalho em comum em nenhum momento do dia, o fuso horário passa a ser um dos principais empecilhos para o desenvolvimento de um projeto com *Scrum*. Para vencer esse desafio, a solução encontrada foi a comunicação escrita, por meio de ferramentas como email (7), fóruns e listas de discussão ((14), (16)).

- Dificuldade de Comunicação: *Scrum* é um framework que se baseia na comunicação face a face e, em equipes distribuídas, esse é um desafio que precisa ser superado. 12 dos 18 trabalhos pesquisados citaram esse problema e as técnicas utilizadas para vencer essa dificuldade foram: uso de ferramentas centralizadoras de discussão, tais como fóruns e listas ((14) e (16)), *tracking systems* ((15)) e *wikis* ((16) e (17)), além de e-mails (14); uso de ferramentas de comunicação instantânea ((11), (12) e (14)) e de videoconferência, como o *Skype* ((8), (13), (14) e (16)). Uma solução simples, mas muito eficaz, foi o uso de linguagem clara para que a comunicação seja facilitada ((11), (12) e (15)).
- Dificuldade de Colaboração: A dificuldade quanto à colaboração entre os membros de um projeto *Scrum* está intimamente ligada às dificuldades de comunicação anteriormente citadas. Dentre as soluções mais citadas nos trabalhos pesquisados, as mais utilizadas foram a promoção da cultura do compartilhamento entre os membros ((11) e (13)) e a definição de um repositório único para produtos e processos do projeto ((11), (13) e (16)).

5 CONCLUSÃO

A literatura já tem demonstrado o quanto o *Scrum* pode ser um grande aliado na condução de projetos de desenvolvimento de software. Muitos são os casos de sucesso apresentados nessa área em diversas empresas de todo o mundo. No entanto, ao utilizar o *Scrum* em projetos distribuídos geograficamente, alguns desafios precisam ser enfrentados e solucionados a fim de que o projeto não termine em um grande fracasso. São eles: diferenças culturais e de linguagem, dificuldade de comunicação e de gerenciamento das equipes distantes, além de diferenças de fuso horário que impedem o encontro mesmo que virtual entre os integrantes dos times.

Dentre as soluções mais frequentemente adotadas pelos trabalhos pesquisados, estão o uso de ferramentas colaborativas tais como *wikis*, listas de

discussão, fóruns e repositório único para artefatos do projeto, em conjunto com ferramentas de comunicação instantânea, a exemplo *chats* e videoconferências.

Essas ferramentas aliadas a métodos de gerenciamento de equipes multiculturais demonstraram ser eficazes na obtenção do sucesso dos projetos estudados, mas nem por isso são as únicas alternativas de solução. Para trabalhos futuros é importante estudar novas formas de manter o canal de comunicação entre essas equipes como, por exemplo, a utilização de documentos compartilhados num servidor que demonstrem a evolução diária do produto; promover entre os times o conhecimento cultural de cada região; buscar ou desenvolver outras ferramentas que auxiliem o *Scrum* em equipes distribuídas. Outro fato bastante evidenciado hoje é o crescimento e a popularização das redes sociais, tais como o *Facebook*, o *Twitter* e o *LinkedIn* que, dentre outras, surgem como um possível meio de auxiliar a superar o obstáculo da comunicação em equipes geograficamente distribuídas.

REFERÊNCIAS

1. SANTOS, Aline et al. Estudo baseado em Evidências sobre Dificuldades, Fatores e Ferramentas no Gerenciamento da Comunicação em Projetos de Desenvolvimento Distribuído de Software. In: EXPERIMENTAL SOFTWARE ENGINEERING LATIN AMERICAN WORKSHOP, 7, 2010, Goiania, GO. **Anais...** Goiania, GO, 2010. p. 20 – 29.
2. ROSE, Tanara Priscilla R.; MELLO, Carlos Henrique P. Proposta de Sistemática para Gestão de Projetos Baseada na Metodologia Ágil *Scrum*. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 30., 2010, São Carlos, SP. **Anais...** São Carlos, 2010. SP p. 4.
3. SCHWABER, Ken. **GUIA DO SCRUM**. 2009.
4. ADAPWORKS. **Certified ScrumMaster Training**. The Zen of *Scrum*. 2011.
5. SUTHERLAND, Jeff et al. **Distributed Scrum: Agile Project Management with Outsourced Development Teams**. In: Proceedings of the 40th Hawaii International Conference on System Sciences, 40, 2007.

6. SANTOS, Aline et al. **Experiência Acadêmica de uma Fábrica de Software utilizando *Scrum* no Desenvolvimento de Software**. In: WORKSHOP BRASILEIRO DE MÉTODOS ÁGEIS (AGILE BRAZIL), 2010, PUC/RS, Porto Alegre, RS. p. 86-98.
7. LIMA, Adailton M.; REIS, Rodrigo Q. Compartilhamento de Informações sobre Processos em Ambientes Descentralizados de Desenvolvimento de Software. In: WORKSHOP DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE, 2., 2008, Campinas, SP. **Anais...** Campinas, SP, 2008. p. 71 – 80.
8. CHALEGRE, Virgínia C. et al. Estudo de Caso da Utilização de *Scrum* no Desenvolvimento Distribuído de Software. In: CONFERÊNCIA BRASILEIRA SOBRE MÉTODOS ÁGEIS DE DESENVOLVIMENTO DE SOFTWARE - AGILE BRAZIL 2010, 1., 2010, Porto Alegre, RS. **Anais...** Porto Alegre, RS, 2010. p. 126 – 136.
9. CAVALCANTI, Eric; MACIEL, Tereza M. de Medeiros; ALBUQUERQUE, Jones. Ferramenta OpenSource para Apoio ao Uso do *Scrum* por Equipes Distribuídas. In: WORKSHOP DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE, 3., 2009, Fortaleza, CE. **Anais...** Fortaleza, CE, 2009. p. 51 – 60.
10. SIAKAS, Kerstin V.; SIAKAS, Errikos. Agile Software Development in Distributed Environments. In: INDUSTRIAL STREAM OF THE 16TH EUROPEAN SOFTWARE PROCESS IMPROVEMENT (EUROSPI), 16, 2009, Alcalá, Madrid, Spain. p. 8.19 – 8.31.
11. DURRANI, Rabia; JAVED, Muhammad A. **Agile Software Development Practices in Distributed Environments**. IDT WORKSHOP ON INTERESTING RESULTS IN COMPUTER SCIENCE AND ENGINEERING, 2009, Mälardalen University, Sweden.
12. WOODWARD, Elizabeth; SURDEK, Steffan; GANIS, Matthew. **A Practical Guide to Distributed Scrum**. IBM, 2010. Disponível em: http://pt.scribd.com/doc/42021760/A-practical-guide-to-Distributed-Scrum#outer_page_52. Acesso em: 27 ago. 2011.
13. HUZITA, Elisa H. M. et al. Um Conjunto de Soluções para Apoiar o Desenvolvimento Distribuído de Software. In: WORKSHOP DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE, 2., 2008, Campinas, SP. **Anais...** Campinas, SP p. 101-110.
14. SOARES, Felipe S. F. et al. Adoção de *SCRUM* em uma Fábrica de Desenvolvimento Distribuído de Software. In: WORKSHOP DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE, 1, 2007, João Pessoa, PB. **Anais...** João Pessoa, PB, 2007.

15. BAVANI, Raja. **Critical Success Factors in Distributed Agile for Outsourced Product Development**. Computer Society of India, 2009. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.167.430&rep=rep1&type=pdf>
16. SUTHERLAND, Jeff et al. **Fully Distributed Scrum: The Secret Sauce for Hyperproductive Offshored Development Teams**. In: Agile 2008, Toronto, Canadá, 2008.
17. SUTHERLAND, Jeff et al. **Fully Distributed Scrum: Linear Scalability of Production between San Francisco and India**. In: Proceedings of 2009 Agile Conference, 2009, San Diego, CA, USA. pp. 339–344.
18. HOSSAIN, Emam; ALI BABAR, Muhammad; PAIK, Hye-young. **Using Scrum in global software development: A systematic literature review**. In: Fourth IEEE International Conference on Global Software Engineering, 2009. p. 175-184.
19. LEHTONE, Ismo. **Communication Challenges in Agile Global Software Development**. University of Helsinki, Helsinki, Finlândia, 2009.
20. HUZITA, Elisa et al. Um Ambiente de Desenvolvimento Distribuído de Software – Disen. In: WORKSHOP DE DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE, 1., 2007, João Pessoa, PB. **Anais...** João Pessoa, PB, 2007. p. 31-38.
21. MINOLI, Mariano; VALERIA DE CASTRO, J. G. **Reduciendo Distancia en proyectos de Desarrollo de Software Global Ágiles con técnicas de ingeniería e requisitos**.
22. TEVELL, Erik; ÅHSBERG, Mathias. **Positive and Negative Quality Effects in Distributed Scrum Projects-An Industrial Case Study**. 2011.
23. CIBOTTO, Rosefran et al. Uma Análise da Dimensão Sócio-Cultural no Desenvolvimento Distribuído de Software. In: WORKSHOP OLHAR SOCIOTÉCNICO SOBRE A ENGENHARIA DE SOFTWARE, 5., 2009, Ouro Preto, MG. **Anais...** Ouro Preto, MG, 2009. p. 96-16.