

METODOLOGIAS ÁGEIS PARA O DESENVOLVIMENTO DE SOFTWARE: APLICAÇÃO E O USO DA METODOLOGIA SCRUM EM CONTRASTE AO MODELO TRADICIONAL DE GERENCIAMENTO DE PROJETOS

AGILE METHODOLOGIES FOR SOFTWARE DEVELOPMENT: APPLICATION AND USAGE OF SCRUM METHODOLOGY IN CONTRAST TO THE TRADITIONAL MODEL OF PROJECT MANAGEMENT

Daisy Eliana dos Santos Silva¹, Ingredy Thaís de Souza², Talita Camargo³

RESUMO: Este artigo foi fomentado estudos bibliográficos sobre metodologias ágil e tradicional. Com objetivo de apresentar os principais problemas encontrados no século XXI, no que diz respeito ao desenvolvimento de projetos de *software* e as metodologias disponíveis. Serão apresentadas as formas de trabalho das metodologias tradicional e a ferramenta SCRUM, onde seus valores e práticas serão apresentados de modo a identificar e justificar suas principais diferenças. Demonstrar um quadro comparativo entre algumas metodologias de desenvolvimento ágil e tradicional, os quais serão comparados e analisados entre os processos das metodologias ágeis em relação às metodologias tradicionais. Com intuito de proporcionar condições favoráveis para avaliação e escolha da metodologia que atenda as necessidades dos clientes no desenvolvimento de *software*.

PALAVRAS-CHAVE: Metodologia Ágil, SCRUM, Metodologias Tradicionais, Desenvolvimento de *software*.

ABSTRACT: *This article was promoted for bibliographic studies about agile and traditional methodologies. In order to present the main problems encountered in the 21st century, with regard to the development of software projects and methodologies available. It will be presented the forms of traditional methodologies and the SCRUM, where their values and practices will be presented in order to identify and justify their main differences. Demonstrate a comparative table between some agile and traditional development methodologies, which are compared and analyzed between the processes of agile methodologies in relation to traditional methodologies. In order to provide favourable conditions for evaluation and choice of methodology that meets the needs of clients in software development.*

KEYWORDS: *Agile Methodology, SCRUM, Traditional Methodologies, Software Development.*

¹ daisyess@ig.com.br. Graduanda em Ciência da Computação pela Universidade Guarulhos.

² ingredy.souza@outlook.com. Graduanda em Ciência da Computação pela Universidade Guarulhos.

³ contatotalitacamargo@yahoo.com.br. Graduanda em Ciência da Computação pela Universidade Guarulhos.

INTRODUÇÃO

As empresas estão numa corrida constante para acompanhar as inovações e oferecer os melhores projetos que atendam as necessidades do mercado. Com o advento da internet e o avanço tecnológico no século XXI, tornou-se muito difícil manter-se sempre à frente dos concorrentes, uma vez que a tecnologia está aí para todos usufruir das informações que chega em tempo real.

Considerando as metodologias tradicionais e ágeis de gerenciamento de projetos disponíveis no mercado, cabe a cada empresa independente do seu porte escolher a que ira lhe retornar um resultado mais eficiente, atendendo suas necessidades.

Este artigo apresenta as metodologias tradicional e ágil de projetos e descreve os principais problemas encontrados no século XXI. Mostra um quadro comparativo para uma análise em termos quantitativos e qualitativos dos processos desenvolvidos entre a metodologia ágil SCRUM em relação à metodologia tradicional em cascata com conhecimentos do *Project Management Body of Knowledge* (PMBOK).

O QUE É PROJETO?

Segundo MAXIMIANO, Antônio (2009) projeto é uma palavra derivada do latim que significa “lançar adiante”, as ideias dos Romanos permanecem vivas e esta linguagem continua na prática na gestão de organização que se entende como temporário ou sequência de atividade que tenha começo, meio e fim.

Para os autores KERZNER, Harold (2006) e MARTINS, Washington Franco; WOILER, Samsao, (2011) ao projetar um sistema é necessário ter as fases operacionais e as informações necessárias de como será o sistema antes mesmo de iniciar o desenvolvimento de algum *software* incluindo até a interface visual do usuário e o armazenamento dos dados, obtendo as informações quantitativas e qualitativas e uma arquitetura básica de todo o sistema fica fácil de fazer a realização e definição dos custos e prazos de entregas.

O QUE É METODOLOGIA?

Metodologia está definida no dicionário Aurélio como uma palavra derivada do latim que significa o

estudo científico dos métodos, de maneira que faça um conjunto de regras e processos próprios para resolução de problemas.

DENNIS, Alan; WIXON, H. Barbara (2011) diz que uma metodologia de desenvolvimento tem a função de formalizar a ordem de desenvolver um *software*, organizar a fase do ciclo de vida do mesmo e ter o equilíbrio entre os processos e os dados. Há vários tipos de metodologia de desenvolvimento e cada empresa pode construir a sua própria conforme a suas necessidades, porém todas elas têm o mesmo objetivo de formalizar a fase de desenvolvimento e organização dos dados (informações).

DESENVOLVIMENTO DE PROJETOS DE SOFTWARE NO SÉCULO XXI.

O mercado está cada vez mais dinâmico e as relações empresariais e os negócios estão muito competitivos, quem possui maior acesso a informação, possui vantagens sobre os concorrentes. Para atender as necessidades da empresa e criar um sistema é fundamental ter conhecimento nas ferramentas e metodologias presentes para oferecer a melhor alternativa em desenvolvimento de projetos de *software*, pois muitos dos projetos corporativos de Sistema de Informação (SI), não chegam sequer a ser implantados e são abandonados antes da conclusão, segundo uma pesquisa realizada em 1996 pelo *Standish Group* o *chaos report* (relatório caos).

Dificuldades enfrentadas baseado em 8.380 projetos Standish Group (1995)	
Respeitaram os prazos e os custos e possuíam todas as funcionalidades especificadas.	16,2%
Dos projetos foram cancelados antes de estarem completos.	31%
Foram entregues, com prazos maiores, custos maiores ou com menos funcionalidades do que especificado.	42,7%

Quadro 1 - Dificuldades enfrentadas baseado em 8.380 projetos

Fonte: <http://www.univasf.edu.br/~mario.godoy/Aulas-Top-Av-Eng-Software/TAES-AULA%20%20-%20Metodologias%20Ageis.pdf>

As organizações enfrentam certas dificuldades diante das metodologias disponíveis para o gerenciamento de projetos de *software*. As empresas de grande porte e entidades governamentais têm como prática o detalhamento de vários processos por relatórios, planilhas e gráficos, sendo comumente utilizada a metodologia tradicional para projeto de *software*.

Por outro lado as empresas pequenas e médias encontram dificuldades em utilizar a metodologia tradicional, por ser muito caro a manutenção e controle, e o longo prazo para entrega.

A metodologia ágil com custo baixo e entregas rápidas pode contribuir no desenvolvimento de projeto de *software* para qualquer tipo de empresa e ser agregada à metodologia tradicional.

Vale apenas resaltar que o sistema de informação por mais detalhado que seja, poderá haver falhas diante da complexidade dos sistemas das empresas no cenário contemporâneo.

METODOLOGIA TRADICIONAL

O gerenciamento de projetos constitui-se uma tarefa de fundamental importância no processo de desenvolvimento de *software*. Em algumas organizações, um projeto só é aprovado após a realização de um estudo de exequibilidade, viabilidade financeira e técnica de um plano preliminar ou equivalente de análise. A definição do ciclo de vida do projeto permite identificar igualmente quais as ações de transição no final do projeto que serão ou não, incluídas com o objetivo de ligar o projeto às operações de rotina da organização. O ciclo de vida do desenvolvimento de um sistema passa por quatro fases, o planejamento, análise, o projeto e a implementação. A metodologia tradicional deste trabalho está baseada em cascata com os conhecimentos do PMBOK que resulta em muita documentação.

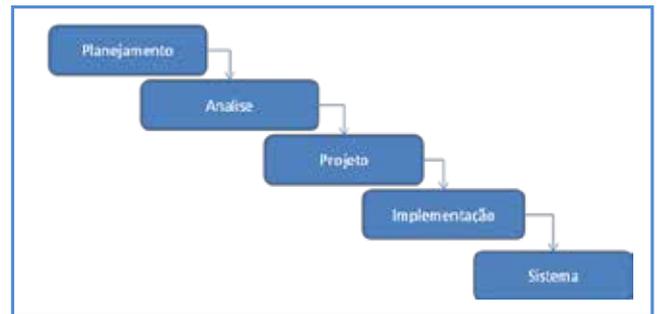


Figura 1- Metodologia de Desenvolvimento em Cascata

Fonte: Dennis, Alan e Wixom, Barbara Haley – Análise e Projeto de sistemas – 2º ed. Pág. 8

Há a disponibilidade de quatro disciplinas para o planejamento de processo usando o *CMMI (Capability Maturity Model Integration)*, ou seja, Integração do Modelo de Maturidade de Controle no desenvolvimento, manutenção de produtos e serviços.

Engenharia de Sistemas: os engenheiros de sistemas concentram-se em transformar necessidades, expectativas e restrições de clientes em produtos e em suportar esses produtos ao longo da respectiva vida;

Engenharia de Software: os engenheiros de *software* concentram-se na aplicação de metodologias sistemáticas, disciplinadas e quantificáveis ao desenvolvimento, operação e manutenção de *software*;

Desenvolvimento integrado de produtos e processos: abordagem sistemática que visa uma colaboração, em tempo oportuno, de *stakeholders* relevantes, ao longo da vida do produto, com objetivo de satisfazer necessidades, expectativas e requisitos dos clientes.

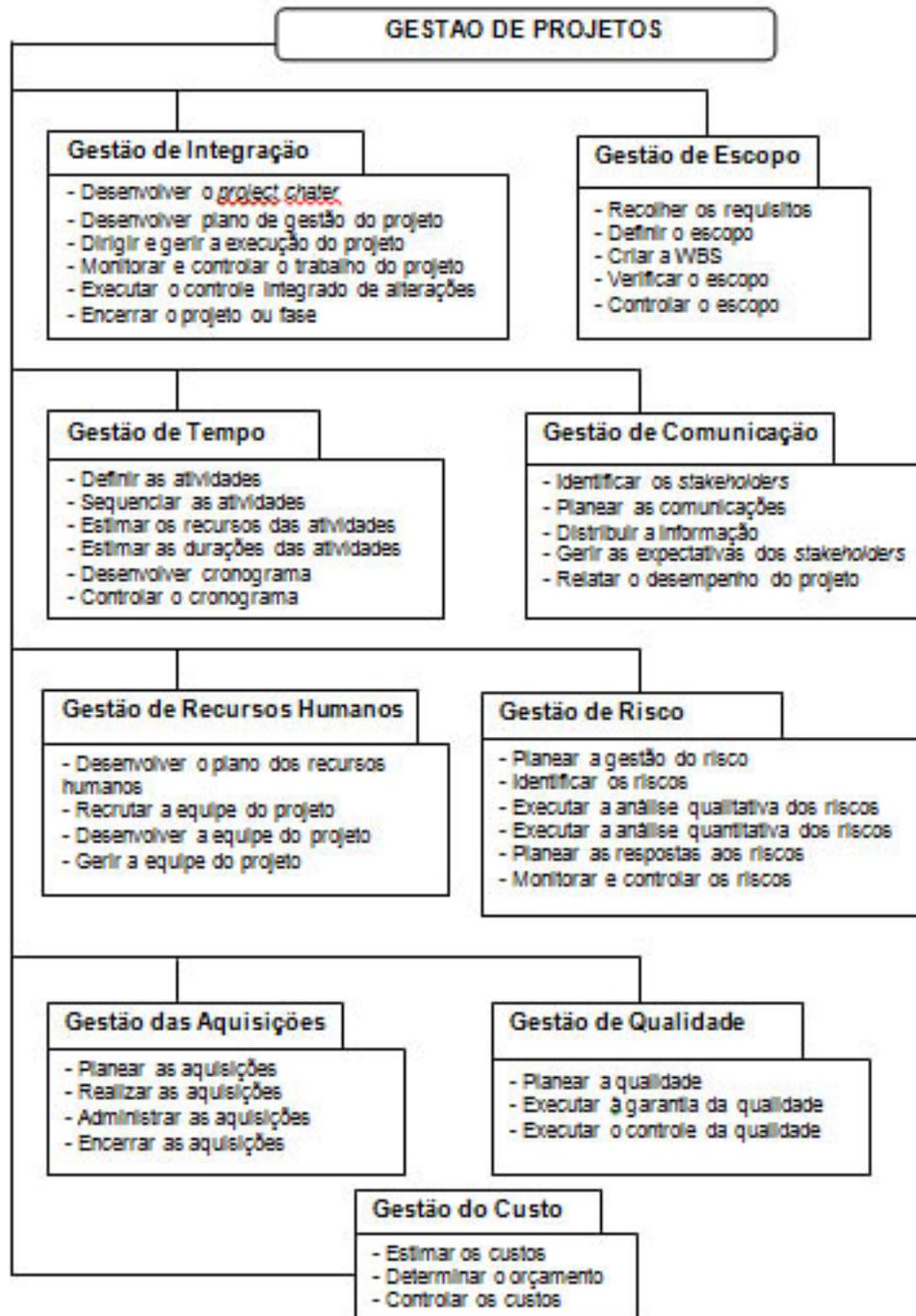
Uso de fornecedores: abordagem sistemática da análise, seleção e gestão dos fornecedores de um projeto ou organização. As melhores práticas da disciplina de uso de fornecedores têm de ser selecionadas em conjunto com as melhores práticas usadas para produzir produtos.

Para fazer uma análise e projeto de sistemas, precisa conhecer as principais habilidades em projetos que são divididos em fases, com objetivo de proporcionar um melhor controle de gestão e com as ligações adequadas às operações rotineiras.

Segundo o *Project Management Body of*

Knowledge (PMBOK) Guide (guia para o conjunto de conhecimento de gerenciamento de projetos), organizam os 42 processos da gestão de projetos

que integram os cinco grupos de processos em nove áreas de conhecimento.



Fonte: Miguel, Antonio - Gestão de Projetos de Software - 4ª ed. pág. 39
METODOLOGIAS ÁGIL

No início da década de 1990 existiam inúmeros projetos que resultavam em retrabalhos e que por consequência acabava estourando o prazo de entrega definido inicialmente no escopo do projeto. Os retrabalhos eram consequências de projetos que se utilizavam sua maior parte do tempo para documentar ao invés de desenvolver e testar.

Existiram ainda tentativas de melhoria nas metodologias de desenvolvimento utilizadas na época, bem como os métodos de programação estruturados ano de 1980 e as metodologias orientadas a objetos, porém os problemas se mantiveram.

Em meio a este cenário de projetos conturbados surgiu um grupo de desenvolvedores de *software* que tinham por objetivo reestruturar as metodologias utilizadas em projetos e focar a equipe de desenvolvimento apenas na programação e testes do projeto.

No ano de 2001 um grupo de dezessete especialistas em métodos de desenvolvimento ágil se reuniu com o intuito de padronizar seus processos, unificaram princípios comuns dentre os métodos ágeis de desenvolvimento formando assim a aliança Ágil.

O manifesto ágil denominou os métodos de desenvolvimento em metodologias ágil de desenvolvimento e definiu quatro princípios a serem seguidos por essa metodologia.

Valores		
Interação entre indivíduos	mais do que	Processos e Ferramentas
Produto funcionando	mais do que	Documentação Extensa
Colaboração com o cliente	mais do que	Termos Negociados (contrato)
Resposta às mudanças	mais do que	Cumprimento de planos
Ágil		Tradicional

Quadro 2 - Quadro de valores entre Metodologias Ágil e Tradicional

Fonte: <http://www.scrumj.org/tag/metodos-tradicionais>

As ferramentas, documentações, negociações de contratos e demais processos utilizados nas metodologias de desenvolvimento tradicionais não são excluídos totalmente das metodologias ágil, apenas são colocados em segundo plano quando comparados com os indivíduos e interações, tornando-

se característica principal das metodologias ágil fácil adaptação e orientado por pessoas.



Figura 3- Indivíduos e interações

Fonte: <http://www.devmedia.com.br/modelos-de-desenvolvimento-agil/27660>

METODOLOGIA ÁGIL SCRUM

Existem várias metodologias de desenvolvimento classificadas como ágil, dentre elas a metodologia SCRUM se destaca por ser um processo de desenvolvimento *software* incremental em ambientes complexos, onde os requisitos não são claros ou mudam com muita frequência.

O SCRUM é fundamentado na teoria de controle de processo e tem por objetivo aperfeiçoar a previsibilidade e controlar os riscos de um projeto.

Transparência, inspeção e adaptação são os três pilares que sustentam a metodologia SCRUM.

A transparência é a garantia que todos os processo que envolvam o resultado sejam claras para ambas as partes envolvidas no projeto.

A inspeção é realizada durante todo o projeto e tem por objetivo detectar qualquer variação e ajustar o processo evitando assim problemas futuros.

A necessidade de adaptação surge da inspeção e tem por finalidade adaptar o processo para qualquer variação detectada na inspeção.

A equipe do SCRUM é composta por três partes:



Figura 4 - Equipe Scrum

Fonte: <http://blog.myscrumhalf.com/wp-content/uploads/2012/03/papeisScrum2.png>

Product Owner é o proprietário do produto que representa a empresa que será aplicada o projeto.

ScrumMaster é o papel assumido pelo gerente do projeto.

A equipe de desenvolvimento que é composto por um grupo de até 7 pessoas e que são responsáveis pela análise, programação e testes do projeto.

Os requisitos do projeto são organizados em uma lista de tarefas organizada de acordo com a prioridade de cada item, sendo que os itens que tem maior importância devem estar no topo da lista, esta lista deve ser constantemente atualizada sempre priorizando os itens com maior importância.

O SCRUM trabalha com desenvolvimento incremental, dividindo seus processos em *sprints*. *Sprints* é o nome como é chamado às interações que ocorrem no SCRUM, ou seja, o período de trabalho para cada fase incremental.

Cada *sprints* tem duração em média de 30 dias e tem seu objetivo claro e definido, conhecido por toda a equipe. Dento de cada *sprints* acontecem reuniões diárias com duração média de 15 minutos, onde proporciona ao SCRUM master a atualização do status do projeto e auxilia na tomada de decisões do mesmo.



Figura 5 - Ciclo de Vida do Projeto Scrum

Fonte: <http://www.sstecnologia.com.br/>

Se o projeto tiver mais de um *Sprint* definido, cada *Sprint* deve conter uma nova implementação no produto, cabendo ao proprietário do projeto ao final de cada *Sprint* a decisão de implantar o produto que já esta desenvolvido ou tomar esta decisão mais tarde em outro final de *Sprint*. A cada final de *Sprint* o produto que foi definido para ser desenvolvido deve estar pronto, codificado e testado.

COMPARATIVO ENTRE AS METODOLOGIAS TRADICIONAL E ÁGIL

CARCTERÍSTICAS	GERENCIAMENTO TRADICIONAL	GERENCIAMENTO ÁGIL
Ter definido a priori	Escopo	Tempo (<i>sprints</i>)
Responsável pela organização para atingir os objetivos do projeto	Gerente de projeto	SCRUM master
Frequência de reuniões de status	Dependendo da complexidade / necessidade do projeto, alinhar a frequência.	Diárias
Escopo	Bem definido nas fases iniciais do projeto e formalização através da WBS (Work Breakdown Structure).	Escopo é definido em alto nível e os requisitos são priorizados e definidos de forma iterativa. Necessita de maior controle de planejamento.
Tempo	Cronograma detalhado para realização de todo o projeto.	Cronograma orientado a produto com entregas incrementais de 2-4 semanas.
Custo	Monitoração das alterações para que não altere o custo planejado.	Maior controle em função da rapidez na incorporação de alterações.
Qualidade	Processos de verificação, validação e plano de testes.	Programação em pares, testes incrementais e refatoração.
Riscos	Análise de riscos durante todo o ciclo de vida do projeto.	Aplica-se o mesmo conceito do gerenciamento tradicional.
Comunicação	Documentação e formal.	Implícita, Interpessoal e colaborativa.
Recursos Humanos	Papéis claros e bem definidos.	Confiança nos membros da equipe e ambiente colaborativo.
Aquisição	Controle por contrato e escopo bem definido e documentado.	Presença do cliente, volatilidade de requisitos e pouca documentação.
Integração	Plano do projeto detalhado e controle total do projeto pelo gerente.	Plano do projeto evolutivo e gerente do projeto atuam como facilitador.

Quadro 3 – Quadro Comparativo entre PMBOK X SCRUM

Fonte: Adaptado da Revista Mundo Project Management - nº 36 - Dez/Jan 2011 - p.44

CONCLUSÃO

O estudo comparativo entre as práticas do PMBOK *Guide* e do SCRUM dão a base para que se faça uma análise entre as diferenças e/ou semelhanças. Através das informações analisadas e propostas é possível perceber que ambas as metodologias aparecem para melhorar as que já existem, porém cada uma tem posições diferentes, enquanto as práticas do PMBOK *Guide* pregam a integração da organização em um nível mais estratégico e formal, o SCRUM norteia a operacionalização da gestão de projetos de maneira informal e colaborativa. As práticas podem ser complementares, onde pontos falhos de uma podem ser supridos por pontos fortes da outra.

Apesar da complexidade dos sistemas e a dificuldade das organizações em implantarem projetos de software no cenário contemporâneo, o caminho mais indicado para se obter melhores resultados é a gestão de projetos, desde que forneça condições efetivas para que as organizações alcancem seus objetivos estratégicos, respeitando o escopo, o prazo, o custo, a natureza do processo e o negócio da empresa.

Ao contratar uma empresa específica para desenvolver o projeto de sua empresa deve-se saber o que é mais importante, ter uma documentação especializada de todo os processos sem ter a necessidade da entrega rápida ou ter o sistema desenvolvido com agilidade e entregue por partes. Ambas as metodologias tem sua eficiência, porém em determinados casos não atinge o esperado do cliente, que é ter o seu sistema funcionando o mais rápido possível. Os quadros comparativos entre as duas metodologias mostram que uma foca em documentação, enquanto a outra em desenvolvimento.

Toda empresa pode adquirir ambas as metodologias desde que possa utilizar apenas uma por projeto e que a escolha seja baseada em atender a necessidade principal da empresa.

REFERÊNCIAS BIBLIOGRÁFICAS

BELMIRO, N. J. **Metodologia de desenvolvimento de sistemas**. São Paulo: Érica Editora Ltda, 1993.

BECK, K. et al. **Manifesto para o desenvolvimento ágil de software**. c2001. Disponível em: <<http://manifestoagil.com.br/index.html>>. Acesso em: 05 maio 2013.

CHAOS Tuesday. Disponível em: <<http://blog.standishgroup.com/>>. Acesso em: 19 de maio 2013.

DENNIS, A.; WIXON, H. B.. **Análise e projeto de sistemas**. 2. ed. Rio de Janeiro: LTC, 2011.

FERNANDES, A. A.; ABREU, V. F.. **Implementando a Governança de TI: da estratégia a gestão dos processos e serviços**. 3. ed. Rio de Janeiro: Brasport, 2012.

FURTADO, A. **Pontas do iceberg do caos no desenvolvimento de software**. Disponível em: <<http://www.microsoft.com/brasil/msdn/Tecnologias/Carreira/DesenvolvimentoSoftware.aspx>>. Acesso em: 19 maio de 2013. KERZNER, H. **Gestão de projetos: as melhores práticas**. Tradução Lene Belon Ribeiro. 2.ed. Porto Alegre: Bookman, 2006.

LEITE, A. F. **Metodologia de desenvolvimento de software**. Disponível em: <<http://www.devmedia.com.br/metodologia-de-desenvolvimento-de-software/1903>>. Acesso em: 12 mar. de 2013. .

MARTINS, W. F.; WOILER, S. **Projeto: planejamento, elaboração, análise** - 2. ed. São Paulo: Atlas, 2011.

MAXIMIANO, A. C.M. **Administração de projetos: como transformar idéias em resultados**. 3. ed. -São Paulo: Atlas, 2009.

MAXIMO, A. C. M. **Administração de projetos: como transformar ideias em resultados**. 3. ed. São Paulo: Atlas, 2008.

MIGUEL, A. **Gestão de projetos de software**. 4. ed.. [S.l]: Editora FCA, 2010.

RODRIGUES, R.; ROST, R. **SCRUM (metodologia para o desenvolvimento de ágil de software)**. Disponível em: <<http://rafaelrgi.files.wordpress.com/2007/11/scrum.pdf>> 29/03/2012>. Acesso em 12 de mar. 2013.



SCHWABER, K. **Guia do scrum**. [S.l.]: Scrum Alliance, 2009.

SOMMERVILLE, L. **Engenharia de software**. 8.ed. São Paulo: Pearson Addison Wesley, 2007.