

## ANÁLISE DE RISCOS PELO USO DE MÉTODOS ÁGEIS NA GESTÃO DE PROJETOS DE DESENVOLVIMENTO DE SOFTWARE

### RESUMO

Métodos Ágeis (MA) têm promovido melhorias em diversos aspectos da gestão de projetos de desenvolvimento de software. No entanto, sua aplicação envolve riscos, que podem ser maiores em certas organizações ou contextos de projetos de software. O objetivo deste trabalho é analisar os principais riscos pela utilização de MA em empresas desenvolvedoras de software. O trabalho foi desenvolvido como pesquisa exploratória e qualitativa, com entrevistas abertas e semiestruturadas, com gestores de projetos de software. Resultados do trabalho mostram que diferentes tipos de riscos podem ocorrer e impactar de forma diferente. No entanto, as conclusões indicam que são apenas duas as principais causas de riscos: (i) engajamento de clientes e (ii) experiência e conhecimento com práticas ágeis.

**Palavras-chave:** Projetos de Desenvolvimento de Software; Métodos Ágeis para Gestão de Projetos de Software; Riscos em Projetos de Software.

## RISK ANALYSIS FOR USE IN METHODS OF AGILE SOFTWARE DEVELOPMENT PROJECT MANAGEMENT

### ABSTRACT

Agile Methodologies (AM) are offering improvements on several aspects of software development projects. However, its application involves risks, which are higher for some organizations and software projects. The objective of this work is to identify the main risks of applying AM in software development companies. The work involved an exploratory and qualitative research carried out by open and semi structured interviews with software project managers. Results show that many different risks can occur and with different impact on software projects. However, the main reasons behind all sort of risks are due to two aspects: (i) client involvement and (ii) experience and knowledge with agile practices.

**Keywords:** Software Development; Agile Methods for Project Management, Risks on Software Development Projects.

Leonardo Rocha de Oliveira<sup>1</sup>  
Geraldo Santoro Gomes<sup>2</sup>  
Fernando Policarpo de Lima<sup>3</sup>

<sup>1</sup> Doutor em Tecnologia de Informação pela University of Salford, Inglaterra. Professor da Faculdade de Administração, Contabilidade e Economia (FACE) da Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS. Brasil. E-mail: [leorochadeoliveira@gmail.com](mailto:leorochadeoliveira@gmail.com)

<sup>2</sup> Mestre em Administração pela Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS. Brasil. E-mail: [gsgomes39@gmail.com](mailto:gsgomes39@gmail.com)

<sup>3</sup> Graduando em Engenharia de Produção pela Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS. Brasil. E-mail: [fernandopolicarpo90@gmail.com](mailto:fernandopolicarpo90@gmail.com)

## 1 INTRODUÇÃO

Visando maior competitividade, produtividade e lucro, empresas dos mais diversos setores da economia buscam constantemente por melhorias e inovações nas formas de realizar suas atividades de trabalho. Empresas desenvolvedoras de software também se inserem neste contexto, e Métodos Ágeis (MA) para gestão de projetos de desenvolvimento de software têm sido apontados como alternativa para promover melhorias nas atividades de trabalho destas empresas, em aspectos como velocidade de desenvolvimento, diminuição de erros de escopo, maior alinhamento com os objetivos de negócios de clientes, maior motivação dos times de projeto e capacidade para gerar um produto final com maior qualidade (Hiranabe, 2008; Larman & Basili, 2003; Poppendieck & Poppendieck, 2006).

A revisão de literatura indica que MA não devem ser vistos como soluções definitivas para problemas com projetos de software, e que certas práticas podem não ser adequadas a determinadas organizações ou contextos de projetos (Larman & Basili, 2003; Mølkken-Østvold & Jørgensen, 2005; Schwaber, 2004). Projetos de software são reconhecidos pela inerente incerteza quanto à serem entregues aos clientes com o custo, escopo, tempo e qualidade planejada (PMI, 2009; Sheffield & Lemétayer, 2013). É comum que projetos sofram variações de até 60% nos custos e prazos estimados, devido a, em grande parte, erros de planejamento nas fases iniciais do projeto (Cohn, 2005; Sheffield & Lemétayer, 2013).

A fase inicial de planejamento e de estimativas em projetos de software é bastante dependente da experiência de gestores, os quais podem ser levados a tomar decisões sem total asserção sobre os objetivos e requisitos indicados por clientes (Cohn, 2005; Keith & Cohn, 2008; Larman & Basili, 2003). O resultado dessas decisões leva a ocorrência de alterações de escopo (Kniberg, 2007), com impacto em diversos aspectos do projeto, tais como custo, tempo, qualidade e, principalmente, insatisfação do cliente (Waters, 2007). É também comum a saída de membros do time de desenvolvimento, o que pode levar a perda de conhecimento especializado, com prejuízos financeiros e atrasos nas entregas (Magne Jørgensen, 2002).

Aceitar projetos significa assumir riscos, os quais são maiores quando os requisitos inicialmente estimados são alterados durante o ciclo de desenvolvimento, e empresas desenvolvedoras de software devem estar preparadas para essa realidade, pois é comum em projetos de software (Jones, 2007; M. Jørgensen & Shepperd, 2007; Schwaber, 2004). Requisitos podem evoluir, levando a criação de

software com escopo, preço, prazo e qualidade diferentes do que o cliente esperava contratar (Paetsch, Eberlein, & Maurer, 2003). MA têm sido indicados como alternativa para resolver ou minimizar esses problemas.

Não existem garantias de que a adoção de MA gere benefícios a processos de software, mas é importante conhecer os riscos pela sua adoção (Cockburn, 2002; Highsmith, 2004; Keith & Cohn, 2008). Embora existam exemplos de aplicações bem sucedidas de MA em empresas desenvolvedoras de software, ainda são poucas as referências acadêmicas e sem cunho comercial indicando diretrizes para garantir o sucesso com sua utilização (Sheffield & Lemétayer, 2013). É reconhecido o fato de que MA são muito dependentes da relação entre cliente e desenvolvedores (Boehm & Turner, 2003; Fowler, 2005; Schwaber, 2010). Também é bem referenciado o fato de que, se não forem utilizados corretamente, MA podem criar mais problemas do que soluções (Cockburn, 2002; Fowler, 2005; Kerzner, 2009; Schwaber, 2010).

Minimizar riscos com a aplicação de MA é atualmente um desafio para empresas que desenvolvem software. Empresas brasileiras vêm utilizando diversas metodologias em projetos de software, tanto para atividades de engenharia de desenvolvimento como para gestão de processos. No entanto, estas ainda não oferecem diretrizes para eliminar riscos em projetos de software.

O objetivo deste trabalho é analisar os principais riscos pela utilização de MA em empresas desenvolvedoras de software. Para isso, foi revisada a literatura sobre metodologias e práticas ágeis, para identificar o conjunto de riscos por sua aplicação nas diversas etapas dos processos de trabalho de empresas desenvolvedoras.

## 2 MA NA GESTÃO DE PROJETOS DE DESENVOLVIMENTO DE SOFTWARE

MA tem origem a partir da experiência de profissionais da área de gestão de projetos, com inspiração nas demandas inicialmente criadas pela produção enxuta em processos de manufatura de produtos industrializados (Hiranabe, 2008; Larman & Basili, 2003; Waters, 2007). No caso da indústria de manufatura, o objetivo era de aumentar o *mix* de produtos e flexibilizar as linhas de produção (Larman & Basili, 2003). Mais tarde, MA foram também adotados na gestão de projetos de software, e desde então vem se difundindo no dia a dia de empresas desenvolvedoras.

O movimento pela adoção de MA na gestão de projetos de software começou oficialmente com a publicação do Manifesto Ágil (*Agile Manifesto*) em fevereiro de 2001 (Beck, 2001). A elaboração deste

manifesto envolveu esforço internacional para reunir profissionais com conhecimento especializado na área. O resultado foi a criação de uma lista de valores para gestão de projetos de software, a qual veio a inspirar as práticas ágeis atuais de engenharia ágil (Beck, 2004). Resultados desta iniciativa são modelos conhecidos como Programação Extrema (XP), SCRUM e Software Enxuto (*Lean Software*), os quais possuem como principais valores (Beck, 2004; Waters, 2007):

- Indivíduos e iterações são mais importantes que processos e ferramentas;
- Software funcionando é mais importante que documentação;
- Colaboração do cliente é mais importante que negociação de contrato;
- Responder a mudanças é mais importante que seguir um plano rígido.

Com a divulgação do manifesto ágil, novas práticas, valores, técnicas e metodologias para gestão de projetos ágeis começaram a se difundir, indicando diretrizes para adoção de MA (Alliance, 2001; Beck, 2004; Waters, 2007). MA seguem um conjunto de regras e valores que indicam seis como principais fatores de sucesso em projetos de software (Beck, 2001, 2004):

- Equipes ágeis funcionam como uma única entidade - todos no time devem estar alinhados ao mesmo objetivo e não existe espaço para indivíduos que fazem sua parte e não se preocupam se os colegas conseguem executar as deles. Se alguma coisa falhar, todos falham, pois é responsabilidade de todos o sucesso do projeto e o bem comum.
- Equipes ágeis trabalham com iterações curtas - não existe definição precisa de fases do projeto, mas as iterações entre os times devem ocorrer com frequência de dias ou semanas, e se alguma tarefa não for concluída, esta deve ser passada como prioridade para a próxima iteração.
- Equipes ágeis devem entregar algo a cada iteração – cada iteração possui um grupo de funcionalidades a ser entregue ao cliente, as quais devem ser incrementadas nas iterações seguintes e testadas de acordo com as exigências de qualidade do projeto.
- Equipes ágeis focam em prioridades de negócio – times de desenvolvimento devem entregar funcionalidades que agregam valor ao negócio do cliente, e para isso devem trabalhar de forma alinhada com a visão do negócio e saber priorizar as funcionalidades

mais importantes e as que devem ser feitas primeiro.

- Equipes ágeis inspecionam e adaptam - o planejamento tende a mudar após o término de cada iteração e o time de desenvolvimento deve estar consciente se subestimou ou superestimou algumas tarefas, pois o conhecimento sobre o projeto deve crescer em cada iteração. Esse conhecimento deve ser agregado aos times do projeto e clientes, oferecendo oportunidades a todos de opinar e gerar mudanças, as quais podem se refletir em uma tarefa a ser inserida (ou removida) do projeto, ou mesmo ter sua prioridade alterada na ordem de entrega.
- Planejamento ágil - equipes ágeis devem seguir três horizontes, que são (i) as entregas, (ii) as iterações e (iii) as datas atual e final. As entregas são planejadas no início do projeto e a cada iteração, indicando funcionalidades e prazos. O planejamento ágil lida com pequenas tarefas, com prazos estimados em horas, e que juntas compõem uma funcionalidade testada e pronta para ser entregue ao cliente.

Times ágeis trabalham como uma equipe única visando um objetivo comum, mas com divisão de trabalhos e responsabilidades (Cohn, 2005). As equipes trabalham com iterações curtas e o planejamento e estimativas são refeitos em cada iteração, visando entregar um produto funcionando para receber retorno do cliente (Boehm & Turner, 2003). O foco em prioridades do negócio deve ser levado em conta e devem sempre ser entregues primeiro as funcionalidades de maior prioridade e valor para o cliente, o qual precisa estar trabalhando junto com as equipes de desenvolvimento (Boehm & Turner, 2003; Keith & Cohn, 2008; Schwaber, 2004).

Estudar os valores e princípios ágeis é fundamental para obter o conhecimento necessário para identificar quando, onde e como MA podem ser usados com sucesso em projetos de software (Boehm & Turner, 2003; Schwaber, 2004). Estes são motivos que levaram a elaboração deste trabalho, que busca analisar os principais riscos com a utilização de MA em empresas desenvolvedoras de software.

### 3 MÉTODO DE PESQUISA

O trabalho apresenta uma pesquisa exploratória e qualitativa, a qual é indicada para casos em que o pesquisador não tem uma ideia clara dos problemas que devem ser enfrentados na investigação (Cooper & Schindler, 2004). Para identificar fatores de risco em projetos de software

com MA foi inicialmente revisada a literatura. O trabalho de pesquisa também contou com a opinião de especialistas para validar os fatores de risco indicados na literatura no contexto de empresas desenvolvedoras de software. Este tipo de pesquisa permitiu analisar a aplicação de MA nas empresas dos entrevistados, bem como os principais riscos apontados com a utilização. Os resultados da análise de conteúdo das entrevistas foram confrontados com a revisão de literatura para um maior entendimento sobre a situação das empresas analisadas.

A coleta de dados foi realizada com entrevistas abertas a profissionais de gestão de projetos de software. O critério de seleção dos entrevistados considerou dois principais aspectos: (i) mínimo de 6 anos de experiência como gestor de projetos de software; e (ii) mais de dois anos com MA. A população de gestores de projetos de software com experiência em MA foi representada pela amostra de 14 entrevistados, os quais foram selecionados por conveniência, ou seja, pela proximidade com os pesquisadores e disponibilidade em participar da pesquisa.

As entrevistas foram direcionadas pelo instrumento semiestruturado para coleta de respostas em profundidade, permitindo que os entrevistados se sentissem à vontade para comentar sobre o foco de investigação, e com liberdade quanto ao tempo e forma de resposta (Malhotra, 2006). A análise de conteúdo das entrevistas foi focada em captar a percepção dos entrevistados em relação a ocorrência e impacto de riscos pela aplicação de MA em empresas desenvolvedoras de software.

As entrevistas foram realizadas de forma que as opiniões do primeiro entrevistado fossem revisadas pelo segundo, e assim respectivamente. O objetivo foi a busca de consenso sobre os fatores de risco e os motivos de ocorrência e impacto. Portanto, em cada nova entrevista, a opinião do entrevistado era confrontada com a dos profissionais previamente entrevistados, bem como com a literatura.

### 3.1 Objetivos de Análise e Proposições da Pesquisa

Os objetivos de análise da pesquisa contemplam aspectos gerais e específicos sobre riscos pelo uso de MA em projetos de software. Os riscos foram analisados quanto ao (i) Impacto e (ii) Probabilidade de Ocorrência. O (i) Impacto analisa a gravidade e abrangência dos danos e, em conjunto com a (ii) Probabilidade de Ocorrência, se pode obter um mapeamento do grau de risco por falhas nas diversas atividades dos processos de desenvolvimento de projetos de software com MA.

A construção do instrumento de pesquisa foi elaborada de forma a permitir a identificação de eventos não planejados que podem gerar prejuízos

no decorrer do projeto, com destaque aos especificamente relacionados com a utilização de MA. Estes riscos contemplam aspectos das equipes de trabalho ao planejamento de projetos, a aplicação de práticas de desenvolvimento ágil, gerenciamento e controle dos projetos. A análise de risco contempla o ciclo de vida de projetos de software e considera as fases de planejamento, desenvolvimento, processos de controle e acompanhamento, qualidade do produto e entregas do software. Esta definição buscou identificar a origem, o motivo e categorizar a ocorrência e impacto do risco.

As dimensões e fatores de análise propostas para representar MA foram agrupadas de forma a ponderar sobre as seguintes proposições:

**Proposição 1** – quanto maior a experiência prévia dos líderes de projeto e times de desenvolvimento com MA, menor serão os riscos com sua utilização em empresas desenvolvedoras de software;

**Proposição 2** – quanto maior o engajamento do cliente no projeto, menores serão os riscos com o uso de MA por empresas desenvolvedoras de software.

A análise de riscos e teste das proposições foi realizada a partir de instrumento de pesquisa utilizado para guiar o processo de entrevistas. A estrutura do instrumento seguiu as recomendações de Cooper e Schindler (2004) e apresenta três tipos de questões de mensuração: (i) questões gerenciais (ou demográficas); (ii) questões de classificação e; (iii) questões de direcionamento. Os objetivos de cada tipo de mensuração são:

(i) Mensuração Gerencial - busca analisar o perfil do entrevistado e posição na empresa, atuando como variáveis de controle para os resultados da pesquisa.

(ii) Mensuração de Direcionamento – buscam informações sobre a experiência do entrevistado com os riscos associados às diferentes práticas e resultados de aplicações de MA em processos de desenvolvimento de projetos de software.

(iii) Mensuração de Classificação – busca aprofundar a análise e identificar justificativas para as opiniões dos respondentes, inclusive dando liberdade para que sejam incluídos aspectos não previstos no instrumento.

Na mensuração de direcionamento o instrumento analisa opiniões sobre três Dimensões de Análise (DA), que são: (DA-1) conhecimento das equipes sobre MA em gestão de projetos de software; (DA-2) papel do Cliente nos projetos e; (DA-3) sobre aspectos culturais das relações inter e intra organizacionais em relação à filosofia ágil.

Cada DA é constituída por Fatores de Análise (FA), os quais são apresentados como questões para os respondentes (Cooper & Schindler, 2004). A versão final do instrumento de pesquisa está no Anexo 1, e a seguir estão detalhes sobre as DA e FA no instrumento.

- DA-1 – Conhecimento sobre MA – explora a experiência do entrevistado sobre MA e as práticas que utiliza. Os Fatores de Análise (FA) propostos para esta dimensão são:
  - FA-1 - experiência sobre as relações entre integrantes dos times de projetos (Beck, 2004; Cohn, 2005; Schwaber, 2004);
  - FA-2 - experiência sobre o papel dos gerentes e líderes de projetos (Beck, 2004; Cohn, 2005; Schwaber, 2004);
  - FA-3 – experiência com atividades de planejamento de projetos, tais como definição de requisitos, estimativas, distribuição de tarefas e prazos de iterações (Beck, 2004; Cockburn, 2002; Cohn, 2005; Schwaber, 2004).
  - FA-4 – experiência com atividades de acompanhamento e controle de projetos, tais como condução de reuniões, revisões de funcionalidades e motivação dos times de projeto (Beck, 2004; Cockburn, 2002; Cohn, 2005; Schwaber, 2004).
  - FA-5 - experiência com práticas de engenharia do desenvolvimento ágil, tais como adaptação a mudanças, programação em pares, desenvolvimento orientado a testes e design incremental (Beck, 2004; Cohn, 2005; Schwaber, 2004);
- DA-2 – Papel do Cliente (*Product Owner*) - explora o conhecimento e experiência do entrevistado sobre riscos associados ao envolvimento do cliente em projetos ágeis. Esta dimensão contempla apenas um Fator de Análise (FA):
  - FA-6 – Envolvimento do Cliente: percepção sobre os riscos associados ao perfil, tipo de envolvimento e experiência do cliente em trabalhar com MA (Beck, 2004; Cockburn, 2002; Cohn, 2005; Schwaber, 2004).

- DA-3 – Aspectos Culturais - explora o conhecimento e experiência do entrevistado em relação aos riscos associados a aspectos culturais das instituições e times de projeto em relação à filosofia ágil. Esta dimensão possui apenas um FA:
  - FA-7 – Filosofia Ágil nas organizações: percepção sobre os riscos associados às mudanças na forma de gestão e de desenvolvimento de projetos ágeis, tais como quanto a forma de documentação, gestão de mudanças de escopo e definição dos termos de contrato (Beck, 2004; Cockburn, 2002; Cohn, 2005; Schwaber, 2004).

O texto das questões que compõem cada DA e FA estão no instrumento de pesquisa no Anexo 1.

## 4 ANÁLISE DE RESULTADOS

Os resultados para classificar os riscos quanto ao impacto e probabilidade de ocorrência em projetos de software foram analisados com base em escala Likert de 5 pontos (5 – Muito Alto; 1 – Muito Baixo). A partir das justificativas para as respostas em escala se buscou identificar os motivos e formas de evitar a ocorrência de riscos. Mais detalhes sobre os resultados obtidos neste trabalho estão detalhados a seguir.

### 4.1. Perfil dos Entrevistados

As primeiras nove questões do instrumento foram utilizadas para identificar o perfil dos entrevistados em aspectos como função na empresa, tempo de experiência profissional e tempo de uso de MA. O resumo dos resultados sobre o perfil dos entrevistados está no Quadro 1, e cabe destacar que todos possuíam 6 ou mais anos de experiência com projetos de software, mas não mais de 4 anos com o uso de MA.

	Cargo na Empresa	Tempo na Empresa	Experiência na área	Experiência com MA	Formação Acadêmica	Formação Graduação
P1	Desenvolvedor sênior	5 anos	10 anos	4 anos	Especialização	Informática
P2	Desenvolvedor sênior	3 anos	6 anos	2 anos	Graduação	Informática
P3	Desenvolvedor sênior	2 anos	6 anos	2 anos	Graduação	Informática
P4	Gerente de projetos	2 anos	6 anos	2 anos	Especialização	Administração
P5	Desenvolvedor sênior	1 ano	7 anos	2 anos	Graduação	Informática
P6	Gerente de projetos	7 anos	12 anos	4 anos	Especialização	Engenharia
P7	Desenvolvedor sênior	4 anos	7 anos	3 anos	Especialização	Administração
P8	Desenvolvedor sênior	8 anos	12 anos	3 anos	Especialização	Informática
P9	Desenvolvedor sênior	3 anos	7 anos	2 anos	Graduação	Informática
P10	Desenvolvedor sênior	8 anos	10 anos	3 anos	Especialização	Administração
P11	Desenvolvedor sênior	4 anos	11 anos	2 anos	Especialização	Informática
P12	Gerente de projetos	9 anos	12 anos	3 anos	Especialização	Administração
P13	Gerente de projetos	3 anos	16 anos	4 anos	Mestrado	Informática
P14	Gerente de projetos	2 anos	15 anos	4 anos	Mestrado	Administração

Quadro 1 - Perfil dos entrevistados.

Todos os entrevistados demonstraram experiência com a utilização de SCRUM e práticas de Programação Extrema (XP), tais como integração contínua, design incremental e *Planning Poker*. Embora não tenha sido considerado como critério de seleção, todos trabalham com projetos distribuídos, com equipes de desenvolvimento no país e exterior. Dos 14 entrevistados, 10 trabalham em empresas sediadas no Parque Tecnológico da Pontifícia Universidade Católica do Rio Grande do Sul (TecnoPucrs), sendo que 5 destes trabalham com projetos em que o cliente é a própria empresa para a qual trabalham. Embora este aspecto possa influenciar na percepção sobre a Proposição 2 (papel do cliente), não foi notado qualquer viés na Proposição 1 (experiência com MA). Portanto, foi tomado um cuidado especial nas entrevistas com estes 5 especialistas para evitar a influência do viés de trabalharem com cliente interno. Os outros 9 especialistas trabalham em empresas desenvolvedoras com clientes no Brasil e exterior (US e Índia).

#### 4.2 Riscos Quanto a Experiência com MA (DA-1)

Os entrevistados opinaram de forma consensual, e de acordo com a revisão de literatura,

confirmando que um dos maiores riscos em projetos ágeis está na falta de experiência e conhecimento com MA. Os comentários destacaram que conhecimentos e competências devem estar principalmente com gerentes de projetos, e por dois motivos: (i) para que tenham capacidade de liderar os processos ágeis; e (ii) para que possam atuar como multiplicadores junto a clientes e times de desenvolvimento. Portanto, a Proposição 1 de que a experiência prévia com MA influencia positivamente em minimizar riscos com o uso de MA em empresas desenvolvedoras foi confirmada consensualmente pelos 14 entrevistados.

##### 4.2.1 Riscos Quanto ao Papel dos Times de Projeto (FA-1)

A análise de riscos sobre o papel dos times de projeto (FA-1) foi realizada por 8 questões de classificação no instrumento (1.1 a 1.8) que abordam aspectos internos das relações pessoais na empresa desenvolvedora. Essa análise foi direcionada pela questão 7.1 do instrumento, a qual representa o valor ágil de substituir o excesso de ferramentas e processos por comunicação face a face nos times de projeto. Segundo Cohn (2005), a comunicação nos times de desenvolvimento deve ocorrer de forma a incentivar o envolvimento de

todos nas discussões, promovendo o engajamento e aprendizado, para manter o pessoal com conhecimento equivalente e de alto nível sobre o projeto.

Segundo os entrevistados, o item com mais destaque e classificado com alto risco de ocorrência e muito alto impacto é referente a problemas históricos de comunicação entre os times (1.3). Equipes que não se comunicam têm dificuldade de manter o conhecimento alinhado, gerando conflitos e dificuldade para realizar tarefas compartilhadas. Esse risco é maior quando problemas de comunicação estão enraizados na empresa desenvolvedora. Segundo Cockburn (2001), pessoas que desenvolvem em conjunto devem inclusive trabalhar fisicamente próximas para minimizar barreiras de tempo, custo e formalidade nas comunicações.

A questão 1.1 aborda o risco dos membros da equipe não terem o nível de senioridade suficiente para utilizar MA. Esse risco foi classificado como o segundo mais importante, com alta probabilidade de ocorrência e alto impacto nos resultados do projeto. Nas questões de classificação 7.5 e 7.6 os entrevistados acrescentaram o fato de que este risco é ainda maior no caso de equipes que trabalham geograficamente distribuídas, especialmente quando a língua e fuso horário são acrescentados como barreiras. O custo da transferência de ideias nesses ambientes causa perda de oportunidades para sugestões de melhorias e inovação nos projetos, as quais são geradas por iterações bem sucedidas entre os times de projeto.

#### **4.2.2 Riscos Quanto ao Papel dos Gerentes de Projeto (FA-2)**

No FA-2 foram analisados os riscos relativos ao papel de gerentes de projetos em empresas desenvolvedoras. Equipes ágeis se diferem das tradicionais por serem auto gerenciáveis, e por isso é esperado que todos tenham a iniciativa para decidir o que é melhor para o projeto e saber tomar decisões (Highsmith, 2004). Portanto, gerentes de projetos não devem focar em atividades de gestão e controle, mas em liderança para ajudar os times a conduzir os trabalhos e remover impedimentos. Equipes ágeis devem focar em prioridades de negócios, trabalhando em conjunto com times do cliente, e entregando valor ao negócio, o mais rapidamente possível (Albino, Souza, & Prado, 2014). Esta análise foi realizada por 4 questões (2.1 a 2.4) de direcionamento e uma de classificação (7.4).

Os entrevistados concordaram com a revisão de literatura e apontaram que os itens 2.1 e 2.2 são os de maior ocorrência e impacto em projetos ágeis. Isso se justifica por uma cultura de

qualidade e controle de processos em empresas desenvolvedoras que dificulta o entendimento de que gerentes atuem com foco em fornecer os meios para que suas equipes façam seus trabalhos bem feitos, e de acordo com preceitos de MA. “Gerentes devem focar em seu papel como *ScrumMaster*, priorizando atividades de facilitador”. No entanto, para isso é necessário conhecer o *SCRUM* em profundidade, a ponto de ser capaz de ensinar e transmitir segurança aos clientes e times de desenvolvimento. As respostas à questão 7.4 reiteram a importância do envolvimento eficiente do gerente de projetos em ambientes ágeis, especialmente quando ocorrem necessidades de mudanças. Como definido por Highsmith (2004), gerente de projetos deve trabalhar removendo impedimentos da equipe e facilitando o trabalho de todos. O maior motivo para riscos sobre o papel de gerentes se deve ao fato de se renderem a aspectos culturais rígidos de qualidade e controle de processos, com pouco espaço para atuar de acordo com MA.

#### **4.2.3 Riscos Quanto ao Planejamento em Projetos Ágeis (FA-3)**

No FA-3 foram analisados os riscos em atividades de planejamento em projetos ágeis. Para isso foram utilizadas 10 questões (3.1 a 3.10) de direcionamento e uma (7.2) de classificação. MA priorizam a entrega de software funcionando sobre a documentação excessiva dos trabalhos dos times de projeto. Segundo Cockburn (2002), documentações, requisitos, diagramas e planos servem como guias, mas não permitem prever o futuro. Os entrevistados concordam que documentação não deve ser a base do projeto. Deve apenas ser entregue o acordado com os clientes, e garantindo o necessário para manter saudável o andamento do projeto. Riscos com falhas de documentação foram associados à experiência dos gestores, bem como no perfil do cliente em se envolver e saber negociar as contratações.

O item apontado como de maior risco de ocorrência e impacto foi devido ao fato de times de projeto não terem conhecimento suficiente do que estão fazendo nas reuniões de planejamento, e acabarem subestimando necessidades de trabalho (3.7). Os principais motivos causadores desse risco são a falta de envolvimento do cliente e dificuldades para fornecer informações com clareza para que os times de desenvolvimento possam detalhar suas tarefas e executar o planejamento ágil.

#### **4.2.4 Riscos Quanto ao Acompanhamento de Projetos Ágeis (FA-4)**

A análise de riscos por falhas no acompanhamento foi realizada por 7 questões de classificação (4.1 a 4.7), as quais representam atividades das iterações ao longo do projeto. Essa análise foi aprofundada pela questão de classificação 7.4 ao final do instrumento. A literatura aponta para a necessidade por um ambiente colaborativo, no qual todos envolvidos possam acessar e compartilhar informações e conhecimento sobre o projeto (Kniberg, 2007; Waters, 2007). O acompanhamento das iterações pode ser feito por relatórios, gráficos, quadros informativos e demais formas de compartilhar informação. O objetivo deve ser sempre de atender às necessidades para comunicar resultados de projetos.

Os entrevistados indicaram que o risco por falhas de acompanhamento de maior ocorrência e impacto em projetos ágeis está relacionado ao fato dos membros dos times não saberem usar corretamente as reuniões diárias para comunicar seus impedimentos (16.3). Esse risco está ligado principalmente a baixa maturidade de desenvolvedores com MA. Outro risco apontado como de alta ocorrência e impacto é devido ao fato de que não sejam apresentadas funcionalidades totalmente prontas ao final das iterações (4.4). Embora seja alto o risco de ocorrência de que funcionalidades estejam 90% (ou quase completas) para apresentação nas interações (4.6), o impacto deste risco pode ser reduzido se o problema for bem apresentado ao cliente pelos gerentes de projeto, mantendo a confiança e satisfação com os times de empresas desenvolvedoras. A questão 7.4 buscou identificar causas de falhas em entregar as funcionalidades completas nas iterações, sendo que as principais são: (i) *user stories* mal escritas, (ii) dificuldade em priorizar funcionalidades, (iii) dificuldades de acompanhamento, e (iv) utilização equivocada de práticas ágeis. Também foi comentada a necessidade para que gerentes atuem de forma a evitar a perda de foco nas iterações, com discussões buscando culpados para problemas do passado.

Os entrevistados se mostraram alinhados com a revisão de literatura em apontar que a diminuição na documentação de processos não implica em falhas de gestão e acompanhamento de projetos. Também concordaram que a experiência de gerentes com MA é condição facilitadora para receber sugestões de clientes por mudanças com receptividade, mas ao mesmo tempo sendo realista quanto ao esforço e custo para que sejam realizadas. Os entrevistados foram unânimes em indicar que a confiança do contratante no contratado é crucial para minimizar riscos de acompanhamento, especialmente nas fases iniciais de planejamento e especificação de requisitos.

#### 4.2.5 Riscos Quanto a Engenharia de Desenvolvimento Ágil (FA-5)

Este fator analisa os riscos associados a falhas por falta de conhecimento para aplicação de práticas de engenharia de desenvolvimento ágil, e foi realizada por 6 questões (5.1 a 5.6) de direcionamento e uma de classificação (7.8). Embora a literatura apresente metodologias com técnicas e boas práticas para aplicação em projetos ágeis, é comum que sejam customizadas e/ou mal utilizadas no dia a dia de projetos. Por exemplo, Schwaber (2004) e Kniberg (2007) citam que a Programação Extrema (XP) é uma metodologia de engenharia de software que, quando usada junto com o SCRUM, fornece um nível de qualidade extremamente alto a projetos de software. No entanto, é necessário conhecimento especializado para aplicação.

Todos os entrevistados comentaram sobre a média ocorrência do risco de que a programação em pares (5.1) não gere benefícios e acabe desperdiçando o tempo de outra pessoa. Este risco foi apontado com alto impacto, pois indica falta de experiência de desenvolvedores, com reflexos em vários outros aspectos do projeto.

O risco por falhas no uso de práticas de desenvolvimento orientadas a testes (5.2) foi indicado como de alta ocorrência, devido a percepções de que exigem muito esforço para serem aplicadas no dia a dia de projeto. No entanto, estes riscos foram indicados com médio impacto, pois não causam grandes reduções nos erros de aplicação. Um dos resultados mais peculiares foi sobre a análise de risco do *design* incremental (5.3), o qual foi apontado com ocorrência muito baixa, pois a falta de definição clara do futuro é natural em projetos ágeis. No entanto, apresenta alto impacto, pois as indefinições podem ser grandes o suficiente para prejudicar em decisões importantes sobre o futuro de projetos. O risco pela necessidade de que a definição e uso de padrões de codificação exijam senioridade (5.4) foi classificado como de alta ocorrência e médio impacto, pois falhas dos times de desenvolvimento podem ser compensadas pelo papel do gerente. O risco de integração contínua (5.5) foi identificado como de média ocorrência e impacto, pois apesar das reuniões diárias consumirem tempo que poderia ser melhor utilizado em outras atividades, são úteis para reduzir retrabalhos, promover o entendimento sobre o projeto e motivar os times.

Quanto a análise geral sobre o uso de modelos e práticas para aplicação de MA (5.6), todos os entrevistados já tinham usado práticas da programação extrema (XP) e SCRUM, embora alguns com maior tempo de experiência. Foi

também indicado que ambas possuem muitas características em comum, até mesmo por buscarem o mesmo objetivo de facilitar que sejam respeitados os valores e princípios ágeis. Segundo os entrevistados, o risco com a adoção de MA depende muito do aspecto citado por Schwaber (2004), de que os modelos para aplicação não devem ser incorporados como um processo rígido, mas adaptados aos diferentes perfis de cada organização. Também houve consenso ao indicar que não se deve modificar um modelo comprovado como SCRUM ou XP de forma leviana, mas que é vantajoso identificar e justificar alterações para atender ao perfil do projeto.

A questão 7.8 buscou analisar o contexto de riscos com práticas de engenharia de desenvolvimento a partir da necessidade apontada nas referências pelo uso de testes integrados ao longo do ciclo de vida do projeto. A percepção dos entrevistados é de que testes realizados na empresa desenvolvedora podem gerar resultados diferentes no ambiente do cliente. Portanto, além de conhecimento especializado para realização e gestão de testes de software, é também necessário engajamento do cliente, com papel de parceiro, para elevar a qualidade do produto.

#### **4.2 Riscos Quanto ao papel do Cliente em MA (DA-2)**

A revisão de literatura é forte em apontar sobre a importância do envolvimento do cliente para o sucesso de projetos ágeis e os entrevistados foram unânimes em confirmar a Proposição 2 de que o engajamento do cliente influencia positivamente em minimizar riscos com a adoção de MA por empresas desenvolvedoras.

A análise de riscos sobre dificuldades com o engajamento de clientes (FA-6) foi realizada por 6 questões de classificação (6.1 a 6.6), as quais representam atividades chave em que se espera a participação do cliente. Esta análise foi direcionada pela questão 7.3 do instrumento, a qual representa o valor ágil de colaboração do cliente sobre negociação de contrato. Segundo Waters (2007) e Schwaber (2004), o envolvimento do cliente é um dos pré-requisitos para trabalhar de forma iterativa e incremental. De forma geral, os entrevistados apontaram como muito alto o impacto de riscos por falhas de envolvimento de clientes, e com reflexo nas diversas etapas e processos de projeto ágeis. “Se isso realmente ocorrer, o impacto é muito alto em diminuir as chances de sucesso em projetos de software.”

O risco pela falta de envolvimento do cliente na definição do produto (6.1) foi também apontado como de alto impacto, e sua ocorrência deve necessariamente ser minimizada durante todo

o projeto. A ocorrência é menor quando as equipes de desenvolvimento e clientes são da mesma empresa, mas medidas como treinamentos para conscientização de clientes e eficiência nas práticas de comunicação podem minimizar a ocorrência deste risco.

O risco de falhas de envolvimento nas atividades das iterações (6.2), nas quais são apresentados os resultados dos trabalhos dos times de desenvolvimento e espera-se julgamento do cliente, também foi apontado como de alto impacto e ocorrência. Outro risco indicado como de alto impacto e ocorrência foi por falhas do cliente em conseguir comunicar uma visão clara do produto (6.4). Sem isso não se consegue visualizar necessidades de negócios do cliente, o que é importante para definir e priorizar a entrega de funcionalidades (Cohn, 2005). Se o cliente não se dedica de forma adequada ao projeto e não fornece uma visão clara do produto, todas as definições de planos e funcionalidades devem ser comprometidas.

O risco do cliente não dar importância para a priorização dos requisitos (6.3) foi apontado de forma similar ao risco de mudanças em excesso (6.5). Ambos foram indicados como de muito alto impacto, e os motivos apontados como causa para este comportamento de clientes estão ligados a um contexto histórico e cultural de rigidez em seguir cláusulas de contratos e planejar com antecipação. Esta visão foi aprofundada nas respostas da questão de direcionamento, que buscou analisar o princípio ágil de que a colaboração com o cliente é mais importante do que negociações de contrato (7.3). Cockburn (2002) e Cohn (2005) citam que clientes devem trabalhar em conjunto com times de desenvolvimento durante todo o projeto, e com os mesmos objetivos, como se fossem um só time. Os entrevistados indicaram que é comum encontrar clientes que não sabem exatamente o que querem no início de um projeto, mas têm sua visão aprimorada ao longo do tempo. O principal resultado da análise sobre o papel de clientes foi sobre a necessidade de confiança entre clientes e desenvolvedores. No entanto, mesmo que exista uma relação antiga e de confiança entre contratante e contratado, isso não deve superar a necessidade de envolvimento do cliente em agregar valor e promover as chances de sucesso em projetos de software.

#### **4.3 Riscos Quanto a Aspectos Culturais (DA-3)**

Apesar da quantidade de referências e boas práticas disponíveis na literatura para embasar conceitos e orientar a aplicação de MA, aspectos culturais ainda são barreiras para adoção que geram riscos em projetos de software (Sheffield & Lemétayer, 2013). Nesta seção foram analisados os

riscos com MA em relação a aspectos culturais das pessoas e organizações. A análise foi realizada com base em 9 questões de direcionamento (7.1 a 7.9) e para maior entendimento sobre os motivos e justificativas para as respostas nas questões de classificação (DA-1 e DA-2).

Segundo os entrevistados, MA ainda são considerados como novidade no mercado de software, com impacto especial sobre o papel de clientes. Foi também indicado que é comum encontrar resistências ainda maiores nas etapas iniciais de adoção de MA, tanto por empresas desenvolvedoras como clientes, e principalmente nas hierarquias superiores. A abordagem tradicional é de que empresas contratantes e contratadas usem o contrato como mecanismo para proteção de seus interesses. Por outro lado, MA indicam que negociações entre as partes devem ser mais importantes do que negociações de contrato (7.3). Este aspecto cultural e suas implicações foi apontado como o de maior ocorrência e impacto em projetos ágeis.

O segundo maior risco de ocorrência e impacto foi quanto a aspectos culturais referentes ao valor ágil de que respostas a mudanças são mais importantes do que seguir um plano rígido (7.4). Schwaber (2004) cita que iterações não devem ser blocos fechados e protegidos de novas ideias, e que requisições de mudanças originadas pelo cliente devem ser consideradas pelos gerentes e discutidas nas iterações. Os entrevistados indicaram que esse risco está também relacionado a desafios em documentar, controlar, fazer medições e gerenciar as comunicações sobre o que é indicado e realizado nas mudanças. A experiência de gerentes e clientes com MA foi apontada como papel chave em realizar estas tarefas com a habilidade e competência necessárias para minimizar a ocorrência e impacto de riscos sobre falhas em priorizar a entrega de funcionalidades sobre complexa documentação (7.2). A utilização de tecnologia e mecanismos para gestão de comunicações podem também facilitar o compartilhamento de informações e de conhecimentos entre os envolvidos, diminuindo o risco de ocorrência e impacto de mudanças no sucesso de projetos ágeis. A sobrecarga de tarefas e estresse com atividades do dia a dia foi também indicada como um dos motivos para que, mesmo o mínimo de documentação exigida por MA, seja vista como uma tarefa trabalhosa, que consome tempo e exige detalhes para que seja propriamente realizada.

Quanto a riscos em priorizar interações entre pessoas em detrimento do uso de ferramentas (7.1), os entrevistados foram unânimes em indicar que processos rígidos e ferramentas não diminuem riscos de falhas de planejamento e acompanhamento de projetos. O principal aspecto

indicado pelos entrevistados para reduzir a ocorrência e impacto da necessidade de constantes iterações está no envolvimento, maturidade e conhecimento do cliente e times de desenvolvimento com práticas ágeis. Este aspecto deve estar alinhado com a capacidade técnica, maturidade, experiência e motivação dos times de desenvolvimento em realizar as mudanças necessárias.

Questões culturais referentes ao uso de equipes distribuídas geograficamente no mesmo país (7.5) e em países diferentes (7.6) não foram apontadas como riscos especificamente relacionados a projetos ágeis. “São riscos com ocorrência e impacto similar em projetos de software gerenciados com métodos tradicionais.” Caso estes projetos sejam gerenciados com MA, as exigências sobre o envolvimento de clientes e experiência de gerentes continuam sendo as de maior risco. Segundo a literatura, equipes separadas geograficamente multiplicam problemas de comunicação, principalmente pela falta do contato pessoal e diferenças com fuso horário e idioma. Equipes grandes também levam a problemas de comunicação que podem ser minimizados pelo aumento do esforço de gerentes em acompanhar processos e gerenciar comunicações. No entanto, estes aspectos representam riscos apontados com ocorrência e impacto similar em projetos tradicionais. Solução apontada para minimizar estes riscos foi o uso do SCRUM of SCRUMS, o qual oferece práticas para melhorar o uso de SCRUM em equipes grandes e distribuídas.

Situação similar foi apontada quanto aos riscos de aspectos culturais em relação a definição de projetos com custo fixo (7.7) e execução de testes integrados no ciclo de vida (7.8). Ambos foram apontados com ocorrência e impacto similar em projetos tradicionais e ágeis, mesmo em relação aos tipos de problemas que causam. A definição de custo geralmente leva a falhas em entregar o que o cliente espera. Por outro lado, a execução integrada de testes não garante a qualidade do produto final e exige conhecimento especializado para que seja realizada. A vantagem de projetos ágeis está apenas na capacidade de priorizar a entrega de funcionalidades com maior velocidade, qualidade e valor ao negócio do cliente, o que exige as já conhecidas condições de envolvimento do cliente e conhecimento com MA.

## 5 CONCLUSÕES

Os entrevistados concordam com a literatura em indicar MA para proporcionar melhorias em relação a métodos tradicionais de gestão de projetos de software. Os objetivos de

melhorias para as empresas em que trabalham contemplam aspectos como: desenvolver produtos agregando maior valor ao negócio do cliente; motivar os times para se engajar nos projetos, compartilhando conhecimento e incentivando a participar com opiniões nas iterações e; reduzir tempo e esforço para desenvolver o projeto.

A revisão da literatura foi capital para construir a versão inicial do instrumento de pesquisa. Embora os autores já possuíssem experiência com MA, a literatura aprofundou conhecimento sobre valores, princípios e riscos em casos de uso de MA. A versão final do instrumento está no Anexo 1 e foi validada quanto à capacidade de explorar as percepções dos entrevistados sobre a ocorrência e impacto dos principais tipos de riscos em projetos ágeis. Embora o instrumento tenha sido apontado como moderadamente extenso, as entrevistas foram avaliadas como extremamente satisfatórias, em especial pela capacidade de gerar conteúdo e entendimento sobre riscos em projetos ágeis.

As conclusões em relação ao objetivo geral de analisar os principais riscos com a utilização de MA por empresas desenvolvedoras de software foram bastante similares ao indicado na revisão de literatura. No entanto, o instrumento de pesquisa foi o diferencial de destaque como contribuição do trabalho, pois permitiu mapear a realidade das empresas dos respondentes quanto a riscos com MA. O agrupamento das questões, considerando objetivos de mensuração (gerenciais, classificação e direcionamento), dimensões e fatores de risco em projetos ágeis foi também validado pelos especialistas e entendido como a grande contribuição do trabalho. A utilização do instrumento no dia a dia de empresas desenvolvedoras foi reconhecida por identificar aspectos específicos a serem observados, e que podem se apresentar de forma diferente como riscos em projetos de empresas desenvolvedoras.

Tal como indicado na literatura, a maturidade das equipes com MA foi indicada como fundamental para o sucesso de projetos ágeis. Profissionais experientes são diferenciais para assumir responsabilidades, especialmente em momentos de pressão e dificuldade. Portanto, a Proposição 1 de que o conhecimento de gerentes impacta positivamente em reduzir riscos em projetos ágeis foi unanimemente comprovada. Mudanças ocorrem, em alguns casos são drásticas, e exigem entendimento e integração de todos para minimizar riscos em projetos de software.

A Proposição 2 indica que o envolvimento do cliente é chave para minimizar riscos em projetos ágeis, e também foi consensualmente confirmada. Promover entendimento sobre necessidades de negócios a serem suportadas pelo software a ser

desenvolvido é papel do cliente. Segundo os entrevistados, as avaliações de ocorrência e impacto dos principais riscos por falhas de envolvimento de clientes estão propriamente representadas no instrumento e permitem identificar aspectos peculiares de projetos, de empresas desenvolvedoras e de clientes.

Conclusões quanto a aspectos culturais sobre riscos com MA destacaram que falhas de comunicação são os principais motivos para ocorrência de riscos em projetos ágeis. Riscos por aspectos culturais se manifestam de forma diferente em cada projeto, pois envolvem aspectos específicos dos times de desenvolvimento, clientes, gerentes de projetos, práticas de engenharia ágil e modo como são tratadas as mudanças. Práticas de Programação Extrema (XP) e SCRUM não foram indicadas como de fácil entendimento e aplicação, mas capazes de gerar resultados muito satisfatórios para representar e comunicar necessidades específicas de projetos de software e de empresas desenvolvedoras.

Na elaboração deste trabalho, especialmente na etapa de entrevistas, diversas ideias surgiram para aprofundar o conhecimento sobre riscos com MA, e que poderiam ser desenvolvidas em futuros trabalhos. Por exemplo, estudos de caso em empresas com equipes internas de desenvolvimento poderiam ser comparados com projetos contratados de empresas terceirizadas. Projetos em diferentes áreas, tais como em empresas do setor financeiro ou de tecnologia, as quais são esperadas como culturalmente mais conscientes sobre dificuldades em projetos de software, poderiam ser comparados com projetos em outros setores. Também é esperado que o instrumento desenvolvido neste trabalho como conceitual seja analisado por pesquisa quantitativa e de ampla cobertura para validação como modelo teórico.

## REFERÊNCIAS

- Albino, R. D., Souza, C. A. D., & Prado, E. P. V. (2014, Janeiro/Abril). Benefícios alcançados por meio de um modelo de gestão ágil de projetos em uma empresa de jogos eletrônicos. *Revista de Gestão e Projetos*, 5.
- Alliance, A. (2001). Principles: The Agile Alliance. Retrieved Feb 1, 2011, from <http://www.agilealliance.org/principles.html>
- Beck, K. (2001). Manifesto for Agile Software Development. Retrieved Nov 1, 2011, from <http://agilemanifesto.org/>

- Beck, K. (2004). *Extreme Programming explained: embrace change*. EUA: Addison-Wesley.
- Boehm, & Turner, R. (2003). *Balancing Agility and Discipline: A Guide for the Perplexed*. EUA: Addison-Wesley.
- Cockburn, A. (2002). *Agile software development*. EUA: Addison-Wesley.
- Cohn, M. (2005). *Agile Estimating and Planning*. EUA: Prentice Hall.
- Cooper, D., & Schindler, P. (2004). *Método de Pesquisa em Administração*. Porto Alegre: Bookman.
- Fowler, M. (2005). The New Methodology. Retrieved Feb, 2011, from <http://www.martinfowler.com/articles/newMethodology.html#PredictiveVersusAdaptive>
- Highsmith, J. (2004). *Agile software development ecosystems*. EUA: Addison-Wesley.
- Hiranabe, K. (2008). Kanban Applied to Software Development: from Agile to Lean. Retrieved Mar, 2011, from <http://www.infoq.com/articles/hiranabe-lean-agile-kanban>
- Jones, C. (2007). *Estimating Software Costs : Bringing Realism to Estimating: Bringing Realism to Estimating*: McGraw-hill.
- Jørgensen, M. (2002). A review of studies on expert estimation of software development effort. *The Journal of Systems and Software*, 70(2), 37-60.
- Jørgensen, M., & Shepperd, M. (2007). A Systematic Review of Software Development Cost Estimation Studies. *Software Engineering, IEEE Transactions on*, 33(1), 33-53. doi: 10.1109/TSE.2007.256943
- Keith, C., & Cohn, M. (2008). How to Fail With Agile: Twenty Tips to Help You Avoid Success. Retrieved Mar, 2011, from <http://www.mountangoatsoftware.com/articles/40-how-to-fail-with-agile>
- Kerzner, H. R. (2009). *Project Management: A Systems Approach to Planning, Scheduling, and Controlling* (10ª ed.): Wiley.
- Kniberg, H. (2007). *Scrum and XP from the Trenches: How We Do Scrum*. EUA: C4Media Inc.
- Larman, C., & Basili, V. R. (2003). Iterative and Incremental Development: A Brief History. *Computer*, 36, 47-56.
- Malhotra, N. K. (2006). *Pesquisa de Marketing: Uma Orientação*. Porto Alegre: Bookman.
- Mølken-Østfold, K., & Jørgensen, M. (2005). A comparison of software project overruns - flexible versus sequential development models. *Software Engineering, IEEE Transactions on*, 31(9), 754-766. doi: 10.1109/TSE.2005.96
- Paetsch, F., Eberlein, A., & Maurer, F. (2003). Requirements Engineering and Agile Software Development. In: Proceedings of the Twelfth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises. *IEEE Computer Society*, 308-324.
- PMI. (2009). *Um Guia do Conhecimento em Gerenciamento de Projetos* (4ª ed.). EUA: Project Management Institute.
- Poppendieck, M., & Poppendieck, T. (2006). *Implementing Lean Software Development: From Concept to Cash*. EUA: Addison-Wesley Professional.
- Schwaber, K. (2004). *Agile Project Management with Scrum*. EUA: Microsoft Press.
- Schwaber, K. (2010). Control Chaos. Retrieved Mar, 2011, from <http://www.controlchaos.com>
- Sheffield, J., & Lemétayer, J. (2013). Factors associated with the software development agility of successful projects. *International Journal of Project Management*, 31(3), 459-472. doi: Doi 10.1016/J.Ijproman.2012.09.011
- Waters, K. (2007). Agile Software Development Made Easy. Retrieved Mar, 2011, from <http://www.agile-software-development.com/2007/09/disadvantages-of-agile-software.html>