



Pós-Graduação em Ciência da Computação

RICKSON CÉSAR DA CRUZ SILVA

**“UM ESTUDO SOBRE A INFLUÊNCIA DE FATORES HUMANOS E
CULTURAIS EM PROJETOS DE DESENVOLVIMENTO DE SOFTWARE
ÁGEIS”**



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE
2017

RICKSON CÉSAR DA CRUZ SILVA

**“UM ESTUDO SOBRE A INFLUÊNCIA DE FATORES HUMANOS E
CULTURAIS EM PROJETOS DE DESENVOLVIMENTO DE SOFTWARE
ÁGEIS”**

Este trabalho foi apresentado ao programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre Profissional em Ciência da Computação.

Orientadora: Profª. Carina Frota Alves

RECIFE
2017

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S586a Silva, Rickson César da Cruz
Um estudo sobre a influência de fatores humanos e culturais em projetos de desenvolvimento de software ágeis / Rickson César da Cruz Silva. – 2017.
105 f.: il., fig., tab.

Orientadora: Carina Frota Alves.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2017.
Inclui referências e apêndices.

1. Engenharia de software. 2. Desenvolvimento de software. 3. Métodos ágeis. I. Alves, Carina Frota (orientadora). II. Título.

005.1 CDD (23. ed.) UFPE- MEI 2017-140

Rickson César da Cruz Silva

**Um Estudo sobre a Influência de Fatores Humanos e Culturais em
Projetos de Desenvolvimento de Software Ágeis**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre Profissional em 06 de março de 2017.

Aprovado em: 06/03/2017.

BANCA EXAMINADORA

Prof^a. Carla Taciana Lima Lourenço Silva Schuenemann
Centro de Informática / UFPE

Prof. George Augusto Valença Santos
Universidade Federal Rural de Pernambuco

Prof^a. Carina Frota Alves
Centro de Informática / UFPE
(Orientadora)

*Dedico este trabalho a Deus pai criador,
N.Sra.mãe que nunca desampara seus filhos ,
aos meus pais Elisiário e Maria José, à
minha esposa Paula Fernanda e aos
meus pequenos Vinícius e Henrique.*

Agradecimentos

Muitas pessoas contribuíram direta ou indiretamente para obtenção deste título, portanto, não posso deixar de agradecer:

Ao pai criador de infinita bondade, que sempre nos dá força, sabedoria, inteligência e saúde para suportar todos os dias de batalha, sob a sua proteção.

Aos meus pais, Elisiário e Maria José por sempre nos proporcionar o alicerce para toda e qualquer caminhada, não há palavras para agradecer.

À minha amada esposa, Paula Fernanda, pelo apoio constante e por ser fiel escudeira mesmo quando estive distante fisicamente, obrigado e Deus te abençoe.

Aos meus pequenos Vinícius e Henrique, por entenderem quando estava tão distante, não podendo participar das brincadeiras e diversões e dando apoio constante a finalização desse sonho.

Aos meus irmãos Tiago e Junior (in memoriam), que mesmo distante fisicamente a sua presença é constante nos nossos caminhos.

A reitoria do Instituto Federal de Sergipe por apoiar e incentivar os servidores, para qualificação e prestação de serviço com qualidade à comunidade acadêmica.

À orientadora Prof. Carina Frota, que mesmo à distância, conduziu de maneira ímpar esta caminhada, sempre presente para qualquer auxílio e incentivo, muito obrigado !

Aos amigos da turma MPROF2014, em especial aos companheiros de hospedagem Fabiano, Marcelo, Norton e Alex, pela companhia e incentivo durante a jornada de viagens.

Aos amigos do DTI-IFS, pelo apoio em todas as viagens !

Em fim, agradeço a todos que direta ou indiretamente contribuíram para este título, obrigado a todos!

Resumo

Ao longo de décadas, a demanda por software de qualidade aumentou consideravelmente. Com o rápido crescimento da indústria de software surgiu a necessidade de criar processos eficientes para que as organizações desenvolvedoras continuassem competitivas no mercado. Diversas opções de métodos foram criadas nas últimas décadas. Assim, os Métodos Ágeis (MA) emergiram como uma alternativa aos métodos tradicionais de desenvolvimento de software. Mais produtivos e adaptáveis, os MA têm sido amplamente usados para desenvolver software de diferentes domínios. Neste contexto, estudos recentes demonstram que a adoção de Métodos Ágeis apresenta novos desafios para as equipes de desenvolvimento de software. O objetivo desta dissertação é investigar a influência de fatores humanos e culturais no sucesso em projetos de software. Foi realizado um mapeamento sistemático (MS) como método de pesquisa, permitindo uma avaliação organizada e rigorosa dos artigos sobre o tema. Cinco questões de pesquisa foram propostas para identificar os artigos relevantes utilizando busca automática, manual e *snowballing*. Foi definida uma *string* de busca geral e *strings* de busca específicas para cada um dos quatro engenhos de pesquisa utilizados (ACM Digital Library, Elsevier ScienceDirect, IEEEXplore Digital Library, Springer Link). Os dados extraídos dos artigos selecionados foram processados e mapeados de maneira sistemática. Vinte e dois artigos foram avaliados segundo os critérios mencionados para responder às questões de pesquisa, de forma a proporcionar uma visão quantitativa geral de cada artigo, sendo identificadas informações sobre o tipo de busca, a distribuição por ano, a origem das publicações, os principais pesquisadores, o tipo de estudo e a quantidade de publicações por país. Para complementar os resultados do mapeamento sistemático, foi realizado um survey com 115 desenvolvedores de software ágil para entender como os fatores humanos e culturais afetam os projetos em que eles estão envolvidos. A principal contribuição deste trabalho é apresentar uma visão abrangente sobre a influência dos fatores humanos e culturais na área de desenvolvimento de software a partir dos Métodos Ágeis. Além disso, os resultados gerados por esta pesquisa poderão auxiliar o mercado em crescimento, bem como subsidiar pesquisas futuras na área.

Palavras-Chave: Mapeamento Sistemático. Survey. Métodos Ágeis. Desenvolvimento de Software.

Abstract

Over the decades, the demand for quality software has increased considerably. With the rapid growth of the software industry, the need arose to create efficient processes for the development organizations to remain competitive in the market. Several method options have been created and used since then. Thus, agile methods (AM) emerged as an alternative to replace traditional software development methods. More productive and adaptable, AMs have proven to meet market demands. In this context, recent studies have shown that the adoption of Agile Methods presents new challenges for software development teams. The objective of this work was to investigate the influence of human and cultural factors on the success of agile software projects. A systematic mapping was used as a research method, allowing an organized and rigorous evaluation of the articles on the subject. Five research questions were created that were used to identify the relevant works using automatic search, manual and snowballing. A general search string and search strings have been defined for each of the four search engines used (ACM Digital Library, Elsevier ScienceDirect, IEEEXplore Digital Library, Springer Link). Data extracted from the selected articles were processed and mapped in a systematic way. Twenty-two articles were evaluated according to the criteria to answer the research questions, in order to provide a qualitative overview of each article, identifying information about the type of search, the distribution per year, the origin of publications, the main researchers, the type of study and the number of publications by country. To complement the results of the systematic mapping, a survey with 115 developers of the agile software was made, looking for an understanding of how the human and cultural factors affect the project in which they are involved. The main contribution of this work is to present a comprehensive view on the human and cultural factors in the area of agile software. In addition, the results generated by this research may help the growing market, as well as support future research in this area.

Keywords: Systematic Mapping. Survey. Agile Methods. Software Development.

Lista de Figuras

Figura 1 - Resultados do relatório Chaos 2015	17
Figura 2 - Modelo Cascata.	21
Figura 3 - Valorização das relações humanas no trabalho	26
Figura 4 - Processo Scrum.....	34
Figura 5 - Processo XP.....	36
Figura 6 - Etapas do Mapeamento Sistemático.	43
Figura 7 - Fluxograma metodológico da presente dissertação de mestrado.....	48
Figura 8 - Número de artigos identificados para a seleção dos estudos primários.....	55

Lista de Quadros

Quadro 1 - Strings de busca básica desenvolvidas para esta dissertação de mestrado. 49

Quadro 2 - Strings específicas de busca para cada Engenho de busca utilizado. 50

Lista de Gráficos

Gráfico 1 - Distribuição dos artigos por ano.	58
Gráfico 2 - Origem das publicações.	58
Gráfico 3 - Distribuição quanto aos tipos de estudo.....	60
Gráfico 4 - Distribuição de publicações por país..	61
Gráfico 5 - Metodologias e princípios ágeis utilizados por artigo.	61
Gráfico 6 - Distribuição do número de artigos por questão de pesquisa.	62
Gráfico 7 - Tipo empresa etapa do survey.....	73
Gráfico 8 - Cargo/papel participante etapa do survey.	74
Gráfico 9 - Experiência participante etapa do survey.....	75
Gráfico 10 - Formação equipe participante etapa do survey.	76
Gráfico 11 - Abordagens utilizadas participante etapa do survey.	77
Gráfico 12 - Fatores sucesso (1) participante etapa do survey.	78
Gráfico 13 - Fatores sucesso (2) participante etapa do survey.	79
Gráfico 14 - Fatores humanos (1) participante etapa do survey.....	80
Gráfico 15 - Fatores humanos (2) participante etapa do survey.....	81
Gráfico 16 - Fatores organizacionais (1) participante etapa do survey.	82
Gráfico 17 - Fatores organizacionais (2) participante etapa do survey.	83

Lista de Tabelas

Tabela 1 - Resultado final da seleção dos artigos para o MS.....	55
Tabela 2 - Distribuição do número de artigos por engenho de busca.....	55
Tabela 3 - Apresentação geral dos artigos selecionados para esta pesquisa.....	56
Tabela 4 - Principais pesquisadores.....	59
Tabela 5 - Tipos de estudos identificados para o MS.....	59
Tabela 6 - Fatores críticos de sucesso em projetos ágeis identificados nos artigos	63
Tabela 7 - Desafios quanto aos aspectos técnicos	65
Tabela 8 - Desafios quanto aos aspectos organizacionais em projetos ágeis identificados	67
Tabela 9 - Desafios quanto aos aspectos humanos em projetos ágeis identificados.....	69

Lista de Abreviaturas

AM – *Agile methods*

ASD – *Agile Software Development*

CE – Critérios de exclusão

CI – Critérios de inclusão

FDD - *Feature Driven Development*

EPPI-Center – *Evidence for Policy and Practice Information and Co-ordinating Centre*

IID - *Iterative and Incremental Development*

MA – Métodos ágeis

MS – Mapeamento Sistemático

PDS – Projetos de Desenvolvimento de Software

QP – Questões de pesquisa

RSL – Revisão Sistemática da Literatura

TDD – *Test Driven Development*

XP – *Extreme Programming*

Sumário

1	INTRODUÇÃO	15
1.1	Motivações.....	16
1.2	Objetivos Gerais	18
1.2.1	<i>Objetivos Específicos.....</i>	18
1.3	Questões de Pesquisa.....	18
1.4	Estrutura da Dissertação	19
2	REFERENCIAL TEÓRICO	20
2.1	Contexto.....	20
2.2	Fatores humanos no desenvolvimento de software	23
2.3	Fatores culturais no desenvolvimento de software.....	27
2.4	Métodos Ágeis (MA)	31
2.4.1	<i>Scrum</i>	33
2.4.2	<i>Extreme Programming</i>	35
2.4.3	<i>Feature Driven Development (FDD)</i>	39
2.4.4	<i>TDD (Test-Driven Development)</i>	40
2.5	Resumo do Capítulo	41
3	MÉTODO DE PESQUISA	43
3.1	Mapeamento Sistemático	43
3.1.1	<i>Introdução ao Mapeamento Sistemático</i>	43
3.1.2	<i>Etapas do Mapeamento Sistemático (MS).....</i>	45
3.1.3	<i>Questões de Pesquisa (QP)</i>	47
3.1.4	<i>Estratégias de Busca</i>	48
3.1.5	<i>Critérios de Inclusão e Exclusão.....</i>	50
3.1.6	<i>Extração dos Dados e Síntese dos Estudos</i>	51
3.2	Survey com Desenvolvedores de Software Ágil	52
3.3	Ameaças à Validade	52
3.4	Resumo do Capítulo	53
4	RESULTADOS DO MAPEAMENTO SISTEMÁTICO	54
4.1	Sobre a Busca e Seleção dos Artigos.....	54

4.2	Caracterização dos Artigos Selecionados para Mapeamento Sistemático (MS) ..	57
4.3	Respostas às Questões da Pesquisa	62
4.3.1	<i>Visão geral das questões da pesquisa respondidas.....</i>	62
4.3.2	<i>QP1- Quais são os fatores críticos de sucesso em projetos de desenvolvimento de software ágeis?</i>	62
4.3.3	<i>QP1.1 - Quais são os desafios relacionados aos fatores técnicos?</i>	64
4.3.4	<i>QP1.2 - Quais são os desafios relacionados aos fatores culturais?</i>	66
4.3.5	<i>QP1.3 - Quais são os desafios relacionados aos fatores humanos?</i>	68
4.3.6	<i>QP2 - Como os fatores humanos e culturais influenciam o sucesso de projetos de desenvolvimento de software ágeis?</i>	69
4.4	Resumo do Capítulo	71
5	PESQUISA DE OPINIÃO COM PRATICANTES.....	72
5.1	Perfil dos Participantes	73
5.2	Resumo do capítulo	85
6	CONCLUSÃO.....	86
6.1	Análise Crítica dos Resultados.....	86
6.2	Principais Contribuições.....	88
6.3	Limitações e Dificuldades para a Realização da Pesquisa.....	89
6.4	Trabalhos Futuros	90
	REFERÊNCIAS	91
	APÊNDICE A	98
	APÊNDICE B.....	105

1 INTRODUÇÃO

Nas últimas décadas, enquanto as forças do mercado, os requisitos de sistemas, as tecnologias de implementação e perfil das equipes de desenvolvimento se transformavam de forma acelerada, um estilo diferente de desenvolvimento de software mostrou suas vantagens sobre o tradicional. Este estilo ágil de desenvolvimento aborda diretamente os problemas de mudança rápida. Uma ideia dominante no desenvolvimento ágil é que a equipe pode ser mais eficaz em responder às mudanças se reduzir o custo da movimentação de informações entre pessoas e reduzir o tempo decorrido entre tomar a decisão de ver as consequências dessa decisão (DYBÅ e DINGSOYR, 2008; COCKBURN, 2013).

Tanto no Brasil como em todo mundo, as metodologias ágeis demonstraram significativo avanço no uso acadêmico e industrial. As metodologias de desenvolvimento de software ágeis melhoram o desenvolvimento do projeto, incrementam a qualidade do produto e resultam em maior satisfação do cliente (MACHADO; MEDINA, 2009). Muitos estudos abordam esta área discutindo principalmente os benefícios encontrados ao utilizar abordagens ágeis. No entanto, ao fazer uma busca mais profunda percebe-se que tais estudos muitas vezes trazem resultados e colocações apenas quanto ao sucesso e falha de projetos, a qualidade em geral, ou mesmo a produtividade das equipes (SPECTOR, 2012, CAPRETZ; AHMED, 2010).

Para reduzir o custo da movimentação de informações entre as pessoas, a equipe ágil trabalha para colocar as pessoas fisicamente mais perto, substituir documentos por conversas e melhorar a amizade da equipe - seu senso moral e de comunidade - para que as pessoas estejam mais inclinadas a retransmitir informações valiosas rapidamente (COCKBURN, 2013).

O desenvolvimento de software é um processo permeado por relações humanas (PIRZADEH, 2010) e estudos recentes demonstram que habilidades sociais e traços psicológicos da equipe interferem no ciclo de vida do software, identificando traços de personalidade mais adequados para o êxito de um projeto nessa área (MERLI, 2008; CAPRETS, AHMED, 2010b).

Neste cenário, as grandes organizações e empresas vêm constatando a relevância do bem-estar físico, emocional e psicológico dos seus colaboradores e nos benefícios mútuos proporcionados para a equipe. Estudos apontam que os recursos humanos interferem no desenvolvimento de um projeto e podem criar vantagem competitiva no mercado, dependendo

da motivação dos envolvidos (PIRZADEH, 2010). A colaboração dos trabalhos durante o desenvolvimento do projeto pode ser crucial para o sucesso ou fracasso final do projeto. Assim, o investimento em recursos humanos e a compreensão acerca da cultura dos colaboradores também devem ser utilizados como estratégia gerencial, visando eficiência e produtividade (MERLI, 2008). Um alto índice de resultados positivos e de bem-estar organizacional é observado quando os colaboradores são a preocupação central da empresa, e esta questão destaca-se como o novo diferencial no mercado competitivo de software e como tendência em um futuro próximo.

Fatores humanos como a personalidade, as características individuais e os aspectos cognitivos vêm sendo investigados em estudos sobre o desempenho das equipes de software, com destaque para as pesquisas sobre interação, comunicação, relações interpessoais, motivação, personalidade, ambiente de trabalho, dentre outros (SPECTOR, 2012, CAPRETZ; AHMED, 2010).

Investigando a literatura atual sobre a área de desenvolvimento de software, notou-se que poucos estudos realizados pela comunidade acadêmica e profissional se voltam para uma análise mais profunda de como os fatores humanos e culturais impactam o sucesso de projetos ágeis.

1.1 Motivações

De acordo com as exigências do mercado de software, equipes de desenvolvimento de software precisam satisfazer demandas por produtos cada vez mais eficientes e com prazos de entrega cada vez mais curtos. Devido à esta realidade, as empresas de desenvolvimento de software vêm lidando com cenários desafiadores, os quais demandam processos mais eficientes e flexíveis para atender diferentes requisitos e exigências cada vez mais específicas de seus clientes.

Nesse sentido, os Métodos Ágeis surgem como uma alternativa mais adequada para as empresas em comparação aos métodos tradicionais, pois eles exigem um maior envolvimento dos participantes, mais transparência, flexibilidade, agilidade e empenho, sendo possível prever riscos com mais eficácia e entregar o produto ao cliente já corrigido. Dessa forma, existe mais

controle de custo, porque é gerado um maior valor de mercado e um menor custo para a empresa. O grande desafio está justamente na operacionalização do produto, além de ser necessário ter uma equipe realmente engajada no desenvolvimento do mesmo.

Métodos Ágeis têm como vantagem a redução significativa do tempo de entrega da primeira versão do software solicitado, dando ao cliente o tempo necessário para verificar se o que ele pediu foi realmente feito e se o produto funciona de forma eficaz. Existe, ainda, o aumento de controle por parte dos gestores já que o método se baseia no que realmente está sendo produzido (TOMÁS, 2009, p.10).

Ressaltamos na Figura 1, segundo relatório Chaos (2015) realizado pelo Standish Group a comparação entre os projetos que utilizam os MA e o modelo cascata de desenvolvimento. Os projetos de desenvolvimento de software ágeis têm 39% de sucesso enquanto os projetos cascata possuem apenas 11% de chances de sucesso.

Figura 1: Resultados do relatório Chaos 2015.

Método	Sucesso	Desafios	Fracasso
Ágil	39%	52%	9%
Cascata	11%	60%	29%

Fonte: Adaptado de Standish Group, relatório Chaos (2015).

Segundo Livermore (2007), os fatores humanos e organizacionais impactam diretamente na produtividade das equipes de desenvolvimento de software. A motivação para realização desta pesquisa baseia-se no fato de que as empresas já observam a importância do bem-estar físico, emocional e psicológico dos seus colaboradores enquanto instrumento de benefícios mútuos. Em um futuro próximo, o que diferenciará uma empresa da outra será a qualidade de vida de seus colaboradores, ressaltando a produtividade e a qualidade resultantes do bem-estar organizacional. Dessa forma, o nosso questionamento central é como os fatores humanos e culturais podem influenciar o desenvolvimento e consequente sucesso de projetos de software ágeis.

1.2 Objetivos Gerais

O presente trabalho tem o seguinte objetivo:

“Compreender o estado da arte sobre o tema: Fatores humanos e culturais em projetos de desenvolvimento de software ágeis.”

Para alcançar este objetivo, realizar-se-á um mapeamento sistemático (MS), conduzido a partir das recomendações propostas por Kitchenham (2004), sobre como os fatores humanos e culturais influenciam no sucesso de projetos de software ágeis. Além do MS, será realizado um survey com profissionais experientes em desenvolvimento de software ágil para entender como os fatores identificados impactam a rotina de trabalho deles.

1.2.1 Objetivos Específicos

Como objetivos específicos, tem-se:

- I. Conduzir um mapeamento sistemático para identificar quais fatores críticos de sucesso em projetos de software ágeis, relacionados aos aspectos técnicos, culturais e humanos;
- II. Executar um *survey* , através da coleta de dados com questionários sobre o contexto desta pesquisa na visão de especialistas;
- III. Demonstrar os resultados obtidos do MS com os obtidos do survey aplicados a um conjunto de especialistas em métodos ágeis.

1.3 Questões de Pesquisa

Este trabalho visa responder as seguintes questões de pesquisa:

QP1. Quais são os fatores críticos de sucesso em projetos de desenvolvimento de software ágeis?

QP1.1 Quais são os desafios relacionados aos fatores técnicos?

QP1.2 Quais são os desafios relacionados aos fatores culturais?

QP1.3 Quais são os desafios relacionados aos fatores humanos?

QP2. Como os fatores humanos e culturais influenciam o sucesso de projetos de desenvolvimento de software ágeis?

1.4 Estrutura da Dissertação

A presente dissertação organiza-se em seis capítulos: Introdução, Background, Método de Pesquisa, Resultados do Mapeamento Sistemático, Resultados do Survey com Especialistas e Considerações Finais. No Capítulo 1 desenvolve-se a introdução do trabalho, contextualizando o tema central, o problema, as motivações e os objetivos do estudo, e as questões de pesquisa que norteiam toda a investigação. Por sua vez, o Capítulo 2 descreve o contexto sobre os fatores humanos e culturais no desenvolvimento de software e apresenta uma visão geral sobre os Métodos Ágeis. No Capítulo 3, é apresentado o método de pesquisa utilizado. Os resultados obtidos através da execução do mapeamento sistemático estão presentes no Capítulo 4. No Capítulo 5, são descritos os dados obtidos a partir da aplicação do survey. Por fim, o Capítulo 6 apresenta as considerações finais sobre a investigação realizada, bem como as limitações e dificuldades para execução da pesquisa e a discussão crítica sobre os resultados encontrados.

2 REFERENCIAL TEÓRICO

2.1 Contexto

A atual fase da economia mundial, combinada com as novas demandas mercadológicas, vem exigindo das organizações a aplicação de novas metodologias de desenvolvimento de software, como uma das mais poderosas ferramentas de transformação das empresas. Diante desse cenário, as organizações precisam estar atentas e, mais do que isso, precisam estar preparadas para implementar estratégias mais céleres, como opção para tornar o seu desempenho organizacional mais dinâmico e eficiente (MATTOS, 2012).

Atualmente, inúmeras empresas estão aderindo à prática de implantar programas de qualidade, aperfeiçoamento, treinamento, dentre outros, visando a manutenção da sua competitividade. Estas organizações vêm procurando atender às novas demandas de um mercado cada vez mais exigente, em permanente transformação e à procura da excelência. Nos últimos anos têm-se analisado a importância das estratégias organizacionais eficientes para a fidelização de clientes e para o alcance de melhores rendimentos, dentre elas, o desenvolvimento de software.

O crescimento no mercado de engenharia de software trouxe a repetição dos passos das práticas que deram certo, partindo em seguida para a formalização dos modelos de ciclos de vida, onde podem ser consideradas como as principais fases, a definição de requisitos, análise, projeto, desenvolvimento, teste e implantação, sendo que o que diferencia um processo de software de outro é a ordem em que as fases ocorrem, assim como o tempo e a ênfase de dados de cada fase, as atividades presentes e os produtos entregues. (MACEDO, SPÍNOLA, s.d.) Ainda na visão dos autores, no processo de software a primeira escolha a ser feita é o modelo de ciclo de vida e, de acordo com essa escolha é definida a maneira mais adequada na obtenção da satisfação das necessidades do cliente, até quando e como ele receberá a primeira versão operacional do sistema.

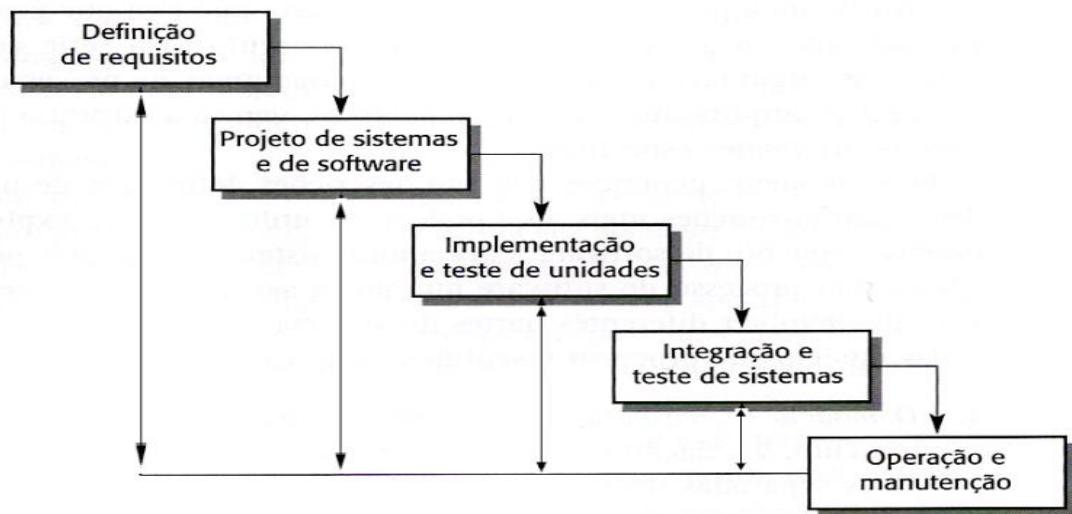
O ciclo de vida do software – esclarecendo que não existe um modelo ideal – deve se comportar de maneira sequencial (fases seguem determinada ordem) e/ou incremental (divisão de escopo) e/ou iterativa (retroalimentação de fases) e/ou evolutiva (software é aprimorado) (MACEDO, ESPÍNOLA, 2013.).

De acordo com Wazlawick (2013, s.p.) “As ideias fundamentais do Modelo Cascata são coerentes em uma primeira abordagem, e podem gerar benefícios relevantes.” O autor ainda

declara que as fases bem definidas ajudam a detectar erros rapidamente, permitindo sua imediata correção, de forma a barateá-lo, além de promover a estabilidade de requisitos, portanto, o projeto só segue em frente quando essas solicitações são aceitas. Além disso, o modelo funciona melhor quando os requisitos são bem conhecidos, se beneficiando de uma abordagem organizada e sistemática.

No modelo Cascata, por exemplo, podemos citar o ciclo de vida retratado na Figura 2, a seguir (MACEDO, SPÍNOLA,2013):

Figura 2: Modelo cascata



Fonte: Elaborado pelo Autor., 2016

Outro indicador positivo está no princípio de que a qualidade deve estar acima das preocupações com o custo ou tempo de desenvolvimento, uma vez que, não havendo mudanças no meio do caminho, a fonte de erros é minimizada, além de ser adequada quando a equipe que está a frente do projeto não precisa ser tão experiente (WAZLAWICK, 2013). Para o autor, o grande problema do projeto está no fato de que ele é totalmente planejado para ser executado sem testes preliminares, ou seja, ele sai do planejamento direto para a execução e, sendo assim, por mais que se torne mais fácil descobrir quando o projeto não funciona, é mais difícil descobrir onde está o erro. Para dar certo, este modelo necessita de ferramentas de análise automatizadas, diagramas e documentos para comprovar sua funcionalidade, porém, essas ferramentas ainda são muito limitadas.

O rápido crescimento da indústria de desenvolvimento de software, juntamente com o aumento da demanda por soluções cada vez mais robustas, as quais são executadas em

ambientes de negócios com considerável volatilidade, têm exigido que as organizações desenvolvedoras tenham processos mais eficientes para que continuem competitivas no mercado (DYBÅ e DINGSoYR, 2008). Por isso, diversos métodos, técnicas e ferramentas têm sido propostos e utilizados (COCKBURN, 2013).

Segundo Salgado (2011), as novas tecnologias, o processo de globalização e as mudanças repentinhas no contexto mundial colocam as empresas em situações de desafios constantes para o desenvolvimento de suas atividades. Por conta disso, cresce a necessidade de buscar metodologias diferenciadas para a melhoria do desempenho organizacional.

Por conseguinte, Zanini (2007) explica que, visando acompanhamento dos avanços tecnológicos, a engenharia de software se estendeu para diversas áreas organizacionais, revolucionando os métodos de gestão. Este movimento faz com que, potencialmente, ampliem-se as possibilidades de criar metodologias mais ágeis e obter maior produtividade, ultrapassando velhos paradigmas. Perboni (2013) acrescenta que o cenário mercadológico está marcado por constantes mudanças, reflexo da incerteza e da grande rapidez com que novas tecnologias são produzidas e as necessidades dos clientes são modificadas. Este contexto está repleto de imprevisibilidade, e por isso, necessita de uma resposta urgente do mundo empresarial, no sentido de que sejam priorizadas a flexibilidade e a adaptação. Essa busca por celeridade e eficiência demandou a adoção de estratégias para inovar os processos organizacionais, tais como os Métodos Ágeis.

Os MA têm tornado-se um modelo alternativo em substituição aos processos tradicionais de desenvolvimento de software. Diversas empresas estão adotando abordagens ágeis com o objetivo de orientar melhor seus esforços, garantindo, dessa forma, agilidade e flexibilidade. O interesse por MA se justifica a partir de diversas pesquisas. De acordo com um estudo realizado pelo *Standardish Group*, em 2013, um elevado número de empresas têm adotado essa nova metodologia, com crescimento em torno de 30%, entre 2010 e 2013. Sendo assim, neste capítulo, são apresentados abordagens no âmbito dos fatores humanos e culturais no desenvolvimento de software, e também os conceitos, definições e as principais especificidades acerca dos Métodos Ágeis.

2.2 Fatores humanos no desenvolvimento de software

A partir da década de 1990, foram realizados estudos buscando investigar a influência de fatores humanos, em especial, o comportamento individual, os aspectos cognitivos e sociais no desenvolvimento de software (JACINTO, 2010). Pesquisas sobre essa temática revelam alguns fatores humanos relacionados à engenharia de software, tais como: personalidade, motivação, trabalho de equipe e comunicação.

Para Oliveira (2011) a nova gestão de recursos humanos é decorrente da valorização dos fatores psicológicos e sociais envolvidos na produtividade, trazendo novo sentido ao gerir pessoas. A partir de então, passou-se a considerar no ambiente de trabalho competências e habilidades, comunicação, motivação, liderança, e desenvolvimento de equipe. Nesse sentido, a ênfase do fator humano nas organizações veio contribuir para novas estratégias para fazer com que os trabalhadores sejam vistos não mais como meros recursos, mas sim como colaboradores.

Quinn et al. (2009) revela que a implantação da gestão com intuito de avaliar os fatores humanos que interferem no desenvolvimento de um projeto cria vantagem competitiva sustentável, uma vez que está enraizada nos colaboradores, e não em recursos físicos. De igual modo, Merli (2008) enfatiza que o investimento nas pessoas deve ser uma estratégia gerencial adotada pelas instituições preocupadas com os processos gerenciais, tendo como principal propósito, que a empresa possa atingir com eficiência seus objetivos. Por essa razão, as empresas precisam ter cuidado ao mapear competências e habilidades humanas e organizacionais, garantindo maior competitividade no mercado em constante evolução.

No aspecto personalidade, Capretz e Ahmed (2010b) mapearam habilidades sociais e traços psicológicos com fator humano que interferem no ciclo de vida do software. Esses estudiosos creem que há traços de personalidade mais adequados para o êxito de um projeto. Encontramos confluência em Jacinto (2010, p. 17) que, ao analisar os estudos de Darcy; Ma (2005) e Mourmant; Gallivan (2007), chegou a algumas conclusões importantes sobre o tema.

Em primeiro lugar, o autor afirma que há uma relação entre as diferenças individuais (tais como a personalidade, os fatores de domínio específico, a demografia, habilidade e experiência de programação) e desempenho do programador. Mourmant e Gallivan (2007), acreditam que o perfil dos profissionais de TI tem impacto direto satisfação com o trabalho e nas mudança de emprego, isso porque os perfis individuais são decisivos na escolha por

processos de trabalho, o que garante, inclusive, a rotatividade de funções dentro de um ambiente profissional.

Estudos focalizados nas habilidades sociais apontam que o desenvolvimento da personalidade está diretamente associado aos fatores genético-individuais, bem como às experiências individuais (PAPALIA; OLDS; FELDMAN, 2009; GOMIDE, 2010). Pinheiro et al (2006) explica que esses fatores acrescidos da dificuldade de as pessoas administrarem os seus problemas cotidianos foram essenciais para a elaboração de programas específicos, com a finalidade de promover um desenvolvimento mais adaptativo aos problemas comportamentais no ambiente organizacional.

Assim, o que se observa nos últimos anos é uma ampliação dos estudos que analisam a influência dos fatores humanos, como a personalidade, as características individuais e aspectos cognitivos no desempenho das equipes de software, destacando-se as pesquisas sobre interação, comunicação, relações interpessoais, motivação, a personalidade, o ambiente de trabalho, dentre outros (SPECTOR, 2012, CAPRETZ; AHMED, 2010).

Pesquisadores sinalizam, ainda, a importância da interação no desempenho em engenharia de software, destacando, sobretudo, o trabalho em equipe, a interação das pessoas com as tecnologias, as capacidades de resolução de problemas, os aspectos cognitivos e os relacionamentos interpessoais (CAPRETZ; AHMED, 2010).

A valorização das relações humanas no ambiente profissional foi decorrente da valorização dos fatores psicológicos e sociais envolvidos na produtividade, trazendo novo sentido à gestão de pessoas. Nesse sentido, a descoberta do fator humano na empresa veio contribuir para novas estratégias para fazer com que os trabalhadores sejam vistos, não mais como meros recursos, mas como colaboradores (RIBEIRO, 2009). A tendência atual é fazer com que todas as pessoas, em qualquer nível dentro da organização, sejam administradoras e não simplesmente executores de suas tarefas. Nesse contexto, a gestão de pessoas tem como objetivo principal, segundo Rocha-Pinto *et al* (2007), “ajudar os colaboradores a descobrir quais são seus interesses de realização, proporcionando, também as condições necessárias para que o trabalho seja executado em consonância com esses interesses, disseminando à cooperação entre as pessoas (p.33).

Para Hermam (2008, p.57), o processo de socialização tende a direcionar a vontade e os valores intrínsecos do indivíduo, que desde sua infância incorpora normas e valores vigentes na família, em seus pares, na sociedade, enfim, estabelecendo funções instrumentais,

consumatórias, ajustadoras ou utilitárias, ego-defensiva, expressão de valores, de controle e apropriando-se dos fatores condicionantes de uma sociedade.

Diante dessa nova realidade, as empresas precisam estar atentas e, mais do que isso, precisam estar preparadas para desenvolver as habilidades dos seus colaboradores, tornando-os talentosos, pois estes devem saber utilizar, consciente e corretamente, todas as possibilidades que o mundo moderno traz, como opção para tornar o seu desempenho mais dinâmico (XAVIER, 2009).

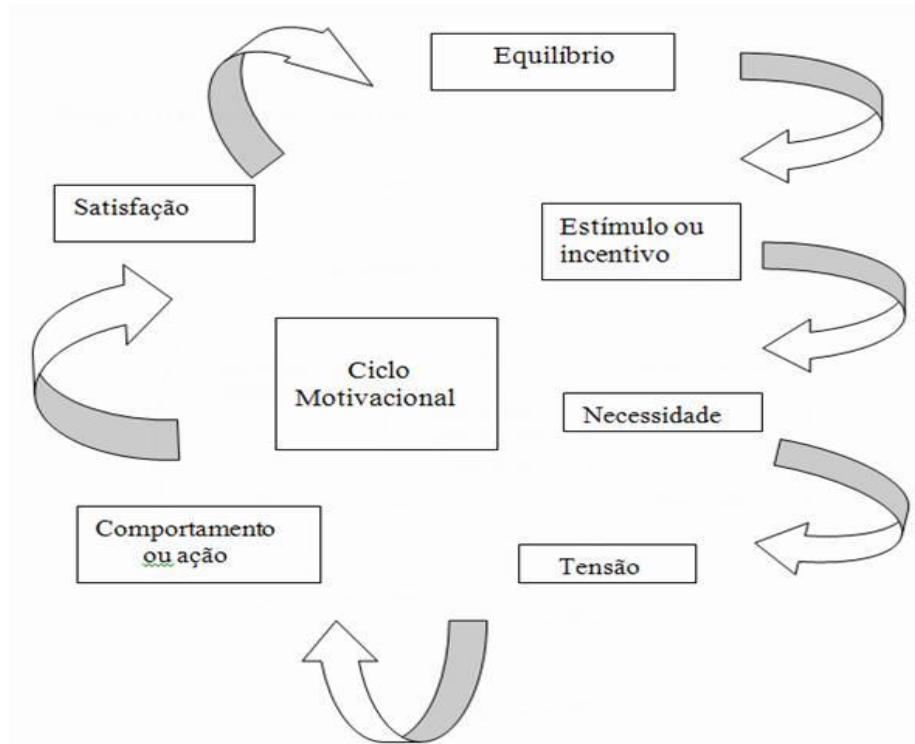
Bittencourt (2012) entende que as competências e as habilidades são conceitos diferentes, embora similares. Para esse autor, competência diz respeito às tarefas diretivas ou profissionais, enquanto habilidades são utilizadas para as tarefas técnicas. Carbone et al (2011) esclarecem que, no desenvolvimento de software, a competência tem o caráter de associar atributos pessoais ao contexto em que são desenvolvidos, isto é, ao ambiente e ao trabalho que a pessoa realiza.

Nesse sentido, Bitencourt (2012) entende ser importante ter como base as reflexões que privilegiam o coletivo (práticas dependentes das pessoas), o desenvolvimento contínuo (visão processual), e a ênfase em ações (práticas de trabalho). Ou seja, por meio da aprendizagem organizacional é possível resgatar questões-chave para o desenvolvimento das competências e que devem ser repensadas visando minimizar as dificuldades de articulação de um software. Para tanto, sugere-se reflexões acerca:

- Do *gap* entre desenvolvimento de competências individuais e coletivas – dificuldade em articular experiências e aprendizagens individuais no âmbito organizacional no intuito de promover competências coletivas;
- Da resistência em se enxergar as competências não como eventos isolados ou programas pontuais, mas como uma estratégia articulada voltada ao desenvolvimento das pessoas e da organização;
- Da preocupação excessiva com a criação de listas de atributos, instrumentos de avaliação e metodologias complexas privilegiando o produto (instrumento de análise) e perdendo o foco na importância do processo (desenvolvimento articulado de competências com base nos diversos programas e estratégias organizacionais) (BITENCOURTT, 2012, p. 12).

Outro autor que estuda as competências e habilidades e sua importância no cenário organizacional é Ubeda (2008), mostrando que a competência enfoca a gestão de recursos humanos, o que representa, na atualidade a valorização das relações humanas no trabalho.

Figura 3 - Valorização das relações humanas no trabalho



Fonte: UBEDA, (2008).

Nesta mesma direção, Pirzadeh (2010) também explicita que os fatores humanos atuam diretamente no desempenho do profissional, afetando-o de diferentes formas, através da interação de fatores organizacionais e interpessoais, associados ainda às características individuais.

No entendimento de Cockburn (2002) o fator humano no desenvolvimento de software influencia a formação de times, focando-se na interação das pessoas e nos indivíduos. Na literatura analisada, os outros fatores humanos envolvidos no desenvolvimento de software: *comunicação, cordialidade, talento e habilidades* (COCKBURN; HIGHSIMTH, 2002), que interagem com a estrutura organizacional e de comportamento individual na organização (MOSCOVICI, 2014). Além disso, esses fatores se entrelaçam no grupo de indivíduos, quando trabalhando junto como um time (COCKBURN, 2002).

No caso específico dos Métodos Ágeis, a literatura mostra que tais abordagens representam um apoio estratégico para que as organizações consigam alcançar maior

desempenho no desenvolvimento de seus sistemas de software com resultados positivos e sucesso. Estudiosos adeptos de MA apontam para os fatores humanos como geradores de interferência na aplicabilidade de tais métodos, esses pesquisadores entendem que problemas nos recursos humanos são capazes de interferir negativamente no cumprimento dos objetivos da empresa, apresentando dificuldades em atender às normas e aos procedimentos que visam corrigir falhas, atingir o mais alto padrão de qualidade e conduzir ao sucesso organizacional (MOE, DINGSØYR, 2008; BECK, 2004; TELES, 2007; PIRZADEH, 2010; COCKBURN, 2002; HIGHSMITH, 2000).

Conforme aprofundamento nas referências teóricas as metodologias utilizadas enfatizam os aspecto humano como potencial grande relevância para impactar nas várias etapas da construção do produto. Conforme descreve Moe e Dingsøyr (2008), há complexidade no desenvolvimento de um software, especialmente, pela interferência das questões humanas no resultado final, uma vez que a efetividade da construção de software depende do desempenho da equipe, e esta sofre influência de diversos fatores, tais como: relações interpessoais, motivação, satisfação, habilidades, competências, comunicação e trabalho em equipe. Isso reforça a relevância em apresentar este estudo científico, com abordagens em métodos ágeis, para demonstrar uma metodologia usual em diversos projetos.

2.3 Fatores culturais no desenvolvimento de software

Segundo Pirzadeh (2010) o desenvolvimento de um software centra-se, notadamente, no ser humano e, por isso, é necessário considerar alguns fatores que o influenciam, como os fatores organizacionais, os interpessoais e as características individuais. O autor descreve esses fatores como categorias:

- Individual: Abrange questões humanas individuais relacionadas à engenharia ou desenvolvimento de software, como características individuais, personalidade, cultura, dentre outros.
- Interpessoal: É relacionada a fatores humanos entre os indivíduos que afetam ou são afetados pela Engenharia ou processo de desenvolvimento de software, como cooperação, aprendizado em grupo, trabalho dos times, dentre outros.
- Organizacional: Inclui fatores humanos relacionados às organizações e ambientes de trabalho, como tomada de decisão nas organizações, consultores, cultura organizacional e outros.

Quando se analisa os fatores descritos anteriormente em comparação aos estudos verificados neste referencial teórico, pode-se descrever o primeiro fator, Individual, como um fator importante para se multiplicar à outros profissionais, através de métodos de cooperação e motivação da gestão de conhecimento entre os indivíduos. Isso reflete na criatividade do profissional que a partir do conhecimento de novas culturas e personalidades, o produto pode refletir um desenvolvimento com aspectos de qualidade e uma tendência para produzir de forma ágil.

Para o segundo fator, Interpessoal, destaca-se justamente o método relatado anteriormente para o fortalecimento individual, haja vista que, neste segundo fator, evidencia claramente a aprendizagem em grupo, trabalho em equipe, dentre outros. Porém, é necessário que o aspecto de engajamento seja fortalecido na questão cultural, para solidificar este valioso aspecto muito presente nas metodologias ágeis. Em harmonia a necessidade de fortalecimento para este fator interpessoal, pode se destacar a forma de comunicação entre os envolvidos no desenvolvimento de software, utilizando práticas de gestão a vista e outras formas para a interação e evolução de todo engajamento da equipe.

Assim, o fator Organizacional, sendo o terceiro aspecto significativo para as questões culturais humanas, traz em seu conceito a tomada de decisão como um ponto importante para o crescimento e motivação na engenharia de software. Com isso, o produto em desenvolvimento necessitará das partes interessadas da organização totalmente envolvidas nas etapas convenientes da engenharia de software. Demonstrando assim uma cultura integrada e significativa para o estudo em questão.

Para Silva (2010), as organizações mais bem sucedidas não só investem em máquinas e equipamentos, como também adotam culturas flexíveis, sensíveis as diferenças sociais e culturais de seus colaboradores, principalmente, quando eles atuam em mercados competitivos. Desse modo, em uma cultura organizacional, faz-se imprescindível os aspectos culturais e civilizatórios de todo o sistema organizacional, a fim de que seja possível oferecer alternativas para problemas inerentes aos processos produtivos, visando o enfrentamento dos desafios do atual mercado.

A cultura organizacional é, portanto, “[...] a capacidade de gestores de suscitar, reforçar ou modificar valores, atitudes e crenças consideradas eficazes através do uso de ritos, cerimônias, símbolos e mitos apropriados.” (FLEURY, 2011, p. 35). Isto é, ela é moldada pelos

colaboradores de uma empresa, a partir de uma perspectiva de interação, visando unir pensamentos e ações, tendo como foco os esforços na busca da mudança organizacional, proporcionando um cenário positivo para a organização.

É importante salientar que ao entender a cultura de uma determinada organização, precisa-se levar em consideração os aspectos culturais e civilizatórios do sistema como um todo, sendo possível oferecer alternativas para crises e problemas inerentes de um sistema que está recursivamente enfrentando os desafios da constante mutação. Pelo exposto, é possível compreender que o fator humano é o conjugado de implicações subjetivas de todos colaboradores, a partir de uma interação satisfatória com a estrutura organizacional, bem como o estilo de liderança dos administradores, que tem influência direta nos relacionamentos interpessoais.

Observa-se que as relações interpessoais desenvolvem-se em decorrência do processo de interação, como bem expressa Moscovici (2011). Desse modo, os gestores podem utilizar estratégias de comunicação para melhorar o desempenho dos seus colaboradores. Uma vez que os bons relacionamentos e os canais de comunicação mais fluentes são essenciais para a construção de um ambiente de trabalho em termos de produtividade, desempenho, comunicação e motivação.

Alguns estudos apontam a falha na comunicação multidisciplinar como principal e maior gerador de conflito (MOSCOVICI, 2014; MEDEIROS, 2002; MACIEL, CAMARGO, 2011). O ato de comunicar é inerente ao ser humano e se faz presente no cotidiano familiar, social, profissional e assistencial através do estabelecimento de contatos (LORENZINI et al, 2015; SANTOS, 2012). Apesar de ser imprescindível o ato de comunicar nas relações humanas, estudos mostram que a comunicação ainda se constitui uma realidade não compreendida em vários contextos sociais (LEITE, 2012). Ao se analisar os artigos sobre a comunicação, no campo do desenvolvimento de software, verificaram-se a comunicação como uma interação entre profissionais, visando à promoção de vínculos no ambiente profissional (CIAMPONE; KURCGANT, 2010).

É por meio de uma comunicação eficiente que a equipe de trabalho pode interagir somar, dividir, ampliar conhecimentos, tornando-a bem informado para que possa exercer sua função com mais dinamismo e resolver problemas (MACIEL, CAMARGO, 2011). Sendo assim, a verdadeira comunicação se dá pela qualidade das relações estabelecidas, assim como

pela credibilidade que cada pessoa manifesta para com a organização. Portanto, a comunicação equilibra o trabalho, uma vez que é ela que permeia possibilidades, ao mesmo tempo em que oferece alicerce na consolidação do sucesso (MOSCOVICI, 2014).

Outro fator gerador de conflito é a estrutura organizacional. Essa estrutura engloba todos os aspectos da organização formal, e inclui a divisão do trabalho, especialização e hierarquia. Classicamente são definidas na estratégia organizacional, pela gerência e diretoria (PAULA, 2007). No caso do desenvolvimento de software, quando essas estruturas não favorecem o desenvolvimento das atividades, podem vir a existir conflitos entre membros da equipe (PAULA, 2007), o que requer um novo perfil gerencial no qual o gestor adote uma postura mais flexível, comunicativa e democrática (SILVA, 2012; OLIVEIRA, 2010). Tem-se, também, o comportamento individual nas organizações como um dos fatores que mais apareceram nos trabalhos analisados. Toda e qualquer organização depende de pessoas e, como bem enfatiza Chiavenato (2014, p. 99), embora se possa ver as pessoas como recursos, [...]... portadoras de habilidades, capacidades, conhecimento, competências, motivação de trabalho [...] etc., nunca se deve esquecer que as pessoas são pessoas [...] portadoras de características de personalidades[...]. Assim sendo, é esperado que em uma organização de trabalho advenham conflitos oriundos do comportamento individual, ora relacionados ao processo de trabalho, ora relacionados às dificuldades de relacionamentos interpessoais (SILVA, 2012; RODRIGUEZ; TREVIZAN; SHINYASHIKI, 2008), que geram desequilíbrios na equipe de trabalho, bem como perda de qualidade de vida, baixa produtividade, aumento de estresse, conflitos e baixa motivação. Tudo isso gera prejuízo a médio e longo prazo no desenvolvimento de um software.

Esta falta de interação das pessoas (profissionalmente), fatalmente impedirá o alcance das metas e objetivos de qualquer serviço, pois há deficiência na comunicação (MEDEIROS, 2002; SILVA, 2012; CIAMPONE; KURCGANT, 2010). Atualmente, o investimento em estratégias comunicacionais representa um diferencial nos modelos gerenciais tradicionais, pois trata-se de uma ferramenta que visa orientar esforços para planejar, desenvolver e avaliar os diferentes níveis de uma organização, identificando as competências dos seus colaboradores à consecução de seus objetivos organizacionais (TOLEDO; MILIONI, 2010).

Portanto, a fim de alcançar um comprometimento de seus colaboradores no desenvolvimento de software e interação com os objetivos organizacionais, as empresas precisam ter uma cultura organizacional sólida e orientada à colaboração. Os fatores culturais

também precisam ser ponderados no desenvolvimento de software a fim de: “promover mudanças no comportamento das pessoas, apesar das diferenças individuais, são fundamentais para que as pessoas e as organizações não permaneçam estáticas diante de um cenário em constantes obstáculos e oportunidades” (CLARO; TORRES, 2012, p. 23).

2.4 Métodos Ágeis (MA)

Até o início da década de 1990 os sistemas de software existentes eram considerados pouco eficazes e, por isso, passaram a ser questionados em virtude de sua efetividade e praticidade (HIGHSMITH, 2002). A partir de meados dessa década, a indústria sofreu transformações que culminaram com o aparecimento de métodos alternativos aos tradicionais.

Conforme Highsmith et al (2002), especialistas passaram a criar métodos próprios adaptáveis às constantes mudanças mercadológicas. A partir de aprimoramentos desses mecanismos surgiram os MA de Desenvolvimento de Software. Em 2001, a partir do manifesto *for Agile Software Development* (ASD) (em tradução para o português: manifesto ágil) criado por profissionais experientes na área, foi oficializada a origem dos Métodos Ágeis. No manifesto, foram descritos valores fundamentais a serem seguidos no desenvolvimento ágil de software, dentre os principais (AGILE MANIFESTO, 2001):

- Indivíduos e interações são mais importantes do que processos e ferramentas, ou seja, as relações e a comunicação entre os atores de um projeto de *Software* são o melhor recurso do projeto;
- *Software* funcionando é mais importante do que documentação completa e detalhada. Manter o código simples e tecnicamente avançado, reduzindo ao mínimo a documentação;
- Colaboração com o cliente é mais importante do que negociação dos contratos. O contato direto proporciona melhores resultados do que relações contratuais;
- Adaptação a mudanças é mais importante do que seguir o plano inicial. O desenvolvimento de *Software* deve estar pronto para responder às mudanças e a equipe deve ser autorizada a sugerir modificações no projeto a qualquer momento.

Machado e Medina (2009) explicam que os MA são compostos por técnicas adaptáveis no desenvolvimento de software, abrangendo custos, prazos e requisitos do cliente. Para Highsmith (2000), estes métodos são uma resposta para atender as exigências constantes do mercado.

As metodologias ágeis possuem alguns princípios que norteiam sua aplicabilidade, compreendendo fases mais específicas e com maior facilidade de inserção na prática de implementação dos métodos ágeis (AGILE ALLIANCE, 2014). Os princípios são:

- A maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de software de valor;
- Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento;
- Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas;
- Entregar software funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos;
- Pessoas relacionadas a negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto;
- Construir projetos ao redor de indivíduos motivados, dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho;
- O método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa face a face;
- *Software* funcional é a medida primária de progresso;
- Processos ágeis promovem um ambiente sustentável;
- Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes;
- Contínua atenção à excelência técnica e bom *design* aumenta a agilidade;
- Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito;
- As melhores arquiteturas, requisitos e *designs* emergem de times auto organizáveis;
- Em intervalos regulares, o time reflete como ficar mais efetivo, então, ajustam e aperfeiçoam seu comportamento.

Estes princípios; em conjunto com o manifesto ágil; visam auxiliar no desenvolvimento ágil de software. Adicionalmente, o manifesto não rejeita processos e ferramentas, documentação, negociação de contratos e planejamentos. Ele simplesmente quer mostrar que estas características têm importância secundária durante o processo de construção do produto (FOWLER; HIGHSITH, 2001). Os principais representantes dos MA são Scrum e XP (Extreme Programming), através de suas metodologias de gerenciamento ágil de projetos de desenvolvimento de software. Nas subseções a seguir apresentamos uma breve descrição destes métodos.

2.4.1 Scrum

O termo “*Scrum*” deriva de uma estratégia no jogo de *rugby*, desenvolvido por Jeff Sutherland, John Scumniotales e Jeff McKenna na década de 1990, na *Easel Corpotion* (SUTHERLAND, 2005). Este *framework* mantém uma lista de funcionalidade, que é implementada durante o processo de desenvolvimento de software, chamada de *Product Backlog*.

Os ciclos curtos são chamados de *Sprints*, nos quais divide-se o desenvolvimento em intervalos de tempos entre duas a quatro semanas. Em seguida, é feita uma reunião inicial de planejamento, chamada de *Sprint Planning Meeting*, onde os itens da lista são priorizados de acordo com as necessidades do cliente - e não necessariamente quem paga pelo produto.

A equipe de desenvolvimento, isto é, o *Scrum Team*, define quais funcionalidades poderão ser atendidas dentro da interação, de acordo com sua capacidade, ao atribuir uma “estimativa” para cada funcionalidade a fim de transformar o *Product Backlog* em incremento de funcionalidades da *Sprint* (SCHWABER, 2004). A Figura 4 ilustra o processo do *Scrum*.

Figura 4 - Processo Scrum.



Fonte: SCHWABER (2004).

O processo de desenvolvimento *Scrum* é composto por quatro fases: planejamento, preparação, desenvolvimento e entrega. Durante o desenvolvimento das *Sprints*, diariamente o time faz uma reunião de 15 minutos para acompanhar o progresso do trabalho e agendar outros encontros necessários. Nessa reunião diária (*Daily Scrum Meeting*), cada membro do time responde três perguntas básicas (LARMAN, 2004):

- O que eu fiz no projeto desde a última reunião?
- O que irei fazer até a próxima reunião?
- Quais são os impedimentos?

No último dia da *Sprint*, realiza-se uma reunião com o cliente que poderá dar novo direcionamento ao projeto. Esta reunião é chamada de reunião de revisão (*Sprint Review*). Em seguida à reunião de revisão, é feita uma reunião de retrospectiva (*Sprint Retrospective*), cuja avaliação será sobre o que foi aprendido e sobre os ajustes necessários ao projeto, objetivando a melhoria contínua.

Dentre as principais vantagens do processo de *Scrum*, podemos citar a agilidade no desenvolvimento do produto, principalmente devido aos ciclos curtos de planejamento e revisão, que garantem que o software seja desenvolvido por etapas de forma eficaz, e não a partir de um planejamento a longo prazo que não sofre adaptações ou modificações que são muito úteis para adequação do produto final ao que foi solicitado. O processo também garante que haja uma maior motivação da equipe que, a cada reunião diária, deve repensar se a sua participação no projeto está sendo satisfatória. Uma outra vantagem do *Scrum* é a redução de *bugs* no software, isso porque, devido às constantes reuniões de equipe, principalmente à *Sprint*

Review e à *Sprint Retrospective*, que acontecem no final de cada ciclo, é possível avaliar se o produto é eficaz e se está sendo desenvolvido conforme as solicitações do cliente, além de promover a observação de eventuais erros e estimular a sua correção antes do início do próximo ciclo.

Apesar das valiosas vantagens do processo de *Scrum*, também podemos encontrar alguns pontos negativos, principalmente no que se refere à desordem nas funções de cada membro da equipe, visto que os papéis de cada um não são definidos. Isso, apesar de ser bom por garantir uma visão geral do processo pelos envolvidos no desenvolvimento do software, pode gerar alguns problemas relacionados à comunicação interna e, consequentemente, promover uma confusão na realização das tarefas.

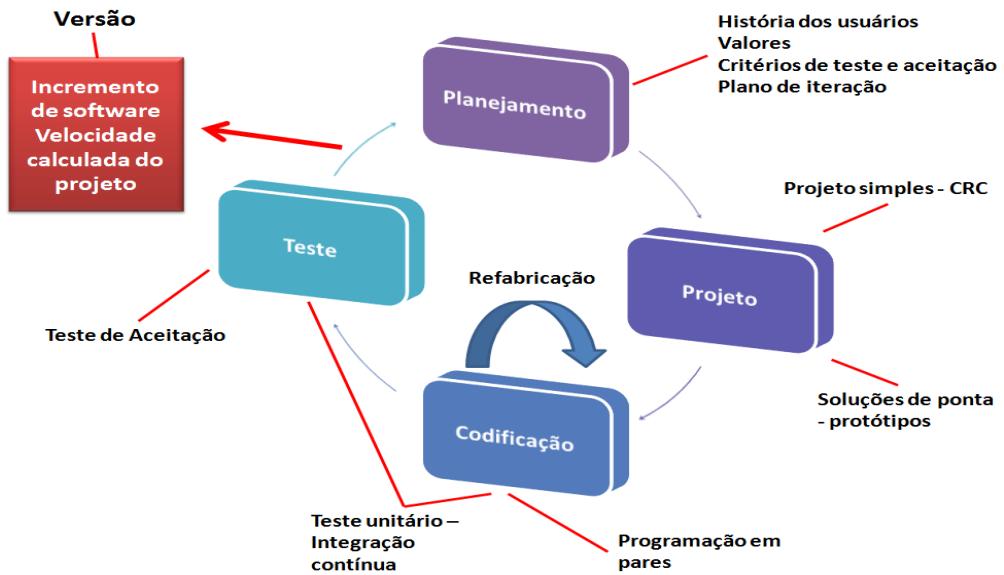
Outro ponto que pode vir a se tornar um problema é a ausência de documentação sobre o andamento do projeto. É fundamental que os aspectos importantes das etapas de desenvolvimento sejam registrados para evitar confusões ao lidar com determinado problema que tenha acontecido em algum momento do processo. Com as informações documentadas, a discussão sadia entre os membros da equipe se torna possível e mais confiável.

2.4.2 *Extreme Programming*

Outra metodologia de desenvolvimento que representa os MA é o *Extreme Programming* (XP), criado através de experiências de desenvolvimento de software em *Smalltalk* (Linguagem de programação orientada a objetos), no início da década de 1990, e passou a atrair o interesse das áreas acadêmicas e industriais (TELES, 2007).

O XP permite a equipes ágeis criarem frequentemente versões de software que possuem características e funcionalidades descritas e priorizadas pelo cliente, organizado em ciclos curtos de *feedback*, conforme ilustrado na Figura 4.

Figura 5 - Processo XP.



Fonte: PRESSMAN (2010).

Observa-se na Figura que o desenvolvimento do XP possibilita a criação de software de alta qualidade, de forma ágil e flexível, aliando o planejamento, o projeto, a codificação e o teste de forma a simplificar e organizar o trabalho. Para tanto, combina, como enfatiza Teles (2007, p. 39), “técnicas comprovadamente eficazes e reduz o risco dos projetos desenvolvendo software de forma iterativa e reavaliando permanentemente as prioridades dos usuários”.

As práticas e regras da XP são conduzidas por quatro valores (BECK *et al.*, 2001):

1. Comunicação: busca manter o melhor relacionamento possível entre clientes e desenvolvedores, preferindo conversas pessoais;
2. Simplicidade: busca implementar o software com o menor número possível de classes e métodos. Utiliza apenas requisitos atuais, evitando, assim, adicionar funcionalidades que podem ser importantes apenas no futuro;
3. Feedback: o desenvolvedor terá informações constantes do código e do cliente;
4. Coragem: sabe-se que não são todas as pessoas que possuem facilidade de comunicação e têm bom relacionamento interpessoal, é preciso coragem para obter e cobrar constantemente um feedback do cliente.

O conjunto de práticas em XP é (BECK, 1999):

- Planejamento: é feito por duas partes, nas quais o pessoal de negócios e o pessoal técnico dialogam entre si para decidir o que é primordial no projeto;
- Desenvolvimento iterativo: deve-se planejar *releases* curtos, contendo os requisitos mais valiosos;
- Estórias: modo de melhorar a compreensão entre clientes e desenvolvedores para que sejam capazes de estabelecer um vocabulário de fácil compreensão, sem termos técnicos;
- *Design* simples: utilizar o *design* mais simples possível, mas que atenda às necessidades, de forma a evitar problemas para outros desenvolvedores e para o cliente. Ao conseguir alcançar os objetivos do *design* simples, certamente será possível prevenir o futuro incerto do projeto e ajudar em aspectos do processo;
- Teste antecipado: o teste se torna parte do processo e torna o *software* muito mais confiável. Isso não significa que deve-se fazer um teste para cada método implementado, e sim para os métodos que podem causar alguma quebra. Os desenvolvedores realizam os testes unitários e, os clientes, os testes de aceitação;
- Refatoração: disciplina de reestruturar o código para melhorar o *design* do projeto e facilitar o trabalho futuro, seja em cima de correções de *bug*, seja com novas funcionalidades que aparecem e que se integram com o resto do código;
- Programação em pares: desenvolvimento feito em dupla em um computador. Cada desenvolvedor tem um papel na programação. Enquanto o que está no comando do computador pensa em como fazer da melhor forma um método, o outro está pensando na estratégia (se o que está sendo feito irá funcionar, casos de erros, formas de simplificar o código);
- Propriedade coletiva do código: todos têm responsabilidade por todo o sistema, de modo que todos conhecem alguma coisa sobre todas as partes do projeto. Todos têm obrigação de melhorar o código caso haja necessidade. Esta prática tira a dependência entre o código e as pessoas;
- Integração contínua: integrar pedaços do sistema, testá-lo e corrigir possíveis problemas é mais fácil do que integrar tudo em uma única vez. Sem esse trabalho contínuo, fica mais difícil descobrir de onde veio o erro e mais complicado de realizar os testes;

- 40 horas na semana: esta prática é questão de eficiência. Trabalhar demais torna a mente cansada e o trabalho fica ineficaz, sem criatividade, e sem cuidados;
- Envolvimento dos clientes: a presença e a colaboração do cliente no projeto são essenciais para responder dúvidas da equipe de desenvolvimento, resolvendo problemas do sistema e trilhando as prioridades dos requisitos; e,
- Padrão de código: deve melhorar a comunicação e diminuir o retrabalho. Com a prática da refatoração e da propriedade conjunta do código, isto fica cada vez mais necessário, visto que se não for dessa maneira, não haverá um *design* simples.

Portanto, XP é um método ágil para times de tamanho pequeno a médio (em relação ao quantitativo de pessoas no grupo) e que desenvolvem software tendo em vista que os requisitos são vagos e que se modificam de forma rápida (BECK, 2004). Desta forma, XP procura valorizar os aspectos gerais dos MA que são ter uma boa relação com as mudanças nos projetos, e o desenvolvimento iterativo e incremental do produto, à medida que os requisitos mudam.

Devido à participação frequente do cliente em todo o processo XP, o desenvolvimento do software atende de maneira mais satisfatória ao que foi pedido, resultando em um produto final com características muito próximas do plano inicial do cliente. Outra vantagem desse mecanismo é a redução no número de erros que o software finalizado pode vir a apresentar, em primeiro lugar devido ao conceito de programação em pares, no qual o projeto é sempre guiado por dois desenvolvedores e, assim, as chances de pontos que necessitam de alteração serem notados é muito maiores. Em segundo lugar, devido à grande quantidade e variedade de testes que são continuamente realizados, de forma a corrigir os erros fragmentadamente, e não de forma centralizada, o que garante um resultado final muito mais limpo e consistente.

Em contrapartida, no desenvolvimento de software usando XP pode ocasionar uma perda de produtividade por parte da empresa desenvolvedora, isso porque os programadores trabalharão sempre em duplas e isso pode comprometer os resultados quantitativos da organização ao final de um ciclo de trabalho. Da mesma forma, o cliente deve sempre disponibilizar uma equipe para participar do processo de desenvolvimento do software, o que pode representar perda de pessoal e de tempo dispendido nessas reuniões. Além disso, todos os envolvidos no projeto devem conhecer técnicas e regras, requerendo uma grande maturidade da equipe desenvolvedora.

É importante ressaltar que a adoção de processos de desenvolvimento como o *Scrum* e o *XP*, que são categorizados como Métodos Ágeis, configura em um Fator Crítico de Sucesso – FCS para a indústria do desenvolvimento de software. Tais abordagens refletem a capacidade de inovação e de integração com as mais novas tendências da Engenharia de Software, garantindo a diferenciação de uma empresa e a consequente refinação do seu potencial competitivo.

Nesse sentido, entender de que forma esses mecanismos podem ser adotados como FCSs para potencializar as chances de sucesso na indústria do software se torna fundamental e decisivo. Evidências disso foram indicadas no Relatório Chaos de 2015, que demonstra o aumento do sucesso dos projetos de desenvolvimento de software nos últimos cinco anos, apontando a utilização dos MAs como fator motivador destes resultados.

2.4.3 *Feature Driven Development (FDD)*

Feature Driven Development foi desenvolvido por Jeff de Luca e Peter Code em 1997, foi elaborado com foco na programação, entrega frequente da versão do produto para os clientes e na utilização de boas práticas durante o ciclo de seu desenvolvimento.

FDD utiliza práticas que não são extremamente rígidas, mas que são básicas para sua definição, sendo elas (BLAHA e RUMBAUGH, 2006):

1. Plano de projeto: documento principal de saída a ser aprovado pelo cliente, nele está definido o escopo, a lista de funcionalidades, os riscos, as métricas para controle do projeto, os critérios de aceitação, dentre outras informações pertencentes ao domínio da aplicação;
2. Lista de funcionalidades: usada para planejar, dirigir, rastrear e reportar o progresso do projeto e está organizada hierarquicamente com os requisitos funcionais;
3. Diagrama de sequência: estabelece um diagrama de classes principal com os objetos de domínio e suas relações, definindo assim, uma arquitetura básica para o modelo do sistema;
4. Inspeções: forma de verificação da qualidade do código e do projeto;
5. Integração (*build*) regular: existe um determinado período de tempo fixo no qual devem ser integradas as características já terminadas, permitindo a verificação de erros e também criando uma versão do produto que pode ser entregue ao cliente;
6. Gerência de configuração: manter versões de todos os artefatos criados.

Tais práticas são incorporadas nas seguintes fases; concepção e planejamento com os processos, desenvolver um modelo abrangente, construir lista de funcionalidades, planejar por funcionalidades e na fase de construção: detalhar por funcionalidade, construir por funcionalidade, por ser uma metodologia bem definida e objetiva, com seus processos integrados.

2.4.4 TDD (*Test-Driven Development*)

Desenvolvimento Guiado por Testes, tradução do termo em inglês *Test-Driven Development* (TDD), é uma das práticas sugeridas pela Programação Extrema (XP). A prática é baseada em um ciclo, no qual o desenvolvedor escreve um teste antes de implementar a funcionalidade desejada e, depois, com o código passando no teste criado, refatora-lo para remover possíveis duplicação de dados e de código (Aniche, 2012).

Simplicidade deve ser também algo intrínseco ao processo; o praticante busca escrever o teste mais simples que falhe e escrever a implementação mais simples que faça o teste passar. Esse ciclo é também conhecido como "Vermelho-Verde-e-refatora" ("Red Green Refactor"), uma vez que lembra as cores que um desenvolvedor normalmente vê quando faz TDD: o vermelho significa que o teste está falhando, e o verde que o teste foi executado com sucesso (Beck, 2004).

Uma consequência da prática de TDD é a bateria de testes de unidade gerada, auxiliando o desenvolvedor a evitar erros de regressão, em que a implementação de uma nova funcionalidade quebra uma outra funcionalidade já existente no sistema. Essa bateria também provê segurança durante as constantes refatorações de código que são feitas durante o processo de desenvolvimento. (Teles, 2004).

A quantidade de código coberto pelos testes também tende a ser alta, uma vez que o desenvolvedor deve sempre escrever um teste antes de implementar uma nova funcionalidade (Aniche, 2012).

2.5 Resumo do Capítulo

Neste capítulo, descrevemos como o rápido crescimento da indústria de software, bem como o aumento de demandas cada vez mais específicas e que exigem grande adaptabilidade e agilidade por parte das empresas de desenvolvimento, ocasionaram uma crescente busca por novas técnicas para aumentar a eficiência na realização de projetos. Nesse contexto, os Métodos Ágeis surgiram como uma alternativa aos mecanismos tradicionais, solidificados no início deste século através do Manifesto Ágil, de 2001, que descreve o papel da interação humana como o melhor recurso em um projeto, defende a necessidade de simplicidade do código e a redução de documentações burocráticas, indica o contato com o cliente como fator de grande importância para o sucesso do produto final, e garante que a adaptabilidade e a abertura a possíveis mudanças é fundamental para um bom processo de desenvolvimento, dentre outros pontos.

Também fizemos uma breve explanação acerca dos princípios dos processos de *Scrum* e de *XP*, dois dos maiores representantes dos MAs, sendo o *Scrum* marcado por ciclos curtos, chamados de *Sprints*, que duram de duas a quatro semanas, pelas constantes reuniões entre a equipe, onde os membros do time avaliam diariamente a sua participação no desenvolvimento do produto, e pelas reuniões de revisão com o cliente, que garantem a adequação do produto às solicitações e uma redução de erros e *bugs* devido aos testes contínuos e frequentes. Já o *XP* segue quatro valores fundamentais: comunicação, simplicidade, *feedback* e coragem, sendo que sua prática é marcada, principalmente, pela grande participação do cliente no projeto, pela preferência por um *design* simples aliado à eficiência, pela programação em pares de desenvolvedores, o que contribui para a percepção de erros e melhorias integradas e contínuas no software, e pela propriedade coletiva do código, a partir da qual todos os envolvidos têm responsabilidade pelo sistema e conhecimento de todas as etapas.

Vimos que adoção dos MAs configura um Fator Crítico de Sucesso para a indústria de desenvolvimento de software, sendo que indicações dos benefícios dessa nova metodologia já foram demonstradas por meio de estudos e relatórios. Alguns pesquisadores defendem que, para uma gestão mais eficiente dos projetos de software ágeis, é importante que as empresas de desenvolvimento observem de que forma os fatores humanos e culturais influenciam esses processos, sendo estes fatores a personalidade, as características individuais, a interação entre equipe e a comunicação dentro da cultura organizacional, dentre outros. Dedicamos algumas páginas, portanto, para o estudo das teorias que garantem que a interferência desses aspectos

está diretamente relacionada à produtividade, ao bom desempenho e à motivação dos colaboradores, agindo, consequentemente, sobre a qualidade do projeto desenvolvido.

3 MÉTODO DE PESQUISA

3.1 Mapeamento Sistemático

3.1.1 *Introdução ao Mapeamento Sistemático*

O mapeamento sistemático (MS) é uma metodologia que é frequentemente utilizada na pesquisa médica, mas que, nos últimos anos, tem sido amplamente adotada na área de engenharia de software. Exemplos de mapeamento sistemático incluem estudos na área de gerenciamento de projetos no desenvolvimento distribuído de software (COSTA, 2014) e mecanismos para guiar estudos empíricos em ES (ALMEIDA; BARREIROS; SARAIVA; SOARES, 2011), dentre outros.

O estudo a partir da elaboração de um MS fornece uma visão tipológica e categorizada dos resultados publicados em determinada área do conhecimento, estruturada através de um resumo visual. Mapeamentos sistemáticos em engenharia de software são recomendados principalmente para áreas de pesquisa onde há falta de estudos primários relevantes e de alta qualidade para condução de uma revisão sistemática da literatura mais completa (KITCHENHAM, et al., 2007).

A partir do MS, se realiza uma revisão mais ampla dos estudos primários em busca de identificar quais evidências estão disponíveis, bem como identificar lacunas no conjunto onde seja direcionado o foco de revisões sistemáticas futuras, e identificar áreas onde mais estudos precisam ser conduzidos (KITCHENHAM, 2004). O estudo de mapeamento sistemático fornece uma visão geral de uma área de pesquisa através da qual é possível identificar tendências, identificando a quantidade, os tipos de pesquisas realizadas, os resultados disponíveis, além das freqüências de publicações ao longo do tempo (PETERSEN et al., 2008). Portanto, o MS mostrou ser o método científico mais adequado para atingir os objetivos deste estudo, uma vez que buscamos identificar de que forma os fatores humanos e culturais influenciam na adoção de Métodos Ágeis para o desenvolvimento de software, de forma a fomentar a discussão acerca do assunto.

Kitchenham e Charters (2007) definem uma revisão sistemática da literatura (RSL) “como um meio de identificar, avaliar e interpretar todas as evidências disponíveis relevantes para uma questão específica, área temática, ou fenômeno de interesse”. E mapeamento sistemático da literatura, “como uma ampla revisão dos estudos primários em um tema específico que visa identificar as evidências disponíveis sobre o tema”.

MS é um método científico que avalia e interpreta todos os estudos disponíveis

referentes a uma ou várias questões, área de pesquisa ou fenômeno (BUDGEN, 2008), identificando as evidências disponíveis no tema e também as lacunas no campo de pesquisa estudado. Este método utiliza a mesma base metodológica do método de pesquisa RSL, mas propõe-se a identificar toda literatura disponível com questões abrangentes e exploratórias, ao invés de abordar questões específicas como os estudos de RSL convencionais (BUDGEN, 2008).

Os estudos de mapeamento são de grande importância para o meio acadêmico, pois fornecem uma visão geral da literatura (KITCHENHAM, BUDGEN, & BRERETON, 2010). Eles podem ser utilizados para identificar literatura disponível antes de realizar RSLs convencionais (KITCHENHAM B. e, 2010). Petersen et. al. (2008) evidenciam que o método de MS é recomendado diante da ausência de estudos primários de alta qualidade e relevância sobre um determinado tema.

A metodologia para MS foi originalmente desenvolvida pelo Centro de Informação e Coordenação de Evidências para Políticas e Práticas (do inglês *Evidence for Policy and Practice Information and Co-ordinating Centre, EPPI-Center*) (PEERSMAN, 1996; OAKLEY et al., 2005). Esses mapas sistemáticos, às vezes denominados "mapas descritivos", são frequentemente usados em um modelo de revisão sistemática em dois estágios como meio de caracterizar inicialmente a base de evidências, seguido pela identificação de subconjuntos menores de estudos que podem ser usados para responder questões específicas através de uma revisão sistemática (PEERSMAN, 1996; OAKLEY et al., 2005; JAMES et al., 2016).

A criação de revisões sistemáticas é um método estabelecido na medicina baseada em evidências. Com o passar dos anos, desmembrou-se em RSL e MS. Em geral, o MS visa obter uma apanhando geral de um determinado tema e dentre seus objetivos podemos destacar:

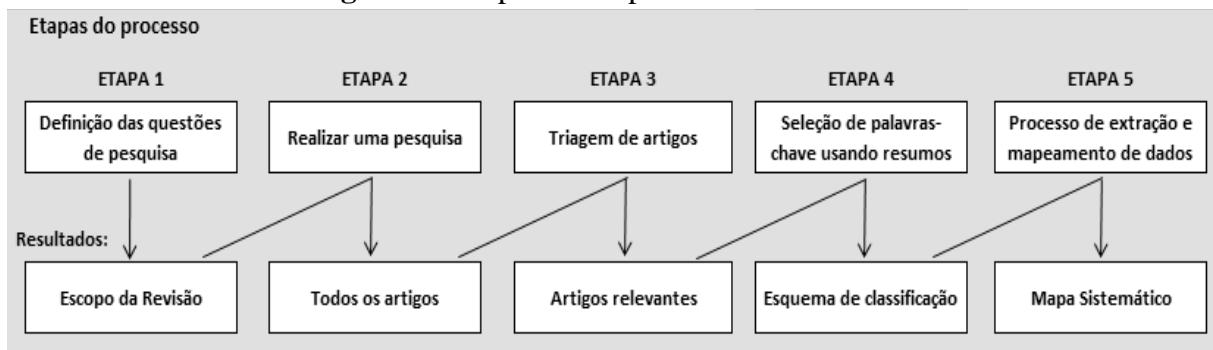
- Criar um esquema de classificação e
- Estruturar um campo de interesse;
- Estruturar o tipo de pesquisa e resultados por categorização de um campo;
- Mostrar frequências de publicações para categorias no esquema;
- Determinar a cobertura em um determinado campo;
- Combinar os resultados para responder a perguntas de pesquisa mais específicas;
- Fornecer um resumo visual mapeando os resultados.

3.1.2 Etapas do Mapeamento Sistemático (MS)

As etapas essenciais do nosso estudo de MS são resumidamente: a definição de questões de pesquisa, a realização de busca de artigos relevantes, a triagem de artigos, o uso de palavras-chave de resumos, a extração e mapeamento de dados. Cada etapa do processo tem uma discussão, sendo o resultado final do processo o mapa sistemático (PETERSEN et al., 2008).

O mapeamento sistemático realizado nesta dissertação seguiu as recomendações disponíveis em Petersen et al., (2008) como referência teórica. A Figura 6 demonstra detalhadamente as etapas do MS.

Figura 6 - Etapas do mapeamento sistemático.



Fonte: Adaptado de PETERSEN et al., (2008).

Segundo Petersen et al., (2008) o mapeamento sistemático está dividido basicamente em 5 etapas

Etapa 1 – Definição das questões de pesquisa: Escopo da Revisão

O principal objetivo de um estudo de mapeamento sistemático é fornecer uma visão geral de uma área de pesquisa, identificar a quantidade, o tipo de pesquisa e os resultados disponíveis nela. Para isso, é fundamental começar com questionamentos relevantes sobre a área pesquisada.

Etapa 2 - Realizar uma pesquisa: Todos os artigos

Assim, os estudos primários são identificados através da utilização de *strings* de busca em bases de dados científicas, de navegação manual, ou publicações de periódicos. Uma boa maneira de criar a sequência de pesquisa é estruturá-los em termos de população, intervenção, comparação e resultado (KITCHENHAM et al., 2007). A estrutura deve, obviamente, ser conduzida pelas questões de pesquisa. Palavras-chave para a sequência de pesquisa podem ser anotadas e utilizadas de cada aspecto identificado nas perguntas do MS.

Etapa 3 - Triagem de artigos: Artigos relevantes

Os critérios de inclusão e exclusão são utilizados para selecionar estudos que são relevantes para responder às perguntas da pesquisa. As questões de pesquisa influenciam os critérios de inclusão e exclusão, levando a classificação apenas de trabalhos relevantes.

Etapa 4 – Seleção de palavras-chave usando resumo: Esquema de classificação

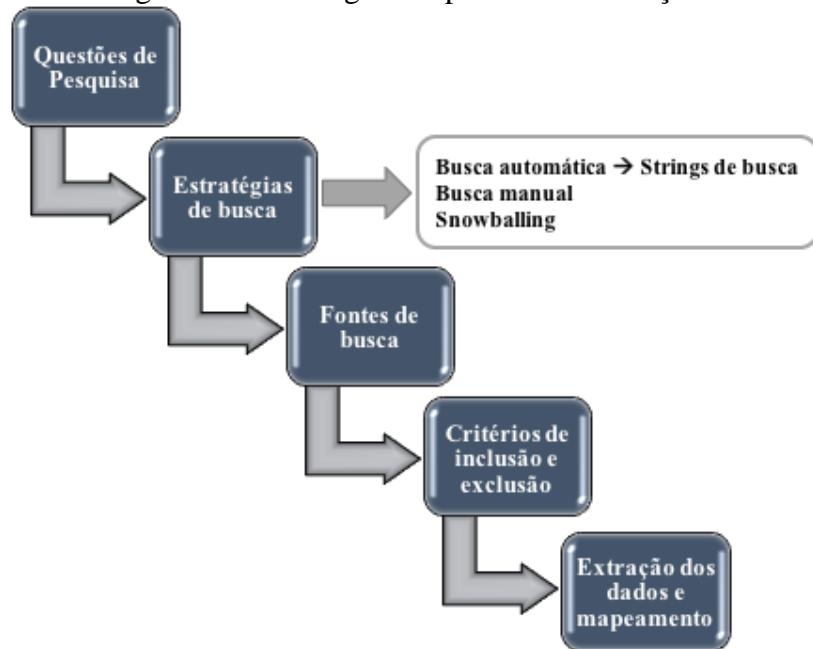
Atribuir palavras-chave é uma maneira de reduzir o tempo necessário para desenvolver o esquema de classificação e garantir que o esquema leva em consideração os estudos existentes. Assim, a atribuição de palavras-chave pode ser realizada em duas fases. Primeiro, os pesquisadores devem ler os resumos e procurar palavras-chave e conceitos que refletem a contribuição do artigo. Ao fazê-lo, também é identificado o contexto da pesquisa. Quando isso é feito, o conjunto de palavras-chave de diferentes artigos são combinados em conjunto para desenvolver um alto nível de compreensão sobre a natureza e a contribuição da pesquisa. Isso ajuda os pesquisadores a criarem um conjunto de categorias representativas da população subjacente. Quando os resumos são de má qualidade para permitir que palavras-chave significativas sejam escolhidas, os pesquisadores podem optar por estudar também as seções de introdução ou conclusão do artigo. Quando um conjunto final de palavras-chave foi escolhido, as palavras podem ser agrupadas e usadas para formar as categorias para o mapa.

Etapa 5 – Processo de extração e mapeamento de dados: Mapa Sistemático

Os artigos relevantes são organizados para a extração de dados em uma tabela, a partir da qual a construção de um esquema de classificação evolui, seja pela adição de novas categorias ou pela mesclagem e divisão de categorias já existentes. Em seguida, a análise dos resultados concentra-se na apresentação das frequências das publicações para cada categoria. Isso possibilita ver quais categorias foram enfatizadas em pesquisas anteriores e, assim, identificar lacunas e possibilidades de pesquisa futura. Ao final, os dados extraídos dos trabalhos relevantes são transformados em esquemas visuais como gráficos e tabelas.

Na presente dissertação, de maneira específica, o MS foi desenvolvido a partir das etapas relacionadas na Figura 7, a seguir:

Figura 7 - Fluxograma metodológico da presente dissertação de mestrado.



Fonte: Elaborado pelo Autor, 2016.

3.1.3 Questões de Pesquisa (QP)

Segundo as motivações para o desenvolvimento desta pesquisa e com a comprovação da sua relevância, realizou-se a primeira etapa do MS que consiste na definição das questões de pesquisa para guiar todo o processo de investigação. As questões definidas nesta seção foram utilizadas como alicerce durante todo o desenvolvimento desta dissertação.

A criação das questões visa gerar uma visão abrangente sobre os fatores críticos de sucesso, e em particular, os fatores humanos e culturais que afetam projetos de desenvolvimento de software ágeis. As questões elaboradas são descritas a seguir, e foram desenvolvidas a partir das motivações descritas no Capítulo 1:

QP1 Quais são os fatores críticos de sucesso em projetos de desenvolvimento de software ágeis?

QP1.1 Quais são os desafios relacionados aos fatores técnicos?

QP1.2 Quais são os desafios relacionados aos fatores culturais?

QP1.3 Quais são os desafios relacionados aos fatores humanos?

QP2. Como os fatores humanos e culturais influenciam o sucesso de projetos de desenvolvimento de software ágeis?

Nossas questões de pesquisa baseiam-se, sobretudo, nos princípios de que os fatores técnicos, humanos e culturais podem vir a configurar os Fatores Críticos de Sucesso das empresas desenvolvedoras de software. De acordo com Livermore (2007), as características pessoais, bem como a cultura organizacional de uma entidade são fatores que atuam diretamente nos resultados obtidos a curto prazo e à produtividade da equipe.

Bullen (1981) define os FCSs como investimentos em áreas que asseguram o bom desempenho e a competitividade das organizações. Ora, se a qualidade de vida dos colaboradores e o bem-estar organizacional são decisivos na obtenção de produtos de qualidade, tal como já foi discutido e defendido por autores como Merli (2008), Oliveira (2011) e Ribeiro (2009), entre outros, nada mais justo do que classificá-los como FCSs dentro do planejamento estratégico de uma empresa de desenvolvimento. De tal forma, as QPs formuladas nesse estudo pretendem subsidiar as discussões acerca do potencial dos fatores humanos e culturais enquanto FCSs na estrutura empresarial do software, de forma a analisar criticamente quais os limites dessa influência e quais os desafios encontrados na adoção de políticas de desenvolvimento pessoal e organizacional dentro da indústria do software.

3.1.4 Estratégias de Busca

Dependendo da dimensão e complexidade do MS, diversas estratégias de busca podem ser utilizadas. A seguir apresentaremos as três modalidades de busca utilizadas na realização desta pesquisa.

A busca automática é definida como a pesquisa de trabalhos primários de maneira automatizada utilizando uma *string* de busca em um número definido de engenhos de pesquisa. A criação de uma *string* de busca é dependente do referencial teórico e contexto da pesquisa, englobando termos que respondam às questões de pesquisa.

A busca manual é realizada sem o auxílio de ferramentas de automatização. Os trabalhos primários são pesquisados em periódicos, meios científicos ou conferências através da leitura diretamente na fonte. A busca manual pode ser útil para que o pesquisador faça uma revisão ampla que aborde o que foi desenvolvido relacionado a área investigada. Para a presente dissertação foi realizada a busca manual e identificado artigos relevantes das seguintes fontes: Computers in Human Behavior, Journal of Systems and Software, International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), International Conference

on Information and Communication Systems (ISICS), estes artigos foram identificados conforme Tabela 3.

A estratégia de busca *snowballing*, conhecido no Brasil como “bola de neve” ou, ainda, como cadeia de informantes, trabalha com amostra não probabilística, onde os componentes iniciais de um estudo indicam novos componentes que por sua vez indicam novos integrantes e assim sucessivamente, até que se alcance o “ponto de saturação”. (PENROD, et al 2003). Dentro de um grupo de artigos pré-definidos pelo tema principal da pesquisa, buscam-se por citações internas dentro do grupo. Ao se aplicar essa técnica de busca, o número inicial de artigos irá depender da abrangência da pesquisa a ser realizada. Por outro lado, é importante selecionar diferentes publicações para essa busca, utilizando um conjunto existente de palavras-chave que auxiliem a compor um bom grupo inicial de trabalhos. A técnica de snowballing foi aplicada no conjunto de referências dos artigos relevantes retornados da busca automática, ou seja, os artigos que passaram na pré-seleção. Para escolher os artigos através do snowballing também foram lidos apenas o título e o resumo de cada artigo referenciado.

A fim de conduzir a pesquisa eletrônica, quatro bases de dados na área da ciência da computação foram selecionadas como fonte de busca. A saber:

- ACM Digital Library (<http://portal.acm.org>)
- Elsevier ScienceDirect (<http://www.sciencedirect.com>)
- IEEEXplore Digital Library (<http://ieeexplore.ieee.org>)
- Springer Link (<http://link.springer.com>)

Posteriormente à definição dos engenhos de busca, foram construídas as *strings* (KITCHENHAM; CHARTERS, 2007; KITCHENHAM E TRAVASSOS, 2007), de maneira a representar as questões para esta pesquisa. Segundo Kitchenham (2007), a *string* de busca deverá ser desenvolvida com base nos termos centrais identificados em cada questão da pesquisa, utilizando-se operadores booleanos como “AND” e “OR”, conforme os Quadro 1 e 2. A pesquisa nos engenhos de busca selecionados na seção anterior foi, então, realizada.

Quadro 1 - Strings de busca básica desenvolvidas para esta dissertação de mestrado.

String Básica de Busca
(“Software” AND (agile OR agility OR scrum OR "XP" OR "extreme programming" OR fdd OR "feature-driven development" OR "featuredriven" OR tdd OR "test-driven development" OR "test-driven" OR lean OR kanban) AND ("challenge" OR"success"))

Fonte: Elaborado pelo Autor, 2016.

Como cada fonte/engenho de busca apresenta sintática própria, foi necessário adequar cada *string* de acordo com o engenho e segundo suas especificações, conforme o Quadro 2.

Quadro 2 - Strings específicas de busca para cada Engenho de busca utilizado.

Engenho de busca	Strings específicas para cada engenho de busca
ACM – Digital Library	"Software"AND (+agile +OR +agility +OR +scrum +OR +XP +OR +extreme +programming +OR +fdd +OR +feature-driven +development +OR +featuredriven +OR +tdd +OR +test-driven +development +OR +test-driven +OR +lean +OR +kanban +AND (challenge OR success)).
Elsevier ScienceDirect	"Software"AND (agile OR agility OR scrum OR exp extreme programming OR fdd OR feature-driven development featured OR tdd OR test-driven development OR test-driven OR lean OR kanban) AND (challenge OR success)) AND LIMIT-TO(cids, "271539,271629","Information and Software Technology,Journal of Systems and Software") AND LIMIT-TO(topics, "model,system,agile").
IEEEExplore Digital Library	"Software"AND (agile OR agility OR scrum OR "XP" OR "extreme programming" OR fdd OR "feature-driven development" OR "featuredriven" OR tdd OR "test-driven development" OR "test-driven" OR lean OR kanban) AND ("challenge" OR "success")
Springer Link	"Software"AND("agile" OR "agility" OR "scrum" OR "XP" OR "extreme programming" OR "fdd" OR "feature-driven development" OR "featuredriven" OR "tdd" OR "test-driven development" OR "test-driven" OR "lean" OR "kanban") AND("challenge" OR "success")

Fonte: Elaborado pelo Autor, 2016.

3.1.5 Critérios de Inclusão e Exclusão

Após a definição da *string* de busca nas bases de dados selecionadas na seção anterior, foram aplicados os critérios de inclusão e exclusão afim de selecionar os artigos para o MS. Objetivou-se ampliar a qualidade dos dados utilizados para esta pesquisa, realizando uma leitura completa e crítica de cada artigo, levando em consideração os critérios de inclusão e exclusão descritos a seguir. Por fim, um conjunto final de artigos que passaram pela avaliação

dos critérios dessa seção, foram submetidos à extração e análise dos dados. Como Critérios de Inclusão (CI) considerou-se:

CI1 – Trabalhos científicos que abordem em seus objetivos, hipóteses, perguntas de pesquisa ou que analise em seus resultados fatores humanos/culturais/técnicos e/ou métodos ágeis;

CI2 – Artigos que respondam pelo menos uma das questões de pesquisa;

CI3 – Artigos completos indexados.

Como Critérios de Exclusão (CE) foram considerados os seguintes:

CE1 – Artigos em outra língua que não o inglês;

CE2 – Artigos cuja área do conhecimento não seja ciência da computação;

CE3 – Artigos cujo tema principal não seja referente ao tema dessa pesquisa;

CE4 - Artigos repetidos;

CE5 – Artigos semelhantes (onde apenas o artigo mais completo e recente entre os dois foi incluído no MS);

CE6 – Artigos não disponíveis ao nosso acesso através do login institucional.

CE7 – Artigos publicados fora do período entre janeiro de 2001 (criação do manifesto ágil) e Setembro de 2016 (período elaboração do mapeamento).

3.1.6 Extração dos Dados e Síntese dos Estudos

Após realizar as etapas anteriores, os artigos relevantes foram classificados para a extração de dados. Para documentar o processo de extração de dados, usamos uma planilha do Excel descrevendo cada categoria do sistema de classificação. Com a inserção dos dados de um artigo na planilha, era sistematizada uma visão qualitativa de cada documento, com informações sobre tipo de busca, engenho de busca, distribuição por ano, origem das publicações, principais pesquisadores, tipo de estudo e quantidade de publicações por país. Além disso, os artigos

foram distribuídos em diferentes categorias - distribuição esta devidamente justificada em cada caso. A partir da análise da planilha final, as frequências das publicações em cada categoria puderam ser calculadas.

A análise dos resultados concentrou-se na apresentação das frequências das publicações para cada categoria. Isto possibilitou verificar quais categorias foram enfatizadas em pesquisas anteriores e, assim, identificar lacunas e possibilidades de pesquisas futuras. Os dados levantados na fase de extração foram codificados em diversos esquemas e gráficos ilustrando as frequências identificadas. O mapeamento dos estudos foi resumido em forma de tabelas e gráficos, conforme será apresentado no capítulo seguinte.

3.2 Survey com Desenvolvedores de Software Ágil

Depois de realizado o mapeamento sistemático, aplicamos um survey com 115 profissionais de TI com experiência em projetos de software ágeis, buscando compreender como os fatores humanos e culturais influenciam os projetos em que eles estão envolvidos. Utilizamos os dados obtidos através do MS para elaborar as questões do survey, de forma a complementar a presente investigação através do levantamento quantitativo de variáveis relacionadas às motivações da pesquisa.

O *survey* é definido por Pinsonneault & Kraemer (1993) apud Freitas (2000) como um método de pesquisa exploratório e descritivo. A pesquisa é feita a partir da aplicação de questionamentos a um público alvo específico. No nosso caso, desenvolvedores de software ágeis. A realização do survey proporciona um aprofundamento dos resultados obtidos com o mapeamento a partir da visão prática de profissionais de TI a respeito do desenvolvimento de software utilizando métodos ágeis.

3.3 Ameaças à Validade

Uma ameaça à validade de constructo é a estratégia de busca utilizada. Apesar da *string* de busca ter sido ajustada para contemplar a maior

quantidade de artigos relevantes, algum estudo em particular, que usa termo diferente dos previstos, pode não ter sido identificado.

Estudos realizados na língua portuguesa não foram contemplados neste estudo, como visto nos critérios de exclusão. Assim, artigos relevantes mas não escritos em inglês, podem ter sido excluídos da pesquisa.

Outra importante limitação de um Mapeamento Sistemático é a cobertura limitada de estudos para a área de investigação, onde, na construção da String de Busca, com a identificação das palavras-chave e sinônimos que a constituem, existe o risco de omissão de estudos relevantes para a pesquisa. Na tentativa de ampliar a cobertura dos estudos existentes, foi feita uma busca automática em 4 engenhos de busca, número considerado suficiente para uma cobertura aceitável (KITCHENHAN, 2007).

Outro fator que pode ser considerado como uma ameaça ao constructo é a pequena quantidade de estudos primários selecionados, apenas 22 (vinte e dois) estudos foram selecionados no final. O fato dos dados deste trabalho terem sido extraídos por apenas um pesquisador também é considerado uma ameaça à validade de constructo. No entanto, o pesquisador já possuía conhecimento prévio na área estudada e as etapas do mapeamento foram supervisionadas pela orientadora do trabalho.

3.4 Resumo do Capítulo

O presente capítulo descreveu os detalhes do método utilizado para o desenvolvimento desta pesquisa. O mapeamento sistemático seguiu regras rigorosas para que cada artigo fosse avaliado segundo os mesmos parâmetros comparativos. Utilizamos o MS por ser esta uma metodologia que fornece uma visão organizada dos resultados encontrados em estudos primários da área de desenvolvimento de software ágil. Para tanto, definimos as questões de pesquisa acerca da influência dos fatores humanos e culturais, considerados em nosso estudo como FCSs, bem como os principais desafios enfrentados durante o desenvolvimento ágil. Feito isso, selecionamos diversos artigos por meio de busca automática e pela técnica de snowballing, a partir dos quais fizemos uma triagem com critérios de inclusão e de exclusão até chegarmos aos artigos realmente relevantes para a nossa investigação. Com a literatura definida, selecionamos os artigos em categorias e fizemos a extração de dados para realizar o mapeamento e discutir os resultados observados. Por fim, um survey foi realizado com o intuito

de complementar os resultados obtidos e aprofundar a discussão acerca de como os fatores humanos e culturais influenciam o desenvolvimento de software ágil.

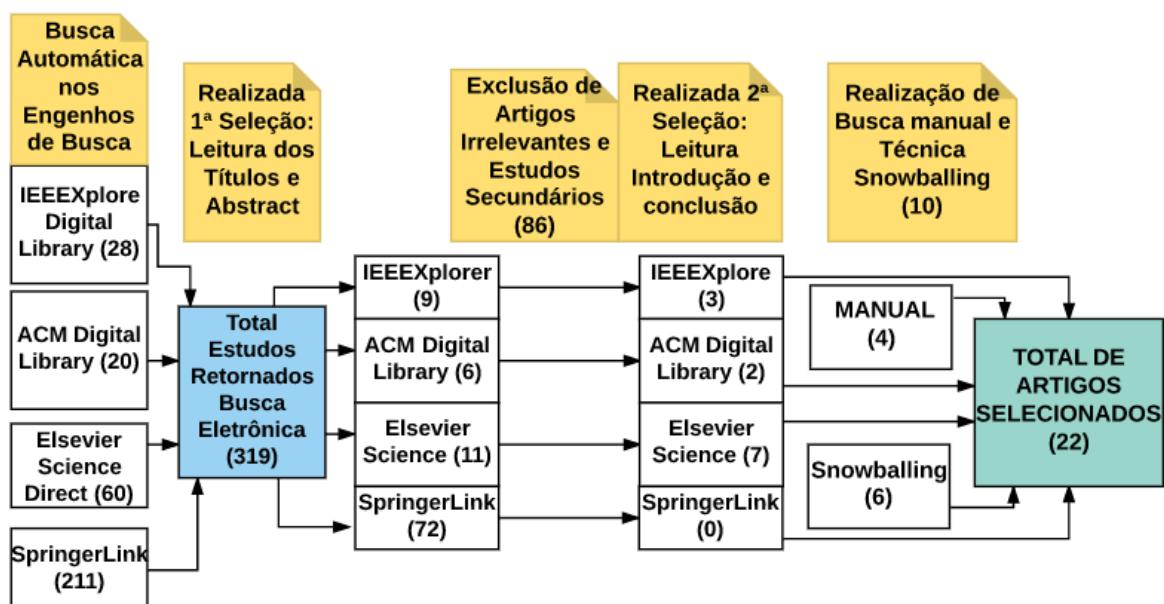
4 RESULTADOS DO MAPEAMENTO SISTEMÁTICO

Este capítulo aborda os resultados obtidos na execução do Mapeamento Sistemático. Os resultados incluem o tipo de busca realizada, os engenhos que se sobressaíram, a apresentação geral de cada estudo primário e, por fim, são respondidas as questões de pesquisa propostas no Capítulo 1.

4.1 Sobre a Busca e Seleção dos Artigos

A partir da metodologia detalhada no capítulo anterior, foram aplicados os passos para a seleção dos estudos primários em uma busca inicial segundo as *strings* de busca e aplicação dos critérios de exclusão e inclusão. Inicialmente, cada engenho de busca identificou um número distinto de artigos que após a aplicação dos critérios e estratégias de busca, revelaram os artigos a serem selecionados pelo mapeamento. A Figura 8 representa um resumo dessa fase inicial, demonstrando o número de artigos levantados em cada fase do processo de busca.

Figura 8 - Número de artigos identificados para a seleção dos estudos primários.



Fonte: Elaborado pelo Autor, 2016.

Após analisar 32 estudos pré-selecionados, foi realizada uma leitura minuciosa de cada um, a partir da qual foi constatado que dez (10) estudos deveriam ser excluídos por não responderem às questões de pesquisa aqui propostas, restando então um total de vinte e dois (22) estudos. Vale frisar que os artigos foram encontrados por meio de busca automática, *snowballing* e manual. Os resultados da seleção estão dispostos na Tabela 1, com seu quantitativo e sua porcentagem destacados respectivamente:

Tabela 1 - Resultado final da seleção dos artigos para o MS.

Tipo de Busca	Quantidade de artigos	%
Automática	12	54,54
Snowballing	06	27,27
Manual	04	18,19
Total	22	100

Fonte: Elaborado pelo Autor, 2016.

Evidencia-se que a busca automática nos engenhos, por meio das *strings* de busca descritas no capítulo anterior, foi o tipo de busca que gerou o maior número de artigos selecionados, chegando a mais 50% dos estudos contemplados para análise, provenientes de diversos periódicos e conferências em bases de dados eletrônicas. Dentre os engenhos de busca encontrados na pesquisa, os que se destacaram aqui foram: ACM, Elsevier, IEEE e Springer. Para uma melhor visualização, os resultados estão distribuídos na Tabela 2.

Tabela 2 - Distribuição do número de artigos por engenho de busca.

Engenho de Busca	Quantidade de artigos
ELSEVIER	06
IEEE	02
SPRINGER	02
ACM	02
Total	12

Fonte: Elaborado pelo Autor, 2016.

Em relação à técnica de *snowballing*, a utilizamos como apoio após realização da busca automática, sendo a segunda técnica com maior número de artigos encontrados, chegando a mais de 20% do seu total.

Assim, os artigos que serviram de base para o *snowballing* foram selecionados na busca automática, ou seja, foram analisados os artigos que faziam parte da lista de suas referências. Assim, foram selecionados aqueles que seriam pertinentes para esta pesquisa, configurando, portanto, na utilização da forma *backward*.

Quanto à busca manual, apesar de ter sido o tipo de busca com menor número de estudos encontrados, menos de 20%, é um tipo de busca mais trabalhosa, pois não são usados métodos automatizados, independentes dos estudos advirem de conferências e/ou periódicos.

Para um melhor entendimento, a Tabela 3 apresenta, em ordem cronológica, os artigos considerados relevantes e que respondem às questões de pesquisa, após passaram pelas etapas de seleção descritas anteriormente.

Tabela 3 – Apresentação geral dos 22 artigos selecionados para esta pesquisa.

ID	Artigos selecionados para mapeamento sistemático	Estratégia de Busca
E01	Cockburn, A.; Highsmith, J. Agile Software Development: The People Factor. <i>Software Management</i> . IEEE Computer Science. 2001.	Snowballing
E02	Siakas, K.V.; Siakas, E. The Agile Professional Culture: A Source of Agile Quality. <i>Software Process Improvement Practical</i> . v.12. p.597-610, 2007.	Snowballing
E03	Chow, T.; Cao, D.B. A survey study of critical success factors in agile software projects. <i>The Journal of Systems and Software</i> . v.81. p.961-971, 2008.	Automática
E04	Tolfo, C.; Wazlawick, R.S. The influence of organizational culture on the adoption of extreme programming. <i>The Journal of Systems and Software</i> . v.81, p.1955-1967, 2008.	Snowballing
E05	Xiohua, W.; Zhi, W.; Ming, Z. The Relationship between Developers and Customers in Agile Methodology. <i>International Conference on Computer Science and Information Technology</i> . IEEE Computer Science. p.566-572, 2008.	Snowballing
E06	Misra, S.C.; Kumar, V.; Kumar, U. Identifying some important success factors in adopting agile software development practices. <i>The Journal of Systems and Software</i> . v.82, p.1869-1890, 2009.	Manual
E07	Moe, N.B.; Dingsayr, T.; Dibá, T. Overcoming Barriers to Self-Management in Software Teams. IEEEComputer Society, 2009.	Automática
E08	Strode, D.E.; Huff, S.L.; Tretiakov, A. The Impact of Organizational Culture on Agile Method Use. <i>Proceedings of the 42nd Hawaii International Conference on System Sciences</i> , 2009.	Snowballing
E09	Tolfo, C.; Wazlawick, R.S.; Ferreira, M.G.; Ferreira, G.; Forcellini, F.A. Agile Methods and Organizational Culture: Reflections about Cultural Levels. <i>Software Process Improvement and Practice</i> , 2009.	Automática
E10	Hajjdiab, H.; Taleb, A.S. Adopting Agile Software Development: Issues and Challenges. <i>International Journal of Managing Value and Supply Chains (IJMVSC)</i> . v.2, n3, 2011.	Snowballing
E11	Hoda, R.; Noble, J.; Marshall, S. The impact of inadequate customer collaboration on self-organizing Agile teams. <i>Information and Software Technology</i> . v.53,p.521-534, 2011.	Automática
E12	Livari, J.; Livari, N. The relationship between organizational culture and the deployment of agile methods. <i>Information and Software Technology</i> . v.53, p.509-520, 2011.	Snowballing

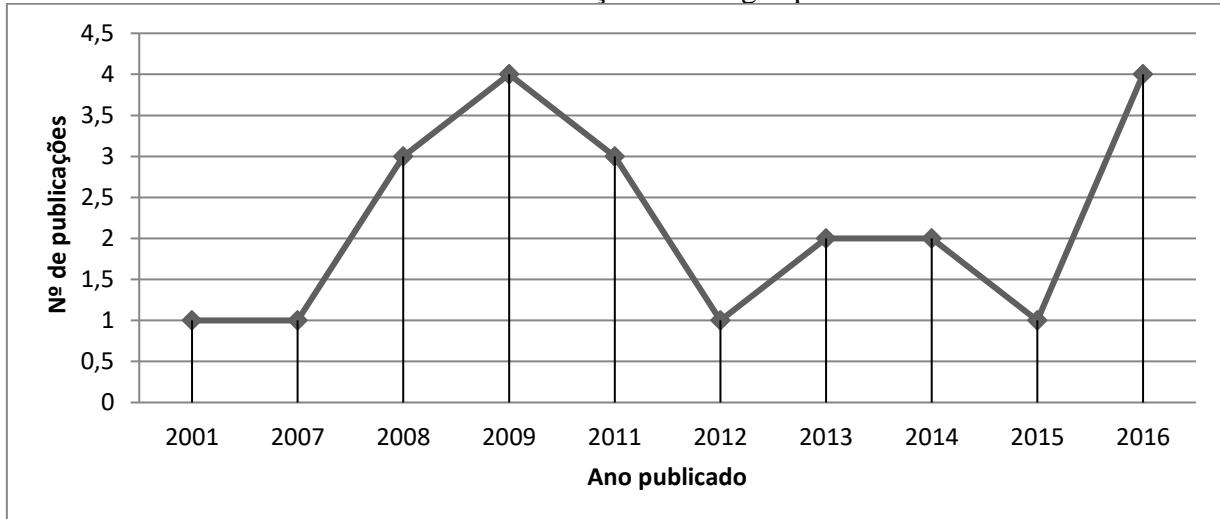
E13	Santos, S.F. Analyzing the Intertwining of Social and Technical Aspects in Agile Methods. ASE International Conference on BioMedical Computing. IEEEComputer Society, p.320-327, 2012.	Automática
E14	Stankovica, D.; Nikolic, V.; Djordjevic, M.; Cao, D.B. A survey study of critical success factors in agile software projects in former Yugoslavia IT companies. The Journal of Systems and Software. v. 86., p.1663-1678, 2013.	Automática
E15	Wiklund, K.; Sundmark, D.; Eldh, S.; Lundqvist, K. Impediments in Agile Software Development: An Empirical Investigation. 14 ^a Conferência Internacional PROFES, 2013.	Automática
E16	Manen, H.V.; Vliet, H.V. Organization-Wide Agile Expansion Requires an Organization-Wide Agile Mindset, 2014.	Automática
E17	Kompella, L. Agile Methods, Organizational Culture and Agility: Some Insights. ACM. p.40-47, 2014.	Automática
E18	Gandomani, T.J.; Zulzalil, H.; Ghani, A.Z.A.; Sultan, A.B.M.; Parizi, R.M. The impact of inadequate and dysfunctional training on Agile transformation process: A Grounded Theory study. Information and Software Technology. v.57, p.295-309, 2015.	Automática
E19	Gandomani, T.J.; Nafchi, M.Z. Agile transition and adoption human-related challenges and issues: A Grounded Theory approach. Computers in Human Behavior. V.62, 2016.	Manual
E20	Gregory, P.; Barroca, L.; Sharp, H.; Deshpande, A.; Taylor, K. The challenges that challenge: Engaging with agile practitioners' concerns. Information and Software Technology. v.77, p.92-104, 2016.	Automática
E21	Gren, L.; Goldman, A. Useful Statistical Methods for Human Factors Research in Software Engineering: A Discussion on Validation with Quantitative Data. 9th International Workshop on Cooperative and Human Aspects of Software Engineering, 2016.	Manual
E22	Lazawanithi, M.R.R.; Alsadoon, A.; Prasad, P.W.C.; Sager, S.; Elchouemi, A. Cultural Impact on Agile Projects: Universal Agile Culture Model (UACM). 7 ^a International Conference on Information and Communication Systems (ICICS), 2016.	Manual

Fonte: Elaborado pelo Autor, 2016.

4.2 Caracterização dos Artigos Selecionados para Mapeamento Sistemático (MS)

Foram encontrados artigos entre os períodos de 2001 a 2016, sendo que os anos de 2002, 2003, 2004, 2005, 2006 e 2010 não foram mencionados, enquanto que os anos de 2009 e 2016, ambos com quatro (04) publicações, foram os anos que obtiveram maior número de estudos. Já os anos de 2001, 2007, 2012 e 2015, apresentaram apenas um (01) artigo cada. Tais dados estão expostos no Gráfico 1:

Gráfico 1 - Distribuição dos artigos por ano.



Fonte: Elaborado pelo Autor, 2016.

Dentre os artigos relevantes, cinco (05) estudos foram publicados em conferências, enquanto que a maioria, dezessete (17), foram publicados em periódicos, conforme descreve o Gráfico 2.

Gráfico 2 - Origem das publicações.



Fonte: Elaborado pelo Autor, 2016.

Durante o período de 2001 e 2016, identificou-se um total de sessenta e três (63) pesquisadores, entre autores e co-autores dos vinte e dois (22) estudos selecionados. A intenção era apresentar os dez que mais contribuíram com seus estudos na Tabela 4 abaixo, porém, este total não foi alcançado, pois apenas quatro (04) autores tiveram um total de dois (02) artigos publicados, chegando a 9% do total dos vinte e dois (22) artigos relevantes. Ou seja, a nossa lista de artigos possui uma grande diversidade de autores.

Tabela 4- Principais pesquisadores.

Nome do autor	Nº de artigos
CAO, D.B.	02
GANDOMANI, T.J.	02
DIBÁ, T.	02
TOLFO, C.	02

Fonte: Elaborado pelo Autor, 2016.

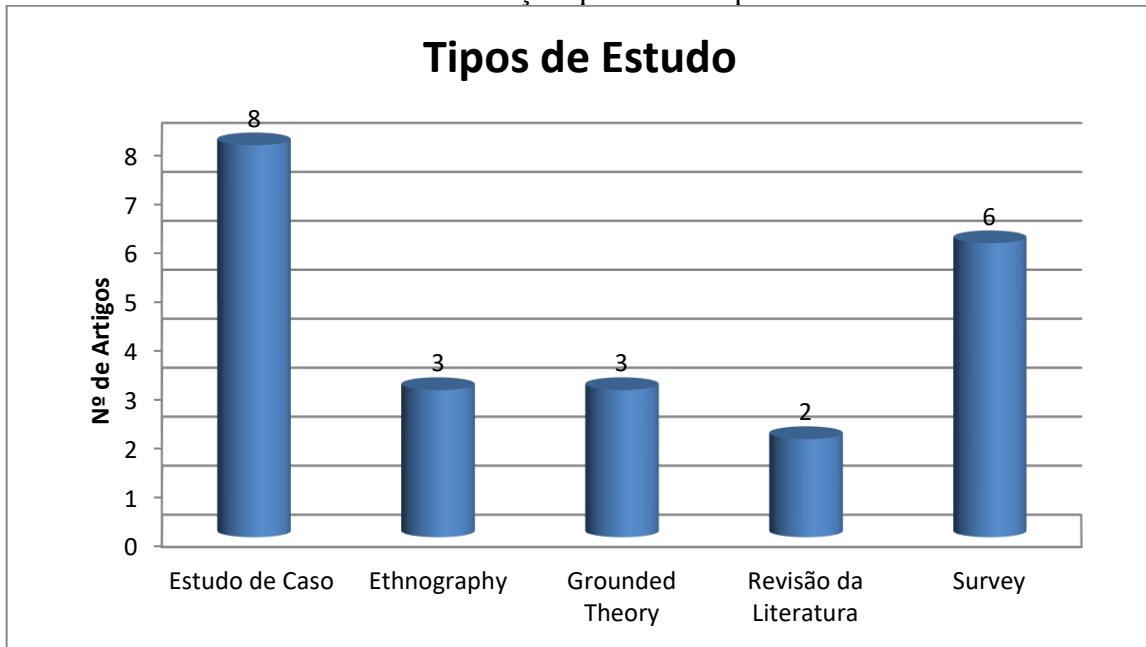
Para se desenvolver uma pesquisa, é indispensável selecionar o método de pesquisa adequado às características do estudo, podendo ser escolhidos diferentes métodos de investigação para possibilitar uma análise qualitativa e quantitativo sobre o fenômeno em estudo. Na Tabela 5, destacam-se os tipos de estudos identificados de acordo com cada publicação.

Tabela 5 - Tipos de estudos identificados nos artigos.

Tipo de estudo	Artigo por ID
Estudo de Caso	E05,E07,E08,E09,E10,E15,E16,E17.
Ethnography	E02,E12,E13.
Grounded Theory	E18,E11,E19.
Revisão da Literatura	E01, E04.
Survey	E03,E06, E14,E20,E21,E22.

Fonte: Elaborado pelo Autor, 2016.

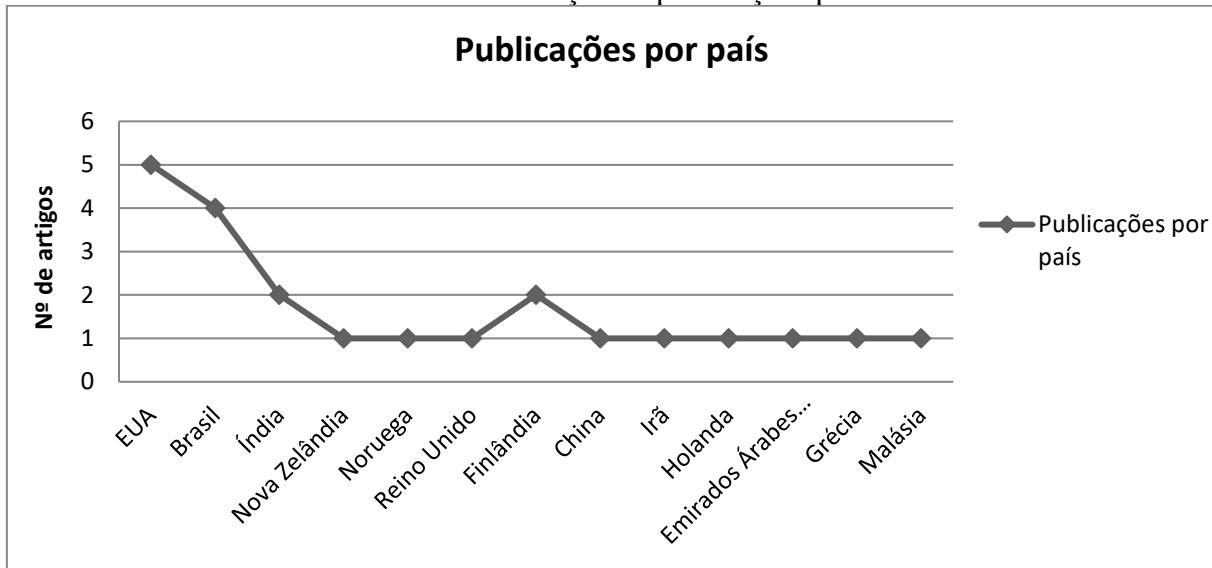
Através desta distribuição pode-se observar que cinco tipos de estudos foram identificados dentre o total de artigos analisados, sendo que a Estudo de Caso 36,36% (8/22), o Survey 27,27% (6/22), Grouded Theory (3/22) e Ethnography (3/22) ambos com (13,64%) foram os quatro procedimentos metodológicos mais adotados. Estes quatro métodos de pesquisa contemplam (63,64%) a dos artigos primários. Estes estudos empíricos foram distribuídos entre 4 métodos diferentes e na revisão da literatura, como representados no Gráfico 3:

Gráfico 3 - Distribuição quanto aos tipos de estudo.

Fonte: Elaborado pelo autor a partir dos dados do MS.

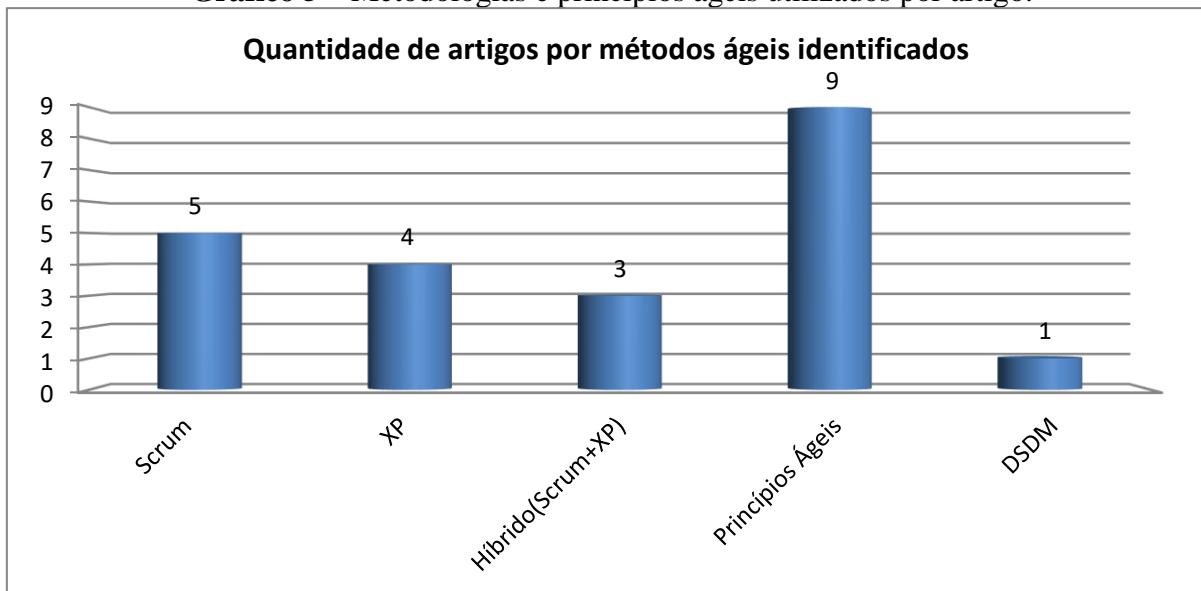
Sobre as instituições onde os autores desenvolveram suas pesquisas descritas nos vinte e dois (22) artigos, temos um total de vinte e nove (29) instituições de pesquisa. São elas: Univ. Fed. Pernambuco; Univ. Victoria Wellington; Univ. Malaya; Univ. Taylor; Univ. Central Lancashire; Univ. Aberta; Univ. Capella; Univ. Pristina; Univ. Nis; Siemens PLM Software, CA;; Univ. Federal de Santa Catarina; Prog. Eng. Tec., CA ; Univ. Islâmica Azad; Inst. Tec. Índia; Univ. Carleton; Univ. Abdu Dhabi; Univ. Massey; Univ. Marludem; Univ. Norueguesa Cienc. Tec.; Univ. VU Amsterdam; Univ. China; Faculdade de Medicina, Zunyi Zunyi; Univ. Gálev; Univ. Gotemburgo Gotemburgo; Univ. São Paulo; Univ. Charles Sturt; Univ. Örebro; Univ. Walden; Univ. Oulu; Univ. South Califórnia; Inst. Educ. Tec. Salónica.

Assim como o número de instituições foi elevado, percebe-se no Gráfico 4 que muitos países estão representados. Este fato mostra que, apesar da quantidade de artigos ser enxuta, há grande diversidade de países que apresentaram estas pesquisas, podendo assim ter uma visão ampla e globalizada sobre o desenvolvimento ágil de software. Importante frisar que alguns países tiveram destaque, como Estados Unidos da América (5 artigos, 22,72%), Brasil (4 artigos, 18,18%), Finlândia (2 artigos, 9,1%) e Índia (2 artigos, 9,1%).

Gráfico 4 - Distribuição de publicações por País.

Fonte: Elaborado pelo Autor, 2016.

O Gráfico 5 apresenta as metodologias identificadas nos artigos selecionados. Com isso, nove artigos adotam os princípios dos métodos ágeis nas organizações citadas pelos estudos, bem como a adoção de práticas do Scrum em cinco artigos, XP em quatro artigos, Híbrido (práticas das duas metodologias) em três artigos e o DSDM (Metodologia de Desenvolvimento de Sistema Dinâmico) com um artigo.

Gráfico 5 – Metodologias e princípios ágeis utilizados por artigo.

Fonte: Elaborado pelo Autor, 2016.

4.3 Respostas às Questões da Pesquisa

Nesta seção, as questões de pesquisa serão respondidas individualmente. Ressalta-se que os artigos selecionados não respondem todas as questões de pesquisa. O critério para selecionar os artigos foi que, cada estudo responda pelo menos uma questão de pesquisa.

4.3.1 Visão geral das questões da pesquisa respondidas

Quando observadas as questões respondidas, percebe-se no Gráfico 6, que todas foram contempladas, porém com variáveis em sua quantidade, das quais a questão 1.2 foi a que mais acumulou respostas advindas de diversos estudos, totalizando 17 artigos e, logo após, as questões 1.1 e 1.3, com 8 artigos cada. Já a questão 1 foi a que menos obteve respostas, com apenas 6 ocorrências nos artigos relevantes.

Gráfico 6 - Distribuição do número de artigos por questão de pesquisa.



Fonte: Elaborado pelo Autor, 2016.

4.3.2 QP1- Quais são os fatores críticos de sucesso em projetos de desenvolvimento de software ágeis?

Com relação à QP1, apenas 6 artigos descrevem de forma explícita fatores críticos de sucesso em projetos ágeis.

Os artigos E02, E03, E06, E11, E14 e E15 discutem, um total de 10 fatores críticos de sucesso em projetos ágeis, os quais estão dispostos na Tabela 6. Dentre estes 4 artigos relevantes, o que mais apresentou fatores críticos foi o E03, com 3 fatores, seguido do E06 e E14, com 2 fatores, e do E2, E11 e E15 com 1 fator identificado em cada estudo.

Tabela 6 - Fatores críticos de sucesso em projetos ágeis identificados nos artigos.

Fator Crítico de Sucesso (FCS)	Estudo
Adoção de Técnicas Ágeis	E03
Apoio/Comprometimento da Alta Gestão	E03,E14
Atendimento dos Requisitos Especificados	E06
Cronograma do Projeto Adequado	E03
Custo do Projeto Adequado	E06
Definição Clara dos Processos de Negócio	E06
Equipe Composta por Especialistas	E03
Gerente de projetos Experiente	E15
Participação Ativa do Cliente no Projeto	E02,E03,E06,E11,E14
Tomada de decisão em tempo hábil	E06
Total de Estudos (FCS)	10 FCS em 6 estudos (E02,E03,E06,E11,E14,E15)

Fonte: Elaborado pelo Autor, 2016.

De acordo com Nielsen (2002), fatores críticos de sucesso são áreas do projeto que devem funcionar perfeitamente, a fim de não comprometer o resultado e a qualidade da implementação. O artigo E03 chama a atenção para a determinação de uma métrica avaliativa que evidencie o controle de determinado FCS, caracterizando que não basta determiná-lo, como também saber se está sendo atingido e mantido corretamente.

Métodos ágeis possuem como uma de suas características a participação do cliente dentro de todo o processo de desenvolvimento, desde a discussão sobre

recursos do produto ao fornecimento rápido de feedback para a equipe de desenvolvimento em contato constantemente.

Quanto aos colaboradores, pode-se destacar dentre esses fatores: comprometimento da gerência; processo de gerenciamento; estratégia correta e cronograma; processo de negócios; pessoas; técnica; trabalho novo para a equipe; coordenação. Já sobre os clientes, os fatores críticos são: envolvimento do cliente; atender requisitos e mudanças do cliente. Enquanto que, nos fatores críticos de sucesso relacionados ao projeto foram identificados: tipo de projeto; projeto metódico; natureza do projeto; técnicas de engenharia de software.

Abaixo apresentamos algumas práticas identificadas pelos estudos relacionados, com intuito de subsidiar equipes ágeis na satisfação dos fatores críticos de sucesso:

- É importante que toda organização tenha conhecimento e aceite a metodologia ágil (E03, E14);
- A gestão deve aceitar em delegar as decisões para os profissionais experientes dentro da equipe de desenvolvimento, podendo também acordar limites com antecedência e mantê-los de forma transparente para a equipe, dessa forma haverá uma redução significativa na tomada de decisões, não atrasando os projetos (E15, E03);
- Equipes ágeis possuem como características comunicação informal, para que seja eficiente é sugerido que o tamanho ideal da equipe seja entre 7 e 10 membros (E03);
- As entregas incrementais podem reduzir os riscos dos projetos, deve ser planejada e executada em sequência. Isso permitirá que a equipe de desenvolvimento tenha *feedback* rápido do cliente da solução contratada (E06).

4.3.3 *QP1.1 - Quais são os desafios relacionados aos fatores técnicos?*

Os estudos E02, E04, E05, E10, E13, E14, E15, E22, apresentam 14 desafios relacionados aos aspectos técnicos em projetos ágeis que devem ser levados em consideração, os quais estão dispostos na Tabela 7:

Tabela 7 - Desafios quanto aos aspectos técnicos.

Aspectos técnicos	Estudo
Qualificação da equipe	E02
Desenvolvimento de código simples e funcional	E04
Pair programming (programação em par)	
Requisitos inadequados	E05
Consultor Ágil (Agile Master)	
Equipe resistente a mudanças	
Equipe subdimensionada	E10
Falta de profissionais experientes em métodos ágeis	
Desenvolvimento IID (<i>Iterative and Incremental Development</i>)	E13
Ausência de tecnologia e ferramentas adequadas	E14
Testes exaustivos	E15
Falta de processo de métodos ágeis	
Gestão transversal (painei)	E22
Reuniões improdutivas	
Total de desafios técnicos / Total de Estudos	14 Desafios em 8 estudos (E02,E04,E05,E10,E13,E 14,E15,E22)

Fonte: Elaborado pelo Autor, 2016.

Como é colocado em E13, o fato dos métodos ágeis se concentrarem mais em fatores humanos do que em fatores técnicos, alinha-se com o fato de que alguns deles não contribuem na definição do processo de desenvolvimento, nas questões técnicas ou mesmo nos padrões de projeto, como é feito em algumas das abordagens tradicionais.

Segundo os autores, ao analisar os métodos Scrum e XP, ambos apresentam ciclo de vida com desenvolvimento iterativo e incremental, no entanto, enquanto o Scrum está mais focado na excelência e na cadeia de mercado para adoção e uso (através de certificações), o XP tem mais elementos para fundamentar o ciclo de vida de iteração e incrementos.

Em relação ao processo de desenvolvimento, apenas o XP apresenta uma definição de alto nível de tarefas individuais, emparelhadas e em equipe; ou seja, não há definições profundamente técnicas com fluxo de trabalho definidos (workflows), papéis especializados e

bem definidos. Executar com êxito essas ações depende muito da experiência de cada indivíduo e da equipe.

No Estudo E22, os resultados mostraram que os métodos Scrum e XP preferencialmente devem ser implementados em organizações de tipo pequeno e médio e não como um método autônomo em uma organização globalmente distribuída. Embora o custo em implementá-los seja baixo e possuam adaptabilidade como característica forte, esses métodos exigem treinamento de funcionários e de mudança cultural da organização.

Corroborando os dados encontrados, Carvalho, Abrantes e Cameira (2011) em seu estudo que propôs melhorias para a metodologia de desenvolvimento de sistemas de uma empresa do setor de software com a implementação do Scrum, colocam que um fator positivo encontrado na experiência foi o auxílio da alta direção da empresa em orientar as mudanças internas no sentido da implantação da nova estrutura organizacional, o treinamento em Scrum oferecido para todos os profissionais, além de campanha eficiente que orientou todos nas transformações que seriam realizadas.

Também como desafio técnico é válido ressaltar a técnica chamada Programação em Pares é um método de programação em que duas pessoas trabalham juntas em uma estação de trabalho. Possui objetivo de revisar o código, com intuito de identificar problemas, bugs e simplificação do projeto. Para a organização sem cultura ágil definida pode significar desperdício de recurso, mas é uma oportunidade da troca de conhecimento entre desenvolvedores, melhorando a experiência da equipe.

O artigo E10 comenta quanto à sobrecarga de trabalho dos desenvolvedores e pode ser resolvida pelo planejamento do método ágil durante um período de tempo com quantidade reduzida de projetos por pelo menos 6 meses. Isso permitirá que a equipe possa investir mais tempo em treinamento e melhoria das técnicas ágil. Considera também como investimento, pois terá como contrapartida no futuro a eficiência da equipe.

4.3.4 QP1.2 - Quais são os desafios relacionados aos fatores culturais?

Identificamos 17 estudos que apresentaram desafios relacionados aos fatores organizacionais em projetos ágeis, foram evidenciados um total de 22 fatores, que estão organizados na Tabela 8.

Tabela 8 - Desafios quanto aos aspectos organizacionais em projetos ágeis identificados.

Desafios organizacionais	Estudo
Organização orientada para os resultados	E02
Confiança mútua entre equipe e organização	E01,E02
Compromisso da Equipe	E02
Interação entre equipe	E02, E07
Liderança democrática da organização	E02,E04
Mudança cultural	E02,E04,E05,E10,E12,E17,E20,E21
Colaboração do cliente as etapas do projeto	E03,E06,E11
Satisfação do cliente com as <i>sprint</i>	E06
Tamanho da equipe inadequado	
Compromisso com o cliente	E06,E11
Estilo de gestão centralizada	E07
Alinhamento da estrutura a equipe	
Negligência aos projetos	E09
Metas da organização alinhadas com a equipe	E13
Apoio da alta gestão	E14,E15
Cultura organizacional política	E14
Mentalidade ágil da organização	E16
Investigação	
Adaptação a métodos ágeis	E20
Disponibilidade de modelos de sistema	E22
Problemas na estrutura e colaboração associado a um grande projeto	
Dificuldades na coordenação de projetos de software ágil	
Total de desafios organizacionais / Total de Estudos	22 Desafios em 17 Estudos (E01,E02,E04,E05,E06,E07,E9,E10,E11,E12,E13,E14,E15,E16,E17,E20,E22)

Fonte: Elaborado pelo Autor, 2016.

Em análise da metodologia XP, no artigo E04, além dos fatores críticos culturais da sua implementação serem apontados – como a predominância do trabalho individual, falta de comunicação e integração entre desenvolvedores de software e resistência da empresa em relação às mudanças – os autores citam importantes fatores, como políticas da empresa relacionadas à motivação dos funcionários, coesão na equipe de desenvolvimento e liderança democrática, dentro do ambiente organizacional, o que poderá exigir mudanças profundas na empresa. Em adição, o artigo E05 discute a importância do contato com o cliente para a

melhoria da qualidade do software, onde a equipe XP deve incentivar o cliente trazer as mudanças de requisitos na reunião de planejamento e, em seguida, os desenvolvedores respondê-los rapidamente, alterando ou reconstruindo o código do software.

Em análise da aplicação do Scrum, o artigo E10 mostrou 3 fatores não descritos pela literatura até a data de sua publicação em 2011: “Pressão exagerada da empresa” – relacionada com a sobrecarga dos desenvolvedores –, “Burocracia da organização” – relacionados com a cultura da organização. O primeiro fator, de acordo com os autores, pode ser solucionado com o planejamento da adoção ao método ágil durante um tempo sem que ocorra pressão mínima de trabalho, e sem comprometer com nenhum projeto novo por pelo menos seis meses. Este investimento será compensado mais tarde com o aumento da eficiência da equipe com o método ágil.

Por outro lado, fator burocracia da organização pode exigir mudanças na mentalidade dos gestores, portanto não podem ser resolvidos rapidamente, mas de maneira gradual, pelos desenvolvedores e gerentes superiores através de diálogo, reuniões e revisão das políticas visando a melhoria da organização. E essa é uma das vantagens principais dos métodos ágeis, a capacidade de adaptação com base na cultura e no ambiente da organização.

Os resultados estão de acordo com Melo e Ferreira (2010), em seu estudo de caso que analisou empiricamente a adoção de métodos ágeis e seu impacto no aprendizado, qualidade do código-fonte, produtividade e satisfação do cliente no Banco Central do Brasil, observaram que as principais dificuldades enfrentadas na implantação de métodos ágeis não estão relacionadas ao aprendizado das práticas ágeis e sim com a necessidade de mudança da cultura organizacional. Os autores explicitam que enquanto apenas o projeto de desenvolvimento de software pensar e agir de forma ágil e o restante da organização manter os vícios e culturas derivadas dos processos tradicionais não será possível usufruir realmente dos benefícios ágeis (MELO & FERREIRA, 2010).

4.3.5 *QPI.3 - Quais são os desafios relacionados aos fatores humanos?*

Para responder a referida pergunta, oito estudos desenvolveram e discutiram sobre a temática. Assim, foi encontrado um total de 13 desafios quanto aos aspectos humanos em projetos ágeis, sendo dois citados mais de uma vez. Sua distribuição para melhor visualização encontra-se na Tabela 9 abaixo:

Tabela 9 - Desafios quanto aos fatores humanos em projetos ágeis identificados em 8 artigos.

Desafios humanos	Estudo
Cultura hostil	E01,E05,E09,E19
Conhecimento limitado	E10,E06,E14,E19
Envolvimento do cliente insuficiente	E11
Medo da demissão	E18
Dificuldade de comunicação	E06
Capacidade de trabalhar em equipe	E06
Pouca relevância a adoção ágil	E20
Total de desafios organizacionais / Total de Estudos	7 Desafios em 8 artigos (E01,E05,E06,E09,E10,E11,E14,E19)

Fonte: Elaborado pelo Autor, 2016.

Em E19, os autores citam que o conhecimento limitado sobre as metodologias ágeis não é apenas visto entre os membros da equipe, mas também entre gerentes e clientes. Além dos efeitos negativos deste problema na transição para métodos ágeis, leva a outras consequências adversas que também são prejudiciais para o sucesso da organização, como baixa colaboração, mentalidade equivocada da equipe e expectativas irrealistas.

No artigo E18, os resultados mostraram que embora a formação dos membros da equipe atue como uma solução para superar os desafios relacionados com fator humano quando empresas se encontram em transição para adoção de métodos ágeis, a resistência das pessoas à mudança afeta o processo de formação, bem como outras atividades relacionadas. Isso influencia fortemente a qualidade e o sucesso do treinamento antes e durante a transição. Assim, aspectos humanos têm sido relatados como fatores significativos que afetam todas as etapas do processo de desenvolvimento de *software*.

O E18, que conduziu entrevistas on-line semiestruturadas de perguntas abertas com 35 especialistas em projetos ágeis de 13 países diferentes, propiciou achados interessantes que apontam que "medo de perder o emprego" e percepção errada leva a baixa colaboração na formação. Várias outras preocupações semelhantes a este respeito foram relatadas por alguns dos outros participantes.

4.3.6 QP2 - Como os fatores humanos e culturais influenciam o sucesso de projetos de desenvolvimento de software ágeis?

Conforme Artigo E08, de acordo com o Manifesto Ágil (Agile, 2001), é mais interessante ter pessoas habilidosas e motivadas do que tecnologias e ferramentas de alto custo. Métodos ágeis enfatizam a satisfação do usuário através da sua participação constante no

desenvolvimento, resposta aos requisitos em constante mudança, entrega frequente das requisições, desenvolvimento de software iterativo e auto gerenciamento das equipes.

Neste tipo de organização, consulta, participação, consenso e compromisso são características primordiais com a cultura ágil. Assim, mudar a cultura de uma organização não é uma tarefa trivial, pois exige mudança dos participantes, bem como mudanças na forma de agir, hábitos e costumes já implantados.

No Artigo E02, os autores citam que a abordagem ágil pode ser considerada uma cultura, possui características diferenciadas através de um conjunto de práticas que inclui conhecimentos, princípios e valores que visam a interação entre os membros da organização. O XP por exemplo chama a atenção para quatro valores que são: comunicação, simplicidade, feedback e coragem, a base para os doze princípios que são traduzidos em práticas. Estas práticas são os artefatos para a cultura XP.

Tendo esta consistente cultura, é importante um alinhamento entre os membros e diversos setores da organização para evitar confrontos entre cultura de um determinado grupo/equipe com toda a organização.

Analizando os resultados, nota-se que em todos os estudos se evidencia o fator humano (equipe de trabalho, gestão de pessoas e exigências do mercado/cliente) como fator crítico de sucesso assim como a engenharia do software propriamente dito. Isso ocorre por uma das diferenças básicas entre os projetos ágeis e clássicos, onde o primeiro trabalha para o indivíduo, auxiliando-o, enquanto o segundo tende a ser autônomo. Desse modo, ao mesmo tempo que os projetos ágeis se colocam como altamente adaptáveis e maleáveis às exigências do mercado, e justamente por essa característica, se mostram com a necessidade um maior comprometimento quanto ao domínio técnico da equipe e auto gestão dos projetos e ambiente de trabalho.

Seguindo essa linha de raciocínio, para garantir o controle de qualidade em todas as partes do processo, as diferentes metodologias ágeis adotam medidas que envolvem não apenas a qualidades técnicas, mas também do indivíduo, explicitando a importância deste. Para obtenção do sucesso em projetos ágeis é notório que a definição e divisão de atividades, determinação de valores, atitudes e comportamentos norteadores como responsabilidade, respeito e liderança, são exemplos de estratégias de gestão da equipe de trabalho que vão além do domínio da técnica, e são aplicáveis em vários ambientes de trabalho que não o do desenvolvimento de software. Determinando estas limitações e características dos membros da

organização, é possível esboçar o perfil de equipe e de projetos em que a metodologia ágil melhor se aplica, assim como os setores que serão incluídos no processo.

4.4 Resumo do Capítulo

No que tange o capítulo, Resultados do MS, observou-se em todo contexto científico as abstrações relativas aos resultados encontrados a partir de concepções dos estudos, levantamentos bibliográficos e alusões direcionadas ao tema pesquisado, as influências e fatores são abordados nas literaturas pesquisadas.

Ocorreu uma organização de fatos para cada questão de pesquisa abordada neste MS relacionados ao tipo de busca realizada, aos engenhos que sobressairam e outros aspectos relacionados a apresentação geral de cada estudo. Com isso, são apresentados quantitativos e informações relevantes quanto aos artigos selecionados por tipo de busca e por tipo de engenho disponibilizado. Por consequente, 22 artigos são apresentados de forma sistemática para facilitar o entendimento e as classificações definidas.

Assim, finaliza-se com a caracterização dos artigos selecionados dentro de um período de 2001 a 2016, relevando as informações sobre os pesquisadores, tipos de estudos identificados e finalizando com a visão geral das questões da pesquisa respondida.

5 PESQUISA DE OPINIÃO COM PRATICANTES

No perpassar deste estudo, foi elaborada uma pesquisa quantitativa tendo como público alvo profissionais da área de TI com experiências em projetos ágeis de desenvolvimento. Foram extraídas opiniões e conclusões relevantes para a materialização do contexto influência dos fatores humanos e culturais da organização em projetos ágeis de software, com o objetivo de acrescentar informações foi elaborado um Survey, respondido por 115 (cento e quinze) participantes, sendo 33 (trinta e três) de funcionários de empresas privadas, 79 (setenta e nove) servidores de órgãos públicos e 3 (três) consultores ou prestadores de serviços autônomos.

Contudo, foi elaborado um questionário online com o total de 10 (dez) perguntas, divididas em 3 (três) etapas. Na primeira etapa, foram coletadas 2 (duas) informações relacionadas à identificação do respondente.

Em continuidade a pesquisa, foi realizada a segunda etapa, foram 5(cinco) perguntas sobre o cargo/função, experiência do respondente em projetos ágeis e afirmações sobre os principais fatores técnicos, humanos e culturais da organização, este *survey* foi elaborado com base na escala likert, utilizada habitualmente em questionários para pesquisas de opinião.

Também disponibilizadas 2 (duas) perguntas abertas com intuito de coletar a opinião do entrevistado acerca do assunto.

Na escala Likert os respondentes apresentam o seu nível de concordância a partir de uma afirmação. Para a terceira etapa, as afirmações foram no tipo múltipla escolha, em 5 (cinco) opções, “Não concordo totalmente”, “Não concordo parcialmente”, “Não concordo e nem discordo”, “Concordo parcialmente” e “Concordo totalmente”.

A grande vantagem da escala de Likert é sua facilidade de manuseio, pois é fácil a um pesquisado emitir um grau de concordância sobre uma afirmação qualquer. Adicionalmente, a confirmação de consistência psicométrica nas pesquisas que utilizaram esta escala contribuiu positivamente para sua aplicação nas mais diversas pesquisas (COSTA, 2011 apud SEVERINO, 2014 p. 5).

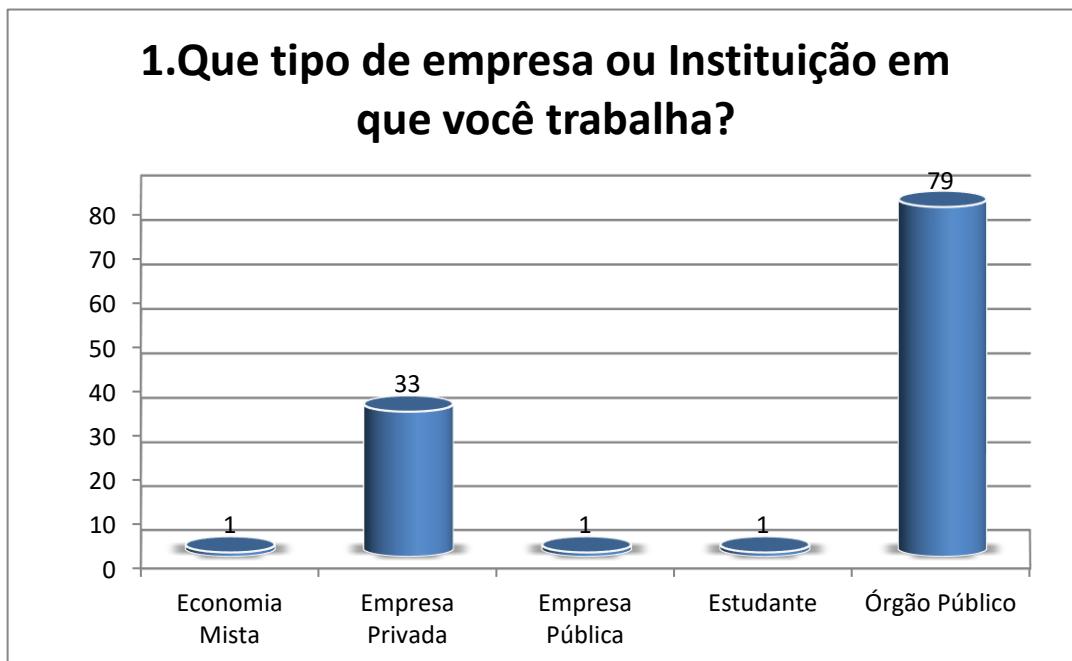
Foi utilizada a aplicação do Google Forms como recurso tecnológico para a criação do formulário de questões online e, após a sua publicação, os respondentes tinham em sua primeira etapa dados abertos para informações pessoais com os tipos descritivos.

As perguntas criadas para este survey foram embasadas no mapeamento sistemático, em práticas ágeis abordadas e identificadas nos estudos primários. Desta forma buscou-se explorar alguns fatores e aspectos para fortalecer os dados extraídos do mapeamento sistemático.

5.1 Perfil dos Participantes

Conforme pode ser visto no Gráfico 7, é apresentado o quantitativo total de 115 respostas, tendo os profissionais de órgãos públicos 79 (setenta e nove) e empresas privada 33 (trinta e três) questionários respondidos principalmente.

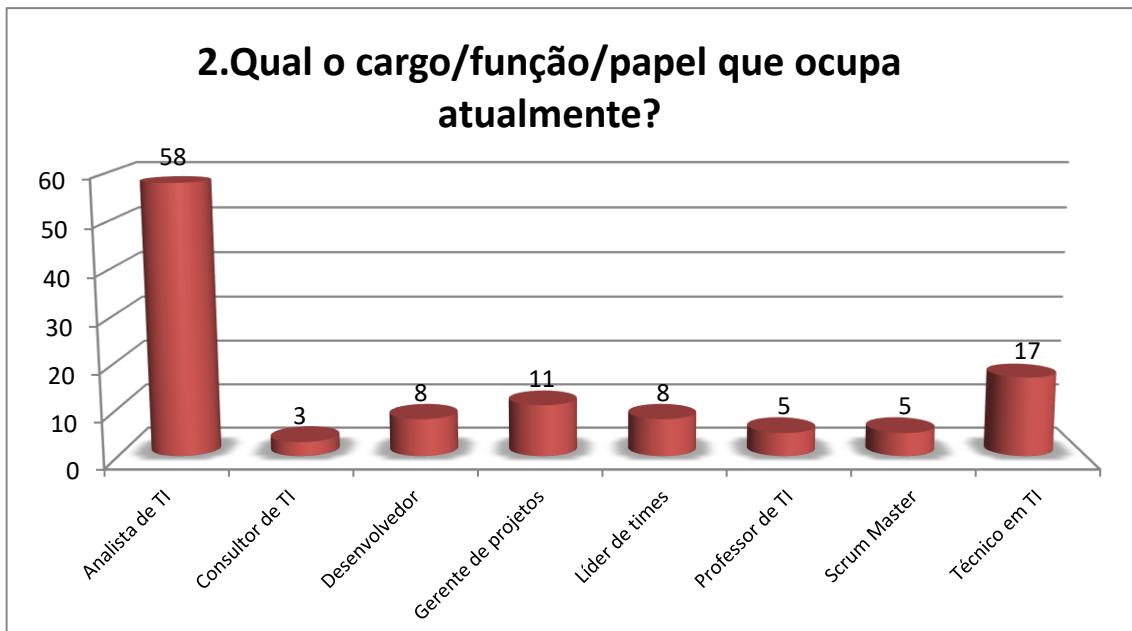
Gráfico 7: Tipo empresa etapa do survey.



Fonte: Elaborado pelo Autor, 2016

O questionário foi destinado a profissionais da área de Tecnologia da Informação que desempenham ou possuem experiências anteriores em projetos ágeis de software. No gráfico 8, solicitamos ao participante informações sobre o seu cargo/função/papel ocupado atualmente:

Gráfico 8: Cargo/papel participante etapa do survey.

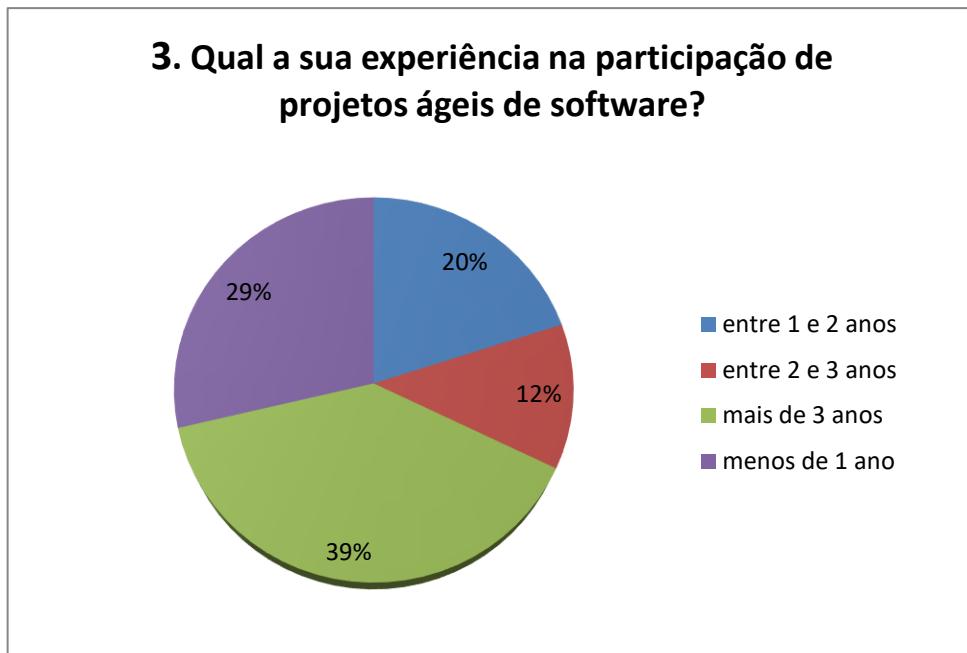


Fonte: Elaborado pelo Autor, 2016

Obtivemos o maior público do cargo analista de TI com 58 (cinquenta e oito) respostas, seguido do técnico em TI 17 (dezessete respostas), é importante ressaltar que na administração pública federal estes cargos são de profissionais que desempenham a atividade desenvolvedor ou gerente de projetos, é válido frisar que houve participação de gerente de projetos com 11 (onze) participantes e Scrum Master 5 (cinco) participantes, reforçando assim a qualidade da pesquisa.

Em relação a experiência dos participantes, 39% dos participantes informaram que possuem mais de 3 anos de experiência na participação em projetos ágeis de desenvolvimento, 29% possuem menos de 1 ano, 20% apresentam entre 1 e 2 anos e 12% entre 2 e 3 anos de experiência, conforme mostramos no gráfico 9 abaixo:

Gráfico 9: Experiência participante etapa do survey.

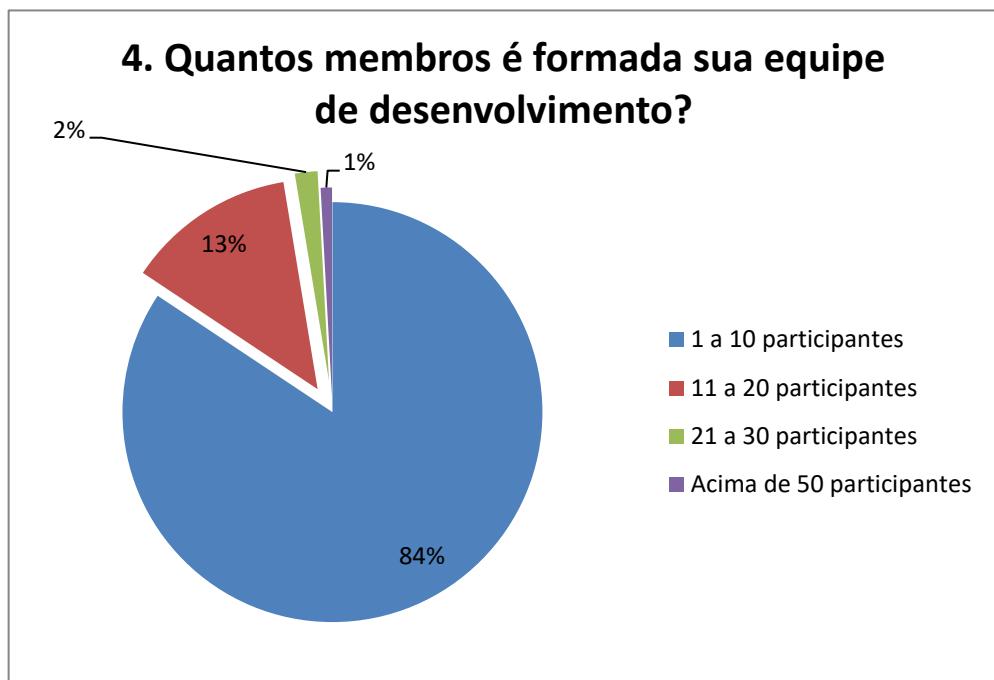


Fonte: Elaborado pelo Autor, 2016

Pode-se evidenciar, diversos estudos do mapeamento sistemático (E02, E03, E06, E10) apresentam o fator experiência e domínio da tecnologia como importante para o bom andamento dos projetos ágeis. Também podemos destacar fatores como auto organização da equipe (E15,E10) e liderança descentralizada (E02,E04), identificamos que é essencial a participação de membros experientes na equipe para conclusão com êxito dos projetos de *softwares*.

Em relação a quantidade de envolvidos nas equipes de desenvolvimento dos participantes obtivemos 84% dos participantes são de equipes entre 1 e 10 participantes e 13% são de equipes entre 11 e 20, conforme demonstrado no gráfico 10 .

Gráfico 10: Formação Equipe desenvolvimento participante etapa do survey.

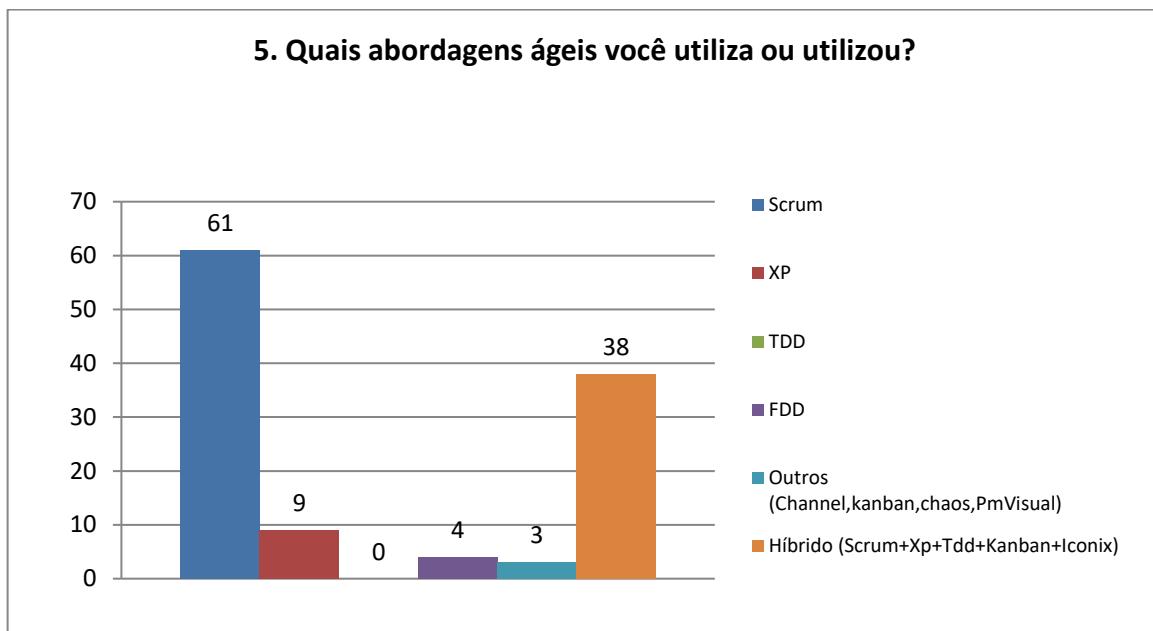


Fonte: Elaborado pelo Autor, 2016

Equipes ágeis contam com forte comunicação informal. Para que isso seja eficiente, os artigos (E06, E10, E14) sugerem que o tamanho ideal para a equipe seja entre sete e dez membros. Este tamanho aparece para minimizar os riscos dentro da equipe. Com base em especificações do projeto, uma equipe menor pode funcionar com menos problemas. Em qualquer caso, os riscos associados ao tamanho da equipe deve ser considerado, comunicado e acordados previamente.

No gráfico 11 buscamos informações acerca das metodologias utilizadas pelos participantes, conforme resultado abaixo:

Gráfico 11: Abordagens utilizadas participante etapa do survey.



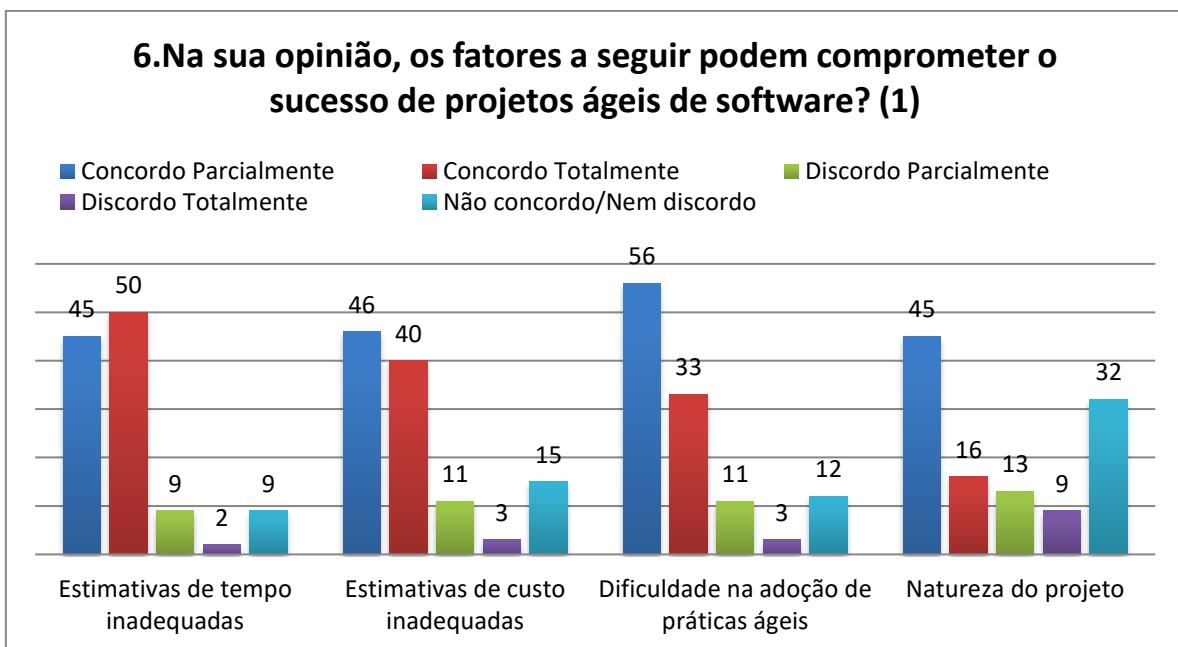
Fonte: Elaborado pelo Autor, 2016

Cada uma dessas metodologias tem a sua particularidade e práticas sugeridas mas muitas vezes o que vemos hoje em dia são os modelos hybridos, que são na verdade uma mescla dessas metodologias/frameworks onde as melhores práticas de cada metodologia é aplicada a um processo customizado. É preciso analisar a necessidade e a maturidade da equipe para então escolher um *framework* ou práticas ágeis que lhe traga o benefício esperado.

Adotar uma metodologia ágil pode trazer muitos benefícios, mas especialistas costumam dizer que o ágil não é a “bala de prata”, ou seja, apenas aplicar o SCRUM ou algumas de suas práticas ágeis por si só não vai resolver os problemas, mas deixar claro as fraquezas do projeto para facilitar a identificação das falhas e atuar sobre elas. Cabe então a equipe atuar de forma pró-ativa e trabalhar nas mudanças para que os benefícios supere as dificuldades.

No gráfico 12 abaixo analisamos os fatores citados pelos participantes que podem comprometer o sucesso de projetos ágeis, estimativas de tempo e custo inadequadas houve uma expressiva opinião dos participantes com 50 (cinquenta) respostas, 56 (cinquenta e seis) participantes concordaram parcialmente com dificuldade na adoção de práticas ágeis e natureza do projeto com 45 opiniões que concordaram parcialmente.

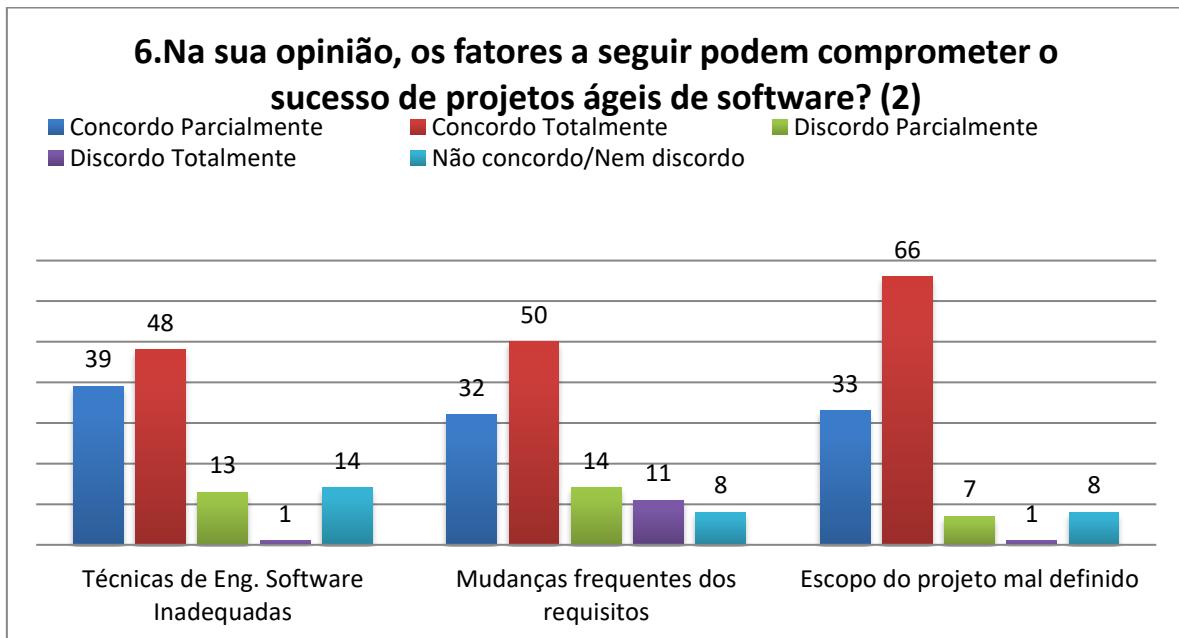
Gráfico 12: Fatores sucesso (1) participante etapa do survey.



Fonte: Elaborado pelo Autor, 2016

Mediante as técnicas de Eng. de software inadequada, mudanças frequentes nos requisitos, escopo do projeto mal definido os participantes apresentaram concordância total e expressiva na pesquisa realizada conforme demonstrado na pontuação apresentada no gráfico 13. O Artigo (E14) do mapeamento sistemático também aborda como dificuldade o escopo mal definido, corroborando com os participantes da pesquisa.

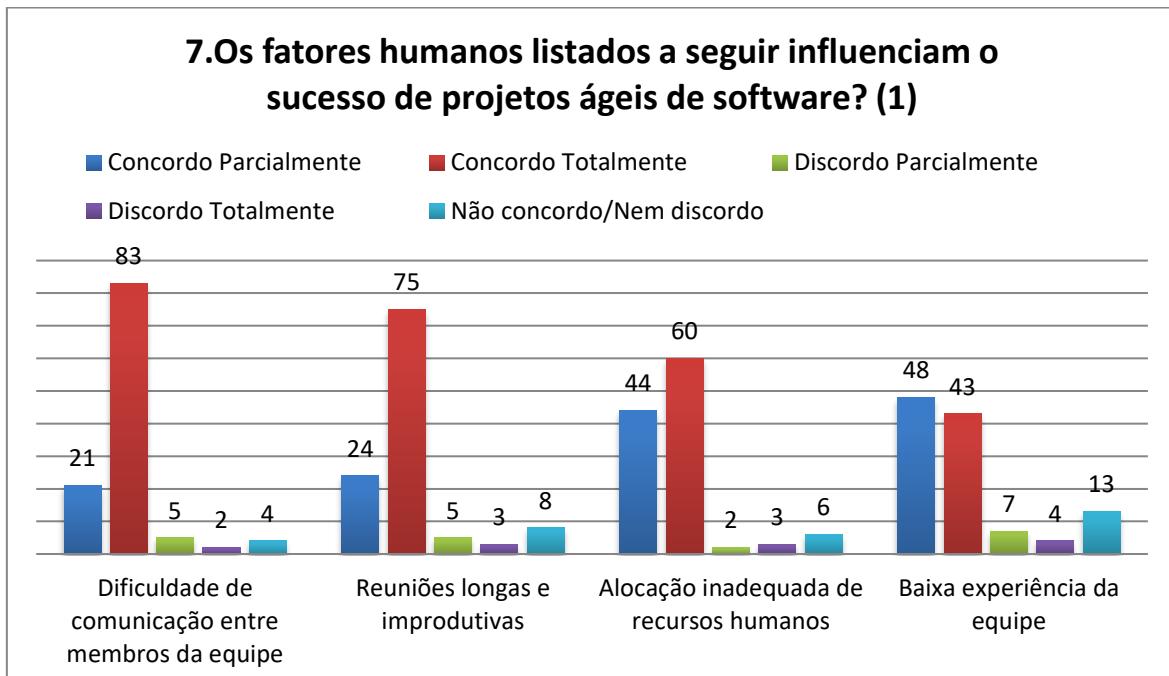
Gráfico 13: Fatores sucesso (2) participante etapa do survey.



Fonte: Elaborado pelo Autor, 2016

No gráfico 14, as questões relacionadas a dificuldade de comunicação entre os membros da equipe, reuniões longas e improdutivas e alocação inadequada de recursos humanos por terem atingido maioria das respostas com concordância total, os participantes concordaram parcialmente com o fator baixa experiência da equipe.

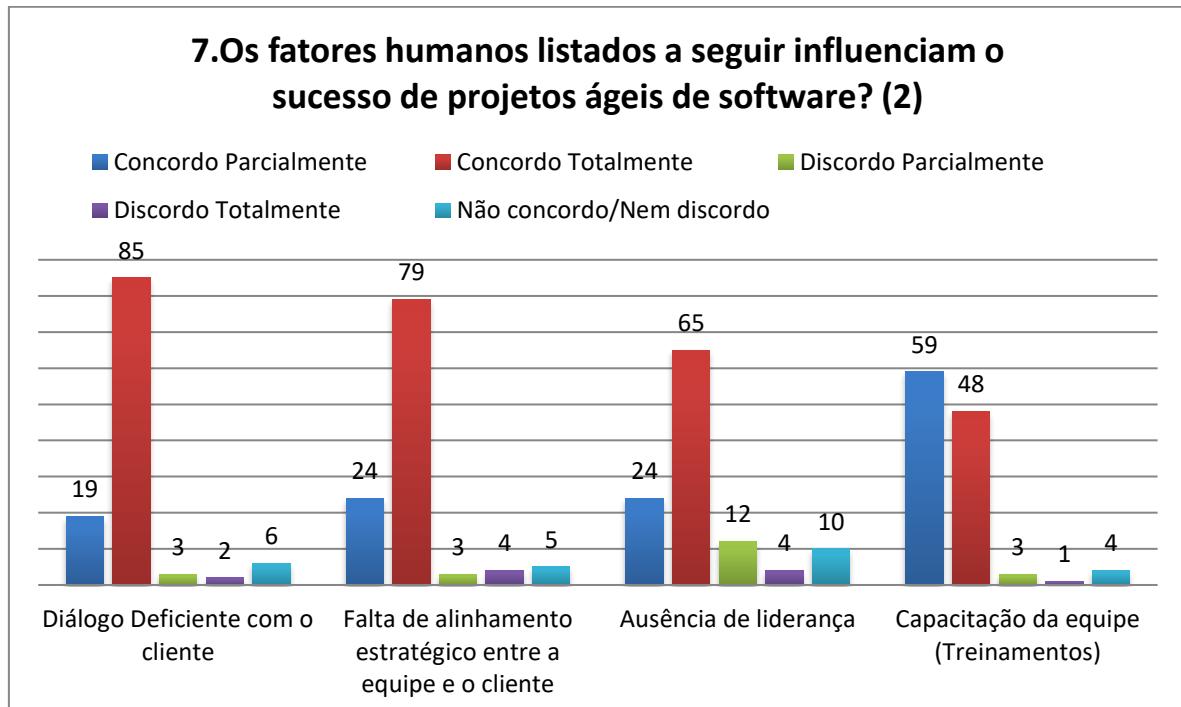
Gráfico 14: Fatores humanos (1) participante etapa do survey.



Fonte: Elaborado pelo Autor, 2016

No gráfico 15, a capacitação da equipe não tiveram total concordância, isto é, a maioria dos participantes responderam uma concordância parcial, esse fato é contrário a opinião dos artigos E02,E03,E04,E06,E10, que consideram treinamentos em métodos ágeis essencial ao sucesso do projeto. O diálogo deficiente com o cliente, falta de alinhamento estratégico entre a equipe e o cliente e ausência de liderança são fatores listados com a concordância total entre os participantes da pesquisa. Destaca-se que o cliente é fator primordial ao sucesso do projeto conforme demonstrado nos artigos analisados.

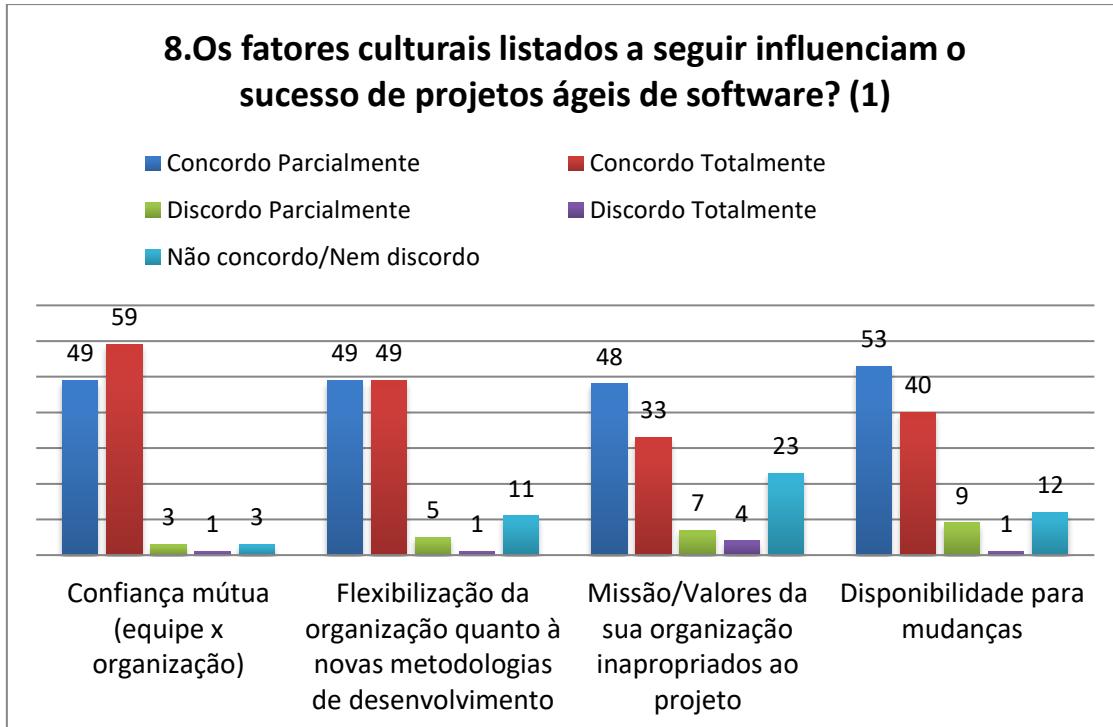
Gráfico 15: Fatores humanos (2) participante etapa do survey.



Fonte: Elaborado pelo Autor, 2016

No gráfico 16, os participantes concordaram totalmente que a confiança mútua (equipe x organização) é um fator essencial ao sucesso do projeto, porém, existem fatores que representam uma concordância parcial expressiva nesse gráfico, podendo ser um ponto de dificuldade do entendimento são eles: flexibilização da organização, missão e valores inapropriados e disponibilidade para mudanças, fatores estes que representam um expressivo envolvimento da cultura organizacional com os projetos de software.

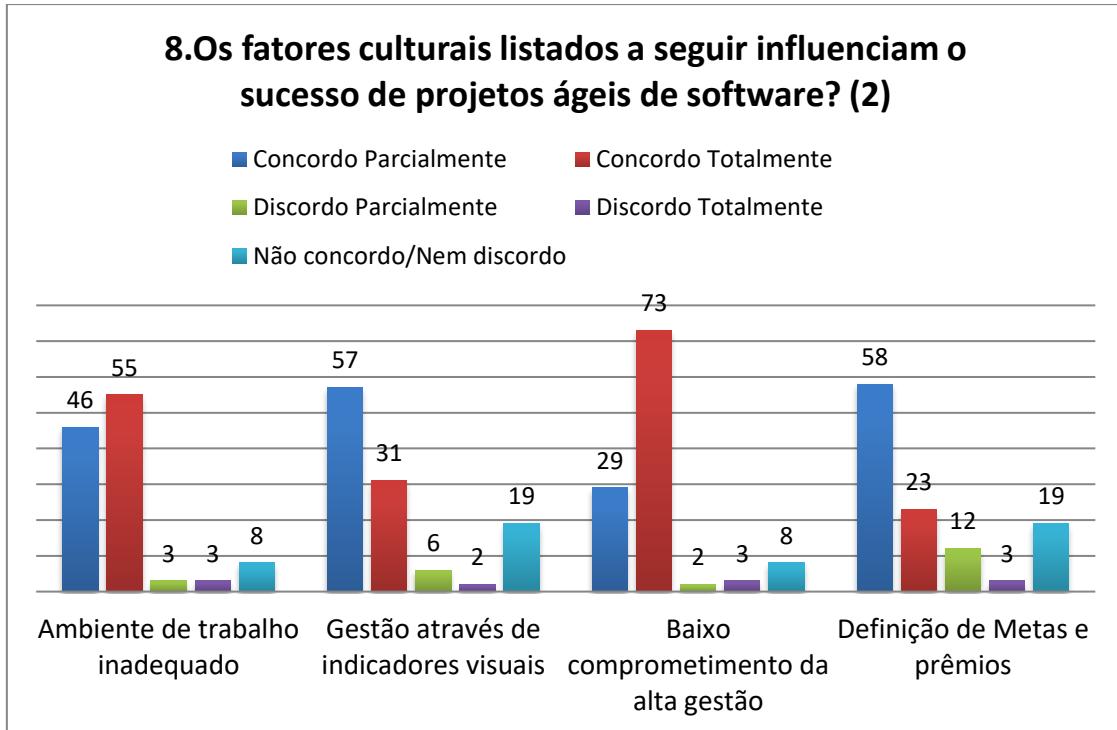
Gráfico 16: Fatores culturais (1) participante etapa do survey.



Fonte: Elaborado pelo Autor, 2016

O baixo comprometimento da alta gestão retrata uma concordância total como destaque neste gráfico 17, visto que a organização necessita de um engajamento mínimo para proporcionar o sucesso do projeto ágil de software, conforme abordado no artigo E20. Existem fatores significativos para uma concordância parcial no envolvimento do profissional em discutir “definição de metas e prêmios”, auxiliando no contexto de sucesso na utilização dos projetos ágeis, uma vez que a equipe deve estar melhor estruturada neste fator cultural.

Gráfico 17: Fatores culturais (2) participante etapa do survey.



Fonte: Elaborado pelo Autor, 2016

Nos itens abaixo, foram elaboradas perguntas subjetivas para os participantes da pesquisa que demonstraram sua opinião referente aos desafios encontrados na sua experiência em projetos ágeis, demonstrando, assim, relatos diversificados destas experiências. Nesta pesquisa, com intuito de transmitir maior qualidade e objetividade às informações, optamos por listar 5 respondentes que possuem o título Scrum Master, trazendo maior experiência com as metodologias ágeis e em equipes de desenvolvimento de software.

9. Na sua opinião, quais são os 3 principais desafios encontrados nos projetos ágeis que você participa ou participou ?

As respostas obtiveram um reflexo significativo para a pesquisa em questão, demonstrando assim argumentos vivenciados na prática e também declarações que são geralmente comuns no ambiente tecnicista.

Respondente 5: "Falta de apoio/confiança da alta gestão, definição de métricas para acompanhamento do desenvolvimento e disponibilidade do cliente para comunicação com a equipe de desenvolvimento."

Respondente 65: "Gerenciamento da equipe, até que esta atinja um nível de maturidade suficiente para se tornar auto-gerenciável; Manter o cliente o mais próximo e disponível possível durante o desenvolvimento; Promover uma mudança cultural/organizacional na empresa, que inclui o apoio da alta gerência."

Respondente 87: "1 - A aceitação dos gestores saírem da metodologia antiga para tentar uma metodologia ágil 2 - Escolha errada dos membros que farão parte da equipe (Deveria avaliar melhor o perfil(conhecimento técnico E psicológico) de quem entrará a equipe) 3 - Falta de treinamento em projetos ágeis, a equipe levou tempo descobrindo por conta própria."

Respondente 96: "Ter o cliente próximo para auxiliar no processo de desenvolvimento. Manter os membros da equipe estimulados e motivados. Manter a realização da daily meeting no modo como está especificado no Scrum Guide."

Respondente 102: "1) Falta de conhecimento da metodologia ágil; 2) Resistência da estrutura de controle (gerentes, administradores, chefes, donos...) 3) Feedback dos clientes, que não tem interesse de participar do desenvolvimento ("já paguei, faça seu trabalho, só quero o resultado")"

10. Na sua opinião, de que forma os fatores humanos e culturais da organização podem influenciar no sucesso de projetos ágeis?

Respondente 5: "São, ao meu ver, os fatores que mais impactam no projeto. Um bom relacionamento da equipe, um líder comprometido, e a disponibilidade para a mudança organizacional são fundamentais para o sucesso do projeto."

Respondente 65: "Uma empresa que possui muitos fluxos de processos "amarrados" pode atrapalhar a "filosofia" ágil de entrega e melhoria contínua. Burocracia demais é impedimento para o andamento das sprints e entregas."

Respondente 87: "Culturais:quando não há disponibilidade de recursos e envolvimento do alto escalão das empresas. Humanos: falta de conhecimento dá forma de trabalhar com projetos ágeis, ou seja, não tem conhecimento/experiência. Pelo fato de serem acostumado a projetos previsíveis, possuem aversão a mudança de perspectiva de trabalho"

Respondente 96: “Acredito que projetos ágeis se adequam bem a organizações da Administração Pública, desde que sejam colocados PO e Scrum Master, devidamente capacitados e com perfil para seguir a metodologia. A cultura da Administração pública, na qual os servidores efetivo possuem razoável grau de autonomia para desempenhar sua função, favorece a formação de times auto gerenciáveis. Por outro lado, a questão da hierarquia administrativa pode ser bem equilibrada com a presença de um líder que desempenhe papel de Scrum Master, desde que devidamente consciente de que é o Zelador do Processo e que seja um líder colaborador e não autoritário. No que se refere ao tamanho do time, a realidade mais comum dos “orgãos público é dispor de equipes pequenas e estas se tornam mais efetivas, se com autonomia, em projetos ágeis uma vez que irá prestigiar preferencialmente comunicação e resultado efetivo ao invés de formalização documental.”

Respondente 102: “Uma cultura bem definida influencia diretamente as pessoas na organização assim como remove as pessoas que não se encaixam ao modelo da mesma. Quando temos uma cultura bem definida e acompanhada pelos líderes, temos uma organização muito mais engajada e com melhores resultados.”

5.2 Resumo do capítulo

Este capítulo teve como objetivo descrever os resultados obtidos com a realização do *survey* no tocante a avaliação dos fatores críticos, humanos e técnicos, bem como os fatores culturais no tocante a adoção das metodologias ágeis. Foram apresentados fatores identificados na literatura através do mapeamento sistemático, buscando opiniões de profissionais experientes em projetos ágeis e seus relatos de experiência.

Conclui-se que foram apresentadas constatações para a influência de fatores técnicos, humanos e culturais em projetos de desenvolvimento de software ágeis que dos fatores apresentados nos gráficos 12 a 17 representam mais de 50% de concordância total para o seu sucesso efetivo. Destaca-se também que a concordância parcial teve expressiva manifestação nos resultados, demonstrando assim que diversos fatores identificados na literatura correspondem a fatores relevantes para este *survey*.

6 CONCLUSÃO

6.1 Análise Crítica dos Resultados

O objetivo principal de um mapeamento sistemático é reunir e catalogar um corpo de evidências para descrever o estado de conhecimento de um tópico ou questão em particular. Este catálogo ou mapa forma um recurso pesquisável que é publicado ao lado do relatório de MS e pode ser analisado para permitir aos leitores realizarem pesquisas futuras. Os resultados facilitam a análise de pesquisadores sobre o estado da arte em determinada área e sintetizam conhecimento para realização futura de revisões sistemáticas da literatura (JAMES et al., 2016).

A síntese de conhecimento permite aos pesquisadores identificar áreas da base de evidências que estão suficientemente representadas para permitir um MS significativo. Os mapas sistemáticos podem também ser utilizados para identificar evidências para outros estudos secundários, revelando deficiências nas evidências obtidas. Estas deficiências podem permitir que os autores façam recomendações de mudanças na prática ou ressaltem a necessidade de estudos futuros (JAMES et al., 2016)

A presente dissertação, ao utilizar o MS como método, foi capaz de identificar subcategorias de pesquisa e planejamento para o sucesso de projetos de software ágeis com base nos fatores humanos e culturais. Além disso, identificou-se lacunas no conhecimento dessa área, o que poderá subsidiar novos estudos. Outros trabalhos na área das Ciências da Computação também utilizaram o MS como metodologia e encontraram resultados importantes, consolidando os conceitos existentes sobre os temas pesquisados (JACINTO, 2010; JALALI, WOHLIN, 2010; OLIVEIRA, 2015).

O presente MS, identificou apenas 22 artigos. Esse número de trabalhos identificados indica que não existe ainda uma literatura sólida sobre o tema porém há diversas lacunas identificadas devem ser preenchidas por trabalhos futuros.

Em relação ao método de busca, a busca automática por strings identificou mais de 50% dos artigos do presente MS, o que corrobora com os estudos de Oliveira (2010), Jalali e Wohlin (2010) que também criaram suas strings de busca e obtiveram um número maior de artigos retornados através desse modelo.

Diante das questões de pesquisa desenvolvidas para este MS, encontrou-se uma frequência de artigos que as responde abaixo do esperado. A maior parte dos artigos não

responderam a Questão de Pesquisa 1, sobre fatores críticos de sucesso em projetos ágeis. Apenas seis trabalhos (E02, E03, E06, E11, E14 e E15) responderam esta questão, compondo uma lista de 14 fatores críticos para o sucesso em projetos ágeis, dentre eles fatores humanos e culturais. Percebe-se assim, que as empresas e seus colaboradores foram os temas que mais obtiveram fatores críticos de sucesso em projetos ágeis, o que leva a acreditar que à medida que estes fatores forem levados em consideração e resolvidas as problemáticas existentes em cada item, o sucesso é provável que ocorra. Para tanto, não basta apenas estes membros fazerem sua parte, os estudos já citados nesta questão mostram que tanto o projeto em si, como seus clientes fazem parte destes críticos fatores, mostrando que há uma inter-relação entre os temas propostos para organização destes.

A Questão 1.1, referente aos desafios relacionados aos aspectos técnicos envolvidos em projetos ágeis, foi respondida por oito artigos (E02, E04, E05, E10, E13, E14 E15 e E22), que apresentaram 14 desafios quanto aos aspectos técnicos que devem ser levados em consideração em projetos ágeis. Percebe-se que estes desafios possuem uma íntima relação com a gestão empresarial, a qual deve estar preparada para acolher e solucionar os mais diversos problemas, principalmente quando mudanças são impostas, neste caso a adoção de projetos ágeis, com o intuito de aprimorar a qualidade da empresa. Outro fato que merece chamar atenção, é que esta gestão trabalha em prol da empresa, porém para que isso ocorra, seus colaboradores devem ser trabalhados e vistos como membros primordiais para o alcance de tal objetivo. Dessa forma, é justificável que os estudos deram ênfase a estes desafios.

Dezessete trabalhos (E01, E02, E04, E05, E06, E07, E09, E10, E11, E12, E13, E14, E15, E16, E17, E20, E21 e E22) responderam à questão 1.2 sobre quais eram os desafios relacionados aos aspectos culturais envolvidos em projetos ágeis. Apesar do total de desafios ter sido 22 (vinte e dois), é identificado que a mudança organizacional foi citada várias vezes, fato que merece ser frisado e discutido, por sua notoriedade em diversos artigos. Em muitos dos outros desafios, estes estão relacionados também a mudanças comportamentais que irão facilitar o desenvolvimento de projetos ágeis em empresas que pretendem melhorar sua qualidade como um todo.

No que se refere à questão 1.3, sobre quais são os desafios relacionados aos aspectos humanos envolvidos em projetos ágeis, oito artigos conseguiram responde-la (E01, E05, E06, E09, E10, E11, E14 e E19) apontando 7 desafios. Quando comparamos estes desafios quanto aos aspectos humanos, percebe-se que existe uma relação educacional em sua grande maioria.

Consequentemente, tais obstáculos podem ser ultrapassados a partir do momento que os indivíduos se demonstrem mais abertos e aceitem as mudanças que estão ocorrendo, adquirindo assim, mais conhecimento para que possam se tornar sujeitos pró-ativos no desenvolvimento de projetos ágeis. Além disso, aspectos culturais também foram apontados como desafios dentro dos fatores humanos, demonstrando a necessidade de se ultrapassar essa barreira.

A questão menos respondida, por fim, foi a QP2 acerca de como os fatores humanos e culturais impactam no sucesso dos projetos ágeis. Apenas dois artigos responderam a este questionamento (E02 e E08) e isto demonstra uma lacuna do conhecimento sobre essa área. Quando revisamos a questão anterior, sobre os desafios quanto aos aspectos humanos e comparamos aos fatores humanos nesta questão, pode-se perceber que a solução das QP1.3 encontram-se justamente na QP2, pois neste item são apresentados comportamentos salutares que busquem a qualidade empresarial dentro da introdução de projetos ágeis. Além disso, esses fatores somente são alcançados quando o ser humano, seja ele membro da equipe ou cliente, está disposto a colaborar sendo um participante ativo em tal desenvolvimento.

No geral, nossos resultados são consistentes e atestam a grande relação existente entre o desenvolvimento de software ágeis, fatores técnicos, humanos e culturais da organização. Nossa estudo identificou a necessidade de melhorar a robustez da avaliação empírica das pesquisas existentes, a falta de suporte de ferramentas de gestão e múltiplas pistas sobre os fatores técnicos, humanos e culturais que influenciam no sucesso de projetos de software ágeis que merecem uma investigação mais aprofundada.

6.2 Principais Contribuições

A maioria das publicações relevantes identificadas não responderam às cinco perguntas de pesquisa levantadas por essa dissertação de mestrado, identificando assim, uma carência na literatura sobre a influência dos fatores humanos e culturais no sucesso dos projetos de software ágeis. Destacamos que este é o primeiro trabalho na literatura brasileira que avalia, através de um MS, a área de software ágeis à luz dos fatores técnicos, humanos e culturais.

A principal contribuição deste trabalho é apresentar uma visão abrangente sobre o impacto de fatores técnicos, humanos e culturais em projetos de software ágeis, discutindo as principais abordagens, metodologias e técnicas de gestão da equipe que vem sendo utilizados.

Este mapeamento consolida os conceitos existentes sobre os fatores humanos no desenvolvimento, execução e sucesso dos software ágeis.

Concluímos que o MS é uma estratégia metodológica primordial para o melhor entendimento e visualização de temas em ciência da computação. Além disso, os resultados gerados por esta pesquisa poderão auxiliar o processo de desenvolvimento de *software* ágeis, bem como subsidiar pesquisas futuras.

6.3 Limitações e Dificuldades para a Realização da Pesquisa

Este estudo utilizou rigoroso método de pesquisa através do mapeamento sistemático conforme descrito no Capítulo 3. Entretanto, esta seção apresenta as limitações e dificuldades encontradas no planejamento e execução do MS

Como todo mapeamento sistemático a limitação consiste em encontrar todos os artigos relevantes existentes sobre determinada área pesquisada, foram utilizados quatro engenhos de busca relativos a área que estava sendo pesquisada. Com intuito de atacar essa lacuna, foram realizadas buscas manuais e a técnica snowballing em referências bibliográficas de artigos descartados ou que não atendiam a nenhuma QP.

Quanto à *string de busca*, a mesma foi elaborada por diversos termos, porém, algum termo importante pode não ter sido incluído. O objetivo da *string* inicial foi atingir o máximo de artigos evitando assim esta dificuldade, entretanto, é possível que existam artigos que não tenham sido incluídos como relevantes por não possuírem termos da *string* em seus títulos ou resumos/*abstract*.

Foram identificadas dificuldades na extração das informações dos estudos relevantes. Alguns artigos selecionados não apresentam informações claras e objetivas sobre as QP desse estudo, exigindo grande esforço na interpretação e análise dos resultados, podendo gerar um viés de resultados e análises críticas.

Esta pesquisa optou por não realizar uma avaliação da qualidade nos artigos selecionados, por não ser uma atividade obrigatória para os mapeamentos. Portanto, foram selecionados artigos que respondiam às perguntas de pesquisa, mas que poderiam não obedecer as melhores práticas de pesquisa científica. Isso dificultou a análise dos dados.

6.4 Trabalhos Futuros

Com a progressiva evolução do mercado de desenvolvimento de software, há um crescente incremento da atenção a questões referentes aos fatores humanos e culturais na produtividade, efetividade e sucesso do software desenvolvido. Nesse sentido, novas pesquisas sobre o tema serão necessárias para manter o direcionamento das grandes empresas desenvolvedoras nos fatores humanos, visando melhores posicionamentos no mercado e satisfação da sua equipe de trabalho.

Sob a luz do que foi abordado na presente dissertação, recomenda-se ampliar este MS para trabalhos futuros, focando em outros aspectos específicos dos fatores técnicos, culturais e humanos, incluindo um maior número de artigos. Além disso, outros estudos apresentam a possibilidade se serem desenvolvidos a partir desse MS, como o desenvolvimento de estudos relacionados as subáreas dos métodos ágeis, trabalhos desenvolvidos com o objetivo de encontrar soluções para a lacunas evidenciadas na presente pesquisa, trabalhos que utilizem métodos estatísticos nos dados levantados pelo MS, gerando novos conhecimentos à partir dos estudos investigados nessa dissertação. Por fim, recomenda-se a elaboração de um estudo de caso buscando vivenciar o processo de desenvolvimento de software da organização, bem como elaboração de revisão sistemática da literatura sobre a influência dos fatores técnicos, humanos e culturais no sucesso dos projetos de software ágeis a fim de completar os conhecimentos levantados pelo presente MS, estendendo o protocolo a novos engenhos de busca e busca manual a novas conferências e periódicos.

REFERÊNCIAS

AGILE ALLIANCE. Disponível em: <<http://www.agilealliance.org/>>. Acesso em 20 de Março de 2016.

AGILE PROJECT MANAGEMENT: creating innovative products. Boston: Addison-Wesley, 2004.

ANICHE, Mauricio Finavarro. Como a prática de TDD influencia o projeto de classes em sistemas orientados a objetos. 2012. 75 f. Dissertação (Mestrado) - Curso de Ciência da Computação, Departamento de Instituto de Matemática e Estatística, IME-USP, Universidade de São Paulo, São Paulo, 2012.

ARRUDA, L. V. (2012) Desenvolvimento Ágil de Software: uma análise sintética a partir da metodologia Kanban. Anais do VII CONNEPI – Congresso Norte Nordeste de Pesquisa e Inovação. Palmas, Tocantins. Disponível em: <<http://propi.ifto.edu.br/ocs/index.php/connepi/vii/paper/viewFile/3644/961>>. Acesso em 19 de abril de 2017.

ATKINSON, R.; ATKINSON, R.; SMITH, E.; BEM, D. Introdução à psicologia de Hilgard. Porto Alegre: Artmed. 1999.

BECK, K. Extreme Programming Explained: Embrace Change. Página 21 – 26. 2.ed. [S.l.]: Addison-Wesley, 2004.

BELFORT, Rheyelle Mota et al. Gestão por competências: um novo modelo de gerenciamento artigo gestão, informação e sociedade. Revista UNI. Imperatriz (MA), ano 2, n.2, p.39-53, janeiro/julho. 2012.

BELLUSCI, Silvia Meirelles. Doenças profissionais ou do trabalho. São Paulo: Editora Senac São Paulo, 2007.

BENNIS, W.G. Organizational development: its nature, origin and prospects. Mass. Addison, Wesley Publishing, 2010.

BITENCOURT, Claudia. A gestão de competências gerenciais e a contribuição da aprendizagem organizacional. Salvador: ENANPAD, 2012. (Tese de Doutorado).

BRANDÃO, H. P., & GUIMARÃES, T. A. Gestão de competências e gestão de desempenho: tecnologias distintas ou instrumentos de um mesmo constructo? Revista de Administração de Empresas, 2010.

CAMPOS, G.W.S. Considerações sobre a arte e a ciência da mudança: revolução das coisas e reforma das pessoas. São Paulo: Hucitec, 2007.

CARBONE, P. P. et al. Gestão por competências e gestão do conhecimento. Rio de Janeiro: Fundação Getúlio Vargas, 2011.

CIAMPONE, M. H. T.; KURCGANT, P. Gerenciamento de Conflitos e Negociação. Rio de Janeiro: Guanabara/Koogan, 2010.

CHIAVENATO, Idalberto. Recursos humanos: o capital humano das organizações. 9.ed. Rio de Janeiro: Elsevier, 2014.

COCKBURN, A. (2002) “Agile Software Development”. Boston: Addison Wesley. COCKBURN, A. HIGHSITH, J. (2001) “Agile Software Development: The People Factor,” Computer, pp. 131-133.

DEJOURS, Christophe. Psicodinâmica do trabalho. São Paulo: Atlas, 2010.

DIAS, Vera Lúcia Cavalcanti et al. Liderança e motivação. Rio de Janeiro: FGV, 2010.

DYBÅ, T.; DINGSØYR, T. Empirical Studies of Agile Software Development: A Systematic Review. *Information and Software Technology*, Butterworth-Heinemann Newton, MA, USA, v. 50, n. 9, p. 833-859, August 2008.

FAGUNDES, Maria Tereza. Aprendizagem e inovação organizacional. São Paulo: Atlas, 2012.

FLEURY, Maria Isabel Leme. Cultura e poder nas organizações. São Paulo: Atlas, 2011.

FORTUNATO, J. C. Motivação no trabalho. *Maringa Management: Revista de Ciências Empresariais*, v.2, n.1, p.20-25, jan/jun, 2008.

FOWLER, M.; HIGHSITH, J. Disponível em: <<http://www.drdobbs.com/opensource/the-agile-manifesto/184414755>>. Acesso em 20 de Março de 2016.

FREITAS H.; OLIVEIRA M.; SACCOL A. Z.; e MOSCAROLA J. O método de pesquisa survey. São Paulo/SP: Revista de Administração da USP, RAUSP, v. 35, nr. 3, Jul-Set. 2000, p. 105-112. Disponível em: <http://www.ufrgs.br/gianti/files/artigos/2000/2000_092_RAUSP.PDF>. Acesso em 08 de Fevereiro de 2017.

GARCIA, Lenise Aparecida Martins. Competências e habilidades: você sabe lidar com isso? Brasília: UNB, 2010.

HART, Chris. Doing a literature review: Releasing the social science research imagination. Sage, 1998.

HERMAM, F. Administração e serviços. São Paulo: Freitas Bastos, 2008.

HIGHSITH, Jim. Adaptative management: patterns for the e-business era. *Cutter IT Journal*, [S.I.], v. XII, n. 9, sep. 1999.

HIGHSITH, Jim. Agile software development ecosystems. Boston: Addison-Wesley, 2002.

JACINTO, Shirley da Silva. Um mapeamento sistemático da pesquisa sobre a influência da personalidade na engenharia de software. Dissertação de mestrado. 2010, 106 f. Centro de Informática da Universidade Federal de Pernambuco, Recife, setembro, 2010.

JALALI, S., WOHLIN, C. Agile Practices in Global Software Engineering - A Systematic Map. 2010 5th IEEE International Conference on Global Software Engineering DOI: 10.1109/ICGSE.2010.14, 2010.

JAMES, K.L; RANDALL, N.P.; HADDAWAY, N.R. A methodology for systematic mapping in environmental sciences. Environmental Evidence The official journal of the Collaboration for Environmental Evidence, 2016.

JESUS FILHO, Aldo de, G. Porque é preciso comunicar?. São Paulo: Atlas, 2010.

KITCHENHAM, B. & Charters, S. (2007), Guidelines for performing systematic literature reviews in software engineering, Technical Report EBSE-2007-01, School of Computer Science and Mathematics, Keele University.

KITCHENHAM, B., et al. Systematic literature reviews in software engineering—a tertiary study. *Information and Software Technology*, v. 52, n. 8, p. 792-805, 2010.

KITCHENHAM, B., Charters, S., Budgen, D., Brereton, P., Turner, M., Linkman, S., Jørgensen, M., Mendes, E., and Visaggio, G. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical report, Keele University, UK.

KITCHENHAM, B.; DYBA, T.; JORGENSEN, M. Evidence-based software engineering. In ICSE'04: Proceedings of the 26th International Conference on Software Engineering, p. 273-281, Washington, DC, USA, 2004.

KITCHENHAM, B. "Procedures for Performing Systematic Reviews", Technical Report Software Engineering Group, Keele University, Australia, 2004.

LÉVY, Pierre. As tecnologias da inteligência. O futuro do pensamento na era da informática. Rio de Janeiro, 2010.

LIVERMORE, J. A. "Factors that Impact Implementing an Agile Software Development Methodology". SoutheastCon. Proceedings. IEEE.p. 69-72, 2007.

LOPEZ-HERREJON, R. E., LINSBAUER, L., EGYED, A. A systematic mapping study of search-based software engineering for software product lines. *Information and Software Technology*. v. 61, p. 33–51, 2015.

LUZ, Ricardo. Gestão do clima organizacional. Rio de Janeiro: Qualitymark, 2010.

MACEDO, Ana Bárbara Lins de, SPÍNOLA, Rodrigo. Ciclos de vida do software. *Revista Engenharia de software Magazine*, n. 36, 2013.

MACIEL, C. de O; CAMARGO, C. Comprometimento, Satisfação e Cooperação no Trabalho: Evidências da Primazia dos Aspectos Morais e das Normas de Reciprocidade Sobre o Comportamento. RAC, Curitiba, v. 15, n. 3, art. 4, pp. 433-453. 2011

MACHADO, A; MEDINA, M. Competências organizacionais. Rio de Janeiro: Qualitymark, 2009.

MANIFESTO FOR AGILE SOFTWARE DEVELOPMENT. Disponível em: <<http://agilemanifesto.org/>>. Acesso em: 23/11/2015.

MATTOS, R. A. Desenvolvimento de recursos humanos e mudança organizacional, Rio de Janeiro, LTC/ANFUP, 2012.

MAXIMIANO, A. C. A. Teoria geral da administração. Da escola científica à competitividade na economia globalizada. São Paulo: Atlas, 2010.

MEDEIROS, C. A. F. et al. Comprometimento Organizacional: o Estado da Arte da Pesquisa no Brasil. In: EnANPAD - Encontro Nacional dos Programas de Pós-Graduação em Administração 2002, Set. 22-25; Salvador: Bahia; 2002.

MELO, Claudia de O, FERREIRA, and Gisele R. M. Adoção de métodos ágeis em uma Instituição Pública de grande porte - um estudo de caso., In Proceedings of the Brazilian Workshop for Agile Methods (WBMA 2010) in the Brazilian Conference on Agile Methods (Agile Brazil 2010), pp. 112- 125. June, 2010. Disponível em: <http://www.agilcoop.org.br/files/WBMA_Melo_e_Ferreira.pdf>

MELO, Vanessa Pontes Chaves de. A comunicação interna e sua importância nas organizações. São Paulo: Núcleo, 2006.

MERLI, A. M. Gestão por competências. Rio de Janeiro: Qualitymark, 2008.

MOE, N. B.; DINGSØYR, T. Scrum and Team Effectiveness: Theory and Practice. Agile Processes in Software Engineering and Extreme Programming. Lecture Notes in Business Information Processing, 2008, Volume 9, Part 1, 11-20.

MOSCOVICI, Fela. Desenvolvimento interpessoal. Rio de Janeiro: Livros Técnicos e Científicos: S/A, 2014.

MORAIS, Francisco Eduardo de oliveira. Organização e implementação da gestão por projetos. Brasília: CETEB, 2009.

NIELSEN, J. L. (2002), “Critical Success Factor for Implementing an ERP system in a University Environment: A Case study from the Australia, Honours Dissertation”, Brisbane, Australia, Griffith University, 2002, 201p.

OAKLEY, A., GOUGH, D., JAMES, T. The politics of evidence and methodology: lessons from the EPPI-Centre. Evid Policy.v. 1, n. 1, p. 5–31, 2005.

OLIVEIRA, L. L. S. Uma análise das práticas de sustentabilidade em melhoria de processos de negócios: um mapeamento sistemático. 2015, 122f. Dissertação de mestrado – Universidade Federal de Pernambuco, Centro de Informática, Pós-graduação em Ciência da Computação. Recife, PE 2015.

OLIVEIRA, Sílvio L. Sociologia das organizações: uma análise do homem e das empresas no ambiente competitivo. São Paulo: Pioneira, 2011.

PAULA, G.M. Estruturas organizacionais: o papel do gestor de nível intermediário. 2007. 148f. Dissertação (Mestrado da Faculdade de Ciências Empresariais), Universidade FUMEC, Belo Horizonte, 2007.

PEERSMAN, G. A descriptive mapping of health promotion in young people, London: EPPI-Centre, Social Sciences Research Unit, Institute of Education, University of London; 1996. Disponível em: <http://www.eppi.ioe.ac.uk/eppi/Evidence/EPPI_reviews/Health_promotion/Review2/des_map_hp_studies_yp.pdf>. Acessado em 19 de outubro de 2016.

PENROD, J.; PRESTON, D.B., CAIN, R. & STARKS, M.T. A discussion of chain referral as a method of sampling hard-to-reach populations. *Journal of Transcultural nursing*, vol 4. no 2.April, 2003. 100-107p.

PETERSEN, Kai et al. Systematic mapping studies in software engineering. In: 12th International Conference on Evaluation and Assessment in Software Engineering. sn, 2008.

PINHEIRO, Carmen M. de. Autonomia organizacional: Um desafio para administradores. Brasília: UNB, 2006.

PIRZADEH, L. (2010) "Human Factors in Software Development: A Systematic Literature Review". Chalmers university of technology. Göteborg, Sweden.

QUINN, Robert E. et al. Competências gerenciais: princípios e aplicações. Rio de Janeiro: Elsevier, 2009.

REGO, Gaudêncio Torquato do. Comunicação empresarial: comunicação institucional. São Paulo: Summus, 2014.

RIBEIRO, João. Comportamento organizacional. São Paulo: Eduweb, 2009.

RODRIGUES, José dos Reis G. Liderança num ambiente de inovações organizacionais. CEPEAD/FACE/UFMG. 2005.

RODRIGUEZ E.O.L; TREVIZAN M.A.; SHINYASHIKI, G.T. Reflexões conceituais sobre comprometimento organizacional e profissional. *Rev Latino-am Enf.*, v 16, n.3, mai-jun, 2008.

SALLES, T.R. Comunicação nas empresas. São Paulo: McGraw-Hill, 2009.

SALGADO, J. Carlos Z. Psicologia, organizações e trabalho no Brasil. Porto Alegre: ARTMED, 2011.

SCHWABER, K.; SUTHERLAND, J. Guia do Scrum.2011. Disponível em <<http://www.Scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-20Portuguese%20BR.pdf>>. Acesso em 24 março de 2016.

SCHWARTZ, J. I. ,Construction of software. In: *Practical Strategies for Developing Large Systems*. Menlo Park: Addison-Wesley, 1st. ed., 1975.

SENNA, R. A. Desenvolvimento de recursos humanos e mudança organizacional, Rio de Janeiro, LTC/ANFUP, 2012.

SEVERINO, D. S. J.; COSTA, F., J. Mensuração e Escalas de Verificação: uma Análise Comparativa das Escalas de Likert e Phrase Completion. Revista Brasileira de Pesquisas de Marketing, Opinião e Mídia. São Paulo, v. 15, p. 1-16, out. 2014.

SILVA, Reinaldo Oliveira da. Teorias da administração. São Paulo: Pioneira Thomson Learning, 2014.

SILVA, M. J. P. da. Comunicação tem remédio: a comunicação nas relações interpessoais. São Paulo: Edições Loyola, 2012.

SOMMERVILLE, Ian. Software Engineering. 7ed. Harlow: Addison Wesley, 2007. 720p.

STANDISH GROUP, Chaos Manifesto 2015, 2015. Disponível em <https://www.infoq.com/articles/standish-chaos-2015>. Acesso em 05 jan de 2017.

STEFFEN, Juliana Berossa. O que são essas tais de Metodologias Ágeis? Artigo de 23 de Janeiro de 2012. Disponível em https://www.ibm.com/developerworks/community/blogs/rationalbrasil/entry/mas_o_que_s_c3_a3o_essas_tais_de_metodologias_c3_a1geis?lang=em. Acesso em 24 março de 2016.

TAMOYO, Stephen P. Administração, mudanças e perspectivas. São Paulo: Saraiva, 2010.

TELES, Vinicius M. Extreme Programming. São Paulo: Novatec, 2007.

TELES, Vinicius Manhães. Um Estudo de Caso da Adoção das Práticas e Valores do Extreme Programming.2005. 180 f. Dissertação (Mestrado em Informática) , Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2005.

TOLEDO, Flávio e MILIONI, B. Dicionário de recursos humanos. São Paulo: Atlas, 2012.

TOMÁS, Mário Rui Sampaio. Métodos ágeis: características, pontos fortes e fracos e possibilidades de aplicação. Lisboa : Centro de Investigação em Inovação Empresarial e do Trabalho. 2009.

TRAVASSOS, G., BIOLCHINI J. Revisões Sistemáticas Aplicadas a Engenharia de Software. In: XXI SBES – Brazilian Symposium on Software Engineering, João Pessoa, PB, Brasil, 2007.

UBEDA, Cristina Lourenço. A gestão por competência em uma empresa de pesquisa e desenvolvimento. Dissertação de mestrado. Universidade de São Paulo, São Carlos-SP, 2008.

WAZLAWICK, Raul Sidnei. Engenharia de software: conceitos e práticas. São Paulo : Elsevier Editora, 2013

ZANINI, Marco T. Confiança: o principal ativo intangível de uma empresa. Rio de Janeiro: Elsevier, 2007.

APÊNDICE A

Questionário Online utilizado no Survey.

Seção 1 de 2

Um Estudo sobre a Influência de Fatores Humanos e Culturais em Projetos Ágeis de Software

Prezado(a) colaborador(a):
Meu nome é Rickson César da Cruz Silva, aluno do Mestrado Profissional em Ciências da Computação pelo CIn/UFPE, sob a orientação da professora Carina Frota Alves.

Gostaria de contar com a sua valiosa contribuição respondendo nossa pesquisa sobre a Influência de Fatores Humanos e Culturais em Projetos de Software Ágeis.

Tempo estimado: 10 perguntas/Aproximadamente 7 minutos.

Prazo para respostas: 07/02/2017

Em caso de dúvidas, favor entrar em contato:
Rickson (rccs@cin.ufpe.br)

Como forma de agradecimento sortearemos no dia 11/02/2017 um vale presentes para aquisição de livro pelo site Saraiva, aos participantes da pesquisa.

Sua contribuição será de grande relevância para o nosso trabalho!

Nome: (opcional)

Texto de resposta curta

Email: (se quiser participar do sorteio)

Texto de resposta curta

Um Estudo sobre a Influência de Fatores Humanos e Culturais em Projetos Ágeis de Software

*Obrigatório

1. Que tipo de empresa ou Instituição em que você trabalha? *

- Empresa Privada
- Órgão Público
- Outro:

2. Qual o cargo/função/papel que ocupa atualmente? *

- Analista de TI
- Consultor de TI
- Desenvolvedor
- Gerente de projetos
- Líder de times
- Scrum Master
- Outro:

3. Qual a sua experiência na participação de projetos ágeis de software? *

- menos de 1 ano
- entre 1 e 2 anos
- entre 2 e 3 anos
- mais de 3 anos

4. Quantos membros é formada sua equipe de desenvolvimento? *

- 1 a 10 participantes
- 11 a 20 participantes
- 21 a 30 participantes
- 31 a 50 participantes
- Acima de 50 participantes

5. Quais abordagens ágeis você utiliza ou utilizou? *

- Scrum
- XP
- TDD
- FDD
- Outro: _____

6. Na sua opinião, os fatores a seguir podem comprometer o sucesso de projetos ágeis de software? indique o grau de concordância ou discordância: *

	Discordo Totalmente	Discordo Parcialmente	Não concordo/Nem discordo	Concordo Parcialmente	Concordo Totalmente
Dificuldade na adoção de práticas ágeis	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Escopo do projeto mal definido	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estimativas de tempo inadequadas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estimativas de custo inadequadas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mudanças frequentes dos requisitos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Natureza do projeto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Técnicas de Eng. Software Inadequadas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

7. Os fatores humanos listados a seguir influenciam o sucesso de projetos ágeis de software? indique o grau de concordância ou discordância: *

	Discordo Totalmente	Discordo Parcialmente	Não concordo/Nem discordo	Concordo Parcialmente	Concordo Totalmente
Ausência de liderança	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Alocação inadequada de recursos humanos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Baixa experiência da equipe	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Capacitação da equipe (Treinamentos)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Diálogo Deficiente com o cliente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dificuldade de comunicação entre membros da equipe	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Falta de alinhamento estratégico entre a equipe e o cliente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Reuniões longas e improdutivas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

8. Os fatores organizacionais listados a seguir influenciam o sucesso de projetos ágeis de software? indique o grau de concordância ou discordância: *

	Discordo Totalmente	Discordo Parcialmente	Não concordo/Nem discordo	Concordo Parcialmente	Concordo Totalmente
Ambiente de trabalho inadequado	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Baixo comprometimento da alta gestão	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Confiança mútua (equipe x organização)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Definição de Metas e prêmios	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Disponibilidade para mudanças	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Flexibilização da organização quanto à novas metodologias de desenvolvimento	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gestão através de indicadores visuais	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Missão/Valores da sua organização inapropriados ao projeto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

9. Na sua opinião, quais são os 3 principais desafios encontrados nos projetos ágeis que você participa ou participou? *

Sua resposta

10. Na sua opinião, de que forma os fatores humanos e culturais da organização podem influenciar no sucesso de projetos ágeis? *

Sua resposta

[VOLTAR](#)

[ENVIAR](#)

 Página 2 de 2

Nunca envie senhas pelo Formulários Google.

APÊNDICE B

Planilha eletrônica do Microsoft Excel utilizada para organização dos artigos após a seleção: