



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Proposta de metodologia de gestão de riscos, práticas
e medições para projetos geridos por meio de
métodos ágeis**

Marcito Ribeiro Madeira Campos

Dissertação apresentada como requisito parcial para conclusão do
Mestrado Profissional em Computação Aplicada

Orientadora
Profa. Dra. Simone Borges Simão Monteiro

Brasília
2019

Ficha catalográfica elaborada automaticamente,
com os dados fornecidos pelo(a) autor(a)

CM319p Campos, Marcito Ribeiro Madeira
Proposta de metodologia de gestão de riscos, práticas e
medições para projetos geridos por meio de métodos ágeis /
Marcito Ribeiro Madeira Campos; orientador Simone Borges
Simão Monteiro. -- Brasília, 2019.
107 p.

Dissertação (Mestrado - Mestrado Profissional em
Computação Aplicada) -- Universidade de Brasília, 2019.

1. Gestão de projetos. 2. Gestão de riscos. 3. Métodos
ágeis. 4. Insucesso em projetos. I. Monteiro, Simone Borges
Simão, orient. II. Título.



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Proposta de metodologia de gestão de riscos, práticas
e medições para projetos geridos por meio de
métodos ágeis**

Marcito Ribeiro Madeira Campos

Dissertação apresentada como requisito parcial para conclusão do
Mestrado Profissional em Computação Aplicada

Profa. Dra. Simone Borges Simão Monteiro (Orientadora)
CIC/UnB

Prof. Dr. Ari Melo Mariano, Ph.D Prof. Dr. Moacyr Amaral Domingues Figueiredo
Universidade de Brasília Universidade Federal Fluminense

Profa. Dra. Aletéia Patrícia Favacho de Araújo
Coordenadora do Programa de Pós-graduação em Computação Aplicada

Brasília, 24 de Janeiro de 2019

Dedicatória

Dedico este mestrado, primeiramente a Deus, por torná-lo possível.

Dedico também aos meus pais pelo dom da vida.

Dedico também a a todos meus amigos e colegas de mestrado e também todos aqueles inconformados que, assim como eu, lutam para que de forma positiva consigam mudar a própria vida e o meio em que vivem de alguma forma.

Agradecimentos

Sozinhos não somos ninguém e não chegamos a lugar algum.

Ao chegar à conclusão dessa dissertação gostaria de agradecer a todos que me ajudaram a alcançar esse importante objetivo.

À Deus, por me conceder vida, saúde e capacidade física, emocional e intelectual.

Aos meus pais que sempre acreditaram, ensinaram e apoiaram seus filhos.

À minha orientadora Prof.[a] Dr.[a] Simone Borges Simão Monteiro, pelas relevantes orientações fornecidas em todas as etapas dessa pesquisa e pela paciência nessa caminhada.

Ao Prof. Dr Ari Mariano por toda paciência e apoio prestado durante esse período.

A todos os colegas de mestrado que sempre me apoiaram e ajudaram em toda essa jornada.

Finalmente, mas não menos importante, a todo corpo docente do Mestrado Profissional em Computação Aplicada do CIC da UnB, por todos os conhecimentos e orientações compartilhados, que tornaram possível a conclusão dessa pesquisa.

Resumo

Os métodos ágeis de gestão de projetos surgiram como forma de contraponto ao um modelo muito estruturado de gestão conhecido como método tradicional. Devido a sua forma de trabalho agilistas pregam que a utilização desses métodos minimizam os riscos do projeto e impulsionam o sucesso do mesmo, no entanto, mesmo com sua utilização têm-se uma alta taxa de insucesso em projetos. Dessa forma, o objetivo principal desse estudo é apresentar um novo desenho para o gerenciamento de riscos em projetos que utilizam métodos ágeis, visando preencher o gap identificado na gestão de riscos em métodos ágeis. Para tanto foi desenvolvida uma pesquisa contendo 04 constructos baseados em Fatores e 04 constructos baseados em Critérios de insucesso em projetos, que envolveu 82 especialistas em Gestão de Projetos para verificar sua percepção com relação a esses fatores e critérios. Dessa forma foi possível identificar que dois constructos têm um maior impacto no insucesso de um projeto, tais constructos estão relacionados a fatores técnicos e humanos do time.

Com base nos dados levantados foi desenvolvido inicialmente um prisma para Gestão de Riscos que posteriormente originou um quadro kanban de riscos e uma metodologia aplicável ao Scrum. Essa metodologia adiciona duas cerimônias ao Scrum para a avaliação dos riscos do projeto, bem como uma nova forma de avaliação proposta com base no *planning poker*. Também foram propostas práticas e medições visando minimizar o impacto negativo das variáveis mais impactantes ligadas constructos identificados como mais nocivos a um projeto ágil. O estudo foi aplicado a um projeto real para sua validação e se mostrou, ao final do período de análise, uma prática factível e eficiente. Os resultados da aplicação dessa metodologia, bem como as práticas e medições são demonstrados neste trabalho. Ao final também são evidenciadas propostas de trabalhos futuros que podem proporcionar uma melhoria no método.

Palavras-chave: Gestão de projetos, gestão de risco, métodos ágeis, insucesso em projetos

Abstract

Agile project management methods have emerged as a counterpoint to a very structured management model known as the traditional method. Due to their way of working agilists argue that the use of these methods minimizes project risks and boosts the success of the project, however, even with their use they have a high project failure rate. Thus, the main objective of this study is to present a new design for risk management in projects that use agile methods, aiming to fill the identified gap in risk management in agile methods. For this, a research was developed containing 04 constructs based on Factors and 04 constructs based on Criteria of failure in projects, that involved 82 experts in Project Management to verify their perception with respect to these factors and criteria. In this way it was possible to identify that two constructs have a greater impact on the failure of a project, such constructs are related to technical and human factors of the team.

On the basis of the data collected a prism was initially developed for Risk Management that later gave rise to a kanban risk framework and a methodology applicable to Scrum. This methodology adds two Scrum ceremonies for project risk assessment as well as a new evaluation form proposed based on planning poker. Practices and measurements were also proposed in order to minimize the negative impact of the most impacting variables linked to constructs identified as most harmful to an agile project. The study was applied to a real project for its validation and showed, at the end of the analysis period, a feasible and efficient practice. The results of the application of this methodology, as well as the practices and measurements are demonstrated in this work. At the end are also evidenced proposals of future work that can provide an improvement in the method.

Keywords: Project management, risk management, agile methods, project failure

Sumário

1	Introdução	1
1.1	Contextualização	1
1.2	Definição do Problema	3
1.3	Motivação	4
1.4	Objetivo	4
1.4.1	Objetivos Específicos	4
1.5	Estruturação do Trabalho	5
1.6	<i>Design Sciences Research Methodology</i>	5
1.6.1	Identificação do problema	6
1.6.2	Definição dos objetivos da solução	7
1.6.3	<i>Design</i> e Desenvolvimento	7
1.6.4	Demonstração	8
1.6.5	Avaliação	9
1.6.6	Comunicação	9
2	Referencial Teórico	10
2.1	Gestão de projetos	10
2.1.1	Metodologia tradicional ou cascata	11
2.1.2	Metodologia ágil	17
2.2	Gestão de riscos em projetos	36
2.2.1	Gestão de riscos na metodologia tradicional	39
2.2.2	Gestão de riscos em projetos de TI	42
2.2.3	Gestão de riscos em metodologias ágeis	43
2.3	Fatores e critérios de sucesso e insucesso em projetos	44
2.4	Modelagem por equações estruturais	45
2.4.1	Modelagem de equações estruturais por mínimos quadrados parciais (PLS-SEM)	47
2.5	Métricas e indicadores de desempenho	52

3	Metodologia de Pesquisa	55
3.1	Classificação da pesquisa	55
4	Pesquisa para identificar os fatores críticos de sucesso de um projeto	59
5	Metodologia Proposta	69
5.1	Quadro para gestão visual de riscos	70
5.1.1	Riscos de Projeto	72
5.1.2	Riscos da Iteração	73
5.2	Cerimônias	74
5.2.1	<i>Planning Poker</i>	77
5.2.2	<i>Risk Evaluation Poker</i>	78
6	Aplicação da Metodologia	80
6.1	Visão Geral	80
6.2	Avaliação de riscos	87
6.3	Proposta de medições	90
7	Conclusão	92
	Referências	95
	Apêndice	102
A	Questionários acerca de fatores e critérios de insucesso em projetos	103

Lista de Figuras

1.1	Sucesso em projetos de desenvolvimento de Software.	2
1.2	<i>Roadmap</i> da pesquisa, segundo a metodologia DSRM.	6
2.1	Metodologia tradicional - abordagem linear.	11
2.2	Metodologia tradicional - abordagem incremental.	12
2.3	Inter-relação de componentes do Guia PMBOK.	13
2.4	Interação entre grupos de processos do Guia PMBOK.	14
2.5	Áreas de conhecimento e grupos de processo.	14
2.6	Processos do método PRINCE2.	16
2.7	<i>Lean</i> e metodologias e <i>frameworks</i> ágeis.	17
2.8	Metodologia ágil - modelo iterativo.	18
2.9	<i>Processo Scrum</i>	21
2.10	Fluxo <i>Scrum</i>	22
2.11	Gráficos de <i>Burnup</i> e <i>Burndown</i>	25
2.12	<i>Exemplo de quadro Kanban</i>	27
2.13	<i>Princípios e práticas do método Kanban</i>	29
2.14	<i>Variáveis de projeto - Abordagem tradicional x DSDM</i>	31
2.15	Processo do método <i>Feature Driven Development</i>	32
2.16	<i>Agile Unified Process</i>	33
2.17	Princípios, estrutura e processos para gestão de riscos.	37
2.18	Processo de gestão de riscos.	39
2.19	Modelo de gestão de riscos para metodologia <i>Scrum</i>	43
2.20	Crescimento do uso da abordagem PLS.	46
2.21	Modelo conceitual de equações estruturais.	47
2.22	Modelos de medição reflexivo.	49
2.23	Modelos de medição formativo.	51
3.1	Exemplo de gráfico com regiões de rejeição em Teste de Hipóteses.	57
3.2	Cálculo da amostra para 95% de confiança.	58

4.1	Cálculo da amostra para a quantidade aproximada de respondentes.	61
4.2	Configuração do Algoritmo PLS.	61
4.3	Modelo desenvolvido para validação do questionário de fatores e critérios de insucesso em projetos.	63
4.4	Configuração do software para rodar o teste de <i>bootstrapping</i>	64
4.5	Valores atingidos para o critério de rho_A.	65
4.6	Valores de AVE atingidos.	65
4.7	Tabela de de HTMT atingidos pelas variáveis analisadas.	66
4.8	Gráfico de valores de HTMT atingidos pelas variáveis analisadas.	66
5.1	Prisma de gestão de riscos proposto.	69
5.2	Proposta de quadro gestão visual de riscos em projetos ágeis.	71
5.3	Configuração de post its.	72
5.4	Código de cores para riscos de projeto.	78
5.5	Código de cores para riscos de iteração.	79
5.6	Matriz de Probabilidade x Impacto x Detecção.	79
6.1	Estrutura Organizacional do Projeto.	82
6.2	Quadro colaborativo para identificação de riscos de projeto e da Sprint. . .	83
6.3	Quadro de Riscos preenchido na primeira <i>Sprint</i>	84
6.4	Quadro de Riscos preenchido depois da <i>Sprint</i> 3.	86
6.5	Roadmap do projeto.	88
6.6	Risco de projeto identificado.	88

Lista de Tabelas

2.1	Definição de sucesso em projetos de TI [1]	45
5.1	Frequência de utilização das ferramentas/técnicas de riscos. Adaptado de: [2]	75
A.1	Fatores de insucesso - Organizacional	103
A.2	Fatores de insucesso - Pessoas	104
A.3	Fatores de insucesso - Processos	105
A.4	Fatores de insucesso - Técnico	106
A.5	Critérios de insucesso - Organizacional	106
A.6	Critérios de insucesso - Pessoas	106
A.7	Critérios de insucesso - Processos	107
A.8	Critérios de insucesso - Técnico	107
A.9	Critérios de sucesso em projetos	107

Capítulo 1

Introdução

Neste capítulo serão apresentadas as informações principais que contextualizam a problemática a ser tratada, a motivação, a definição do problema e os objetivos geral e específicos da pesquisa.

1.1 Contextualização

O cenário atual demonstra claramente a dependência de empresas e organismos federais da Tecnologia da Informação (TI), seja como forma de impulsionar o crescimento dos negócios seja para atendimento ao público e atingimento de sua missão institucional, como no caso de órgãos públicos.

A evolução tecnológica, a busca por maior eficiência nos processos, as constantes mudanças nos requisitos definidos por entidades regulatórias e a volatilidade dos mercados traz a necessidade de que os projetos de TI possam ser rapidamente adaptados para acompanhar estas demandas. A rigidez do modelo tradicional de gestão, com o excesso de burocracia e necessidade de extensa documentação para mudanças dificulta essa rápida adaptação. Dessa forma os métodos ágeis se tornam uma proposta muito mais atraente para esse tipo de projeto.

Desde de 1994 o *The Standish Group* elabora anualmente o que chama de *CHAOS Report*, um relatório sobre a situação mundial da indústria de desenvolvimento de software e apresenta também uma comparação entre o sucesso de projetos gerenciados da forma tradicional e aqueles gerenciados de forma ágil [3].

Projetos bem-sucedidos são aqueles que são finalizados dentro do prazo e custo estimados e contemplam todos os requisitos solicitados pelo cliente [1]. Eventos ou condições incertas, caso ocorram, podem ter efeito positivo ou negativo em um ou mais objetivos do projeto como escopo, cronograma, custo e qualidade, a estes eventos ou condições incertas dá-se o nome de risco do projeto [4]. Segundo o *CHAOS Report* do ano de 2015

de todos os projetos estudados, gerenciados de forma ágil, apenas 39% obtiveram sucesso [3], conforme ilustrado na Figura 1.1:

METODOLOGIA ÁGIL x TRADICIONAL EM DESENVOLVIMENTO DE SOFTWARE				
Tamanho	Método	Sucesso	Completados com Desvios	Fracasso
Projetos de todos os tamanhos	Ágil	39%	52%	9%
	Tradicional	11%	60%	29%
Projetos de Grande Porte	Ágil	18%	59%	23%
	Tradicional	3%	55%	42%
Projetos de Porte Médio	Ágil	27%	62%	11%
	Tradicional	7%	68%	25%
Projetos de Pequeno Porte	Ágil	58%	38%	4%
	Tradicional	44%	45%	11%
Desenvolvimento de todos os projetos de software de 2011 a 2015, de acordo com a nova base de dados do CHAOS segmentados pela metodologia ágil e tradicional. O número total de projetos de software ultrapassa a marca de 10.000				

Figura 1.1: Sucesso em projetos de desenvolvimento de Software (Fonte: [3]).

É incontestável que as metodologias ágeis trouxeram um enorme ganho para a indústria, como pode ser verificado no quadro acima, que evidencia claramente uma evolução das metodologias ágeis frente a metodologia tradicional com relação as taxas de sucesso para o desenvolvimento de projetos de TI, no entanto, é claramente perceptível que ainda há muito o que evoluir afim de maximizar o sucesso deste tipo de projeto.

Nesse sentido a gestão adequada dos riscos do projeto torna-se premissa para que os projetos se desenvolvam de forma adequada e gerem maior valor possível as partes interessadas. Para que isso ocorra é necessário empenho e comprometimento da organização e equipe do projeto na realização de atividades contínuas de identificação, análise, avaliação e monitoramento dos riscos dos projetos, internos e externos a organização, com vistas a assegurar o bom desempenho do projeto e seu alinhamento com os objetivos do negócio. Dessa forma, conseguir diagnosticar o desempenho do projeto e monitorar os fatores de risco é uma condição fundamental para viabilizar o sucesso do projeto.

No momento atual existem diversas literaturas que tratam a gestão de riscos em projetos, as quais oferecem conjuntos de processos no sentido de facilitar o gerenciamento de riscos de um projeto, tais como a norma ABNT ISO 31000 [5], o *Project Management Body of Knowledge (PMBOK)* [4], o *framework* COSO [6], dentre outros, os quais forneceram grande base para o desenvolvimento dessa pesquisa e serão apresentados no Capítulo 2 deste trabalho.

1.2 Definição do Problema

As literaturas que tratam a gestão de risco, em geral, tem suas orientações e abordagens elaboradas de forma genérica e suas recomendações comumente são superficiais [5][7][8]. Elas apresentam conhecimentos estruturados em torno de conjuntos de princípios e de orientações gerais para a aplicação de conhecimentos e técnicas para todo o ciclo do processo de gestão de riscos, contemplando desde o estabelecimento do contexto, avaliação e análise, até o monitoramento dos riscos.

Por possuírem escopo e objetivos amplos de sua aplicação em diversos contextos, essas abordagens e modelos representam um desafio as organizações para sua implementação, pois apesar de fornecer os princípios e orientações gerais, não aprofundam nas diretrizes de como devem ser implementados [5][7]. Outrossim, listam uma série de ferramentas e técnicas que podem ser utilizadas, no entanto deixam uma lacuna quanto a exemplos práticos de aplicação e combinação entre estas ferramentas e sua utilização de forma prática [8].

A situação se agrava quando se trata de metodologias ágeis, pois atualmente há uma escassez de investigação sobre a aplicação de procedimentos de gestão de risco para métodos ágeis [9]. Métodos ágeis, como o *Scrum*, necessitam ainda de práticas adequadas de gerenciamento de riscos e devem possuir uma estrutura robusta de gerenciamento para que seja possível melhorar a execução do projeto [9]. Processos de desenvolvimento de software ágil, na melhor das hipóteses, abordam riscos de forma implícitas e que metodologias que sofrem da falta de gestão de riscos sofrem com as seguintes deficiências [10]:

- Incapacidade de tomar decisões sobre riscos e recompensas;
- Falha em identificar estratégias adequadas de resposta ao risco com base na exposição ao risco;
- Falta de supervisão no monitoramento de riscos;
- Má compreensão quando se envolve atividades de risco.

Apesar de ser um fator importante para o sucesso do projeto, diversos autores reconhecem que a gestão de riscos muitas vezes é negligenciada em projetos [9]. Dessa forma fica evidenciada a necessidade do aprofundamento dos estudos nessa disciplina dentro das metodologias ágeis.

1.3 Motivação

A fim de diminuir a taxa de insucesso em projetos de desenvolvimento de software, faz-se necessário o aprofundamento dos estudos relacionados às técnicas de Gestão de Riscos em métodos ágeis de forma que possibilitem identificar e antecipar o impacto dos problemas, com objetivo de minimizar as influências negativas dos fatores e critérios que levam um projeto ao insucesso.

Dessa forma a motivação dessa pesquisa está em contribuir para o preenchimento dessa lacuna por meio da avaliação, combinação e seleção de técnicas e ferramentas para desenvolver um método de identificação, diagnóstico, análise e monitoramento de fatores e critérios de riscos que contemple orientações práticas e detalhadas para sua aplicação em projetos regidos por metodologias ágeis.

Com o atingimento dos objetivos abaixo elencados, espera-se que a pesquisa contribua para a área de gestão de projetos com a elaboração de uma metodologia para gestão de riscos em projetos que utilizam os métodos ágeis como forma de gestão/execução, visando integrar a gestão de riscos aos processos ágeis, aumentando sua probabilidade de sucesso, onde os riscos fiquem evidentes para todos os envolvidos e seja criado plano de resposta aos riscos, assim como o monitoramento possa ser realizado e acompanhado. Ainda acredita-se ter grande contribuição para o meio acadêmico devido a carência de material no que tange a riscos em projetos geridos por métodos ágeis.

1.4 Objetivo

O objetivo deste trabalho é apresentar uma nova metodologia para o gerenciamento de riscos em projetos que utilizam métodos ágeis de gestão, apoiado por meio de práticas e medições a fim de garantir o sucesso nesses projetos.

1.4.1 Objetivos Específicos

- Coletar informações sobre os riscos em projetos;
- Pesquisar métodos, técnicas, processos, boas práticas, normas, modelos e ferramentas de gestão de riscos adotadas atualmente;
- Fazer uma identificação preliminar dos fatores críticos que levam ao insucesso dos projetos que utilizam métodos ágeis;
- Elaborar um método de identificação, análise e avaliação dos riscos inerentes aos fatores críticos previamente identificados;

- Desenvolver uma ferramenta para monitoramento e análise crítica dos riscos;

1.5 Estruturação do Trabalho

O presente trabalho está estruturado com intuito de construir o conhecimento de forma incremental sobre a necessidade da gestão de riscos em métodos ágeis. Para tanto este estudo está pautado nas *Design Science*. As pesquisas baseadas nesse campo são baseadas no paradigma explanatório, isto é, faz uma investigação destinada a descrever, explicar e prever a fim de entender um conjunto de problemas, no entanto o grande foco da *Design Sciences* é produzir conhecimento de criação, ou seja, conhecimento que pode ser utilizado na concepção de soluções para determinados problemas [11]. *Design Sciences* é responsável pela concepção e validação de sistemas ainda não existentes, seja pela criação, combinação ou alteração de produtos, processos, softwares ou métodos visando melhorar situações existentes [12].

A *Design Sciences Research* tem como objetivo estudar, pesquisar e investigar o artificial, e seu comportamento tanto no meio acadêmico quanto nas organizações [13]. *Design Sciences Research* compreende um rigoroso processo de produzir artefatos para solucionar problemas, cujo objetivo é criar coisas que servem aos propósitos dos seres humanos [14].

Para atingir o objetivo proposto na pesquisa será necessário compreender como os riscos são tratados hodiernamente em projetos de desenvolvimento de software e quais são os fatores que levam os projetos de TI a fracassar, assim, apenas após a compreensão de todos os preceitos, será possível produzir conhecimento para proposição de melhoria nos métodos utilizados atualmente, com o objetivo de aumento no desempenho dos projetos ágeis. Verifica-se assim, que este estudo está perfeitamente alinhado com o paradigma da *Design Sciences*.

Com base na literatura produzida sobre esta disciplina, surgiu em 2007 uma nova metodologia de pesquisa onde é aplicada a *Design Sciences* para produção de pesquisas em Sistemas de Informação, a esta nova metodologia foi dado o nome de *Design Science Research Methodology (DSRM)* [15].

1.6 *Design Sciences Research Methodology*

Esta nova metodologia busca atingir a 03 objetivos, o primeiro é prover um processo nominal para condução de pesquisas em *Design Sciences*, o segundo é basear-se na literatura anterior sobre *Design Sciences* em Sistemas de Informação e disciplinas de referência e, por fim, proporcionar aos pesquisadores um modelo mental ou *template* para uma estrutura de resultados de pesquisa [15].

A metodologia proposta constitui uma sequência de 06 etapas para a elaboração da pesquisa. Os autores sugerem que estas etapas sejam executadas de forma sequencial, no entanto o diferencial desta metodologia é a utilização de um modelo iterativo, de forma que as atividades de avaliação e comunicação devem servir como balizadores e podem ser utilizadas para se fazer uma revisão nos objetivos e no *Design* proposto.

A Figura 1.2 oferece um *roadmap* para a pesquisa em questão baseado na metodologia DSRM. Pela imagem é possível verificar como se pretende evoluir, bem como cada entrega ou marco esperado ao final de cada uma das etapas da *Design Sciences Research Methodology*. Foi acrescentada uma nota entre as duas primeiras etapas para ilustrar o momento em que deverá ser feita a análise dos fatores e critérios de sucesso levantados na etapa 2.

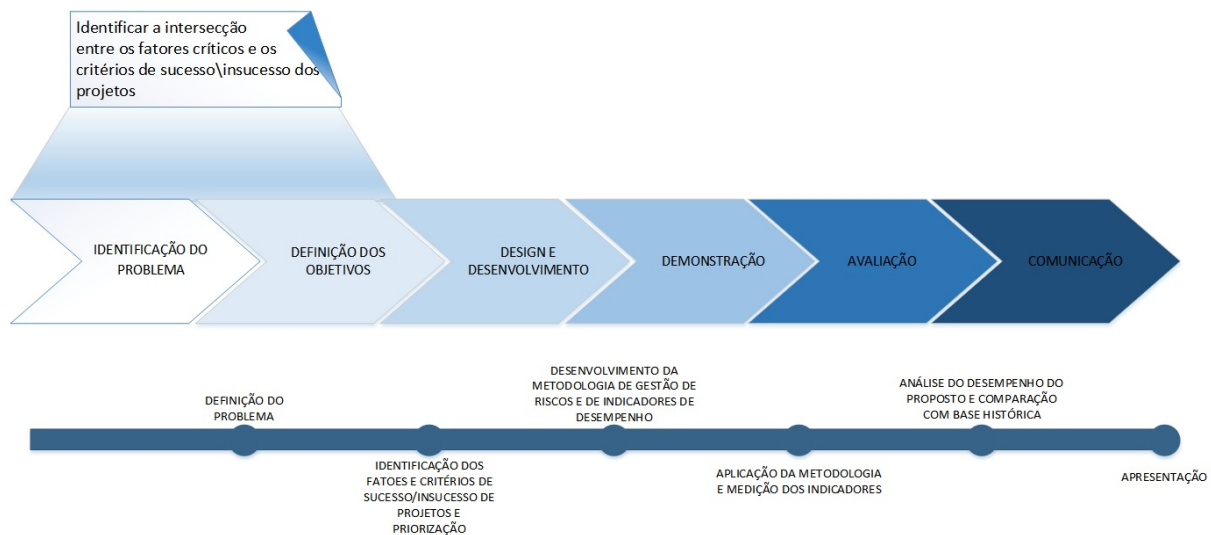


Figura 1.2: *Roadmap* da pesquisa, segundo a metodologia DSRM.

Nas seções seguintes serão descritos, segundo a metodologia DSRM, os fatores da pesquisa e seu detalhamento.

1.6.1 Identificação do problema

Nesta etapa é definido o problema específico da pesquisa e demonstrado o valor da solução. Nesse momento pode ser útil atomizar o problema conceitualmente para que a solução possa captar sua complexidade [15].

Apesar da taxa de sucesso ser consideravelmente maior quando utilizados os métodos ágeis, por meio da figura 1.1 é possível verificar que em 52% dos casos ocorreu algum desvio no projeto, seja por atraso na entrega do produto final, seja um entrega incompleta (com menos requisitos do que os requeridos) ou até mesmo por um aumento significativo

nos custos do projeto. Verifica-se também que em 9% dos casos o projeto foi abortado, totalizando 61% de taxa de insucesso em projetos de desenvolvimento que utilizaram métodos ágeis. Diante desse cenário a pergunta que deve ser respondida é como diminuir a taxa de insucesso em projetos que utilizam métodos ágeis?

1.6.2 Definição dos objetivos da solução

A etapa 2 da metodologia consiste em definir os objetivos da pesquisa. Os objetivos podem ser tanto qualitativos quanto quantitativos, devem ser inferidos a partir da definição do problema, bem como deve se ter em mente que os recursos necessários para sua definição incluem o conhecimento do estado dos problemas e soluções atuais, caso haja, e a eficácia dos mesmos [15].

Diante de todo o exposto torna-se clara a necessidade da elaboração de ferramentas que possam ser aplicadas as metodologias ágeis que venham a suprir a carência do método no que tange a Gestão de Riscos. Neste quadro, o objetivo deste trabalho é apresentar uma nova metodologia de gerenciamento de riscos a fim de diminuir a taxa de insucesso em projetos ágeis.

1.6.3 *Design* e Desenvolvimento

A terceira etapa da metodologia contempla o desenho e desenvolvimento de um artefato, cujo objetivo é solucionar o problema da pesquisa. Conceitualmente este artefato pode ser qualquer objeto que ofereça alguma contribuição para a pesquisa [15]. Inovações podem ser baseadas em novas propriedades de recursos técnicos, sociais ou informacionais, ou até mesmo suas combinações [16]. Esta atividade inclui determinar a funcionalidade desejada para o artefato, bem como sua arquitetura e posteriormente seu desenvolvimento [15].

Diante do problema identificado, para o desenho da solução proposta será necessário um entendimento de quais são os influenciadores para o atingimento do insucesso do projeto. Para tanto, será necessário primeiramente um levantamento dos fatores e critérios de sucesso de projetos em TI, visando assim orientar a elaboração de quais métricas devem ser verificadas objetivando manter o controle sobre estas variáveis para que ao fim do projeto a taxa de sucesso seja aumentada.

A percepção de fracasso no projeto é algo muito específico para cada cliente e comumente é dependente de múltiplas variáveis, o que dá a este tipo de levantamento um viés das ciências sociais e comportamentais. No campo das ciências sociais, apesar de não ser comum, as relações entre múltiplas variáveis e múltiplas causas devem ser pesquisadas [17]. Nos últimos anos a modelagem em equações estruturais tem se tornado a mais importante técnica de análise de dados nas ciências sociais [18]. Ainda segundo o autor, mais

do que uma técnica de análise de dados, a modelagem em equações estruturais, se tornou a linguagem para se formular teorias sobre ciências sociais, bem como para falar sobre o relacionamento entre diversas variáveis. Dessa forma este método foi o escolhido para a realização da pesquisa. Serão identificados constructos de fatores e critérios de sucesso em projetos, sob a perspectiva de diversos profissionais que atuam na área de gestão de projetos e também clientes, a fim de verificar quais são os atributos que mais influenciam na obtenção do sucesso do projeto.

Com base nos resultados da pesquisa aplicada, será possível identificar quais são os maiores influenciadores do sucesso de um projeto e então desenvolver ferramentas para mensuração dos critérios que mais influenciam no sucesso de um projeto.

No entanto, apenas o desenvolvimento dos indicadores não é suficiente para garantir o sucesso do projeto. Para que seja possível potencializar a utilização dos artefatos desenvolvidos é necessário que seja desenvolvido um novo método de avaliação e gestão de riscos aplicado às metodologias ágeis. Dessa forma é sugerida uma nova abordagem para gerir riscos em projetos que utilizam métodos ágeis, amparada sobre 04 pilares:

1. Gerar mais valor para o cliente;
2. Estar sempre alinhado aos objetivos estratégicos do negócio;
3. Reduzir desperdícios; e
4. Melhorar as oportunidades para o projeto.

O que se propõe é o planejamento da gestão de riscos em ondas sucessivas, de tal forma que, no início do projeto deverão ser identificados todos os riscos dado o escopo definido, criando assim um *backlog* de riscos para o projeto. Este *backlog* de riscos inicial deve ser monitorado com as ferramentas de controle comuns de projetos que seguem a metodologia tradicional, verificando e monitorando eventos e seus possíveis impactos no cronograma, escopo, custo e qualidade.

A segunda onda de planejamento do gerenciamento de riscos enxerga os possíveis riscos das atividades referentes a uma iteração, criando então um *backlog* de riscos da iteração. Assim, deve ser possível monitorar eventos e fatores que causem quaisquer impedimentos a equipe de desenvolvimento. O monitoramento se dará por meio dos indicadores desenvolvidos para este projeto.

1.6.4 Demonstração

Como etapa 4, deve ser aplicado o que foi desenvolvido como solução para o problema. Cita que esta demonstração pode envolver a experimentação, prova, estudo de caso ou

qualquer outra atividade apropriada para demonstrar a eficácia do que foi proposto [15]. Como demonstração do artefato desenvolvido como solução, a metodologia será aplicada em um projeto da Universidade de Brasília, assim será possível verificar se o trabalho desenvolvido atende os objetivos para os quais foi proposto.

1.6.5 Avaliação

A quinta etapa descrita na metodologia é a avaliação do desempenho do artefato desenvolvido. Nesta etapa deve ser avaliado quão bem o artefato funciona como solução para o problema. Esta atividade deve envolver comparações com o objetivo esperado e o desempenho observado. Segundo o autor, esta avaliação pode incluir qualquer tipo de evidência empírica ou prova lógica [15].

A metodologia preconiza que esta etapa deve prover insumos suficientes ao pesquisador para que ele decida se há necessidade de voltar a atividade de Design e Desenvolvimento para melhorias no artefato ou, caso sejam resultados satisfatórios, seguir com a comunicação as partes interessadas e implantação em outros projetos [15].

Diante da demonstração da metodologia no passo anterior será possível obter uma avaliação do proposto, com base no *feedback* de clientes e demais partes interessadas, e verificar se o que foi desenvolvido e aplicado atinge os objetivos para os quais foi elaborado. Esta atividade cíclica deverá acontecer enquanto os artefatos não alcançarem a meta esperada.

1.6.6 Comunicação

A sexta e última atividade desta metodologia refere-se à comunicação. Esta atividade consiste em comunicar o problema da pesquisa e sua importância, qual artefato desenvolvido, bem como sua utilidade e inovação, o modo como foi desenvolvido e sua efetividade a outros pesquisadores e outros públicos relevantes [15].

O resultado desta pesquisa será demonstrado através de uma dissertação, a ser apresentada a uma banca avaliadora, bem como através de artigos e/ou publicações para o meio profissional ao qual é destinada.

Capítulo 2

Referencial Teórico

Este capítulo apresenta uma visão geral acerca do tema da pesquisa, expõe os conceitos fundamentais, técnicas e ferramentas utilizadas no desenvolvimento desta pesquisa.

A revisão de literatura está estruturada em cinco seções, sendo que a primeira apresenta os conceitos relacionados à gestão de projetos; a segunda, gestão de riscos; a terceira, equações estruturais e por fim, a última apresenta conceitos relativos a indicadores de desempenho.

2.1 Gestão de projetos

Os projetos podem ser vistos como esforços temporários necessários para se produzir um produto, serviço ou resultado único. Da mesma forma, a Gestão de Projetos é vista como a aplicação de habilidades, conhecimentos, ferramentas e técnicas as atividades do projeto visando alcançar os requisitos para o qual foi proposto [4].

Os projetos existiram durante toda a história da humanidade e exemplos podem ser vistos desde os tempos antigos através de grandes obras arquitetônicas em diversas localidades, como a construção das pirâmides no Egito, os zigurats na Mesopotâmia, o coliseu de Roma, entre outros [19]. No entanto a gestão de projetos tem origem no início do século XX. Inspirados pelas teorias de Frederick W. Taylor, apresentadas em sua obra *Scientific Management*, Henri Fayol e Henry Gantt desenvolveram trabalhos significativos neste campo e atualmente são considerados os pais da gestão de projetos. Henri Fayol propôs, em 1916, cinco funções primárias de um gerente: planejar, organizar, comandar, coordenar e monitorar [19][20]. Henry Gantt estende o trabalho de Fayol por meio da criação de uma ferramenta de apoio a gestão de projetos chamada Gráfico de Gantt, por intermédio da qual processos de trabalho podem ser traçados e controlados [19].

No decorrer dos anos os processos evoluíram, novas ferramentas de apoio foram desenvolvidas, surgiram mais métodos de controle, novas tendências e, conseqüentemente

novos métodos foram propostos. Atualmente pode ser feita claramente a distinção entre dois métodos de gestão de projetos:

- Tradicional ou Cascata, possui um ciclo de vida linear, ou seja, com fases sequenciais e bem definidas [21];
- Ágil, que ocorre de forma iterativa, adaptativa e incremental [21].

Esses dois métodos serão melhor descritos nas Seções que se seguem.

2.1.1 Metodologia tradicional ou cascata

A metodologia tradicional de gerenciamento de projetos data da metade do século XX. Princípios estabelecidos nos anos 1950 descrevem que métodos e procedimentos devem ser aplicados a todos os projetos de uma forma uniforme [22]. Esta implementação uniforme deve garantir robusteza e aplicabilidade a uma enorme gama de projetos, desde os simples e pequenos projetos, até os grandes e mais complexos [22]. Os tradicionalistas defendem um planejamento extensivo, processos codificados e reutilização rigorosa para tornar o desenvolvimento uma atividade eficiente e previsível que amadurece gradualmente rumo a perfeição [23].

Projetos gerenciados de acordo com a abordagem tradicional estão focados na entrega do produto dentro do orçamento e do prazo estimado, preocupam-se mais em estar em conformidade com o planejado do que com entrega de valor ao cliente [21].

Abordagens tradicionais de gerenciamento de projetos são baseadas em processos lineares ou incrementais [21]. Uma abordagem linear busca entregar o produto completo ao final do projeto, como exemplo demonstrado na Figura 2.1.



Figura 2.1: Metodologia tradicional - abordagem linear (Fonte: [21]).

Já a abordagem incremental trabalha de forma cíclica, desenvolvendo o produto do projeto em incrementos até que seja completado ao final do projeto conforme demonstrado na Figura 2.2.

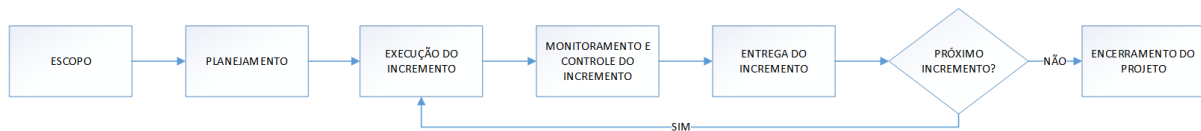


Figura 2.2: Metodologia tradicional - abordagem incremental (Fonte: [21]).

Dois dos métodos de gestão tradicional de projetos mais conhecidos mundialmente são o Guia PMBOK elaborado pelo *Project Management Institute* e o PRINCE2 desenvolvido pelo governo britânico. Ambos métodos são amplamente utilizados a nível mundial.

Project Management Body of Knowledge (PMBOK)

O PMBOK, ou Corpo de Conhecimento em Gerenciamento de Projetos, em português, é um *framework* e fornece uma base sobre a qual as organizações podem criar metodologias, políticas, procedimentos, regras, ferramentas e técnicas e fases do ciclo de vida necessários para a prática do gerenciamento de projetos [4].

O PMBOK define que todos os projetos possuem um ciclo de vida dividido em início, organização e preparação, execução e término. Esse ciclo de vida é suportado por 05 grupos de processos: grupo de processos de Iniciação, grupo de processos de Planejamento, grupo de processos de Execução, grupo de processos de Monitoramento e Controle e o grupo de processos de Encerramento. Por fim, esses grupos de processos estão subdivididos em 10 áreas de conhecimento, conforme verificado na Figura 2.3.

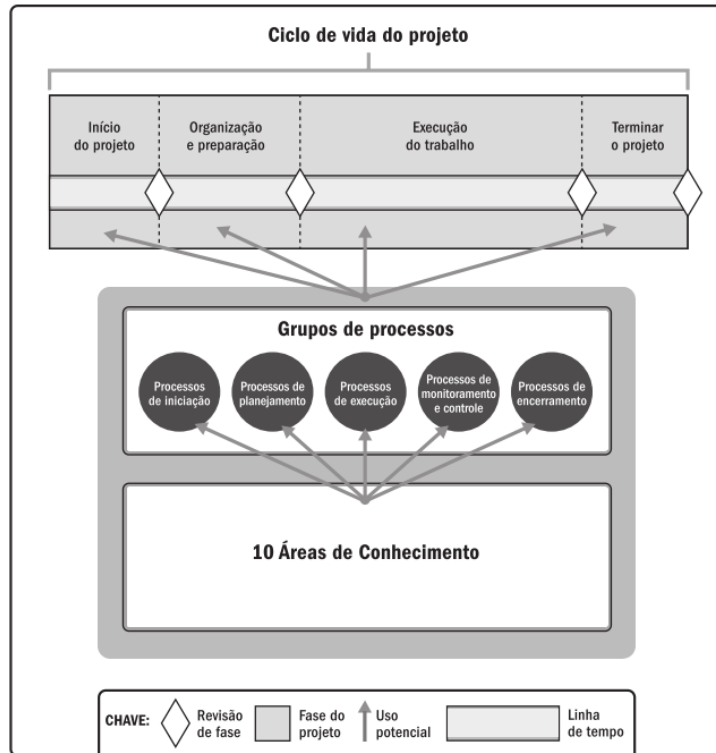


Figura 2.3: Inter-relação de componentes do Guia PMBOK (Fonte: [4]).

Os Grupos de Processos de Gerenciamento de Projetos são agrupamentos de processos de gerenciamento de projetos visando atingir os objetivos específicos do projeto [4].

- Grupo de processos de iniciação contempla os processos realizados para definir um novo projeto ou uma nova fase de um projeto existente, seu principal objetivo é a obtenção de autorização para iniciar o projeto ou fase;
- Grupo de processos de planejamento abrange os processos necessários para definir o escopo, refinar os objetivos e definir a linha de ação necessária do projeto para que se alcancem os objetivos para os quais o projeto foi criado;
- Grupo de processos de execução engloba aqueles processos realizados para concluir o trabalho definido no plano de gerenciamento do projeto e tem como meta atingir os requisitos do projeto;
- Grupo de processos de monitoramento e controle abarca os processos exigidos para acompanhar, monitorar e controlar o progresso e o desempenho do projeto, visando identificar quaisquer necessidades de mudanças no plano;

- Grupo de processos de encerramento compreende os processos realizados ao final de uma fase ou de um projeto e visa concluir ou fechar formalmente um projeto, fase ou contrato.

Os grupos de processos interagem ao longo do ciclo de vida do projeto conforme Figura 2.4

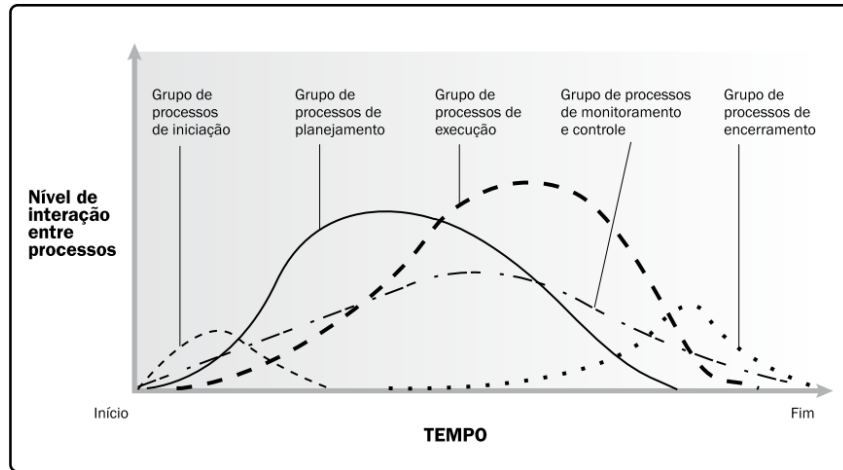


Figura 2.4: Interação entre grupos de processos do Guia PMBOK (Fonte: [4]).

Cada área de conhecimento possui processos em determinada fase do projeto e são distribuídos conforme apresentado na Figura 2.5.

	INICIAÇÃO	PLANEJAMENTO	EXECUÇÃO	MONITORAMENTO E CONTROLE	ENCERRAMENTO
Gerenciamento da integração do projeto	Desenvolver o Termo de Abertura do Projeto	Desenvolver o Plano de Gerenciamento do Projeto	Gerenciar o trabalho e Gerenciar conhecimento do projeto	Monitorar o trabalho e realizar o controle de mudanças	Encerrar o projeto ou fase
Gerenciamento do escopo do projeto		Coletar Requisitos, definir escopo, Criar EAP.		Validar e controlar o escopo	
Gerenciamento do cronograma do projeto		Definir e sequenciar atividades, estimar durações e desenvolver cronograma		Controlar o cronograma	
Gerenciamento dos custos do projeto		Estimar os custos e determinar o orçamento		Controlar os custos	
Gerenciamento da qualidade do projeto		Planejar o gerenciamento da qualidade do projeto	Gerenciar a qualidade	Controlar a qualidade	
Gerenciamento dos recursos do projeto		Planejar o gerenciamento dos recursos e estimar os recursos das atividades	Adquirir Recursos Desenvolver a Equipe Gerenciar a Equipe	Controlar os recursos	
Gerenciamento das comunicações do projeto		Planejar as comunicações do projeto		Monitorar as comunicações	
Gerenciamento dos riscos do projeto.		Identificar os riscos, realizar análise quantitativa e qualitativa e planejar respostas aos riscos		Monitorar os riscos	
Gerenciamento das aquisições do projeto		Planejar as aquisições do projeto		Controlar as aquisições	
Gerenciamento das partes interessadas do projeto	Identificar as partes interessadas	Planejar o engajamento das partes interessadas		Monitorar o engajamento das partes interessadas	

Figura 2.5: Áreas de conhecimento e grupos de processo (Fonte: [4]).

O guia PMBOK propõe um conjunto de práticas e ferramentas podem ser utilizadas para compor um processo de gestão altamente estruturado, com fases sequenciais e muito

bem definidas amparado por um planejamento completo e altamente documentado. Apesar de fornecer as ferramentas e melhores práticas o guia não constitui uma metodologia de gestão, diferentemente do modelo britânico, chamado de *Projects In Controlled Environments* (PRINCE2), o qual fornece um conjunto de regras e passos necessários para realização dos objetivos do projeto.

Projects In Controlled Environments (PRINCE2)

O PRINCE2 é um método genérico, estruturado e não proprietário de gerenciamento de projetos. Trata-se de uma metodologia que integra processos e trata de temas como planejamento, delegação, monitoramento e controle de seis aspectos de desempenho do projeto: custos, prazo, qualidade, escopo, riscos e benefícios [24].

O PRINCE2 é um método baseado em princípios, este possuem sua origem em lições aprendidas, tanto positivas como negativas, em projetos. Esses princípios proporcionam um *framework* de boas práticas e são caracterizados como [24]:

1. Universais, uma vez que é possível aplicá-los a todo o tipo de projeto;
2. Autovalidáveis, pois foram comprovados na prática no decorrer dos anos;e
3. Capazes de dar autonomia, pois oferecem aos envolvidos mais confiança e habilidade de influenciar na maneira de gestão do projeto.

Os sete princípios base do PRINCE2 são:

- Justificação do negócio contínua: significa dizer que a justificativa pela qual o projeto foi iniciado se mantém válida até o final do projeto;
- Aprender com a experiência: significa buscar nas lições aprendidas de outros projetos insumos em que se pode basear as ações do projeto;
- Papéis e responsabilidades definidos: significa identificar todos os envolvidos no projeto e atribuir-lhes os corretos papéis dentro da estrutura do projeto;
- Gerenciar por estágios: a gestão por estágio proporciona maior controle sobre o projeto;
- Gerenciar por exceção: significa estabelecer tolerâncias para cada objetivo do projeto, visando estabelecer os limites da autoridade delegada;
- Foco em produtos: o foco do PRINCE2 está na definição e entrega de produtos, mais especificamente naqueles concernentes a qualidade; e

- Adequar ao ambiente do projeto: significa ser um método genérico suficiente para que se adeque ao ambiente, porte, complexidade, importância, capacidade e riscos dos projetos.

O PRINCE2 propõe a utilização de 07 processos, distribuídos ao longo do ciclo de vida do projeto, os quais fornecem o conjunto de atividades necessárias para direcionar, gerenciar e entregar um projeto com êxito. A Figura 2.6 elenca os processos descritos e as etapas do ciclo de vida do projeto no qual são utilizados.

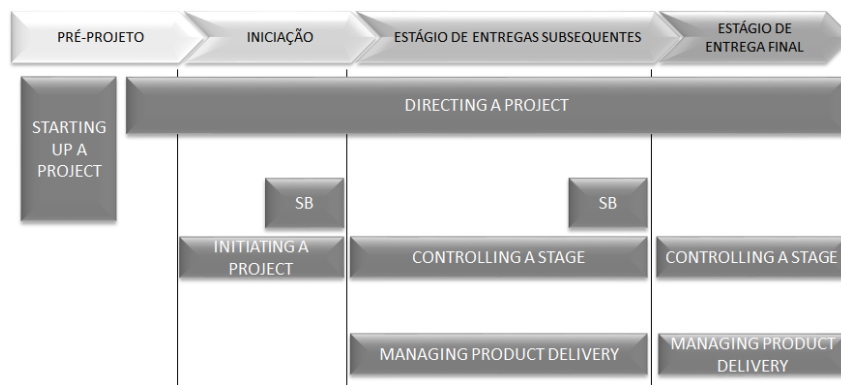


Figura 2.6: Processos do método PRINCE2 (Fonte: [24]).

O processo *Starting up a project* tem como objetivo garantir que todos os requisitos para iniciar o projeto estejam devidamente implementados. O processo *Directing a project* visa possibilitar ao comitê diretivo tomar as decisões necessárias e exercer o controle geral objetivando assegurar o sucesso do projeto, é iniciado após a conclusão do processo *Starting up a project*. O processo *Initiating a project* tem como propósito estabelecer os fundamentos para o projeto visando assegurar o correto entendimento de todo o trabalho que precisa ser feito para entregar os produtos do projeto. O processo *Controlling a stage* tem como objetivo o monitoramento do trabalho resolução de problemas referentes aquele estágio. O processo *Managing product delivery* estabelece os requisitos formais de execução, aceite e entrega do produto. O processo *Managing a stage boundary*, representado pelas letras 'SB' na Figura 2.6 tem com propósito o fornecimento de informações periódicas ao Comitê Diretivo acerca do progresso atual do projeto, visando assegurar a validade da justificativa do projeto e o aceite dos riscos do mesmo. Por fim o processo *Closing a project* tem como objetivo confirmar a aceitação do produto do projeto e reconhecer o atingimento dos objetivos para is quais o projeto foi empreendido [24].

Ambos os *frameworks* recomendam a adoção de uma sequência encadeada de atividades para a gestão do projeto, com passos pré determinados e documentação extensa e abrangente. Esses métodos, para que sejam eficazes, fazem necessário que os requisitos e

objetivos do projeto estejam claros e bem definidos. Nesse sentido, e levando também em consideração a robustez deste modelo de gestão, este tipo de abordagem se torna avesso a maioria das mudanças [21]. Esse tipo de abordagem, orientada sob aspectos de controles de processo, se torna subótima devido ao dinamismo dos ambientes de projeto atuais [25]

Todas as objeções à abordagem tradicional de gerenciamento de projetos, juntamente com os pedidos crescentes de inovações contínuas que afetaram todas as indústrias e com as tendências de redução de custos, resultaram no surgimento de novas vertentes de gerenciamento de projetos, estas abordagens são chamadas de métodos ágeis [22].

2.1.2 Metodologia ágil

Devido a rápida evolução tecnológica e também do mercado, os modelos tradicionais de gestão começaram a perder espaço na indústria de TI. A partir da década de 1990 emerge então um novo modelo de gestão baseado no pensamento *lean* [26]. O pensamento *lean* é a base para todas as metodologias ágeis conhecidas atualmente, como pode ser visto pela Figura 2.7.

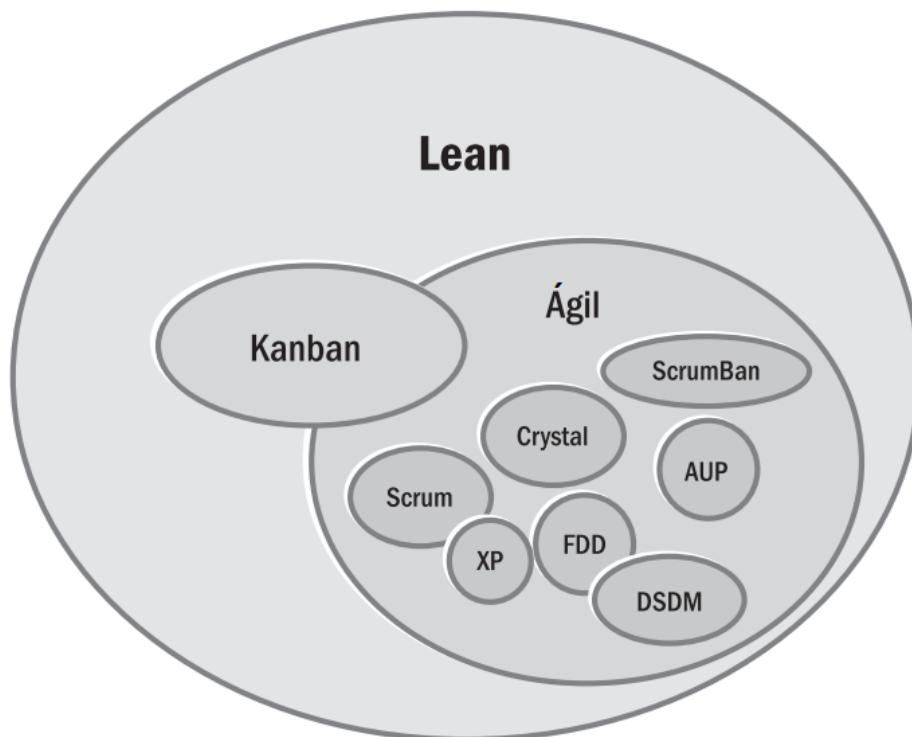


Figura 2.7: *Lean* e metodologias e *frameworks* ágeis (Fonte: [27]).

O termo *lean* foi utilizado pela primeira vez na década de 1980, em um artigo intitulado "*Triumph of the Lean Production System*" e que tinha como objetivo demonstrar

que, ao contrário do que se pensava na época, não era a localidade que influenciava na produtividade e qualidade do que era produzido e sim no modelo adotado nas plantas fabris [28]. O estudo demonstra que não é apenas o emprego de tecnologia de ponta que faz com que a produtividade aumente, mas sim sua combinação com outras filosofias de produção, demonstrando que o pensamento e técnicas aplicadas pelo modelo Toyota de Produção eram mais eficientes do que as utilizadas pela indústria automotiva tradicional [29].

Esse modelo de produção tem como base a total eliminação do "lixo", do desperdício, para isso se baseia em dois pilares [30]:

- *Just-in-time*: esse é um modelo conhecido como estoque zero, onde os itens necessários para determinada execução chegavam na linha de produção apenas na quantidade e no momento necessários;
- Automação ou automação inteligente: no modelo Toyota as máquinas são equipadas com dispositivos de parada para que se evite a produção de grandes lotes de produtos defeituosos, a isso dão o nome de automação inteligente ou automação com toque humano.

Além dos pilares acima descritos, o modelo ainda destaca a importância do trabalho em equipe e busca constante por eficiência [30]. A este novo padrão de produção industrial proposto, juntamente com suas bases filosóficas, foi dado o nome de *lean production system* [29]. Esse é o pensamento que fornece os fundamentos sobre os quais estão estabelecidas as metodologias ágeis de desenvolvimento de software [31].

Enquanto a metodologia tradicional tem o foco no plano, as metodologias ágeis têm o foco nas mudanças [21] e na satisfação do cliente por meio da entrega de valor [32]. A inevitabilidade de desvios nos planos do projeto, sugere que a solução não está em planos iniciais mais sofisticados, mas sim em metodologias que podem facilitar ações para resolver desvios. No ambiente de projetos de TI, essa necessidade de melhorar o processo de planejamento levou cada vez mais as empresas passarem a utilizar um processo que gira em torno de várias iterações através do ciclo de desenvolvimento em oposição a um processo tradicional de planejamento [33]. A Figura 2.8 demonstra o modelo iterativo.

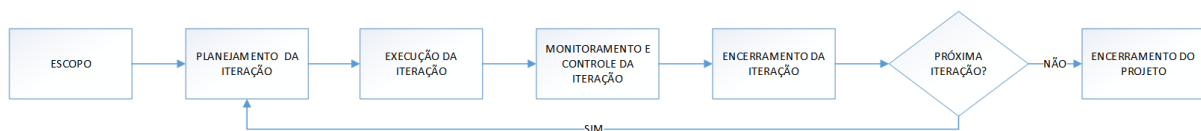


Figura 2.8: Metodologia ágil - modelo iterativo (Fonte: [21]).

Em contraste à metodologia tradicional os métodos ágeis são desenhados para utilizar a mínima documentação possível, facilitando assim a flexibilidade do projeto. Tais pensamentos ficaram conhecidos mundialmente por meio do Manifesto Ágil de desenvolvimento de software. Esse manifesto afirma que o desenvolvimento ágil deve ter o foco em quatro valores fundamentais [34]:

- Indivíduos e interações mais que processos e ferramentas;
- Software em funcionamento mais que documentação abrangente;
- Colaboração com o cliente mais que negociação de contratos; e
- Responder a mudanças mais que seguir um plano.

Apesar de defender que projetos tenham pouca documentação, isso não significa dizer que projetos ágeis não tenham planejamento, tendo em vista que a cada início de iteração deve ocorrer o planejamento da mesma. Um equilíbrio entre métodos tradicionais e métodos ágeis geralmente é apropriado para o planejamento do projeto [33]. Para tal a gestão de riscos poderá atuar como um suporte para análise do quanto de planejamento é necessário, atingindo-se, portanto, um ponto ideal [23]. Certos fatores, como o tamanho do projeto, os requisitos de segurança e os requisitos futuros conhecidos, exigem um planejamento antecipado mesmo em projetos ágeis, enquanto ambientes turbulentos e de alta mudança exigem um planejamento menos antecipado e um maior uso de métodos ágeis [33].

O Manifesto Ágil possui ainda 12 princípios [34]:

1. Nossa maior prioridade é satisfazer o cliente através da entrega contínua e adiantada de software com valor agregado.
2. Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente.
3. Entregar frequentemente software funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.
4. Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
5. Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e o suporte necessário e confie neles para fazer o trabalho.
6. O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face.

7. Software funcionando é a medida primária de progresso.
8. Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
9. Contínua atenção à excelência técnica e bom *design* aumenta a agilidade.
10. Simplicidade: a arte de maximizar a quantidade de trabalho não realizado é essencial.
11. As melhores arquiteturas, requisitos e *designs* emergem de equipes auto organizáveis.
12. Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo.

As mudanças trazem o desperdício de tempo e recursos em projetos tradicionais [21]. No entanto, como se nota pelo princípio número 2, agilistas esperam que mudanças ocorram. Trabalhando com o modelo de planejamento *Just-in-time* acreditam que minimizam os impactos das alterações durante o projeto [21].

Scrum

O *Scrum* foi criado oficialmente em 1993 como uma alternativa mais rápida, eficaz e confiável de criação de softwares [35]. Esse método tem como base a abordagem do *Rugby* para o desenvolvimento de produtos, proposta por Takeuchi e Nonaka que propunha que o novo método para o desenvolvimento de produtos deveria ter seis características básicas: instabilidade integrada, equipes de projeto auto organizadas, fases de desenvolvimento sobrepostas, aprendizagem múltipla, controle sutil e transferência de aprendizado [36].

O *Scrum* utiliza um processo iterativo e incremental que pode ser compreendido por meio da Figura 2.9 [37].

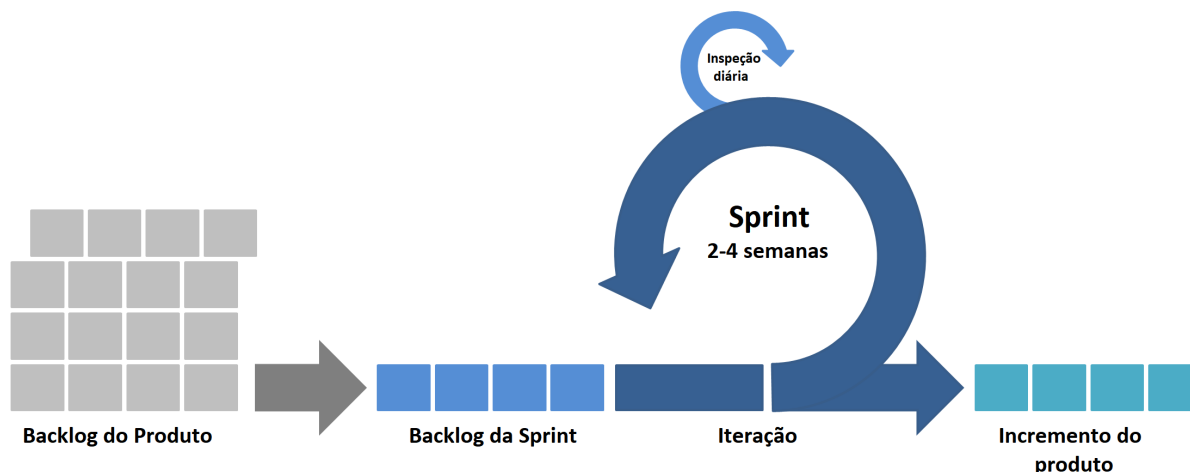


Figura 2.9: *Processo Scrum* (Fonte: [37]).

Na Figura 2.9 o círculo inferior representa uma iteração (*Sprint*), onde ocorrem as atividades de desenvolvimento. A saída de cada iteração é um incremento de produto. Por sua vez, o círculo superior representa a inspeção diária que ocorre durante a iteração, por meio de uma reunião diária, na qual os membros individuais da equipe se reúnem para analisar as atividades dos outros e fazer as adaptações que forem necessárias. Este ciclo deve se repetir até que o projeto seja finalizado [37].

O Scrum funciona da seguinte maneira, no início da iteração o time deve revisar o que deve ser feito durante o projeto (*Backlog* do produto) e então selecionar o que acreditam ser capaz de desenvolver e entregar em uma *Sprint*. Após a seleção o time executor é deixado sozinho para que possa elaborar um plano de ação para executar o que foi selecionado. Ao final de iteração o time apresenta um incremento funcional do produto para as partes interessadas, para que esse possa ser avaliado e inspecionado e, caso seja necessário, sejam sugeridos ajustes e adaptações no projeto [37].

Para que seja possível a implementação do *Scrum* são necessários três papéis [37]:

1. *Product Owner*, ou simplesmente PO: é o responsável por representar os interesses das partes interessadas no projeto ou no resultado do mesmo. Deve garantir que o *Backlog* do produto esteja sempre priorizado de forma que os itens que gerem mais valor ao cliente sejam desenvolvidos na iteração seguinte;
2. *Scrum Master*: é o ator responsável por garantir o processo *Scrum*, ensinar o *Scrum* a todos os envolvidos e, é responsável pela implementação do *Scrum* de forma que se encaixe na cultura da organização e entregue os benefícios esperados. Também por garantir que todos os envolvidos sigam as regras e práticas do *Scrum*;

3. Time de desenvolvimento: é responsável pela execução do projeto de fato, uma vez que é o time que desenvolve as funcionalidades indispensáveis para o atendimento das necessidades das partes interessadas. O time de desenvolvimento deve ser auto organizado, ou seja, os próprios integrantes que determinam a melhor maneira de atuar para atender a meta da *sprint*; deve ser um time multifuncional, o que quer dizer que a equipe possui todas as habilidades necessárias para o desenvolvimento do que se propõe; e deve também ser auto gerenciável, de maneira que, o time por si só deve atingir uma dinâmica de relações saudáveis entre seus membros .

Os papéis do *Scrum* interagem durante a execução do projeto por meio de um fluxo pré-definido, e gera uma série de artefatos, como demonstrado na Figura 2.10.

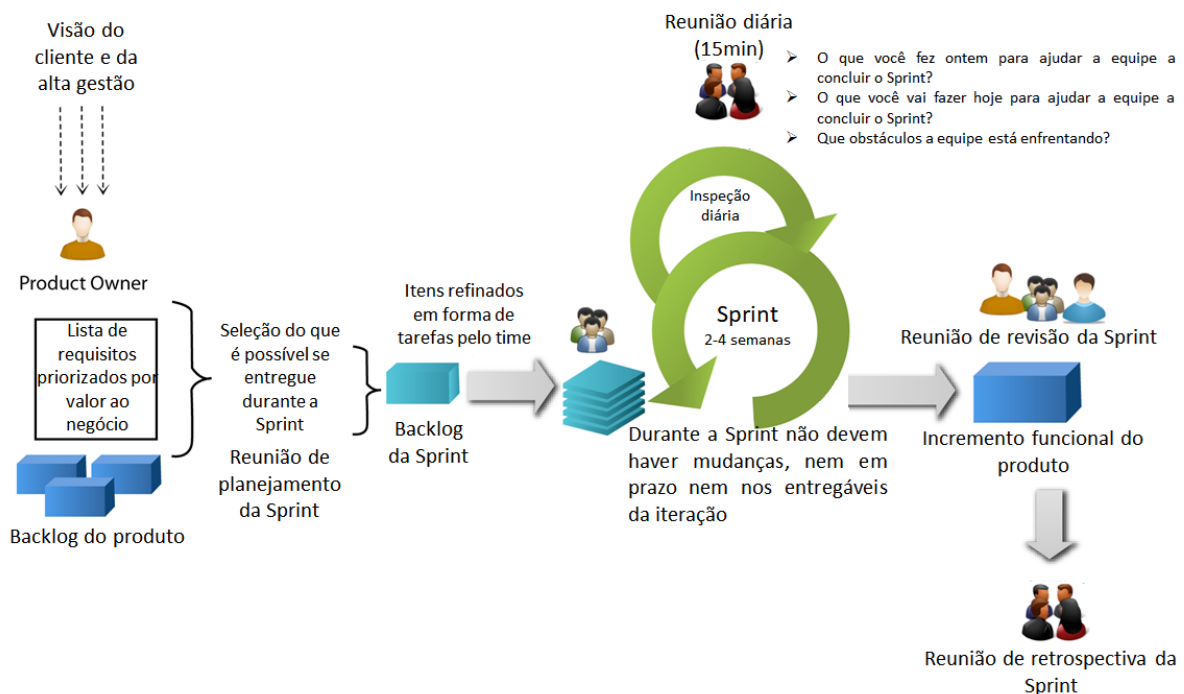


Figura 2.10: Fluxo *Scrum*.

O fluxo começa com a visão do produto por parte da organização e clientes, que é transformada pelo PO em uma lista de requisitos, funcionais e não funcionais, chamada de *backlog* do produto. De forma que seja maximizado o retorno de valor à organização, o PO deve sempre rever essa lista para que a mesma seja mantida ordenada da mais alta a mais baixa prioridade. De posse do *backlog* priorizado o time de desenvolvimento, juntamente com o PO, durante a reunião de planejamento da *sprint*, seleciona os itens que acredita ser possível desenvolver em uma iteração, o que é chamado de *backlog* da

sprint. Após a seleção dos itens que entrarão na esteira de desenvolvimento o time se isola para traçar um plano de ação e refinar cada item de *backlog* da *sprint* [37].

Diariamente o time se reúne para uma inspeção do andamento do trabalho, e devem ser respondidas 03 perguntas [35]:

1. O que você fez ontem para ajudar a equipe a concluir o Sprint?
2. O que você vai fazer hoje para ajudar a equipe a concluir o Sprint?
3. Que obstáculos a equipe está enfrentando?

O objetivo dessa reunião é alinhar o que o time está fazendo e identificar prontamente quaisquer impedimentos que possam afetar a entrega da meta da *sprint*.

Ao final da iteração ocorre a reunião de revisão, onde o time apresenta ao *Product Owner*, e a quaisquer outras partes interessadas, aquilo que foi desenvolvido. Essa reunião de revisão tem por objetivo obter o *feedback* dos *stakeholders*. Por fim, após a reunião de revisão, é feita uma reunião chamada de retrospectiva da *sprint*. Essa por sua vez, visa identificar o que pode ser melhorado para a próxima iteração, sendo portanto uma coleta de lições aprendidas e que faz parte de um ciclo de melhoria contínua do time [35][37].

As cerimônias do fluxo *Scrum* são todas as reuniões que ocorrem durante a iteração e a própria iteração em si, e cada uma delas deve obedecer algumas regras:

- Reunião diária (*Daily meeting*): com duração de 15 (quinze) minutos, todos os integrantes do time de desenvolvimento devem participar e visa promover o alinhamento e, identificar obstáculos e impedimentos ao atingimento das metas da *sprint* [37];
- *Sprint*: a *sprint* é o espaço de tempo que o time utiliza para efetivamente realizar o trabalho. Tem uma duração de no mínimo 02 semanas e no máximo 04 semanas, durante esse período nada deve ser alterado do escopo inicialmente planejado para a *sprint*, nem modificado o tempo da mesma. Toda iteração deve ter uma meta a ser cumprida pelo time de desenvolvimento e, caso seja identificado que a meta não será cumprida, o *Scrum Master*, juntamente com o time de desenvolvimento, pode antecipar o término da *sprint*. Caso o time identifique que pode exceder o planejado, o PO deve ser consultado para selecionar os itens do *backlog* que entrarão na *sprint*. Durante a iteração o time tem duas atividades administrativas obrigatórias: fazer a reunião diária e manter atualizado o *backlog* da *sprint* [37];
- Reunião de planejamento da *Sprint* (*Sprint Planning*): é a reunião de planejamento do ciclo de desenvolvimento, que deve ocorrer no primeiro dia da *sprint* e conta com a participação do PO, *Scrum Master* e do time de desenvolvimento. Tem duração máxima de 08 horas para iterações de 4 semanas, podendo variar de acordo com

a duração da iteração. É dividida em duas partes: na primeira parte o time, em conjunto com o PO define, dentro do que está priorizado no *backlog* do produto, o que irá ser desenvolvido naquele ciclo (meta da *sprint*); na segunda parte o time se isola para refinar o escopo, definir tarefas e elaborar um plano de ação para que seja possível atingir a meta estabelecida [38];

- Reunião de Revisão da *Sprint* (*Sprint Review*): com duração máxima de 04 horas para iterações de 04 semanas, essa reunião tem por finalidade demonstrar ao PO o que o time desenvolveu durante a iteração. Só deverá ser apresentado o que está realmente pronto e funcionando. Ao final da apresentação o PO e todos os *stakeholders* presentes devem passar ao time o *feedback* acerca do que foi apresentado [38];
- Reunião de Retrospectiva da *Sprint* (*sprint Retrospective*): com duração máxima de 03 horas e objetivo principal de discutir o que foi feito durante a *sprint*, o time inspeciona os processos de trabalho, dinâmicas, comportamentos, práticas, ferramentas utilizadas e ambiente, e planeja as melhorias necessárias para a próxima iteração. São basicamente respondidas duas questões chaves [38][37]:

O que foi bem durante a última *sprint*?

O que poderia ser melhorado na próxima *sprint*?

A reunião de retrospectiva é a última cerimônia da iteração, após a mesma o time pode dar início a um novo ciclo de desenvolvimento para dar prosseguimento ao projeto. O processo *Scrum* produz também artefatos, a saber:

- *Backlog* do Produto e da *Sprint* (*Product/Sprint Backlog*): o *backlog* do produto é uma lista de tudo aquilo que é necessário ser desenvolvido no projeto, ou seja, é o escopo do projeto de maneira simplificada e deve sempre estar classificado quanto ao valor do item para o negócio. É uma lista dinâmica que pode ter itens acrescentados ou retirados pelo PO. Ao passo que os itens dessa lista são selecionados para entrar no ciclo desenvolvimento da iteração, estes se tornam o *backlog* da *sprint*. Assim o *backlog* da *sprint* se apresenta como a lista de tudo que será desenvolvido na iteração e pode ser mais detalhada que o *backlog* do produto, chegando até ao nível de tarefas [37].
- Gráficos de acompanhamento: o *scrum* utiliza dois gráficos como formas de acompanhamento, a saber: o gráfico de *burnup*, que demonstra a quantidade de trabalho remanescente para o final do projeto; e, o gráfico de *burndown* que evidencia a quantidade de trabalho restante para a meta da *sprint* [38]. Ambos os gráficos podem ser vistos na Figura 2.11.

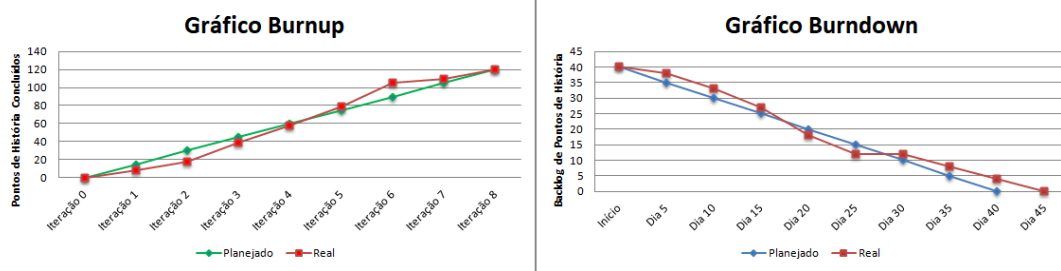


Figura 2.11: Gráficos de *Burnup* e *Burndown*.

- Quadro *Scrum*: foi inspirado na metodologia Kanban surgida em meados dos anos 50, tem como objetivo apoiar a equipe e dar visibilidade ao que está sendo desenvolvido. O quadro contém basicamente três colunas, na primeira são colocados os itens pendentes de desenvolvimento (*sprint backlog*) na *sprint*, na segunda coluna devem ser colocados os itens que estão em desenvolvimento e na última os itens que foram desenvolvidos e estão prontos [35].

Extreme Programming - XP

O XP é uma metodologia de desenvolvimento de software também criada em contraponto a metodologia tradicional. Assim como o *Scrum*, o XP se baseia em pequenas e constantes entregas de forma iterativa e incremental. Durante muitos anos diversos autores elencaram inúmeras práticas, tidas como melhores maneiras de se produzir softwares em ambientes de mudanças constantes, o XP se baseia nestas práticas para compor sua metodologia de desenvolvimento [39]. As práticas que compõem a metodologia de desenvolvimento de software *extreme programming* são [39][40]:

- *Stories*: o planejamento deve ser feito utilizando unidades de funcionalidades visíveis para o cliente, o planejamento não deve ser feito com requisitos e sim com histórias de usuário. Assim que as histórias forem escritas deve-se estimar o esforço de desenvolvimento necessário para sua implementação
- Pequenas entregas: o planejamento deve ser feito para que o sistema entre em produção o mais rápido possível, isso significa que deverão haver pequenas entregas visando o atingimento do objetivo de forma incremental;
- Metáforas: o sistema é definido por metáforas compartilhadas entre o cliente e os programadores;
- *Test-First Programming*: antes de modificar o código o programador deve efetuar um teste, se possível escrever um teste automatizado, assim após modificar o código poderá rodar o teste e verificar se o sistema continua correto;

- Refatoração: o sistema é desenvolvido através de evoluções do projeto existente, não há desperdício de código, ocorre a revisitação para a evolução do mesmo;
- Programação em pares (*Pair Programming*): no XP todo o desenvolvimento é feito em pares, dois colaboradores utilizando uma mesma estação de trabalho;
- Integração contínua: todo novo desenvolvimento deve ser integrado ao sistema que está em produção em um intervalo de poucas horas, todos os testes devem ser feitos e validados ou as alterações são descartadas;
- Propriedade coletiva: qualquer desenvolvedor pode melhorar qualquer código em qualquer lugar do sistema a qualquer momento, se existe oportunidade, nenhum problema é de apenas uma pessoa, o time deve ter o sentimento de coletividade;
- Presença do cliente: o cliente ou seu representante deve estar presente 100% do tempo com o time;
- Semanas de 40 horas: ninguém deve trabalhar frequentemente acima de 40 horas por semana. O colaborador deve trabalhar apenas quantas horas você pode ser produtivo e apenas quantas horas você pode sustentar. Horas extras isoladas usadas com demasiada regularidade são sinais de problemas e devem ser abordados;
- Espaço de trabalho aberto: o ambiente de trabalho deve ser grande o suficiente para comportar toda a equipe de projeto, é importante encorajar a conversação para facilitar o entendimento de todo time;
- Ambiente de trabalho informativo: qualquer observador deve ser capaz de transitar pelo ambiente de trabalho e identificar facilmente do que se trata o projeto;
- *Design* incremental: Todos os dias o *design* do sistema deve ser revisitado, trabalhando sempre de forma incremental, visando torná-lo ajustado para as necessidades que surjam no momento.

As práticas do XP não são obrigatórias, na verdade elas dependem da situação em que o projeto se encontra devem ser escolhidas aquelas que melhor atendam as condições do ambiente de projeto. Elas visam ser um direcionador para um estado ideal de desenvolvimento efetivo [40].

Kanban

O Kanban é um método de organização e gerenciamento de atividades criado como um sistema de controle para o modelo Toyota de produção, e foi utilizado como alternativa aos sistemas computadorizados devido a 03 fatores [41]:

1. Redução de custos para o processamento de informações: a implementação de um sistema informatizado que forneça cronograma de produção para todos os processos e fornecedores, bem como suas alterações e ajustes por controle em tempo real é extremamente dispendioso;
2. Rapidez e precisão na obtenção dos fatos: a utilização do Kanban proporciona aos gerentes uma percepção rápida de cenários contínuos, tal como capacidade de produção, dessa forma permite que promovam atividades para melhorias espontâneas; e
3. Limitar a capacidade de produção excedente: com a limitação da capacidade de produção excedente, evita-se o desperdício causado pelo excesso de produção.

O kanban, com "k"minúsculo é uma palavra japonesa que significa "cartão". A metodologia no entanto é escrita com "K"e diz respeito a um processo de melhoria evolutivo e incremental [42]. Na engenharia de produção há o sistema Kanban, porém, para o desenvolvimento de software, o Kanban é tido como um método e é utilizado para limitar o trabalho em andamento, WiP, do inglês *Work in Progress*, essa metodologia se tornou popular com o crescimento dos métodos ágeis de desenvolvimento de software, pois proporciona um controle visual do progresso do trabalho do time [42]. Um exemplo de quadro kanban, utilizado para desenvolvimento de software, pode ser verificado na Figura 2.12.

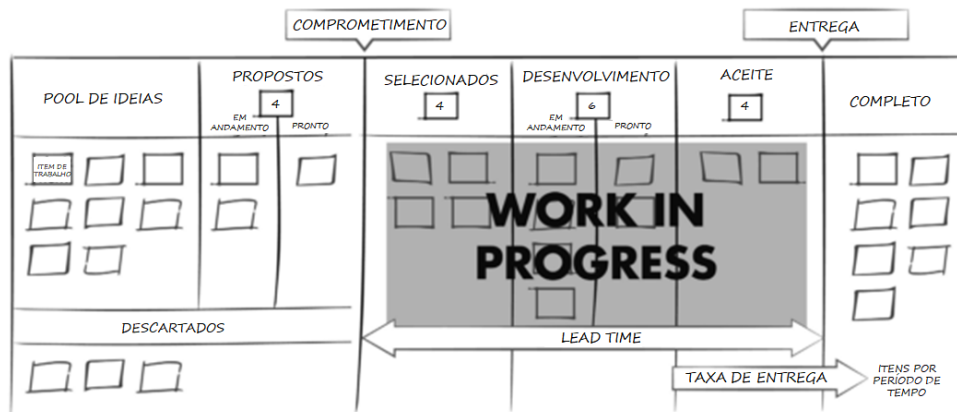


Figura 2.12: Exemplo de quadro Kanban (Fonte: [43]).

O método Kanban se concentra na entrega de serviços para uma organização, mesmo quando existe um produto físico de serviços, de maneira que o valor reside menos no próprio item e mais em seu conteúdo informativo (o software, no sentido mais geral) [43].

Orientado por valores, os criadores do método Kanban defendem nove valores[43]:

1. Transparência: o compartilhamento de informações melhora o fluxo de valor ao negócio;

2. Equilíbrio: os diferentes aspectos, pontos de vista e capacidades devem estar balanceados visando maior efetividade;
3. Colaboração: o método Kanban foi desenvolvido para aprimorar a maneira que as pessoas trabalham em conjunto;
4. Foco no cliente: o método é orientado a entrega de valor ao cliente, dessa forma o ponto central deve ser o cliente e o valor gerado ao mesmo pelo projeto;
5. Fluxo: enxergar o fluxo de trabalho é o cerne do método;
6. Liderança: no Kanban a liderança é fundamental em todos os níveis visando orientar e inspirar o time a gerar mais valor ao cliente e sempre melhorar;
7. Entendimento: o Kanban é um método de melhoria contínua, nesse sentido é importante que todos, time e organização, tenham o entendimento do ponto de início para que seja possível seguir em busca do aperfeiçoamento constante;
8. Consenso: o comprometimento de todos, independentemente de diferenças de opinião ou abordagens, em buscar atingir os objetivos em conjunto é essencial;
9. Respeito: valorizar, entender e demonstrar consideração pelas pessoas é a fundação para todos os outros valores.

Além dos valores, o método possui ainda práticas e princípios, e estabelece três outros direcionadores, chamados de agendas do método, que visam atingir as necessidades da organização. São elas: sustentabilidade; orientação ao serviço; e, capacidade de sobrevivência [43].

- Sustentabilidade: tem como objetivo encontrar um ritmo sustentável e um foco na evolução;
- Orientação ao serviço: tem foco na performance e na satisfação do cliente;
- Capacidade de sobrevivência: preocupa-se em manter a organização adaptativa e competitiva.

Os princípios e práticas estão demonstrados na Figura 2.13.

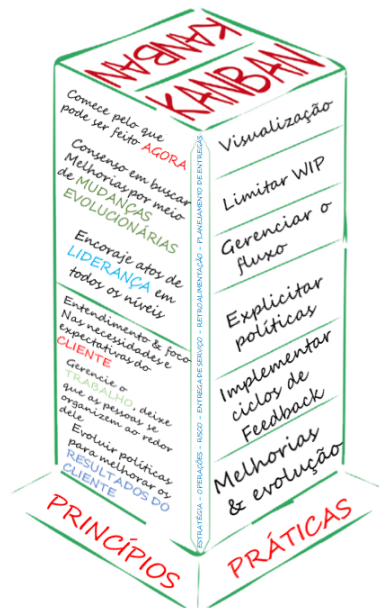


Figura 2.13: Princípios e práticas do método Kanban (Fonte: [43]).

Os princípios podem ser divididos em dois grandes grupos, os relacionados a Gestão de Mudança, que podem ser vistos na parte superior da Figura 2.13, e os relacionados a Entrega de Serviços, que são evidenciados na parte inferior da Figura 2.13 [43].

Princípios relacionados a Gestão de Mudança visam quebrar a resistência da organização e de seus colaboradores as mudanças e incluem [43]:

- Comece com o que pode ser feito agora;
- Consenso em perseguir as melhoras por meio da mudanças evolucionárias; e
- Encorajar a liderança em todos os níveis da organização.

Princípios relacionados a Entrega de Serviços enfatizam que o foco deve voltar para os clientes do serviço e o valor que eles recebem dele e compreendem [43]:

- Entender e focar nas necessidades e expectativas do cliente;
- Gerenciar o trabalho e deixar que o time se organize em torno dele; e
- Evoluir políticas visando melhorar os resultados do cliente e do negócio.

O método ainda preconiza a aplicação de uma série de práticas que envolvem basicamente: **enxergar** o trabalho e políticas para determinar como proceder; e, **aprimorar** o processo de maneira evolutiva, mantendo e amplificando as mudanças positivas e descartando aquelas negativas. Assim, são práticas essenciais do método Kanban [43]:

1. Visualizar: manter políticas e o trabalho visíveis fornece apoio na colaboração e no entendimento do sistema, tornando mais simples a identificação de possíveis melhorias;
2. Limitar o *Work in Progress* (WiP): novos itens não devem ser iniciados até que o item atual esteja completado ou seja abortado, ter muitos itens parcialmente completados significa desperdício. Observar, limitar e otimizar a quantidade de trabalho em andamento é essencial para o sucesso;
3. Controlar o fluxo: o fluxo no Kanban deve maximizar o valor da entrega, minimizar o tempo de entrega e ser o mais previsível possível;
4. Explicitar as políticas: explicitar as políticas é uma forma de articular e definir um processo que vá além da definição do fluxo de trabalho;
5. Implementar ciclos de *feedback*: os *feedbacks* são parte essencial de qualquer processo evolutivo. O método define sete oportunidades específicas de *feedback* chamadas de cadência: revisão estratégica; revisão das operações; revisão dos riscos; revisão da entrega dos serviços; reunião de retroalimentação; reunião Kanban; e reunião de planejamento de entrega.
6. Melhorar colaborativamente, evoluir de forma experimental: o Kanban é fundamentalmente um método de melhoria, ele parte do que existe na organização e, utilizando o paradigma do fluxo *Lean*, busca uma melhoria contínua e incremental.

Assim como os demais métodos, o Kanban deve ser aplicado de acordo com o contexto ao qual a organização se encontra e pode ser aplicado em sua essência ou em conjunto com outros métodos ágeis.

Dynamic Systems Development Method - DSDM

O DSDM surgiu em 1995 no Reino Unido por meio de uma organização formada por gestores, consultores e profissionais de TI, cuja a ideia fundamental de reunir as melhores práticas para o desenvolvimento e codificação de aplicações [44]. A ideia principal do DSDM é que ao invés de fixar a quantidade de funcionalidades do produto que deve ser entregue pelo projeto, ajustando se necessário o prazo de entrega e os custos do projeto, é preferível que sejam fixados o prazo de entrega e o orçamento do projeto e então, caso haja necessidade, ajustar as funcionalidades que devem ser entregues [45], como pode ser visto na Figura 2.14.

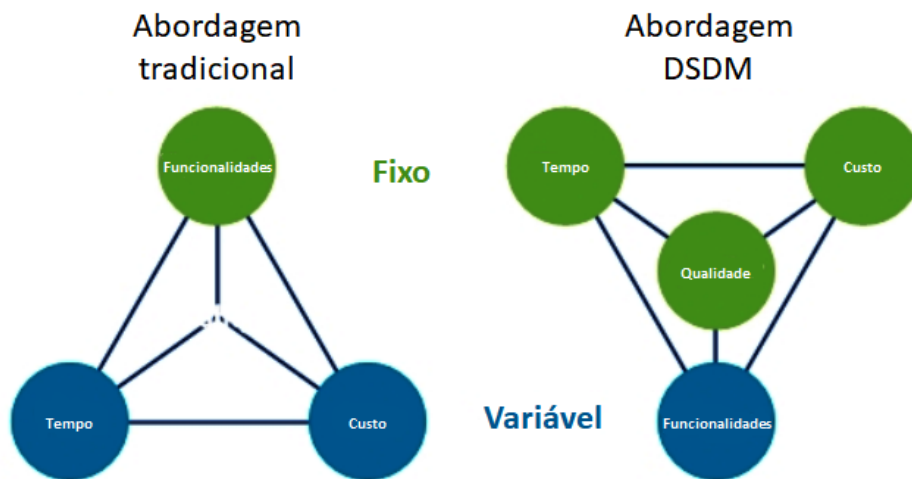


Figura 2.14: Variáveis de projeto - Abordagem tradicional x DSDM (Fonte: [46]).

O *framework* é baseado em nove princípios básicos, sendo que os 04 primeiros definem a fundação em que foi desenvolvido do DSDM e os demais guiaram a elaboração da estrutura do *framework* [44]:

1. O envolvimento ativo do usuário é imperativo;
2. O time DSDM deve ter autoridade suficiente para tomar decisões;
3. O foco é na entrega frequente de produtos;
4. Alinhamento com os propósitos do negócio é critério essencial de aceitação das entregas;
5. Desenvolvimento iterativo e incremental é necessário para convergir em uma solução precisa de negócio;
6. Todas as mudanças durante o desenvolvimento devem ser reversíveis;
7. Os requisitos são delimitados em alto nível;
8. Os testes devem ser integrados durante o ciclo de vida do desenvolvimento;
9. Uma abordagem cooperativa e colaborativa entre todas as partes interessadas é essencial.

O DSDM é estruturado em 07 fases bem definidas, sendo que destas, a primeira e a última são fases em que não há desenvolvimento efetivo e as outras envolvem o desenvolvimento efetivo do projeto: a primeira fase é a de pré-projeto, que visa assegurar que o projeto está estruturado de forma sólida; a segunda fase, a primeira das etapas de desenvolvimento, é a de análise de viabilidade que visa identificar se o sistema está adequado a

abordagem proposta e proporciona uma visão inicial de custos; a terceira fase diz respeito ao negócio e visa assegurar que o negócio contém a fundamentação técnica necessária para suportar todo o projeto; a quarta fase é a primeira fase do ciclo iterativo, é chamada de iteração do modelo funcional, onde análise iniciada no estudo de negócios é feita com mais detalhes, para apoiar essa análise é feita a prototipagem evolutiva da funcionalidade; depois de bem entendida a funcionalidade inicia-se a fase de desenho e construção, que contempla outra iteração do método; posterior a fase de construção inicia-se a fase de implementação da solução, que contempla o envio do sistema para o ambiente de produção e também o treinamento dos usuários; e por fim tem-se a última fase, a fase de pós-projeto que inclui manter a solução entregue operacional [44].

O DSDM pode ser associado a outras abordagens, tais como *Extreme Programming* e PRINCE2 por exemplo, e assim para criar um modelo integrado e mais robusto para os projetos.

Feature Driven Development - FDD

O Desenvolvimento Guiado por Funcionalidade, ou *Feature Driven Development* (FDD), em inglês, é um método de desenvolvimento de software orientado a processo essa abordagem FDD concentra-se nas fases de projeto e construção e incorpora o desenvolvimento iterativo com as práticas consideradas efetivas na indústria. Enfatiza os aspectos de qualidade ao longo do processo e inclui entregas frequentes e tangíveis, acompanhado de monitoramento preciso do andamento do projeto [45]. A Figura 2.15 demonstra o ciclo de desenvolvimento proposto por esse método.

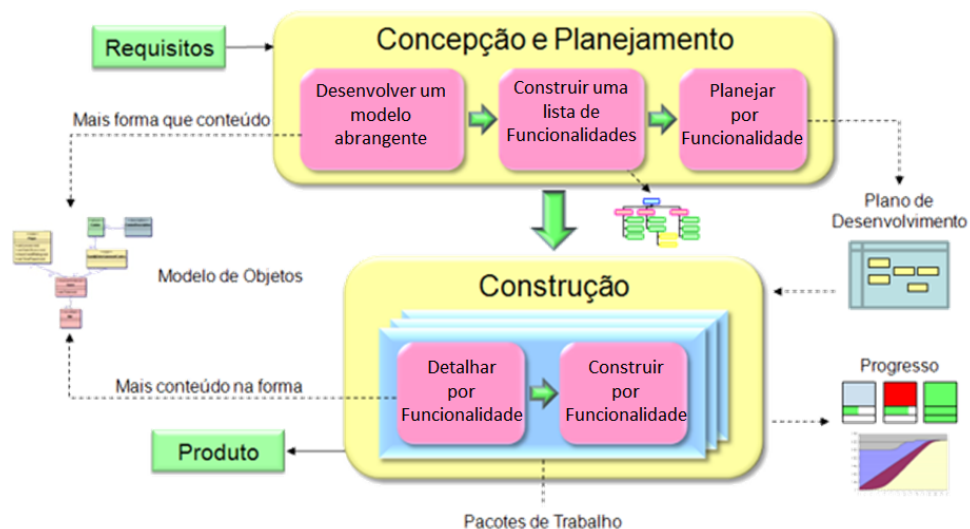


Figura 2.15: Processo do método *Feature Driven Development* (Fonte: [47]).

O primeiro passo, segundo o FDD, é a criação de um modelo abrangente, baseado em tudo que deve ser construído. Com base nesse modelo geral, os desenvolvedores constroem uma lista de funcionalidades necessárias para o atingimento dos objetivos do projeto. Com todas as funcionalidades elencadas, passa-se a fase de planejamento, onde os integrantes do time devem organizar as funcionalidades por ordem de prioridade de desenvolvimento e selecionar aquelas que podem ser desenvolvidas em iterações de até 02 semanas. O quarto passo do método é detalhar as funcionalidades selecionadas para a iteração e por fim desenvolvê-las, os dois últimos passos se repetem até que o sistema seja entregue a contento para o cliente [47].

Agile Unified Process - AUP

O Processo Ágil Unificado, ou *Agile Unified Process*, em inglês, é uma simplificação do *Rational Unified Process - RUP*. O método descreve uma maneira simples e de fácil entendimento para o desenvolvimento de softwares [48]. A figura 2.16 demonstra o ciclo de vida de uma iteração de acordo com o AUP.

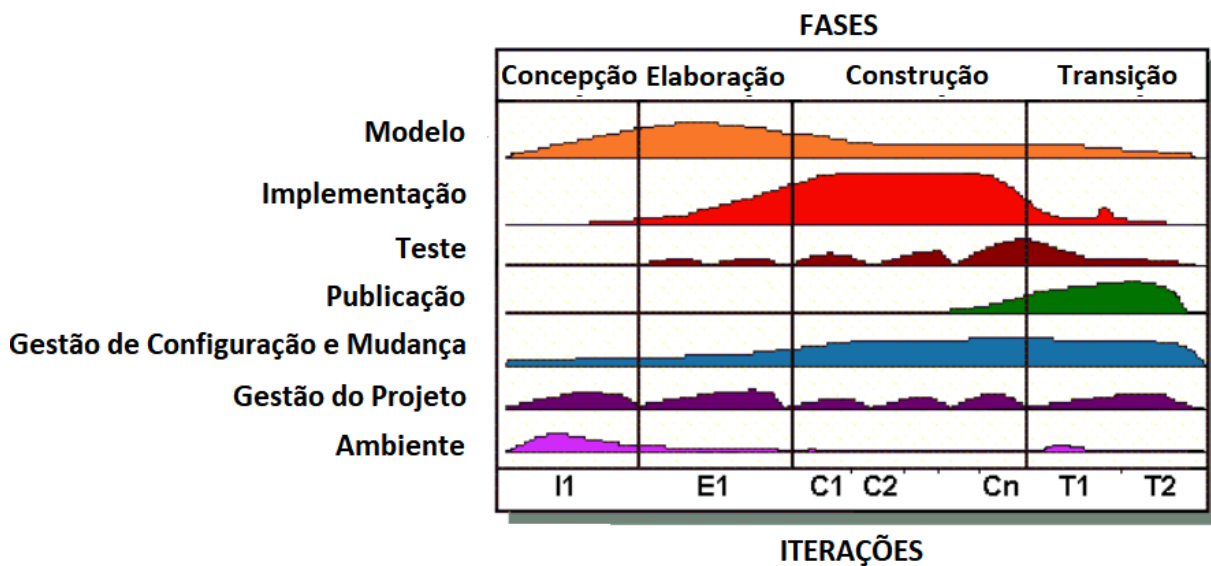


Figura 2.16: *Agile Unified Process* (Fonte: [48]).

O método está dividido em 04 fases: Concepção, cujo objetivo é identificar o escopo inicial do projeto e a aprovação e financiamento necessários; Elaboração, que visa aprovar a arquitetura da aplicação; Construção, essa fase é a fase de desenvolvimento do projeto, é feita de forma incremental, sempre buscando atender as necessidades mais prioritárias dos *stakeholders*; e Transição, que tem como objetivo validar o que foi produzido e colocar o software em ambiente produtivo [48].

Para cada iteração são definidas uma série de etapas, chamadas de disciplinas. Essas disciplinas definem as atividades necessárias a serem seguidas para que seja entregue um produto de valor e que atenda as expectativas das partes interessadas [48].

1. Modelo: nessa disciplina deve-se entender o negócio da organização, o problema que está sendo abordado e identificar uma solução;
2. Implementação: esta disciplina visa transformar o modelo em código executável;
3. Teste: visa assegurar a qualidade e validar o funcionamento do que foi desenvolvido;
4. Publicação: tem como objetivo planejar a entrega do que foi desenvolvido e disponibilizar o sistema aos usuários finais;
5. Gestão de Configuração e Mudança: objetivo desta disciplina é gerenciar o acesso aos artefatos do projeto, inclui o rastreamento de versões de artefatos ao longo do tempo, como também o controle e gerenciamento de alterações neles;
6. Gestão do Projeto: essa disciplina inclui todas as atividades de gestão do projeto, tais como gerenciamento de riscos, coordenar as pessoas e os sistemas fora do escopo do projeto e assegurar que o projeto seja entregue dentro do prazo e dentro do orçamento; e
7. Ambiente: esta é uma disciplina de apoio que visa assegurar o processo adequado, garantindo que as orientações e ferramentas estão acessíveis e disponíveis ao time do projeto.

O AUP foi criado com base em alguns conceitos básicos. Primeiro, as pessoas não lerão a documentação detalhada do processo, no entanto irão querer orientação e treinamento de alto nível de tempos em tempos. Segundo as coisas devem ser descritas e em poucas páginas. Terceiro, concentre-se nas atividades e riscos importantes. Quarto, as equipes devem usar as ferramentas que se sentem mais confortáveis para um determinado trabalho, e não ferramentas impostas pelo processo. E, finalmente, o processo deve ser altamente personalizável para que seja possível adaptá-lo às necessidades de um projeto específico [49].

Crystal Clear

O *Crystal Clear* é uma metodologia que faz parte do conjunto de metodologias *Crystal* de desenvolvimento de software. A família *Crystal* é dividida por cores, as metodologias são aplicadas de acordo com o tamanho e complexidade do projeto e necessidade de comunicação e coordenação, a *Crystal Clear* é aplicável a projetos com times de até 8

peças, o *Crystal Yellow* para times de 10 a 20 pessoas, o *Orange* para times de 25 a 50, o *Red* de 50 a 100, e assim sucessivamente através do *Maroon*, *Blue* e *Violet* [50].

A família de metodologias *Crystal* tem em comum as mesmas prioridades, que são [50]:

- Segurança no resultado do projeto;
- Eficiência no desenvolvimento; e
- Habitabilidade das convenções (o time precisa aceitar o processo).

O *Crystal* tem a equipe do projeto orientada para sete propriedades, sendo as três primeiras essenciais e as demais podem ser adicionadas qualquer ordem para aumentar a margem de segurança do projeto [50]:

1. Entregas frequentes;
2. Melhoria reflexiva;
3. Comunicação próxima;
4. Segurança pessoal (o primeiro passo na confiança);
5. Foco;
6. Fácil acesso a usuários experientes; e
7. Ambiente de desenvolvimento com testes automatizados, gerenciamento de configuração e integração frequente.

O *Crystal Clear* também possui uma série de princípios, cuja ideia central é [50]:

- A quantidade de detalhes necessários nos documentos de requisitos, *design* e planejamento varia de acordo com as circunstâncias do projeto;
- Trabalho intermediários podem ser reduzidos na medida em que caminhos de comunicação curtos, ricos e informais estão disponíveis para a equipe e software testado e funcionando é entregue frequentemente; e
- O time deve ajustar continuamente suas convenções de trabalho para se adequar às personalidades particulares da equipe, ao ambiente de trabalho local e às peculiaridades de tarefas específicas.

O *Crystal* não exige nenhuma técnica específica para ser usado, e pode ser empregado por qualquer pessoa no projeto, sendo o *Crystal Clear* é uma otimização que pode ser

aplicada quando a equipe é composta de duas a oito pessoas sentadas na mesma sala ou em escritórios adjacentes, isso facilita uma "osmótica", o que significa que as pessoas se ouvem mutuamente discutindo diariamente as prioridades, o status, os requisitos e o projeto do projeto [50].

As abordagens ágeis modificam a maneira como equipes de desenvolvimento trabalham e seu relacionamento com as áreas de negócio. Essas modificações geraram muitos benefícios e solucionaram diversos riscos em projetos de desenvolvimento, no entanto riscos ainda precisam ser identificados e analisados como passo inicial na busca por diminuir a probabilidade de fracasso do projeto [51].

2.2 Gestão de riscos em projetos

Os riscos podem ser definidos como o efeito das incertezas nos objetivos do projeto [5]. Como se trata do efeito das incertezas em algum objetivo do projeto, os riscos podem ser classificados como ameaças ou oportunidades [4].

A gestão de riscos requer a identificação e controle da exposição da organização aos riscos que podem impactar seus objetivos de negócio [5]. Tem como objetivo, além de potencializar a ocorrência de eventos que oferecem oportunidades ao projeto e minimizar a possibilidade da ocorrência de eventos negativos ao projeto [4], sistematizar esse processo para que ocorra de forma visível que possa ser repetido e seja aplicável de forma consistente visando apoiar a tomada de decisão fundamentado em um entendimento arraigado dos riscos e seu possível efeito nos objetivos da organização.

A gestão eficaz dos riscos está pautada em princípios, estrutura e processos, que se relacionam conforme identificado na Figura 2.17

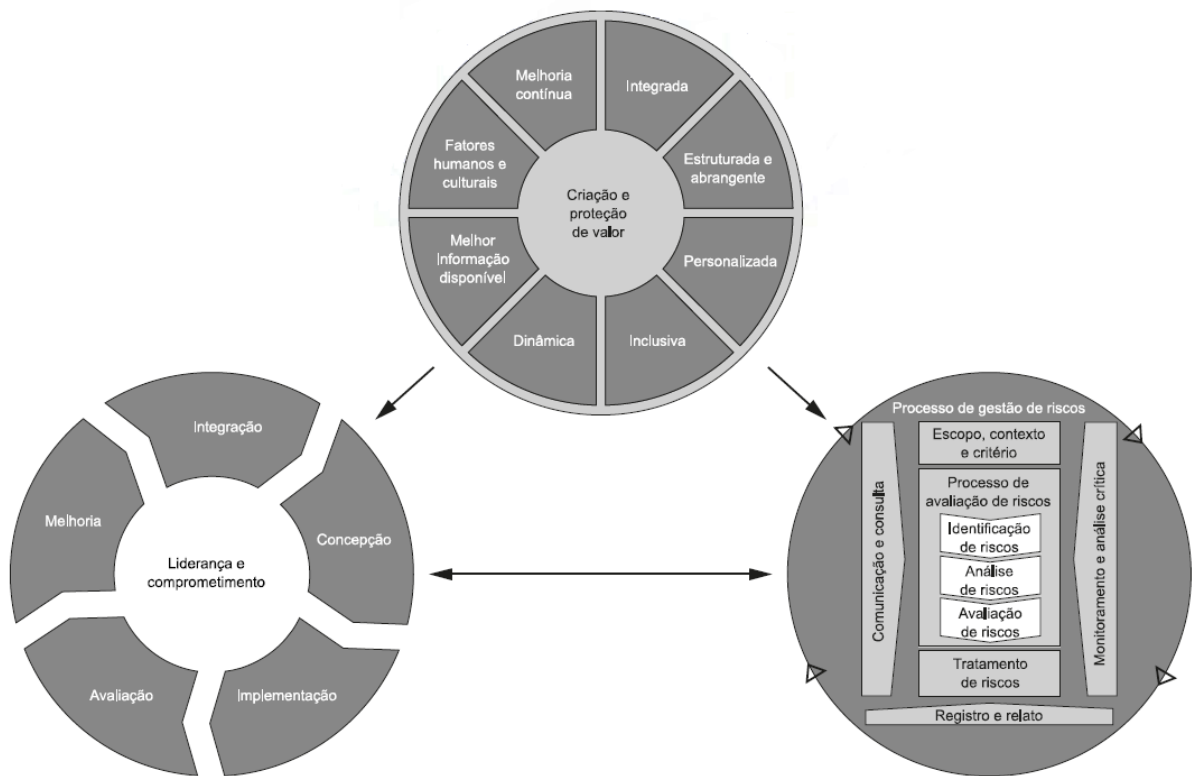


Figura 2.17: Princípios, estrutura e processos para gestão de riscos (Fonte: [5]).

O preocupação central da gestão de riscos é a criação e proteção de valor. Isso significa que a gestão de riscos contribui para a realização dos objetivos e para a melhoria do desempenho do projeto. É tamanha sua importância que o tema aparece no centro dos princípios abordados na ISO 31000:2018. Esses princípios visam oferecer orientações sobre as características da gestão de riscos eficaz e eficiente e também fornecem a base para o gerenciamento de riscos, são eles [5]:

1. *Integrada*: a gestão de riscos é parte integrante de todos os processos organizacionais.

Toda a organização deve ser responsável pela gestão eficaz dos riscos, esta atividade não pode ser dissociada das principais atividades e processos da organização.

2. *Estruturada e abrangente*: a gestão de riscos deve possuir uma estrutura sólida e abrangente.

Este tipo de abordagem estruturada e abrangente contribui para a obtenção de resultados mais consistentes e compatíveis com os objetivos da organização.

3. *Personalizada*: a gestão de riscos deve ser personalizada.

Cada organização, estrutura ou projeto possui uma estrutura única, nesse sentido nem todos os processos são iguais para todos. Dessa forma é necessário personalizar

os processos de gestão de riscos de acordo com os contextos interno e externo da organização com vistas a ter maior eficácia e eficiência nos processos e contribuir para o atingimento dos objetivos da organização.

4. *Inclusiva*: a gestão de riscos é transparente e inclusiva.

Os avaliadores devem envolver todas as partes interessadas no projeto, consultá-las e informá-las acerca dos processos de gestão de riscos do projeto, isso possibilita que seus conhecimentos e pontos de vista sejam considerados e resulta numa melhor conscientização por todos.

5. *Dinâmica*: a gestão de riscos deve ser diligente.

Os contextos interno e externo de organizações mudam constantemente e isso faz com que surjam ameaças e oportunidades aos projetos. A gestão de riscos deve se antecipar e detectar, reconhecer e responder a estas mudanças de forma oportuna.

6. *Melhor informação disponível*: a gestão de riscos baseia-se nas melhores informações disponíveis.

O processo de gestão de riscos tem como sua principal entrada as informações, sejam dados históricos, experiências, retroalimentação dos *stakeholders*, observações, entre outros.

7. *Fatores humanos e culturais*: pessoas desenvolvem projetos e a gestão de risco deve considerá-las.

O comportamento e cultura influenciam consideravelmente todos os aspectos da gestão e da organização.

8. *Melhoria contínua*: a gestão de riscos facilita a melhoria contínua por meio do aprendizado e experiências.

É importante que as organizações desenvolvam e implementem estratégias para evoluir e melhorar a maturidade em gestão de riscos.

Os *frameworks* para gestão de riscos são constituídos por uma sequência de passos bem definidos e estruturados, aplicados de forma sequencial. Apesar de serem passivos de uso em qualquer tipo de projeto [7] a estrutura e forma de aplicação desses *frameworks* está muito próxima do gerenciamento tradicional de projetos com fases bem definidas e sequenciais. Para que seja possível ter um melhor entendimento das diferenças da gestão de riscos entre as metodologias, na Seção 2.2.1 será explorado o gerenciamento de riscos na metodologia tradicional, na Seção 2.2.2 será evidenciado o gerenciamento de riscos em TI e na Seção 2.2.3 será explorado o gerenciamento de riscos em metodologias ágeis.

2.2.1 Gestão de riscos na metodologia tradicional

A metodologia tradicional de gestão de projetos, como visto na Seção 2.1.1, recomenda uma abordagem estruturada, com um planejamento extenso e fases sequenciais e bem definidas. Os *frameworks* atuais de gestão de risco consistem em passos bem delimitados para gerir riscos dentro dos limites suportável pela organização. A norma ABNT NBR ISO 31000 é um dos *frameworks* mais aplicados atualmente. Essa norma surgiu na necessidade de compatibilizar padrões publicados anteriormente, integra diversas metodologias e terminologias e consiste em um processo e estrutura abrangente buscando assegurar que o risco será gerenciado de forma eficaz, eficiente e coerente [52].

A ISO 31000 descreve detalhadamente o processo sistemático e lógico que organizações podem utilizar para gerenciar riscos. Contempla atividades de identificação, análise, avaliação e tratamento do risco. Durante todo o processo devem ser comunicadas e consultadas as partes interessadas e monitorados e controlados os riscos visando certificar que nenhum tratamento adicional seja necessário [5]. A Figura 2.18 apresenta o processo de gestão de risco segundo a norma ABNT NBR ISO 31000:2018.



Figura 2.18: Processo de gestão de riscos (Fonte: [5]).

A primeira atividade no processo de gestão de riscos segundo a norma é o estabelecimento do contexto. Essa atividade visa avaliar a natureza e a complexidade dos riscos, por meio da captura dos objetivos da organização, o ambiente interno e externo, as partes interessadas e os critérios de risco no contexto da organização. Esse processo é seguido pelo processo de avaliação do de riscos, que é dividido em três etapas: *identificação*, *análise e avaliação dos riscos*, que por sua vez são seguidos pela etapa de *tratamento dos riscos*. Estas etapas têm como objetivo [5]:

- *Identificação dos riscos*: o objetivo dessa etapa é gerar uma lista abrangente de riscos baseada nas possíveis fontes de riscos, identificando também as áreas impactadas, eventos e suas causas e consequências potenciais aos objetivos do projeto;
- *Análise dos riscos*: tem como objetivo a compreensão dos riscos, envolve a apreciação das causas e fontes de risco, suas consequências positivas e negativas, e a probabilidade de que essas possam ocorrer;
- *Avaliação dos riscos*: a avaliação dos riscos tem como finalidade apoiar a tomada de decisão. Envolve uma comparação do nível de risco encontrado durante o processo de análise com os critérios estabelecidos durante a identificação do contexto para, assim, definir a necessidade de tratamento do risco;
- *Tratamento dos riscos*: esta etapa objetiva selecionar uma ou mais opções para tomar ações para modificar os riscos, bem como a implementação destas ações.

O Monitoramento e análise crítica é um processo contínuo que visa assegurar e melhorar a qualidade e eficácia da concepção, implementação e resultados do processo de gestão de riscos. O Registro e relato é a atividade de documentação de todo processo e visa fornecer informações para tomada de decisão, melhorar as atividades de gestão de riscos, comunicar as atividades e resultados e apoiar a interação com as partes interessadas.

Existem diversos *frameworks* que tratam da gestão de riscos, dentre eles, um dos mais conhecidos mundialmente é o *Practice Standard for Risk Management* do *Project Management Institute (PMI)*. Este material é uma derivação do *Project Management Body of Knowledge (PMBOK)* e tem uma abordagem abrangente sobre a gestão de riscos, com um conjunto de processos específicos para tal disciplina. Este material divide a gestão de riscos em 06 processos:

- *Planejar o gerenciamento dos riscos*: este processo tem como objetivo desenvolver a estratégia de gerenciamento de riscos do projeto, para definir como os processos de gestão de riscos serão executados e integrados com os demais processos de gestão de projetos;

- *Identificar os riscos*: este é um processo iterativo, pois deve ser revisitado diversas vezes no decorrer do projeto, e tem como objetivo identificar os riscos em sua máxima extensão;
- *Realizar a análise qualitativa dos riscos*: este processo avalia os riscos individualmente identificados e os prioriza com base em suas características;
- *Realizar a análise quantitativa dos riscos*: este processo visa prover uma estimativa numérica sobre o impacto do risco nos objetivos do projeto;
- *Planejar respostas aos riscos*: este é o processo de determinar as ações efetivas que serão tomadas como respostas aos riscos identificados individualmente e para o projeto como um todo, visa assegurar que as chances de sucesso do projeto sejam aumentadas considerando as restrições do mesmo;
- *Monitorar e controlar os riscos*: o objetivo desse processo é monitorar e controlar os riscos do projeto com vistas a identificar riscos residuais, novos riscos, garantir que as respostas aos riscos sejam executadas no prazo certo e avaliar sua efetividade no decorrer do ciclo de vida do projeto.

São consideradas estratégias para respostas aos riscos:

- *Evitar/Explorar*: envolve tomar ações para que uma ameaça não ocorra ou, caso ocorra tenha impacto mínimo nos objetivos do projeto e, no caso de uma oportunidade, assegurar que caso ocorra esta seja aproveitada da melhor forma pelo projeto;
- *Compartilhar/transferir*: envolve transferir o risco a um terceiro que possa melhor trabalhar com a ameaça ou explorar melhor uma oportunidade;
- *Mitigar/aumentar*: esta estratégia envolve tomar ações que visam aumentar a probabilidade e/ou o impacto das oportunidades e diminuir a probabilidade e/ou impacto das ameaças;
- *Aceitar*: esta estratégia é utilizada quando nenhuma das demais se aplica, envolve não tomar ação alguma a não ser que o risco ocorra, no caso da ocorrência do risco deve-se ter preparado um plano de contingência para que o mesmo seja implementado.

Com uma abordagem baseada em processos bem definidos, etapas sequenciais, aplicação sistemática de princípios e abordagens para identificar, avaliar e monitorar os riscos, os *frameworks* atuais, tais como a Norma ABNT NBR ISO 31000, PMBOK, *Practice Standard for Project Risk Management*, *Management of Risk (MoR) framework*, entre

outros, se tornam muito próximos do modelo tradicional de gestão de projetos. No entanto, para todas as indústrias, deve-se levar em consideração que, ao avaliar os riscos relacionados a uma determinada solução, o nível de risco estabelecido baseia-se em conhecimentos básicos, incluindo pressupostos, compreensão fenomenológica, modelos, dados e declarações de especialistas [53]. Assim, para a área de Tecnologia da Informação, assim como em qualquer outro projeto, devem ser consideradas as peculiaridades da indústria quando se pensa em riscos em projetos.

2.2.2 Gestão de riscos em projetos de TI

O desenvolvimento de softwares data da década de 60 [54] e, como em qualquer campo, nos seus estágios iniciais ocorreram alguns desastres. O estudo de muitos destes desastres revelou que os problemas ocorridos poderiam ter sido evitados ou altamente reduzidos caso houvesse uma preocupação inicial na identificação e resolução de seus elementos de maior potencial de risco [55]. Ainda assim atualmente é visto que a taxa de sucesso no desenvolvimento de novas aplicações continua baixa, isso se dá em razão de muitos aspectos e práticas influenciarem para o atingimento do sucesso do projeto [54].

Visto que os softwares são nos dias de hoje componentes essenciais para qualquer ambiente de negócio, dessa forma se faz necessário um esforço maior no gerenciamento dos riscos envolvendo projetos de desenvolvimento de novas aplicações [56]. Riscos em TI podem ser descritos como eventos relacionados a TI que podem causar impacto no negócio, e estes podem acontecer com frequência e impactos incertos, e criar barreiras para o atingimento dos objetivos estratégicos do negócio [57]. Para projetos de desenvolvimento de software os riscos devem estar endereçados não apenas ao atingimento dos objetivos do projeto em custo e tempo, mas também em desempenho e confiabilidade do sistema [56].

Os *frameworks* comuns da indústria podem ajudar a determinar como prevenir ou mitigar riscos comuns em determinados setores específicos [58]. Uma abordagem comum, rápida e de baixo custo de identificar riscos em processos de desenvolvimento de software e avaliar sua exposição são as listas de verificação. Estratégias de controle de riscos visam reduzir, ou até mesmo eliminar a probabilidade de ocorrência das ameaças ao projeto de desenvolvimento bem como se mostram úteis para limitar o seu impacto [59].

Em ambientes que utilizam metodologias ágeis, algumas atividades de gerenciamento de riscos estão intrinsecamente incorporadas no método ágil utilizado. No entanto, a área do processo de gerenciamento de riscos incentiva uma abordagem mais sistemática para gerenciar riscos, tanto técnicos como não técnicos [58].

Dadas as diferenças entre abordagens tradicionais e ágeis, a identificação de riscos de em metodologias ágeis requer uma nova visão, que não depende excessivamente do conhecimento de riscos de desenvolvimento tradicional [51].

2.2.3 Gestão de riscos em metodologias ágeis

O gerenciamento de riscos em projetos ágeis continua sendo uma atividade passiva e implícita que pode ser mal direcionada e muitas vezes incompreendida [10]. As metodologias ágeis afirmam que os mecanismos de *feedback* reduzem o risco (negativo) ao disponibilizar a informação em tempo hábil, para que os membros do projeto possam anunciar resultados negativos [10].

Uma das metodologias ágeis mais conhecidas e aplicadas no mundo é o *Scrum*. Essa metodologia proporciona a redução dos riscos por meio da colaboração com clientes e *stakeholders* durante todo o desenvolvimento do projeto e também pelas entregas frequentes em períodos curtos de tempo [38]. O *Guia para o Conhecimento em Scrum (SBOK)* sugere a identificação diária de possíveis riscos pelo time *Scrum* como prática de gestão de riscos nessa metodologia, como evidenciado pela Figura 2.19 [60]:

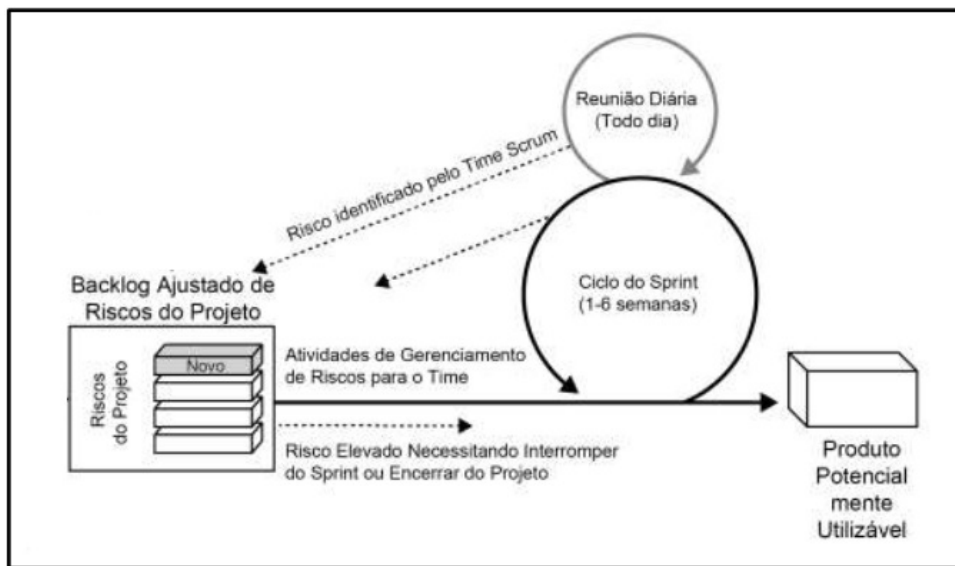


Figura 2.19: Modelo de gestão de riscos para metodologia *Scrum* (Fonte: [60]).

Apesar de frequentemente mencionado, os riscos são tratados de maneira bastante restrita e implícita em projetos ágeis [10]. É consenso que riscos ao projeto são aqueles fatores que impactam no atingimento dos objetivos do projeto e que possivelmente podem levá-lo ao fracasso. Nesse cenário, para que possa ser feito um gerenciamento eficaz dos riscos do projeto é preciso que seja definido previamente o que e quais são fatores que

levam um projeto a não lograr êxito. Dessa forma a Seção 2.3 tem como objetivo esclarecer o que são fatores de sucesso e de insucesso em projetos.

2.3 Fatores e critérios de sucesso e insucesso em projetos

Existem poucos tópicos dentro da gestão de projetos que são tão frequentemente discutidos, e ainda sem chegar a um consenso, como a noção do que seja sucesso de um projeto [61].

A classificação de sucesso ou insucesso de projetos é um constructo multidimensional que envolve diversos fatores [62]. A literatura propõe múltiplas definições de sucesso e insucesso em projetos de desenvolvimento de aplicações bem como descreve diversos aspectos que influenciam na entrega do projeto a contento, como por exemplo: estrutura organizacional; comunicação com *stakeholders*; requisitos do cliente e especificações do sistema; cronograma e orçamento; satisfação do cliente; qualidade; suporte da alta gestão; ferramentas e processos de gestão; metodologias de desenvolvimento de software, dentre outras [54].

Nesse momento se faz necessário que seja feita uma distinção entre alguns conceitos. Primeiramente não se deve confundir sucesso em projeto, que é aquele referente ao atingimento dos objetivos do projeto, e sucesso da gestão do projeto, que é aquele relacionado com os aspectos gerenciais do projeto, tais como custo, cronograma e qualidade. Outra diferenciação a ser clarificada é que existem fatores de sucesso, aqueles que se referem aos insumos para o sistema de gestão que conduzem direta ou indiretamente ao sucesso do projeto ou da organização, e critérios de sucesso que estão ligados as medidas pelas quais o sucesso ou o fracasso de um projeto será julgado [63].

Historicamente o entendimento de sucesso em projetos está ligado ao conceito simplista da tripla restrição, conhecido ainda como *iron triangle* [64], e envolve as variáveis tempo, escopo e custo do projeto. O *The Standish Group*, em 2014, realizou uma pesquisa com 309 respondentes sobre a definição de sucesso em projetos de TI e obteve os seguintes resultados:

Tabela 2.1: Definição de sucesso em projetos de TI [1]

#	Característica	Percentual
1	Dentro do prazo	30%
2	No custo planejado	32%
3	Com escopo completado	26%
4	Alinhado as estratégias organizacionais	29%
5	Gera valor para a organização	52%
6	Satisfaz o cliente	41%
	Todos acima	33%
	Tripla restrição (<i>Iron Triangle</i>)	15%

Diversos autores têm pesquisado quais são os possíveis fatores e critérios de sucesso e insucesso em projetos, no entanto ainda não existe um consenso sobre os fatores e critérios determinantes para o fracasso de um projeto. Dessa forma para que o tema possa ser abordado com consistência e amparo científico é necessário um sólido levantamento visando coletar a visão de inúmeros decisores. Por se tratar de uma pesquisa é necessária validação científica e esta pode ser alcançada por meio da modelagem por equações estruturais.

2.4 Modelagem por equações estruturais

A modelagem por equações estruturais, ou *Structural Equations Modeling (SEM)*, em inglês, pode ser melhor definida como uma classe de metodologias que procura representar hipóteses sobre os meios, variâncias e covariâncias de dados observados em termos de um número menor de parâmetros estruturais definidos por um modelo hipotético. Esse tipo de modelagem representa o híbrido de duas vertentes estatísticas separadas. A primeira é a análise fatorial desenvolvida nas disciplinas de psicologia e psicometria. A segunda é a modelagem simultânea de equações desenvolvida principalmente em econometria, mas com uma história antiga no campo da genética [18]. Pode ser visto como uma forma geral de modelos estatísticos comumente empregados, tais como análise de variância, análise de covariância, regressão múltipla, análise fatorial, análise de caminho crítico, modelos econométricos de equação simultânea e modelagem não recursiva, modelagem multinível e modelagem de curva de crescimento latente [65].

O principal objetivo deste tipo de análise é confirmar hipóteses de pesquisa sobre os meios observados, variâncias e covariâncias de um conjunto de variáveis. As hipóteses devem ser representadas por uma série de parâmetros estruturais, como por exemplo, cargas de fatores, caminhos de regressão, que são o menores do que o número de parâmetros observados [65]. Normalmente este tipo de modelagem é representado por meio de um

diagrama de rede formado pelas relações sugeridas (hipóteses) entre as variáveis latentes e seus indicadores e relações direcionais (regressões) e não direcionais (correlações) [65].

Existem dois tipos de abordagem por modelagem de equações estruturais, a primeira é baseada em covariância e é chamada de *Covariance-based Structural Equations Modeling (CB-SEM)*, cujo objetivo é confirmar ou rejeitar teorias e faz isso determinando o quão bem um modelo teórico proposto pode estimar a matriz de covariância para um conjunto de dados de amostra [66]. A segunda abordagem é por mínimos quadrados parciais, ou *Partial Least Squares (PLS)* em inglês, e é usada principalmente para desenvolver teorias em pesquisas exploratórias, faz isso com o foco em explicar a variância nas variáveis dependentes ao examinar o modelo [66].

Dentre as abordagens de modelagem por equações estruturais, a modelagem por mínimos quadrados parciais, do inglês *Partial Least Squares Structural Equation Modeling (PLS-SEM)*, tem se tornado a técnica de análise multivariável mais aplicada em pesquisa de gerenciamento [67]. A Figura 2.20 demonstra o crescimento do uso dessa abordagem nas áreas de marketing, no gerenciamento de sistemas de informação (*Management Information Systems Quarterly (MIS Quarterly)*) e no gerenciamento estratégico [68].

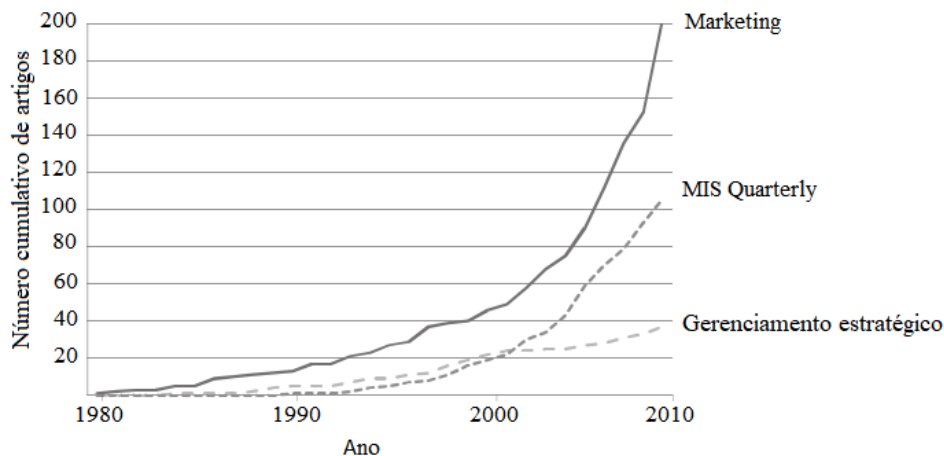


Figura 2.20: Crescimento do uso da abordagem PLS (Fonte: [68]).

Grande parte desse aumento pode ser associado a habilidade dessa abordagem em tratar modelagens problemáticas comuns nas ciências sociais [68]. A abordagem por mínimos quadrados parciais funciona com muita eficiência ao se trabalhar com modelos com muitos constructos e diversos relacionamentos ou com muitos indicadores por constructo, nessas situações o PLS alcança níveis estatísticos mais altos e atinge a convergência com maior frequência do que utilizando-se o CB-SEM [68]. Dessa forma, esta foi a abordagem escolhida para este estudo.

2.4.1 Modelagem de equações estruturais por mínimos quadrados parciais (PLS-SEM)

A modelagem de equações estruturais por mínimos quadrados parciais tem como principal objetivo estimar a variação de constructos endógenos e suas respectivas variáveis de entrada [?]. Essa técnica permite estimar relacionamentos entre $Q(q=1,2,\dots,Q)$ blocos de variáveis expressas por constructos não observáveis, ou seja, a técnica basicamente constrói um sistema de equações interdependentes baseadas em regressões simples e múltiplas. Esse esquema estima a rede de relacionamento entre variáveis latentes e entre as variáveis latentes e suas variáveis manifestas [69].

Essa abordagem é formada por dois submodelos. O submodelo interno, ou também conhecido como um modelo estrutural, especifica as relações entre as variáveis latentes independentes (também chamadas de exógenas, pois explicam o modelo, neste caso ξ_1 e ξ_2) e dependentes (também chamadas de endógenas, por serem explicadas pelo modelo, ξ_3 no modelo apresentado), e o submodelo externo, conhecido como um modelo de medição, que especifica as relações entre as variáveis latentes e as variáveis manifestas (ou indicadores), indicadas pelos retângulos X_{pq} , como pode ser visto na Figura 2.21 [70].

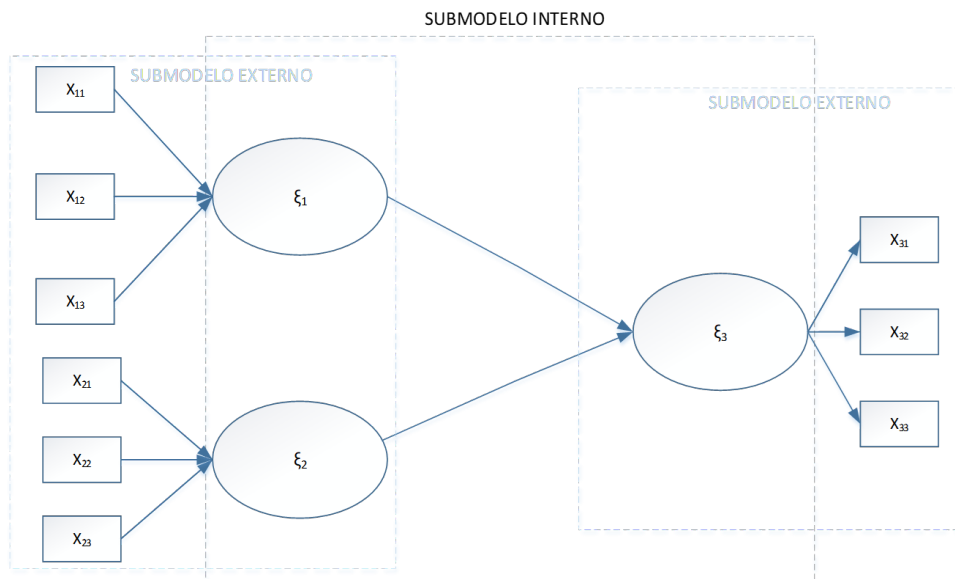


Figura 2.21: Modelo conceitual de equações estruturais (Fonte: [70]).

O modelo estrutural pode ser matematicamente descrito por:

$$\xi_j = \beta_{0j} + \sum_{q:\xi_q \rightarrow \xi_j} \beta_{qj} \xi_q + \zeta_j$$

Onde $\xi_j (j = 1, 2, \dots, J)$ é a variável genérica endógena latente, β_{qj} é o coeficiente genérico que relaciona a q -ésima variável latente exógena a j -ésima variável latente endógena, e ζ_j se refere ao erro da relação interna [69].

Os modelos de medição, por sua vez, podem ser constituídos exclusivamente por indicadores reflexivos ou formativos ou por uma combinação de ambos, sendo que a decisão se um determinado constructo deve ser operacionalizado com indicadores formativos e/ou reflexivos deve basear-se em considerações teóricas [71]. Para qualquer um dos modelos selecionados, devem ser respeitadas as seguintes hipóteses [72]:

1. As variáveis manifestas devem ter média zero e variância 1;
2. Cada conjunto de variáveis manifestas deve ser primeiramente transformado para que seja possível garantir correlações positivas com as variáveis latentes $y_g, g = 1, 2, \dots, G$.

Modelos de Reflexivo

Os indicadores reflexivos podem ser vistos como uma amostra representativa de todos os itens possíveis disponíveis no domínio conceitual da construção, assim, uma vez que uma medida reflexiva determina que todos os itens indicadores são causados pelo mesmo constructo, os indicadores associados a um determinado constructo devem estar altamente correlacionados entre si [66]. Ademais itens individuais devem ser intercambiáveis e qualquer item único geralmente pode ser excluído sem alterar o significado do constructo, desde que se mantenha sua confiabilidade. O fato de o relacionamento passar do constructo a seus indicadores, implica que, caso haja mudança no constructo, todos os indicadores mudarão simultaneamente [66]. A Figura 2.22 indica esse modelo.

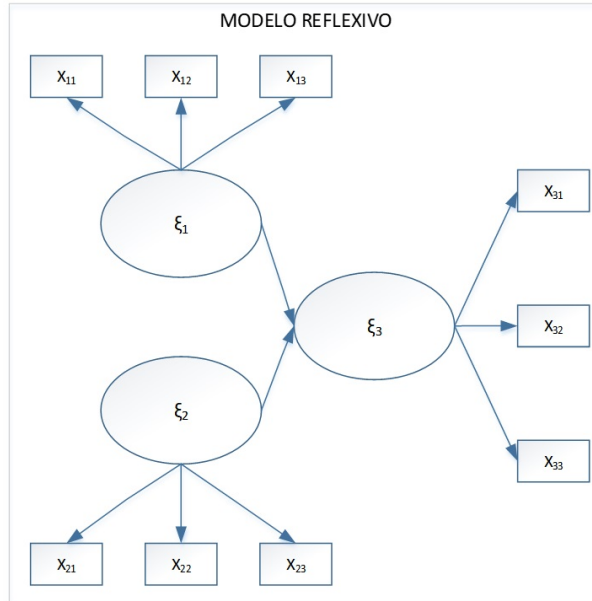


Figura 2.22: Modelos de medição reflexivo.

Nesse modelo, uma variável latente se forma por um conjunto de variáveis manifestas. Dessa forma seu modelo de medida é definido uma regressão linear simples [69]:

$$x_{pq} = \lambda_{p0} + \lambda_{pq}\xi_q + \epsilon_{pq}$$

Onde λ_{pq} é o peso associado a p – ésima variável manifesta no q – ésimo bloco e a imprecisão do processo de medição é representada pelo erro ϵ_{pq} .

De acordo com a abordagem, todas as variáveis latentes são estimadas em forma de combinações lineares dos seus indicadores, nesse sentido, deve obedecer a hipótese de que o erro ϵ_{pq} tem média igual a zero e é não correlacionado com a variável latente do mesmo bloco, nesse sentido, deve obedecer a seguinte relação matemática:

$$E(x_{pq}|\xi_q) = \lambda_{p0} + \lambda_{pq}\xi_q$$

Neste tipo de relacionamento, as variáveis manifestas devem ser unidimensionais no sentido da análise fatorial. O objetivo é que cada conjunto de indicadores se relacione fortemente com sua variável latente [73]. Dessa forma, é preciso garantir a unidimensionalidade e para isso são utilizadas tipicamente 03 técnicas [72]:

- *α de Cronbach*: é uma medida de consistência interna de uma escala unidimensional de vários itens [71], que fornece uma estimativa da confiabilidade com base nas intercorrelações dos indicadores observados [66] e pode ser definido por [74]:

$$\alpha = \frac{\sum_{p \neq p'} \text{cor}(x_{pq}, x_{p'q})}{P_q + \sum_{p \neq p'}, \text{cor}(x_{pq}, x_{p'q})} \times \frac{P_q}{P_q - 1}$$

Onde, P_q se refere ao número de variáveis manifestas presentes no bloco q .

O alfa de Cronbach quantifica o quão bem um conjunto de indicadores impacta um constructo unidimensional, caso os dados formem uma estrutura multidimensional este coeficiente tende a ser baixo. Um problema na avaliação do alfa de Cronbach é que as correlações entre os indicadores e o comprimento da escala são críticas, influenciando o alfa. Além disso, o tamanho da amostra tem um efeito significativo na precisão da estimativa de alfa [71]. Valores deste coeficiente variam entre 0 e 1, valores menores que 0,6 indicam falta de confiabilidade de consistência interna, aqueles entre 0,6 e 0,7 são tidos como aceitáveis para pesquisas exploratórias, entre 0,7 e 0,9 são considerados como satisfatórios, já aqueles acima de 0,90 não são desejáveis, pois indicam que todos os indicadores estão medindo o mesmo fenômeno e provavelmente não é uma medida válida para o constructo [66].

- ρ de Dillon-Goldstein (*Composite reliability*): essa técnica tem foco na variância da soma das variáveis que compõem um bloco. Quando seu valor é superior a 0.7 o bloco é considerado homogêneo. Seu cálculo se dá por meio da fórmula:

$$\rho = \frac{(\sum_{p=1}^{P_q} \lambda_{pq})^2}{(\sum_{p=1}^{P_q} \lambda_{pq})^2 + ((\sum_{p=1}^{P_q} (1 - \lambda_{pq}^2))$$

Onde λ_{pq} é o *loading* do indicador p no bloco q correspondente [69].

- *Análise de componentes principais*: essa é uma técnica que analisa os valores da própria matriz de correlação dos indicadores. Um bloco é considerado unidimensional caso o primeiro valor da matriz de correlação for maior que 1 e os demais menores. Depois dessa verificação é preciso verificar se as outras variáveis do bloco estão positivamente correlacionadas com o primeiro fator, no caso da correlação ser negativa, significa que o indicador é inadequado e deve-se retirá-lo do bloco, este procedimento é conhecido como *bootstrap* [69].

Enquanto o α de Cronbach assume a equivalência paralela das variáveis manifestas, isto é, assume que cada variável manifesta é igualmente importante na definição da variável latente, a técnica do ρ de Goldstein se baseia nos resultados dos modelo ao invés de se basear nas correlações entre as variáveis manifestas.

Modelos Formativo

Os modelos de medição formativos, ao contrário do que acontece com modelos reflexivos, baseiam-se no pressuposto que os indicadores influenciam o constructo [66]. Neste tipo de modelo, os indicadores não são intercambiáveis, dessa maneira, cada indicador compõem um aspecto específico do constructo e em conjunto determinam o significado do constructo, o que implica que a omissão de um indicador altera potencialmente a natureza do mesmo [66].

Na modelagem por meio de diagrama de rede, para o modelo reflexivo, a direção de causalidade flui do constructo para seus indicadores e, no modelo formativo, flui dos indicadores para o constructo como exemplificado na Figura 2.23.

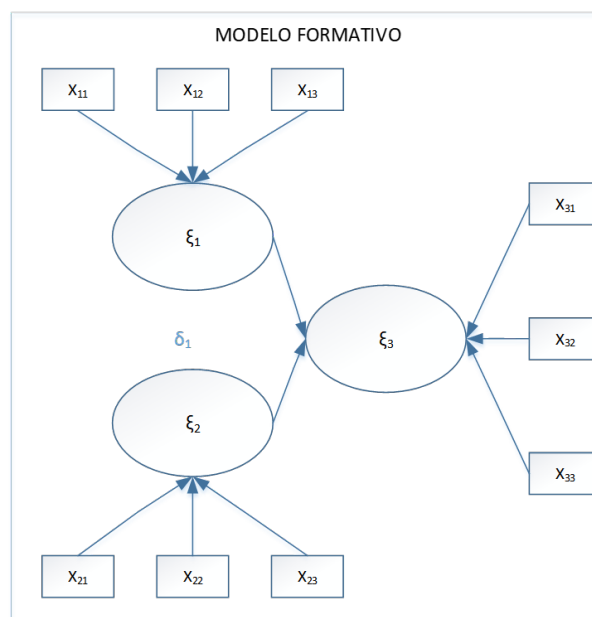


Figura 2.23: Modelos de medição formativo.

Acontrário do modelo reflexivo, neste modelo a variável latente é definida como uma combinação linear das variáveis manifestas correspondentes, assim, cada variável manifesta é uma variável exógena no modelo de mensuração. Além disso, e mudanças em um indicador não implicam mudanças nos outros e a consistência interna não se torna um problema como no modelo anterior. Dessa forma o modelo de medição pode ser expresso por [69]:

$$\xi_q = \sum_{p=1}^{Pq} \omega_{pq} x_{pq} + \delta_q$$

Onde ω_{pq} é o coeficiente que liga cada variável manifesta a sua variável latente correspondente e o erro δ_q representa a fração da variável latente correspondente não contabilizada pelo bloco de variáveis manifestas. A hipótese que deve ser assumida para esse modelo é a seguinte:

$$E(\xi_q | x_{pq}) = \sum_{p=1}^{Pq} \omega_{pq} x_{pq}$$

Modelo Misto

O modelo misto nada mais é que a junção dos modelos formativo e reflexivo em um único modelo, com o mesmo bloco de variáveis manifestas. Independente do tipo de modelo de medição o padrão de valores das variáveis latentes $\hat{\xi}_q$ associado a q -ésima variável ξ_q pode ser calculado como uma combinação linear do seu próprio bloco de variáveis manifestas por meio de uma relação de pesos definida por:

$$\hat{\xi}_q = \sum_{p=1}^{Pq} w_{pq} x_{pq}$$

Onde x_{pq} são variáveis padronizadas e w_{pq} são seus pesos. Esses pesos são produzidos pela convergência do algoritmo e depois transformados de modo a produzir pontuações variáveis latentes padronizadas. A relação de peso deste modelo apenas implica que qualquer variável latente é definida como uma soma ponderada de suas próprias variáveis manifestas, não afeta a direção do relacionamento entre a variável latente e suas próprias variáveis manifestas no modelo externo e por isso não deve ser confundida com o modelo formativo [69].

A abordagem por mínimos quadrados parciais tem sido amplamente utilizada pelas ciências sociais, pois pode lidar com todos os tipos de dados, padronizados ou não, com pressupostos mínimos sobre as características destes [75]. Isto faz com que se torne a técnica ideal para analisar quais são os fatores e critérios que mais influenciam para o insucesso do projeto do ponto de vista dos entrevistados. Diante desse levantamento pretende-se desenvolver métricas e indicadores de desempenho com vistas a minimizar as taxas de insucesso em projetos de TI.

2.5 Métricas e indicadores de desempenho

Os termos medição, medida, métrica e indicador, são frequentemente mal-interpretados e utilizados sem a devida diferenciação, dessa forma se faz necessária sua distinção, a qual é apresentada a seguir [76]:

- *Medição de desempenho*: é a capacidade de medir e interpretar o desempenho de processos, relaciona-se diretamente à capacidade de entendimento sobre o desempenho de processos de negócio e compreende as dimensões de tempo, custo, capacidade e qualidade;
- *Medida*: Medida é a quantificação de dados em um padrão e qualidade aceitáveis (exatidão, completude, consistência, temporalidade). Diz respeito a um dado, como por exemplo uma unidade de medida (kg, cm, km, etc);
- *Métrica*: Métrica é uma extrapolação de medidas. É uma informação, isto é, uma conclusão com base em dados finitos;
- *Indicador*: é uma representação de forma simples ou intuitiva de uma métrica ou medida para facilitar sua interpretação quando comparada a uma referência ou alvo. Representam informações e para serem válidos devem refletir uma métrica, como por exemplo número de reclamações/número de vendas

Infere-se que a existência de métricas bem elaboradas e apropriadas a prover suporte à criação de indicadores de problemas organizacionais são requisitos de uma boa gestão de desempenho [76].

Uma estrutura equilibrada de indicadores deve contemplar indicadores direcionadores e indicadores de resultados. Enquanto o primeiro tipo monitora a causa antes do efeito, possibilitando alterar o curso para alcançar o resultado, o segundo se preocupa em monitorar o efeito e, dessa forma, não permitem alterar determinado resultado [76].

As medições são ferramentas primárias para o gerenciamento de atividades do ciclo de vida de um projeto, avaliando a viabilidade dos planos e monitorando a aderência das atividades do projeto a estes e são também fundamentais na avaliação da qualidade dos produtos e da capacidade dos processos organizacionais [77].

Para um programa de medição suportar efetivamente os objetivos estratégicos da organização este deve ser escalável, automatizado, padronizado e flexível [78]. À medida que ocorrem mudanças no mercado de software, as organizações de desenvolvimento de software e suas necessidades também mudam, dessa forma um bom programa de medição deve possuir mecanismos que garantem sua robustez apesar das mudanças [78].

Essas medições podem ser feitas com base em indicadores de desempenho do projeto ou *Key Performance Indicator (KPI)*, em inglês. Indicadores de desempenho são medidas que fornecem uma estimativa ou avaliação de atributos específicos derivados de um modelo em relação a necessidades de informação pré definidas [77]. No entanto, KPIs não são indicadores quaisquer, em contraste com indicadores genéricos, KPIs tem ligação com as operações do negócio [79], ou seja, devem estar alinhados aos objetivos estratégicos da organização.

Existem 07 critérios principais para a geração de indicadores [80]:

1. *Seletividade ou importância*: esta característica refere-se a capacidade do indicador de perceber características chaves do processo, serviço ou produto a que se referem;
2. *Simplicidade e clareza*: esta característica diz respeito a facilidade de compreensão do indicador nos diversos níveis da organização;
3. *Rastreabilidade e acessibilidade*: esta característica permite ao indicador que o mesmo seja atualizado e seus dados estejam sempre disponíveis (resultados, memórias de cálculo e responsáveis);
4. *Abrangência*: o indicador deve ser representativo, inclusive em termos estatísticos;
5. *Comparabilidade*: o indicador deve ser facilmente comparável com outros referenciais apropriados;
6. *Estabilidade*: o indicador deve ser duradouro, gerado baseando-se em procedimentos padronizados e que permitam previsibilidade de cenários futuros;
7. *Baixo custo*: o indicador deve ser gerado a baixo custo, ou seja, o custo para a geração de informações deve ser o mínimo possível. Também devem ser utilizadas unidades adimensionais ou dimensionais simples, tais como percentuais e unidades de tempo.

Muitas organizações de TI medem coisas que têm pouco ou até mesmo nenhum valor [81]. Medições efetivas devem se concentrar em indicadores vitais e significativos que devem ser econômicos, quantitativos e utilizáveis, caso a organização tenha muitas medidas pode-se perder o foco na melhoria dos resultados [81].

Visando evitar que se tenham muitas medições no processo de gestão de riscos, este estudo tem foco nos fatores que mais influenciam o insucesso de projetos de acordo com uma pesquisa desenvolvida pelo autor. Dessa forma pretende-se evitar uma gama de indicadores desnecessários, focando-se apenas naqueles que têm maior impacto no projeto. Para que seja possível alcançar este objetivo o Capítulo 3 evidencia a metodologia desenvolvida nesta pesquisa.

Capítulo 3

Metodologia de Pesquisa

É possível definir pesquisa como o processo formal e sistemático de desenvolvimento do método científico, cujo objetivo fundamental é descobrir respostas para problemas mediante o emprego de procedimentos científicos [82]. A proposta desta pesquisa é apresentar um novo desenho de gerenciamento de riscos a fim de diminuir a taxa de insucesso em projetos ágeis, com propósito de contribuir com o aumento do índice de sucesso em projetos de desenvolvimento de software. Os tópicos que se seguem apresentam a classificação da pesquisa e sua estrutura.

3.1 Classificação da pesquisa

Pesquisa é um procedimento formal, com método de pensamento reflexivo, que requer um tratamento científico e se constitui no caminho para se conhecer a realidade ou para descobrir verdades parciais [83].

A ciência pode ser dividida em dois campos: as ciências naturais, caracterizada como um conjunto de conhecimento sobre uma classe de objetos ou fenômenos do mundo; e ciências do artificial, classificada como conhecimento acerca de objetos e fenômenos artificiais, ou seja, aqueles criados pelo ser humano [84]. Ainda de acordo com o autor enquanto ciências naturais se preocupam em como as coisas são, as ciências do artificial estão mais preocupadas na produção de artefatos com propriedades específicas e que realizam objetivos.

Do ponto de vista paradigmático o campo da ciência pode ser dividido em três categorias distintas: as ciências formais, tais como filosofia e matemática; as ciências explanatórias, como ciências naturais e sociais; e as denominadas *Design Sciences*, que tem por objetivo desenvolver artefatos para solução de problemas [11]. Esta pesquisa se enquadra dentro do campo das *Design Sciences*, pois tem como objeto principal o desenvolvimento

a produção de conhecimento para a elaboração de um método que venha a solucionar o problema da alta taxa de insucesso em projetos regidos por metodologias ágeis.

Este estudo foi dividido em duas fases e cabe classificá-lo de acordo com cada momento. Para que fosse possível elaborar a metodologia proposta foi utilizada a técnica de levantamento de dados (*survey*), dessa forma foi desenvolvido um questionário, tendo como objeto de pesquisa fatores e critérios de insucesso em projetos. Essa técnica foi escolhida visando assegurar o anonimato respondentes e também não ter desvios e nem diferenciação nas perguntas, foi aplicado de forma *online*, por meio da ferramenta de formulários *Google Forms*, tendo como público alvo gerentes de projeto, líderes de projeto e integrantes de times de projeto com ampla divulgação em nível nacional. Nesta etapa a pesquisa pode ser tipificada como qualitativa, pois tinha como foco investigar com maior profundidade o problema estudado [82].

Para obter o número necessário da amostra de respondentes foi utilizada uma técnica chamada *Statistical Power Analysis*. Esse tipo de estimativa é utilizado por pesquisadores para estimar quantos entrevistados são necessários para responder a pergunta de uma pesquisa [85]. Este tipo de análise estatística utiliza como base o Teste de Hipóteses e depende da significância (α), do tamanho da amostra (N) e do tamanho do efeito (*Effect size* (ES)) da população [86].

Os Testes de Hipóteses são utilizados quando se deseja testar a veracidade de alguma afirmação feita sobre determinada população, utilizando-se de resultados experimentais provenientes de determinada amostra, pois muitas vezes afirmações são derivadas de teorias e a adequação ou não dessa teoria num universo real pode ser verificada ou refutada pela amostra. Então, o objetivo desse teste é fornecer amparo metodológico que permite ao pesquisador apoiar ou refutar a hipótese formulada [87].

A importância da *Power analysis* vem do fato de que diversos pesquisadores do campo das ciências comportamentais e sociais, formulam e testam hipóteses que se espera rejeitar como meios de estabelecer fatos ou fenômenos sobre o estudo [86]. Em um Teste de Hipótese a hipótese H_0 típica, é que uma correlação produto-momento populacional, r , é zero para ser testada na região crítica de $\alpha = .05$. Quando H_0 é testado em uma amostra de tamanho N , r realmente tende a zero, fazendo com que os pesquisadores rejeitem erroneamente H_0 , quando este é verdadeiro, este é chamado de erro Tipo I, cuja probabilidade é controlado por α , nesse caso é de 5% [86]. Outra possibilidade é de não rejeitar uma hipótese falsa, o que chamado de erro Tipo II, cuja probabilidade é chamada de β , dessa forma a escolha de uma hipótese correta se daria por $1-\beta$ [86].

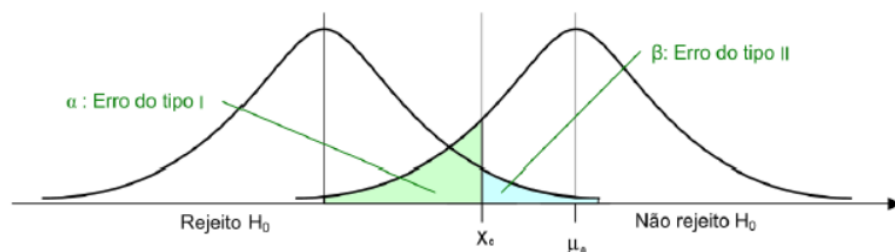


Figura 3.1: Exemplo de gráfico com regiões de rejeição em Teste de Hipóteses (Fonte: [88]).

A Figura 3.1 mostra um gráfico com as regiões dos dois tipos de erros citados. Verifica-se que para a região de α , a esquerda de X_C , o erro ocorre pela rejeição da hipótese verdadeira H_0 , erro Tipo I, e na região de β , a direita de X_C , o erro ocorre pela não rejeição de uma hipótese H_0 falsa, erro Tipo II.

O resultado desse teste depende da magnitude do ES da população e também do tamanho da amostra (N), quanto maior o tamanho da amostra ou o ES, maior é a probabilidade de se rejeitar uma hipótese falsa [86]. Dessa forma, este tipo de análise estatística explora a relação matemática entre quatro variáveis: α , N , *Power* e o *Effect size*. Essa relação é tal que, quando três dessas variáveis estão fixas é possível descobrir a quarta [86].

Para apoio aos cálculos, foi utilizado o software *GPower3.1*. Este software efetua diversos cálculos estatísticos, dentre eles a *Power Analysis*. Para que fosse possível estabelecer a quantidade mínima da amostra, utilizou-se um teste de regressão múltipla linear, com análise a priori. Nesse tipo de análise, o tamanho da amostra (N_e) calculado como função de [89]:

- O nível *Power* requerido ($1-\beta$);
- O nível de significância pré especificado α ; e
- O *Effect size* a ser determinado com a probabilidade ($1-\beta$).

Dessa forma, foi possível simular 02 resultados com diferentes níveis de confiança, um com 95% e outro com 90% de confiança, para estimar a quantidade mínima da amostra para cada um dos respectivos níveis de confiança.

Para uma análise que proporcione 95% de confiança, foram determinados os valores de *Effect size* = 0,15, α = 0,05 e $1-\beta$ = 0,95, dessa forma foi possível estabelecer que seria necessária uma amostra de pelo menos 129 respondentes, como destacado na Figura 3.2.

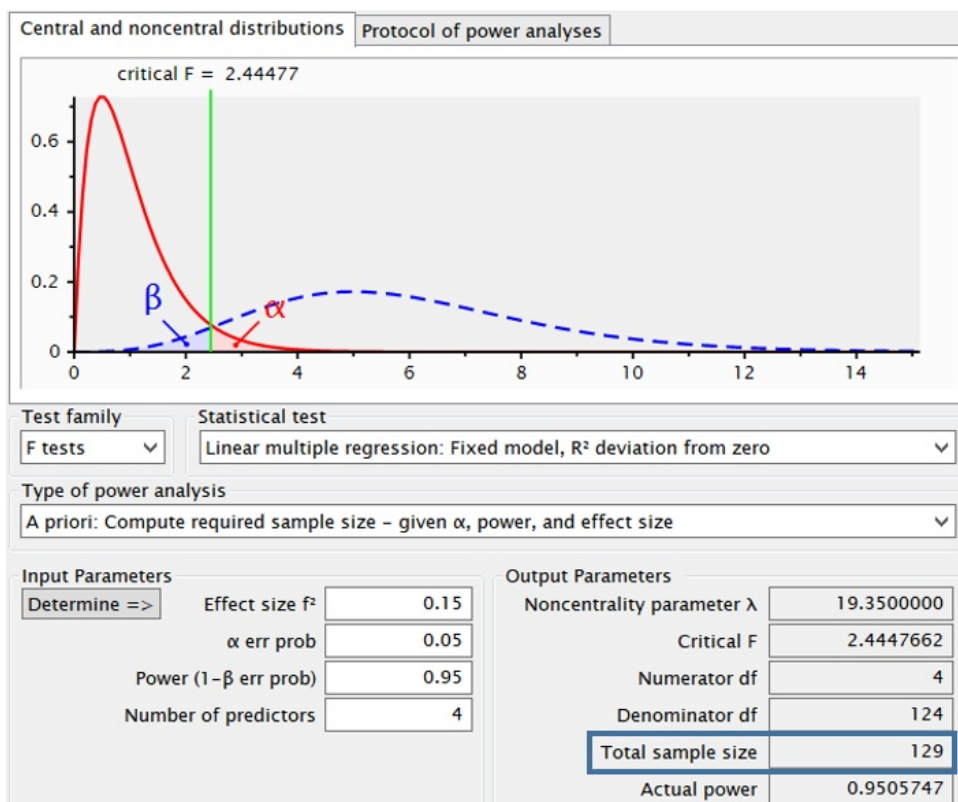


Figura 3.2: Cálculo da amostra para 95% de confiança.

Valendo-se da mesma ferramenta foi verificado que, pra uma análise que proporcione 90% de confiança, é necessária uma amostra de pelo menos 90 respondentes.

Após encerrado o prazo dado para o preenchimento do questionário, as respostas foram coletadas e a estas foi aplicada a modelagem por equações estruturais utilizando o método por mínimos quadrados parciais, dessa forma ao levantamento foi dado um viés estatístico tornando a pesquisa nesse momento do tipo quantitativa. Essa análise gerou insumos para que fosse elaborada a metodologia proposta nesse estudo.

Em uma segunda etapa foi desenvolvido um estudo exploratório, pois teve como principal objetivo desenvolver ideias com vistas a formulação de problemas mais precisos, tendo em vista a pouca exploração da gestão de riscos em métodos ágeis [82]. Essa etapa tem como objeto o desenvolvimento e aplicação de uma metodologia de gestão de riscos voltada para projetos que são geridos por métodos ágeis e teve sua aplicação em um projeto da Universidade de Brasília com o Exército Brasileiro. A partir dos Capítulos seguintes serão exploradas a pesquisa, a metodologia desenvolvida e sua aplicação no referido projeto.

Capítulo 4

Pesquisa para identificar os fatores críticos de sucesso de um projeto

O objetivo deste capítulo é apresentar os resultados obtidos a partir da pesquisa desenvolvida, aplicação do questionário e análise dos resultados. Esses dados servirão de insumos para a elaboração práticas e medições que, juntamente com o modelo proposto no Capítulo 6, tem por objetivo ampliar as chances de sucesso de um projeto.

Inicialmente foi elaborada uma lista com intuito de analisar quais fatores e critérios influenciam o insucesso de um projeto (Apêndice A). Os fatores visavam evidenciar situações no decorrer do projeto que poderiam levá-lo ao fracasso, já os critérios são itens que deveriam verificar situações após a finalização do projeto que objetivavam e obter dos pesquisados qual sua percepção de insucesso no que tange ao que foi entregue. Visando obter um resultado mais assertivo o público alvo dessa pesquisa foi formado por, gerentes de projetos, líderes de projetos e integrantes de times de projeto. Fatores e problemas em projetos ágeis de TI podem ser classificados em quatro categorias: *Organizacionais*, *Pessoas*, *Processos* e *Técnicos* [90]. Dessa forma, estas categorias foram consideradas como constructos, tanto para os fatores como para os critérios de insucesso.

Para os fatores e critérios de insucesso foram elaborados 113 itens. Esses itens foram definidos após um levantamento da literatura acerca do tema em questão e foram distribuídas da seguinte maneira:

- Para a categoria *Organizacional* foram elaborados 16 itens sobre fatores e 5 sobre critérios de insucesso;
- Para a categoria *Pessoas* foram elaborados 25 itens sobre fatores e 8 sobre critérios de insucesso;
- Para a categoria *Processos* foram elaborados 27 itens sobre fatores e 10 sobre critérios de insucesso; e

- Para a categoria *Técnico* foram elaborados 16 itens sobre fatores e 6 sobre critérios de insucesso.

Para que seja possível medir a influência destes constructos é necessário também listar itens de desempenho. Estes itens visam analisar, sob a percepção dos entrevistados, quais são os fatores que influenciam o sucesso de um projeto. A compreensão dos critérios de sucesso do projeto evoluiu a partir do conceito simplista de restrição tripla (custo, tempo e escopo), conhecido como o triângulo de ferro, para algo que engloba critérios de sucesso adicionais, como qualidade, satisfação das partes interessadas e gerenciamento de conhecimento [91]. Dessa maneira, foram elencados 08 itens que englobam critérios como custo, escopo, tempo, qualidade, satisfação e gerenciamento.

Para coletar as opiniões dos decisores foi desenvolvido um conjunto de 05 possíveis respostas, baseadas na escala de Likert, que iam desde *Não tem impacto algum* (refletindo o menor valor da escala, ou seja, 1) a *Está diretamente relacionado (é indissociável)* (refletindo o maior valor da escala, ou seja, 5).

Foram obtidas 82 respostas como resultado da aplicação do questionário de fatores de insucesso em projetos. Com essa quantidade de respondentes e com o apoio do software *Gpower*, foi possível verificar que, com a quantidade de respostas, obtem-se uma confiabilidade de aproximadamente 88% como evidenciado na Figura 4.1. Com este número de respostas foi utilizado o software *SmartPLS* para que fosse possível gerar o modelo e validar as questões de desempenho.

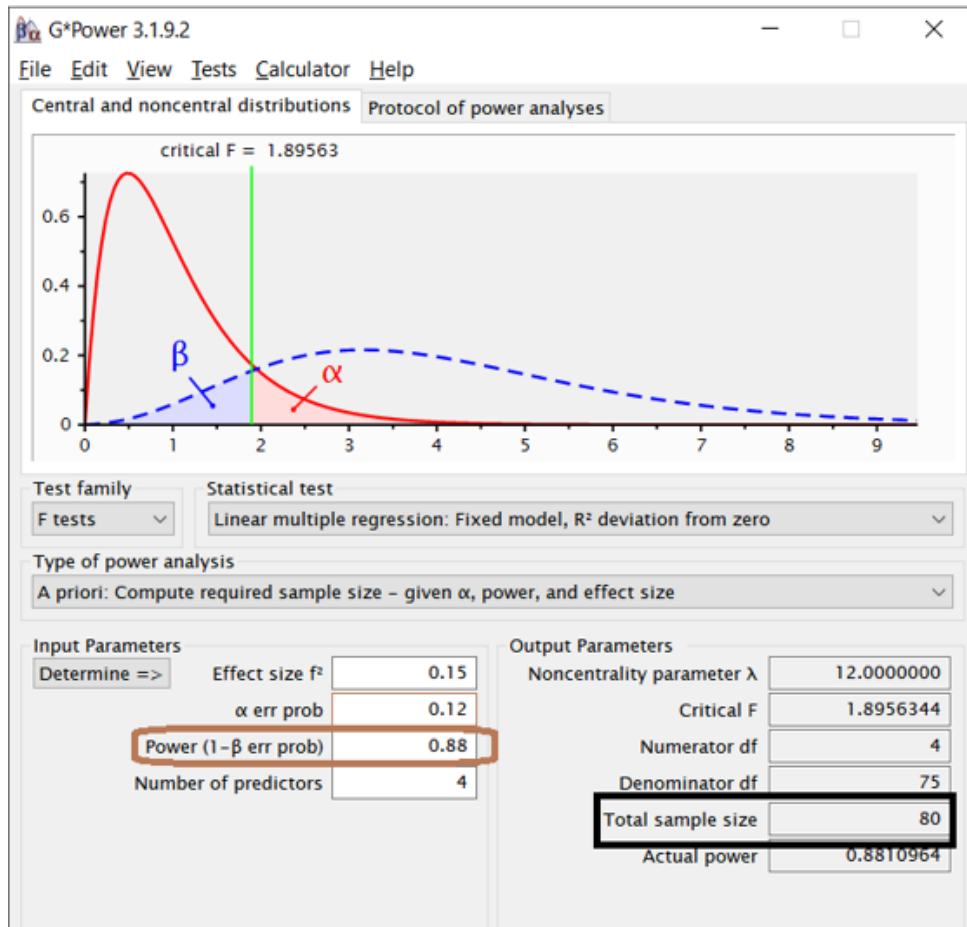


Figura 4.1: Cálculo da amostra para a quantidade aproximada de respondentes.

Para analisar as respostas recebidas pelos especialistas foi gerado um modelo e utilizada a configuração básica do software, ou seja, cálculo por meio do algoritmo PLS, com ponderação pelo caminho, com número máximo de 300 iterações e critério para paragem em 10^{-7} , como verificado por meio da Figura 4.2.

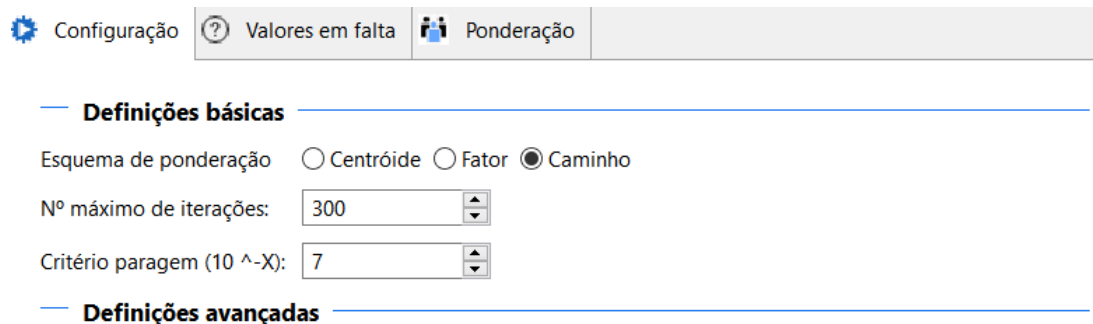


Figura 4.2: Configuração do Algoritmo PLS.

Depois de realizados os cálculos com o algoritmo PLS verificou-se que o modelo necessitava de ajustes para que fosse possível assegurar sua estabilidade. Dessa forma foram retiradas algumas questões que estavam influenciando negativamente o atingimento destes valores. Ao final dos ajustes necessários o modelo apresentou uma redução significativa de questões, que ficaram distribuídas da seguinte maneira:

- Na categoria *Organização-Fatores* ocorreu uma redução de 16 para 07 itens;
- Na categoria *Pessoas-Fatores* ocorreu uma redução de 25 para 06 itens;
- Na categoria *Processos-Fatores* ocorreu uma redução de 27 para 06 itens;
- Na categoria *Técnico-Fatores* ocorreu uma redução de 16 para 05 itens;
- Na categoria *Organização-Critérios* ocorreu uma redução de 05 para 04 itens;
- Na categoria *Pessoas-Critérios* ocorreu uma redução de 08 para 05 itens;
- Na categoria *Processos-Critérios* ocorreu uma redução de 10 para 09 itens; e
- Na categoria *Técnico-Critérios* ocorreu uma redução de 06 para 04 itens.

Depois de retirados os itens que influenciavam negativamente nos constructos o foi gerado o modelo apresentado na Figura 4.3.

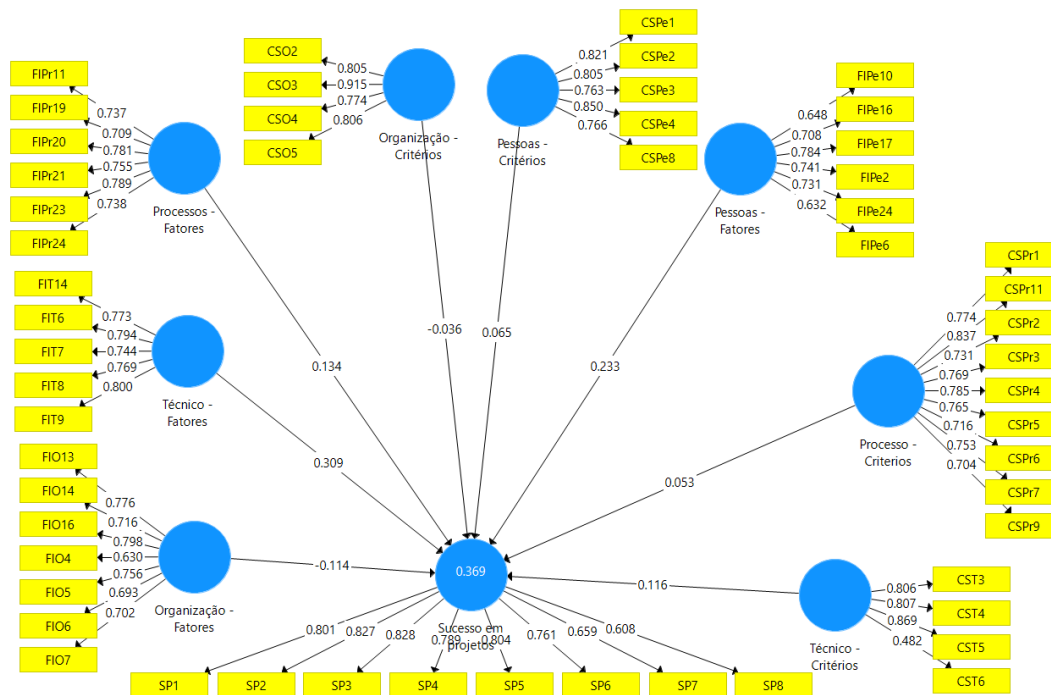


Figura 4.3: Modelo desenvolvido para validação do questionário de fatores e critérios de insucesso em projetos.

Após o ajuste do modelo foi verificada a confiabilidade dos itens, medida pelos pesos dos mesmos com relação aos constructos. Itens com cargas extremamente baixas devem ser cautelosamente revistos, pois podem adicionar muito pouco valor explicativo ao modelo enquanto atenuam e, portanto, enviesam as estimativas dos parâmetros que conectam constructos. De forma geral, itens com cargas inferiores a 0,4 ou 0,5 devem ser descartados [92]. Nesse sentido, verifica-se no modelo que apenas 06 itens apresentam valores de carga inferiores a 0,7, ainda assim, estes estão ainda acima de 0,6 atendendo, dessa maneira, ao requisito de confiabilidade de itens.

Para testar a significância estatística dos resultados, ao modelo foi aplicado um procedimento não paramétrico chamado de *bootstrapping*. O *bootstrapping* é uma abordagem utilizada para validar um modelo multivariado por meio da extração de grande número de sub-amostras e extraíndo-se um de sub-amostras e estimando modelos para cada uma delas [93]. Para o teste de *bootstrapping* foi utilizado o tipo de teste bicaudal com 5000 amostras como pode ser verificado pela Figura 4.4.

Bootstrapping

Bootstrapping is a nonparametric procedure that allows testing the statistical significance of various PLS-SE and R^2 values.

Configuração Mínimos quadrados parciais Valores em falta Ponderação

Definições básicas

Subamostras 5000

Usar processamento paralelo

Mudança de sinal Sem alteração de sinal Alterações ao nível do constructo Alterações individuais

Nível de resultados Bootstrapping básico Bootstrapping concluído

Definições avançadas

Método do intervalo de confiança Bootstrap com base em percentis Bootstrap com distribuição t-student Bootstrap com enviesamento corrigido e acelerado (BCa) Davision Hinkley's Duplo Bootstrap Bootstrap duplo de Shi

Tipo de teste Uma cauda Bi caudal

Nível de significância 0,05

Figura 4.4: Configuração do software para rodar o teste de *bootstrapping*.

Para analisar a confiabilidade e validade dos modelos de medida se faz necessário que a escala deva possuir indicadores com um nível mínimo de confiabilidade, garantindo que a amostra não possua vieses e que as respostas sejam confiáveis. Dessa forma, após a aplicação dos ajustes necessários, foram verificados os valores das variáveis para garantir a estabilidade do modelo final. Assim, foram analisados os resultados atingidos para confiabilidade composta, AVE, Rácio Heterotrait-Monotrait (HTMT) e estatísticas de colinearidade (VIF). Os resultados serão expostos e analisados em seguida.

Na etapa de validação deve ser verificada a validade composta. Dijkstra e Henseler [94] verificaram em seus estudos deficiências apresentadas nos coeficientes de confiabilidade até então apresentados e sugeriram a utilização do ρ_A , essa variável verifica os pesos dos constructos, é determinado de tal forma que os elementos fora da diagonal da matriz de correlação do indicador de uma variável latente são reproduzidos, tanto quanto possível, em um sentido de mínimos quadrados. Para esse critério os valores devem ser superiores a 0.7. A Figura 4.5 indica o atingimento dos valores para a variável em questão.

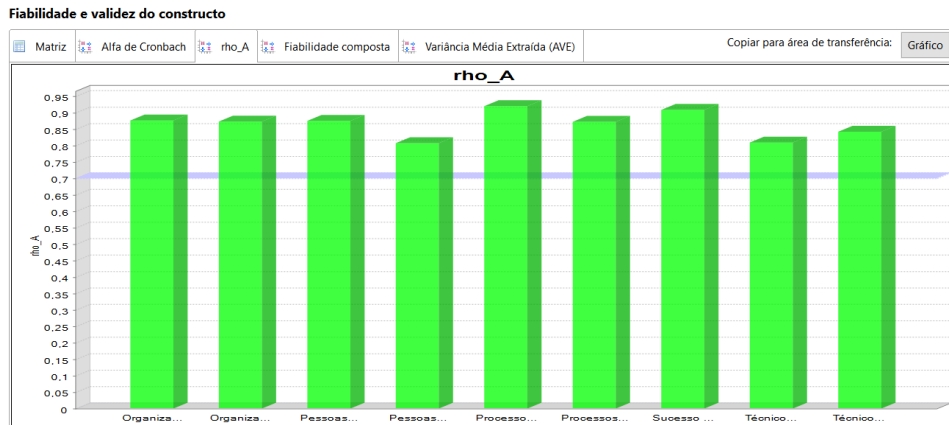


Figura 4.5: Valores atingidos para o critério de rho_A.

Um dos fatores mais importantes a serem verificados é a validade convergente, esse fator é verificado quanto os valores de AVE são maiores do que 0.5. Estes valores acima de 0.5 há indicação de que pelo menos 50% da variância de um constructo foi incorporada ao modelo. A Figura 4.6 demonstra que para todos os constructos analisados esses valores foram atingidos.

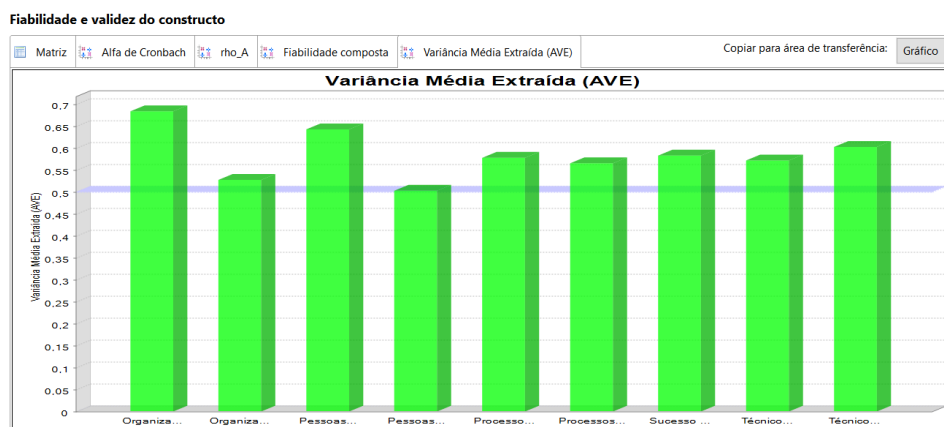


Figura 4.6: Valores de AVE atingidos.

Outra variável a ser analisada é a validade discriminante, essa validade é verificada pela variável HTMT ou Racio Heterotrait-Monotrait. À medida que as correlações aumentam, a diferenciação dos construtos diminui, tornando menos provável que as abordagens indiquem a validade discriminante. Para essa variável são definidos que os valores abaixo de 0.9 asseguram o atendimento a essa validação. As Figura 4.7 demonstra que todos os critérios estão abaixo de 0.9, assegurando o atendimento também a este critério.

Validade discriminante

	Organizaçã...	Organizaçã...	Pessoas - Cr...	Pessoas - Fa...	Processo - ...	Processos - ...	Sucesso em...	Técnico - Cr...	Técnico - Fa...
Organizaçã...									
Organizaçã...	0.352								
Pessoas - Cr...	0.701	0.421							
Pessoas - Fa...	0.279	0.559	0.450						
Processo - C...	0.596	0.382	0.838	0.526					
Processos - ...	0.269	0.715	0.383	0.639	0.435				
Sucesso em ...	0.291	0.363	0.369	0.555	0.402	0.488			
Técnico - Cr...	0.727	0.537	0.582	0.357	0.747	0.490	0.382		
Técnico - Fa...	0.402	0.679	0.441	0.644	0.440	0.772	0.581	0.472	

Figura 4.7: Tabela de de HTMT atingidos pelas variáveis analisadas.

A Figura 4.8 demonstra graficamente o atingimento dos valores necessários para HTMT.

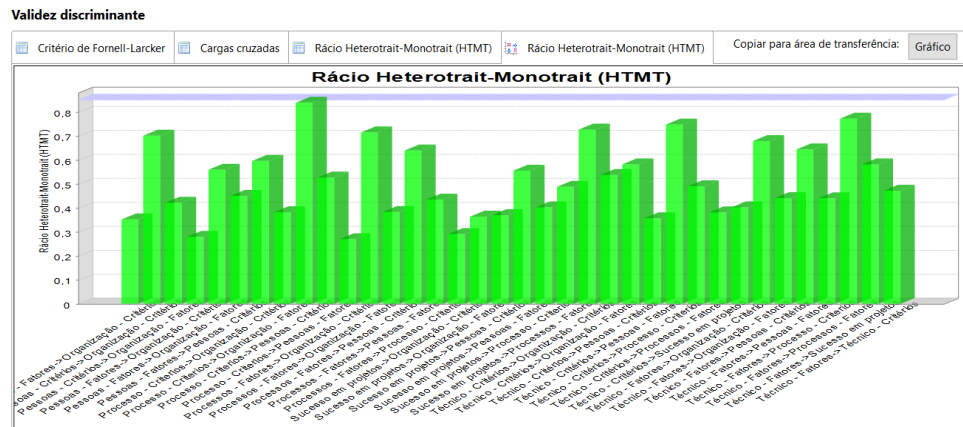


Figura 4.8: Gráfico de valores de HTMT atingidos pelas variáveis analisadas.

Existe também uma outra maneira de ser avaliada a validade discriminante, dessa vez analisando a tabela de cargas cruzadas e analisando se os indicadores possuem cargas fatoriais mais altas em suas variáveis latentes respectivas do que em outras [95]. Essa verificação não será demonstrada, pois a tabela ficou muito grande e, por ter sido utilizada uma versão gratuita do software *SmartPLS*, não foi possível a exportação da mesma de forma que, pelas imagens não seria possível verificar neste documento essa forma de avaliação. Logo, optou-se por apresentar apenas a primeira.

A última verificação a ser feita é a de VIF (*Variance Inflation Factor*). Essa variável indica o efeito que as outras variáveis independentes têm sobre o erro padrão de um coeficiente de regressão. Altos valores de VIF indicam um alto grau de colinearidade ou multicolinearidade, o que pode indicar uma variância "compartilhada" entre as variáveis, diminuindo assim a capacidade de averiguar os papéis relativos de cada variável indepen-

dente [75]. O modelo modificado apresentou valores aceitáveis para o VIF, tanto internos como externos.

Com o modelo validado e ajustado é possível fazer uma análise dos resultados obtidos. Pode ser observado que por meio da Figura 4.3 o constructo que mais influencia na obtenção de sucesso do projeto são os Técnicos-Fatores, seguido por Pessoas-Fatores, já os que menos influenciam são Organização-Critérios seguido Processos-Critérios.

Em uma análise mais aprofundada, nota-se que os fatores que têm mais influência dentro de cada constructo são:

- Para o constructo Técnico-Fatores:
 - FIT9 - Não existe uma estrutura da organização do projeto; e
 - FIT8 - Plano de projeto não é atualizado regularmente;
- Para o constructo Pessoas-Fatores:
 - FIPe17 - Gerente de projeto incompetente; e
 - FIPe24 - Falta de apoio da alta administração;
- Para o constructo Processos-Fatores:
 - FIPr23 - Requisitos mal definidos; e
 - FIPr20 - A data final da conclusão do projeto não está claramente definida;
- Para o constructo Organização-Fatores:
 - FIO16 - Falta de logística; e
 - FIO13 - Falta de compensação para funcionários;
- Para o constructo Organização-Critérios:
 - CSO3 - Não afetar a melhoria da capacidade organizacional; e
 - CSO5 - Os impactos do projeto nos beneficiários não são visíveis;
- Para o constructo Pessoas-Critérios:
 - CSPe4 - Não propiciar satisfação do usuário; e
 - CSPe2 - Não propiciar satisfação da equipe;
- Para o constructo Processo-Critérios:
 - CSCPr11 - Não atinge os critérios de qualidade; e
 - CSPr5 - Não atingir os requisitos do cliente;

- Para o constructo Técnico-Critérios:

CST5 - Não está em conformidade com as normas ambientais; e

CST3 - O projeto não ter boa aceitação.

Além dos constructos que influenciam o sucesso do projeto, é possível também definir quais são os itens considerados critérios de sucesso em projeto. Neste quesito, pode-se destacar os seguintes:

- Constructo critério de sucesso em projetos:

SP3 - Escopo (o produto se comporta da forma como o cliente espera); e

SP2 - Escopo (alcançar todos os requisitos e objetivos propostos).

Dessa forma, verifica-se que não possuir uma estrutura de organização do projeto e a não atualização dos planos de projeto são medidas primárias de sucesso do projeto, no entanto deve-se destacar também que o terceiro critério considerado como primordial para definição de sucesso no projeto é o SP5 que versa acerca da estrutura de custos do projeto. A falta de organização do projeto traz consigo a consequência na entrega do escopo do projeto para o cliente e também consequências financeiras, pois projetos mal estruturados podem render retrabalho ou gastos desnecessários.

No Capítulo seguinte será apresentada a metodologia proposta para Gestão de Riscos em projetos geridos por meio de métodos ágeis.

Capítulo 5

Metodologia Proposta

Esta Seção apresenta o quadro modelo elaborado como proposição de um novo desenho para a gestão de riscos em metodologias ágeis. Baseando-se no prisma apresentado na Figura 5.1 foi possível a elaboração de um quadro visual para a gestão de riscos mapeados durante a execução do projeto (Figura 5.2). A metodologia foi pensada tomando por base o *framework* Scrum, tendo em vista que esse é o método ágil mais utilizado no mundo, atualmente presente em 56% dos projetos [96].

A Figura 5.1 evidencia o prisma de gestão de riscos inicialmente proposto:



Figura 5.1: Prisma de gestão de riscos proposto.

Na Figura 5.1, a parte externa (cinza escuro) faz alusão ao projeto como um todo, a parte interna (verde) refere-se a iteração e a parte central (cinza claro) são os quatro pilares da metodologia. O que se propõe é o planejamento de riscos de forma iterativa, onde os riscos devem ser analisados a cada Sprint e revisados em cada reunião diária do Scrum.

Além do quadro para gestão visual de riscos foram também incluídas duas cerimônias ao processo de gestão ágil de projeto, desenvolvida uma matriz de Probabilidade x Impacto x Detecção e um código de cores para avaliação dos riscos de acordo com sua gravidade com base na matriz previamente descrita. A seção 5.1 apresenta o quadro kanban fruto da estruturação inicial evidenciada na Figura 5.1 e também como se dá a metodologia em si.

5.1 Quadro para gestão visual de riscos

O quadro é formado por dois quadrados e um losango, o quadrado mais externo e o losango visam identificação de ameaças a nível de projeto e a nível de iteração, e o quadrado interno tem como objetivo a identificação de oportunidades. Cada quadrado é dividido internamente em algumas seções com vistas a atender as etapas de identificação, análise e elaboração de planos de respostas a riscos, definidas pelo gerenciamento de riscos.

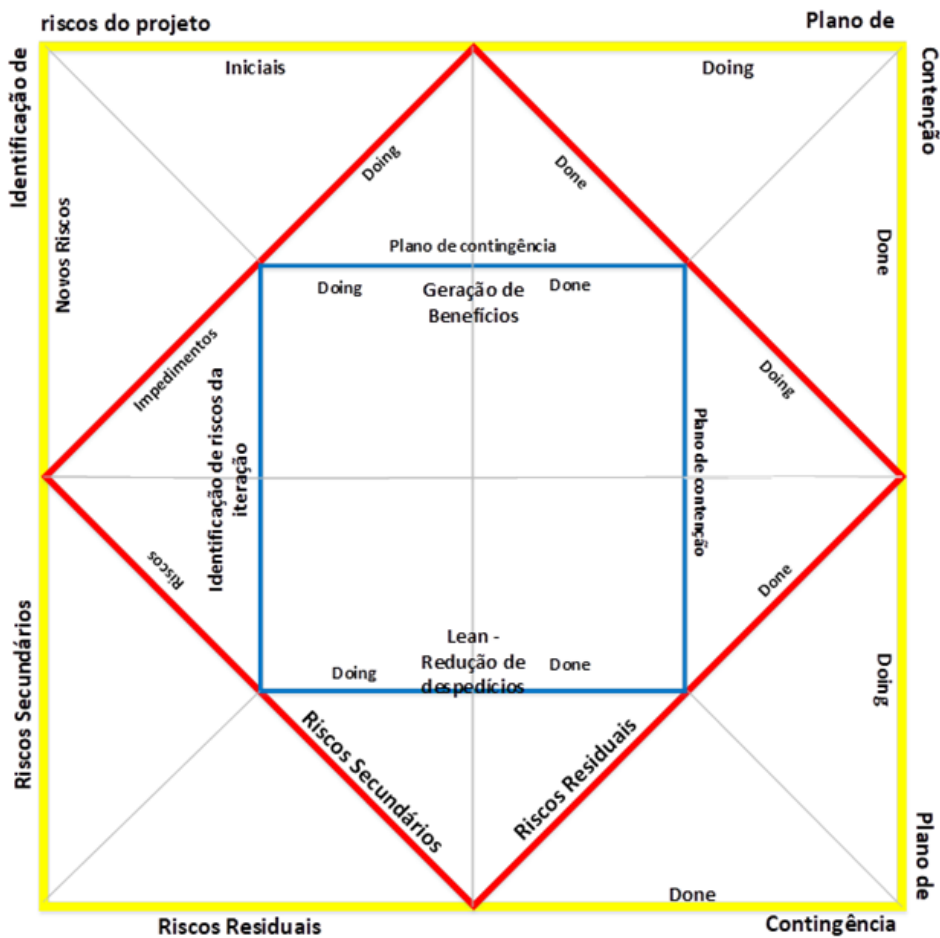


Figura 5.2: Proposta de quadro gestão visual de riscos em projetos ágeis.

O quadro deverá ser alimentado por *post its* que identificarão, de acordo com sua cor, a severidade do risco. As cores deverão seguir o padrão definido para avaliação de riscos descrito na seção 5.2. Para manter um melhor controle sobre os itens colocados no quadro, os cartões deverão ser identificados de acordo com o exemplificado na Figura 5.3, onde são explicitados o ID do risco, a data em que o cartão foi colocado no quadro e também as iniciais do responsável pelo risco ou pela ação de contingência ou contenção.

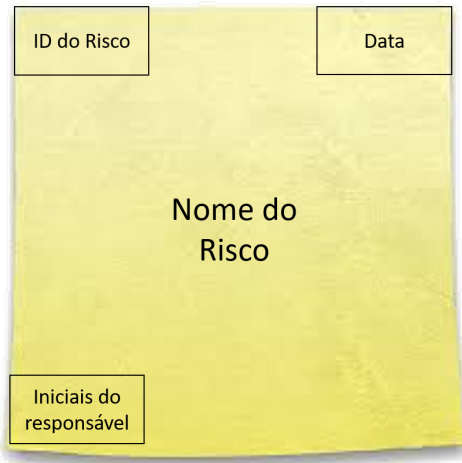


Figura 5.3: Configuração de post its.

Um dos valores definidos pela metodologia Kanban é enxergar o fluxo, no entanto para o quadro proposto, diferentemente da utilização mais comum de fluxo, que usualmente é da esquerda para a direita, o fluxo segue um sentido circular no sentido horário para cada quadrado de levantamento de ameaças (quadrados amarelo e vermelho) e para as oportunidades o sentido da esquerda para a direita. A maneira como o esquema proposto funciona será melhor descrito nas seções seguintes.

5.1.1 Riscos de Projeto

O quadrado mais externo, em amarelo, é o quadro onde serão evidenciadas as ameaças a nível de projeto. Esse quadrado possui 08 divisões internas, descritas a seguir:

- Identificação de riscos do projeto: Nessa parte do quadro deverão ser colocados os riscos identificados durante a cerimônia de levantamento de riscos do projeto e é subdividida em duas partes:
 1. Riscos Iniciais: são aqueles identificados durante a fase inicial do projeto;
 2. Novos Riscos: São aqueles identificados no decorrer da execução do projeto, mas que não tem relação com quaisquer outros riscos ou planos de contenção ou contingência.
- Plano de Contenção: Nesta parte do quadro deverão ser elencados as respostas aos riscos identificados cujo objetivo é minimizar os impactos dos riscos ou evitar que os mesmos ocorram. Assim como na identificação, esta parte está subdividida em duas menores:

1. *Doing*: Itens elencados nessa parte do quadro devem estar sendo executados ou por executar, essa parte funciona como um *backlog* de ações de contenção de riscos;
 2. *Done*: Nessa parte devem ser colocados os itens do plano de contenção que já foram executados.
- Plano de Contingência: Nessa parte do quadro deverão ser colocadas as repostas aos riscos que não puderem ser evitadas e dessa forma, caso ocorram, necessitam de uma solução de contorno. Assim como na identificação, esta parte está subdividida em duas menores:
 1. *Doing*: Itens elencados nessa parte do quadro devem estar sendo executados ou por executar, essa parte funciona como um *backlog* de ações de contingência de riscos;
 2. *Done*: Nessa parte devem ser colocados os itens do plano de contingência que já foram executados.
 - Riscos Residuais: Nessa parte do quadro devem ser elencados aqueles itens que tiveram seu plano de contenção executado, no entanto, ainda assim, permanecem como ameaças ao projeto, mesmo que com impacto minimizado pela aplicação do plano; e
 - Riscos Secundários: Essa é a última parte do quadro e deve ser utilizada caso a execução de algum plano, seja de contenção ou contingência, gere um risco não previsto inicialmente.

5.1.2 Riscos da Iteração

O losango, em vermelho, é o quadro onde serão evidenciadas as ameaças a nível de iteração. Esse losango possui 08 divisões internas, descritas a seguir:

- Identificação de riscos da iteração: Nessa parte do quadro deverão ser colocados os riscos identificados durante a cerimônia de levantamento de riscos da iteração e é subdividida em duas partes:
 1. Riscos: são aqueles itens que se acontecerem poderão atrapalhar o trabalho do time. Para estes deve ser elaborado um plano de contenção com vistas a minimizar seu impacto ou evitar que o mesmo ocorra;
 2. Impedimentos: São aqueles itens que impedem o time de continuar o trabalho e devem ser tratados imediatamente. Para estes deve ser elaborado diretamente

o plano de contingência, pois nesse ponto entende-se que o impedimento já ocorreu.

- Plano de Contenção: Nesta parte do quadro deverão ser elencados as respostas aos riscos identificados cujo objetivo é minimizar os impactos dos riscos ou evitar que os mesmos ocorram. Assim como na identificação, esta parte está subdividida em duas menores:
 1. *Doing*: Itens elencados nessa parte do quadro devem estar sendo executados ou por executar, essa parte funciona como um *backlog* de ações de contenção de riscos;
 2. *Done*: Nessa parte devem ser colocados os itens do plano de contenção que já foram executados.
- Plano de Contingência: Nessa parte do quadro deverão ser colocadas as repostas aos impedimentos levantados e também aos riscos que não puderam ser evitados. Assim como na identificação, esta parte está subdividida em duas menores:
 1. *Doing*: Itens elencados nessa parte do quadro devem estar sendo executados ou por executar, essa parte funciona como um *backlog* de ações de contingência de riscos e impedimentos;
 2. *Done*: Nessa parte devem ser colocados os itens do plano de contingência que já foram executados.
- Riscos Residuais: Nessa parte do quadro devem ser elencados aqueles itens que tiveram seu plano de contenção executado, no entanto, ainda assim, permanecem como ameaças ao projeto, mesmo que com impacto minimizado pela aplicação do plano; e
- Riscos Secundários: Essa é a última parte do quadro e deve ser utilizada caso a execução de algum plano, seja de contenção ou contingência, gere um risco não previsto inicialmente.

Em conjunto com o quadro, as duas cerimônias adicionadas tem o intuito de efetuar o levantamento e análise de riscos em dois níveis, a nível de projeto e a nível de iteração.

5.2 Cerimônias

A primeira cerimônia foi dado o nome de *Overall Risk Meeting* e tem como objetivo identificar e analisar a riscos nível de projeto. Essa cerimônia tem como público alvo os

gestores de mais alto nível do projeto, devendo acontecer uma vez a cada iteração para apreciação geral dos riscos elencados, do ocorrido na iteração e relizar uma nova rodada de identificação e avaliação de riscos relacionados ao projeto.

As ameaças ao projeto, deverão ser colocados em *post its* e indicados dentro do quadrado amarelo, mais externo. Devido ao processo dos métodos ágeis a avaliação dos riscos deve ser feita apenas de maneira qualitativa. Diversas ferramentas podem ser utilizadas para gerenciar riscos em projetos, a Tabela 4.1 demonstra quais as técnicas e ferramentas mais utilizadas em projetos ágeis de desenvolvimento de software [2]

Tabela 5.1: Frequência de utilização das ferramentas/técnicas de riscos. Adaptado de: [2]

Frequência de utilização			
Técnica/Ferramenta	Identificação	Análise	Avaliação
Entrevista semi-estruturada	5	1	-
Questionário	3	1	-
Observação	1	-	-
Grupo focal online	2	-	-
Formulário	1	-	-
Revisão de literatura com base em artigos científicos	4	-	-
Base histórica de riscos típicos - base própria ou compartilhada	3	-	-
Reunião com equipe - <i>Brainstorming</i>	4	2	1
<i>Workshops</i>	1	-	-
Swot	-	2	-
Kanban	1	2	2
Rede Bayesiana	1	-	-
FMEA	-	1	1
Matriz de riscos (probabilidade e impacto)	-	4	4
Livre - não citou ferramenta	4	17	19

Dentre todas as ferramentas e técnicas analisadas foram escolhidas 3 específicas para o desenvolvimento do estudo, a primeira delas é o Kanban, como evidenciado na Figura 5.2, essa foi a primeira técnica e dentre as práticas da mesma utilizam-se a de Visualizar, Controlar o fluxo, Implementação de ciclos de *feedbacks* e a Melhoria colaborativa.

A segunda técnica escolhida, foi o *brainstorming*, este se dá por meio das discussões com os especialistas nas cerimônias propostas, juntamente com a utilização de uma adaptação do método conhecido como *Planning Poker*.

Por fim, a terceira e última técnica escolhida foi o FMEA (do inglês *Failures Mode and Effect Analysis*), por se tratar de uma técnica robusta para a análise de riscos. O FMEA

é uma técnica que utiliza conhecimentos de engenharia, confiabilidade e desenvolvimento organizacional para otimizar projetos, sistemas, processos, produtos e/ou serviços, é utilizado para identificar possíveis modos de falhas e seus impactos sobre o desempenho do projeto [97]. Para a identificação destas possíveis falhas são feitas suposições, onde a primeira a ser analisada é a prioridade dos possíveis problemas. Para que seja possível definir tal prioridade devem ser avaliadas 03 variáveis [97]:

1. Ocorrência: visa identificar a frequência de ocorrência de determinado problema;
2. Gravidade: determina quão sério é o problema;
3. Detecção: visa estabelecer o quão difícil é identificar a ocorrência de determinado problema.

Essas três variáveis foram adaptadas para Probabilidade, ao invés de Ocorrência e Impacto ao invés de Gravidade. Dessa forma foi possível gerar uma matriz de Probabilidade x Impacto x Detecção que é utilizada em conjunto com uma cerimônia para a avaliação dos riscos em potencial.

Já para as oportunidades deverá ser preenchido o quadro azul, mais interno, onde deverão ser verificadas oportunidades sob duas óticas, a primeira é sob o ponto de vista da geração de benefícios para o cliente e a segunda sob o ponto de vista da redução de desperdícios no projeto.

A segunda cerimônia foi chamada de *Sprint Risk Planning* e tem como objetivo avaliar os riscos referentes a iteração. Os responsáveis por essa reunião são o time de desenvolvimento e ela deve ocorrer uma vez por iteração, sempre após a reunião de planejamento da iteração. Assim como ocorre na reunião de levantamento de riscos do projeto, para os riscos da Sprint será feita apenas uma avaliação de forma qualitativa utilizando o *Failure Mode and Effect Analysis (FMEA)*, riscos negativos deverão ser colocados no quadro vermelho intermediário e as oportunidades, caso existam, deverão ser analisadas sob as óticas de geração de benefícios e redução de desperdícios e inseridas no quadro azul, mais interno.

O quadro de riscos, assim como o de desenvolvimento, deve ser verificado a cada reunião diária do time e acompanhado pelo Scrum Master para remoção de impedimentos e aplicação de planos de contenção e contingência necessários.

Para facilitar a identificação dos riscos durante as cerimônias propostas foi desenvolvido um método baseado no *planning poker* que foi chamado de *risk evaluation poker*.

5.2.1 *Planning Poker*

O *planning poker* é uma técnica para estimativas utilizada em grupo muito utilizada para estimar *user stories* no desenvolvimento ágil de software. Apesar de ser utilizado em métodos relativamente novos este método de estimação é derivado do método *Delphi*, desenvolvido na década de 50 como método de elicitación e refinamento de julgamentos de grupo [98].

O método *Delphi* se baseia em três características básicas: a primeira delas é que o método é anônimo e utiliza de questionários formais para obtenção de respostas como forma de evitar o efeito de indivíduos dominantes; a segunda característica é que o método é iterativo e possui ciclos controlados de *feedback*, dessa forma é possível reduzir ruídos entre os participantes; e por fim, o terceiro característica é que o método se baseia em uma resposta estatística de grupo, ou seja, a opinião do grupo deve ser definida como um agregado das opiniões individuais na última rodada [99]. Dessa forma as estimativas podem ser obtidas por meio de uma interação anônima entre especialistas mediante uma série de iterações conduzidas por um mediador.

Com toda a robustez o método *Delphi* se tornou o método muito complexo de estimação em grupo, nesse sentido, na década de 80 surgiu uma derivação desse método que foi chamada de *Wideband Delphi*. Essa derivação propõe menos complexidade e maior interação entre os especialistas, pois para a estimação os especialistas devem fornecer suas estimativas anonimamente, mas posteriormente devem se encontrar para discuti-las e revisá-las no contexto de outras estimativas. O ciclo se repete até que o grupo decida que a média é representativa e satisfatória [98]. Essa técnica deu origem ao *planning poker*, uma técnica ainda mais interativa e mais leve do que ambas as anteriores que permite uma interação face a face e mais discussões [100].

O *planning poker* foi desenvolvido para estimativas de histórias de usuário em um modelo de desenvolvimento ágil de software. Funciona por meio da comparação de uma história com a outra e com o apoio de um baralho com cartas numeradas na sequência de *Fibbinaci*, onde deve ser lido para o time a história, a mesma é discutida pelos desenvolvedores e cada um escolhe uma carta, após todos escolherem as cartas são apresentadas, se houver discordância nas estimativas, a equipe pode discutir suas diferentes estimativas e tentar chegar a um consenso [101]. Dessa forma são estimadas todas as histórias que serão desenvolvidas no decorrer do projeto. Esse modelo serviu de base para o que foi chamado de *risk evaluation poker*, uma maneira interativa de se mensurar e priorizar riscos de um projeto.

5.2.2 Risk Evaluation Poker

O *risk evaluation poker* segue os mesmos preceitos do *planning poker*, no entanto se diferencia do primeiro por utilizar cartões coloridos e um quadro de apoio. Os cartões coloridos indicam, na perspectiva do avaliador, a gravidade do risco quando observado sua probabilidade de ocorrência, impacto no projeto e capacidade de detecção. Para essa avaliação foram sugeridas três cores para riscos de projeto como indicado na Figura 5.4

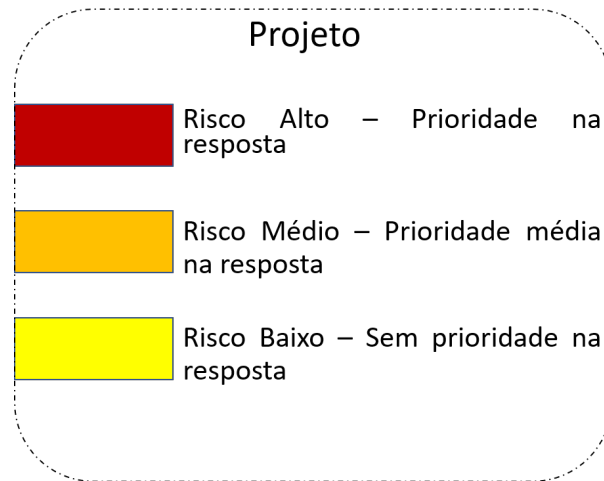


Figura 5.4: Código de cores para riscos de projeto.

Para os riscos de iteração também foram assumidas as mesmas três cores, no entanto há uma especificidade quando tratados os riscos da *Sprint*. Nas iterações os riscos devem ter um prazo estabelecido para sua resposta de acordo com a gravidade, além disso, os impedimentos são sempre considerados como os eventos mais graves e que necessitam resolução imediata. A Figura 5.5 exemplifica o código de cores para iterações, onde X e Y representam o número de dias que o risco deve ser tratado, isto pode variar de acordo com o tamanho escolhido para a iteração.

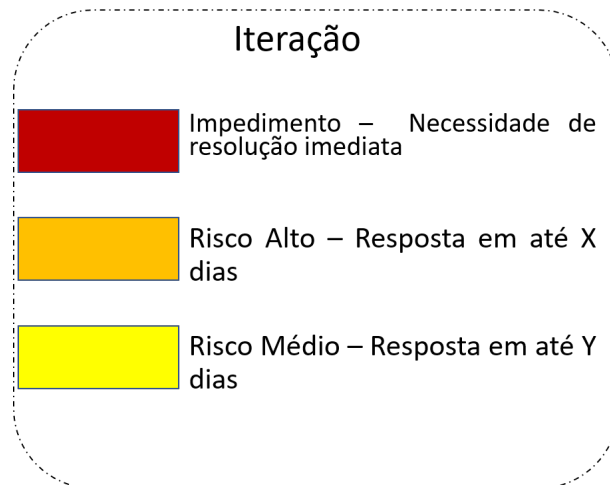


Figura 5.5: Código de cores para riscos de iteração.

Para apoiar os avaliadores na definição da severidade dos riscos, baseado no FMEA, foi criada uma matriz de Probabilidade x Impacto x Detecção, onde foi aplicado o mesmo código de cores definido para o projeto e iterações, como pode ser visto na Figura 5.6. Dessa forma o avaliador pode analisar o quadro e definir, com apoio da matriz, a cor q melhor representa a severidade do risco analisado.

Matriz de Avaliação de Riscos											
Probabilidade											Impacto
0,1	0,001	0,002	0,003	0,004	0,005	0,006	0,007	0,008	0,009	0,01	0,1
0,2	0,004	0,008	0,012	0,016	0,02	0,024	0,028	0,032	0,036	0,04	0,2
0,3	0,009	0,018	0,027	0,036	0,045	0,054	0,063	0,072	0,081	0,09	0,3
0,4	0,016	0,032	0,048	0,064	0,08	0,096	0,112	0,128	0,144	0,16	0,4
0,5	0,025	0,05	0,075	0,1	0,125	0,15	0,175	0,2	0,225	0,25	0,5
0,6	0,036	0,072	0,108	0,144	0,18	0,216	0,252	0,288	0,324	0,36	0,6
0,7	0,049	0,098	0,147	0,196	0,245	0,294	0,343	0,392	0,441	0,49	0,7
0,8	0,064	0,128	0,192	0,256	0,32	0,384	0,448	0,512	0,576	0,64	0,8
0,9	0,081	0,162	0,243	0,324	0,405	0,486	0,567	0,648	0,729	0,81	0,9
1	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1	1
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1	
	Detecção										

Figura 5.6: Matriz de Probabilidade x Impacto x Detecção.

Com o apoio de todas essas ferramentas tem-se uma maneira robusta, simples e dinâmica de gestão de riscos facilmente aplicável e condizente com o que defendem os métodos ágeis. No Capítulo seguinte será evidenciada a aplicação dessa metodologia em um projeto real.

Capítulo 6

Aplicação da Metodologia

Esta seção apresenta a aplicação da metodologia proposta bem como uma proposta de indicadores de acompanhamento, frutos da pesquisa destacada no Capítulo 4. Para isso, esta Seção foi dividida em três tópicos: o primeiro apresenta uma visão geral do projeto e da aplicação da metodologia; o segundo apresenta de forma mais detalhada alguns riscos identificados e como chegou-se a sua pontuação e as suas respostas; por fim, o terceiro tópico apresenta uma sugestão de indicadores a serem observados em projetos com vistas a auxiliar na identificação de desvios.

6.1 Visão Geral

A metodologia apresentada foi aplicada em um projeto de pesquisa da Universidade de Brasília em conjunto com o Exército Brasileiro. Esse projeto é um Termo de Execução Descentralizada entre o Exército Brasileiro, mais especificamente seu Departamento-Geral de Pessoal (DGP) e a Universidade, que tem como objetivo contribuir para a modernização da gestão do Sistema de Pessoal do Exército Brasileiro, de forma que esta agregue o indivíduo e suas atividades às novas necessidades de integração com a tecnologia e a gestão do conhecimento.

Para que fosse possível atingir o objetivo desse projeto foi elaborada uma estruturação que visa o apoio e orientação da transformação organizacional do DPG, por meio de uma logística do conhecimento amparada em conceitos de gerenciamento por competências, gerenciamento por processos e tecnologia da informação, sugerindo mudanças que gerem valor, ganhos e melhorias nos processos e sistemas atuais. Para o desenvolvimento da metodologia proposta se faz necessário o conhecimento da situação atual do DGP e sua arquitetura de processos, para que seja possível a proposição de melhorias nos mesmos. Todas essas melhorias devem ser suportadas pelas aplicações de TI, dessa forma é necessário também a análise dos Sistemas de Informação relacionados aos processos e o

levantamento de macro requisitos relacionados aos processos atuais, originando assim um estudo de demanda e uma arquitetura de TI que darão subsídio para o aperfeiçoamento dos atuais sistemas.

Essa estruturação exigiu que o time de projeto fosse composto por colaboradores com diferentes competências e conhecimentos, assim foi elaborado uma equipe composta por:

- 1 Professora Doutora Pesquisadora Sênior: responsável pela coordenação do projeto;
- 5 Professores Doutores Pesquisadores: responsáveis pelo desenvolvimento de metodologias e validações do trabalho nas diversas frentes;
- 1 Pesquisador especialista: responsável pela gestão do projeto em geral;
- 10 pesquisadores de nível superior (mestrandos e doutorandos) com diferentes papéis: 6 pesquisadores líderes de equipe; 1 pesquisador chefe do escritório de processos/projetos responsável pela coordenação das demandas do dia a dia do time e pela validação sintática dos processos; 1 pesquisador chefe de requisitos responsável pela orientação técnica do time de requisitos e pela validação do trabalho dos mesmos;
- 42 estagiários com diferentes papéis: 6 estagiários para apoio no mapeamento de competências; 16 estagiários responsáveis pela modelagem de processos; 16 estagiários responsáveis pelo levantamento de requisitos; 02 estagiários que realizam atividades desenvolvidas no escritório de projetos e 02 estagiários volante para cobrir eventuais folgas e faltas.

Para que fosse possível a coordenação do time de projeto foram criadas 8 equipes distintas para a execução, além do núcleo de coordenação de pesquisa e da gerência do projeto conforme representado na Figura 6.1.

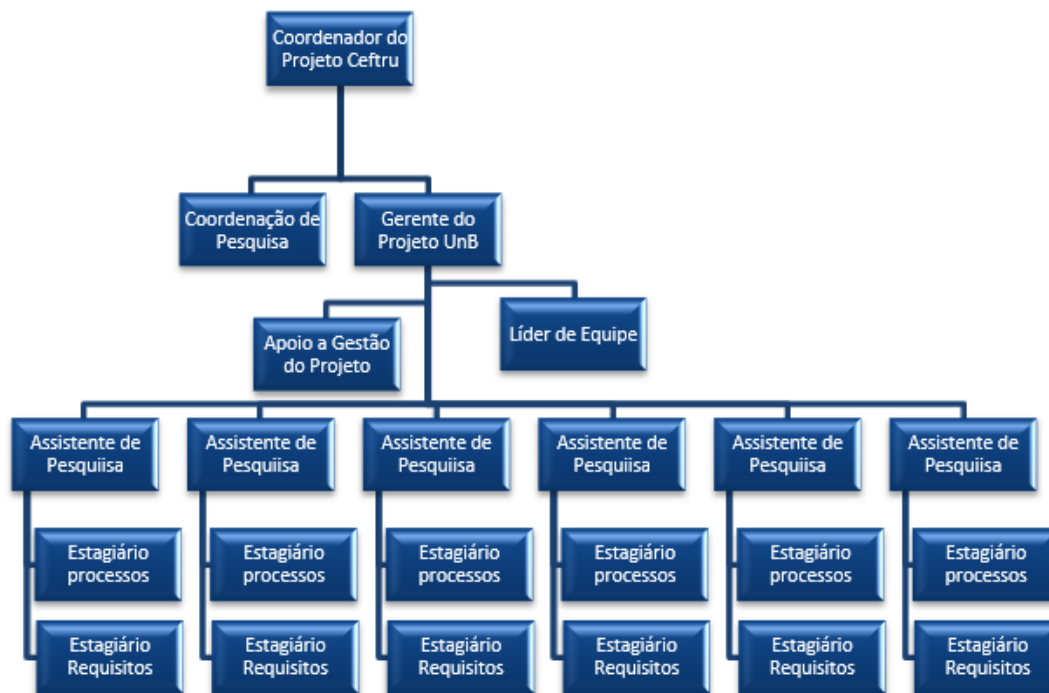


Figura 6.1: Estrutura Organizacional do Projeto.

Essa multidisciplinaridade fez com que o método de execução sugerido para o projeto fosse o *Scrum*, dessa forma tornou-se uma excelente oportunidade para a aplicação desta pesquisa.

As diversas equipes de projeto nem sempre estavam juntos, pois o cronograma de execução das atividades da *Sprint* tiveram que ser defasados para atender as reuniões no DGP. Dessa forma, no primeiro momento houve a necessidade de buscar ferramentas colaborativas para coletar dados acerca dos riscos do projeto que poderiam ser identificados por qualquer colaborador a qualquer momento. Assim, como estratégia para identificação de riscos foi utilizada uma ferramenta colaborativa virtual chamada *Ideaboardz*. Essa ferramenta se trata de um quadro *online* onde os colaboradores podem elencar quais riscos eram identificados durante as *Sprints* a qualquer momento com um acesso simples pelo celular. Outro momento de levantamento de riscos foram as reuniões diárias, onde os apontamentos do time eram também elencados na ferramenta. A ferramenta utilizada permite ainda que seja feita uma votação nos cartões identificados, dessa forma os colaboradores podem votar naqueles cartões que concordam, assim, ao final de um certo período, os cartões com maior número de votos podem ser considerados como os mais críticos, ou os que devem ser tratados com prioridade. A Figura 6.2 demonstra o quadro com riscos identificados no início do projeto.

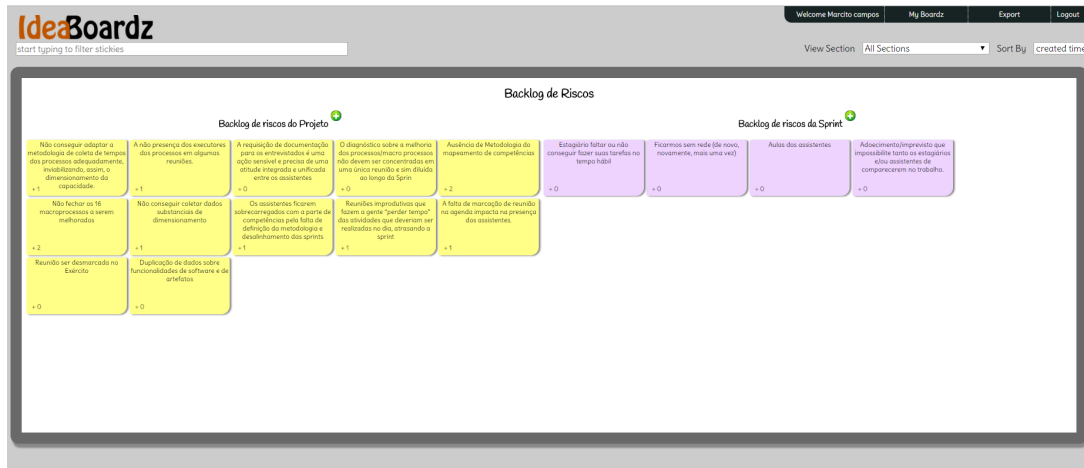


Figura 6.2: Quadro colaborativo para identificação de riscos de projeto e da Sprint.

Essa forma de identificação de riscos facilitou o trabalho, tendo em vista que a ideia inicial era que os riscos fossem identificados nas próprias cerimônias propostas. Dessa maneira nas cerimônias de avaliação e planejamento já havia um *Backlog* de riscos a ser analisado, não sendo necessário ser feito um *brainstorming* na reunião para esse motivo. Isso fez com que os eventos *Overall Risk Meeting* e *Sprint Risk Planning* se tornassem mais simples e objetivos.

Os riscos identificados no quadro eram apresentados durante as cerimônias e, a medida que eram apresentados, os colaboradores envolvidos na reunião utilizavam o *risk evaluation poker*, definido na Seção 5.2.2, para estabelecer, com o apoio da Matriz de Probabilidade x Impacto x Detecção, apresentada na Figura 5.6 e dentro das características do referido risco, sua probabilidade, impacto e facilidade de detecção.

Após avaliados todos os riscos com relação a estes critérios foram traçadas estratégias para conter os mesmos, e colocadas em *post its*, nas mesmas cores do risco correspondente, na área do Plano de Contingência. Também foram identificadas ações de contorno, também identificadas em *post its* nas mesmas cores que o dos riscos, para o caso de ocorrência dos riscos, e colocadas na área do Plano de Contingência. Ao final da primeira reunião obteve-se o quadro identificado na Figura 6.3.

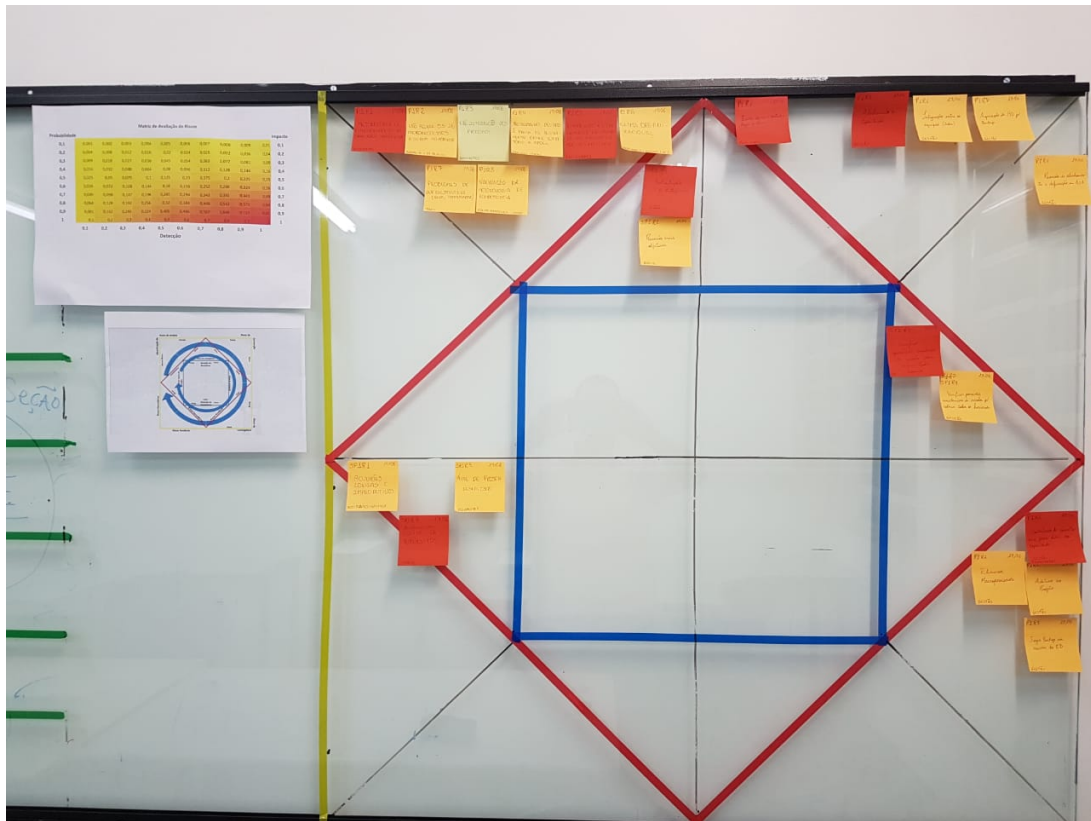


Figura 6.3: Quadro de Riscos preenchido na primeira *Sprint*.

Alguns pontos que dificultaram a implementação dessa nova metodologia de gestão de riscos inicialmente foram o fato do time de projeto não ter tido contato ainda com o Scrum e o Gerente de Projetos que aplicou a metodologia não estava o tempo todo presente no projeto devido a compromissos em outras empresas. Dessa forma foi preciso preparar o time para a utilização tanto do Scrum como da nova metodologia e houve a necessidade de incluir ainda duas imagens de apoio para o time para que se nortearassem quanto ao fluxo definido para a utilização do quadro como também para facilitar a identificação das cores para os *post its* de riscos, assim é possível reparar ainda na Figura 6.3 que foi necessário que fossem colocados a Matriz de Probabilidade x Impacto x Detecção e também uma imagem com o fluxo que deve ser seguido na utilização do quadro. Isso facilitou a aplicação, pois na ausência do Gerente os próprios assistentes conseguiram rodar o método.

Após a primeira rodada de reuniões pôde-se perceber que o time conseguiu identificar muitas ameaças ao projeto e poucas ameaças a Sprint, o que era esperado, tendo em vista que o time nunca havia trabalhado com métodos ágeis antes do projeto em questão, dessa forma conseguiram enxergar mais facilmente ameaças a médio e longo prazo, ao invés de ameaças no curto prazo, como aconteceria caso os riscos impactassem as Sprints.

Das ameaças inicialmente identificadas, quando analisadas sob a ótica de Probabilidade x Impacto x Detecção, 02 riscos se encontravam no quadrante vermelho, o que significa que são de alto impacto, alta gravidade e de complexa detecção, 05 riscos que se enquadravam na categoria média e 01 risco na categoria baixa de baixo Impacto x Probabilidade x Detecção.

Para os riscos mais graves foi possível identificar planos de contenção, no entanto para apenas 01 foi possível prever um plano de contingência. Para 03 riscos da categoria médio foram elaborados planos de contenção e de contingência, inclusive para um destes já foi determinada a solução. Os riscos de baixo impacto não influenciaram no momento da análise.

Para os riscos referentes a *Sprint* foi identificado apenas um risco tido como grave, para o qual foi elaborado um plano de contenção e um plano de contingência, e dois riscos de média gravidade, destes apenas um teve como resultado da análise um plano de contenção e contingência.

O quadro era avaliado diariamente e medidas tomadas para que os riscos não viessem a afetar o projeto ou, caso ocorressem, tivessem seu impacto minimizado por meio de ações de contingência. Ao final da terceira *Sprint* existiam mais riscos de projeto críticos e os identificados para a iteração permaneceram inalterados, como pode ser identificado na Figura 6.4.



Figura 6.4: Quadro de Riscos preenchido depois da *Sprint 3*.

Pela Figura acima é possível notar que final da terceira Sprint a quantidade de riscos graves subiu de 02 para 04, foram desenvolvidos novos planos de contenção e, dentre estes, 02 foram efetivamente executados. Nota-se também que foram retirados 02 *post its* de riscos de criticidade média, 01 de plano de contenção, 01 de plano de contingência do quadro em comparação com a Sprint 1 e também um risco descoberto no decorrer das iterações. Estes *post its* foram alocados a esquerda do quadro, para que se mantivesse um histórico dos riscos e dos planos. A retirada dos riscos do quadro significa que a implementação dos planos efetivamente eliminou o risco identificado.

Apesar de não ter sido aplicado durante um longo espaço de tempo a metodologia mostrou-se satisfatória do ponto de vista de identificação, avaliação e monitoramento dos riscos. As duas cerimônias não tomaram muito tempo das iterações e o *Risk Evaluation Poker* trouxe uma forma dinâmica de avaliação e consenso entre os decisores, sendo essencial no processo de avaliação e foi extremamente favorável ao processo como um todo, todas as discussões geradas pelo método de avaliação proporcionaram um entendimento melhor de cada um dos riscos identificados e, conseqüentemente, uma avaliação mais assertiva. O quadro Kanban trouxe um modelo de gestão à vista, apesar de ter um desenho incomum quando se comparado aos tipos de quadros tradicionalmente utilizados em mé-

todos ágeis, a dinâmica proposta proporcionou reflexões diárias sobre os planos e deixou claro as pendências de projeto no tocante a gestão de riscos e evidenciava nitidamente a necessidade de ações por parte de todos os colaboradores envolvidos.

6.2 Avaliação de riscos

Para exemplificar melhor o funcionamento da metodologia nesse tópico será abordado como foi executada a análise, pontuação e desenvolvimento de respostas de contenção e contingência para um determinado risco. No entanto, para que seja possível exemplificar o risco identificado é preciso que seja melhor contextualizado o projeto.

Os projeto em questão tinha como objetivos:

- Identificar os processos que fazem parte da Diretoria de Gestão de Pessoas do Exército Brasileiro, por meio de sua cadeia de valor;
- Selecionar 200 processos dentre aqueles identificados durante o mapeamento da Cadeia de Valor para modelagem em sua forma inicial (AS-IS);
- Priorizar 16 processos dentre os 200 inicialmente selecionados, para que, a partir de seu entendimento e mensuração, sejam propostas melhorias (TO-BE);
- Mapear as competências dos executores dos processos modelados, e a posteriori identificar GAPS de competências;
- Propor uma análise de dimensionamento da força de trabalho para os processos melhorados; e
- Propor uma racionalização dos sistemas de TI em uso no DGP para os processos melhorados.

A duração total prevista para o projeto era de 21 meses e os objetivos do projeto foram distribuídos para que se preenchessem os 21 meses com entregas assim definidas: a primeira aconteceria 3 meses após o início do projeto; a segunda, terceira, quarta e quinta entregas ocorreriam de 4 em quatro meses; e, por fim, a sexta e última entrega do projeto ocorreria 2 meses após a quinta.

Para que fosse possível cumprir o cronograma e efetuar todas as entregas previstas no projeto foi desenvolvido um *roadmap* no início do projeto com a projeção do que se esperava no decorrer de sua execução, o qual pode ser visto na Figura 6.5.

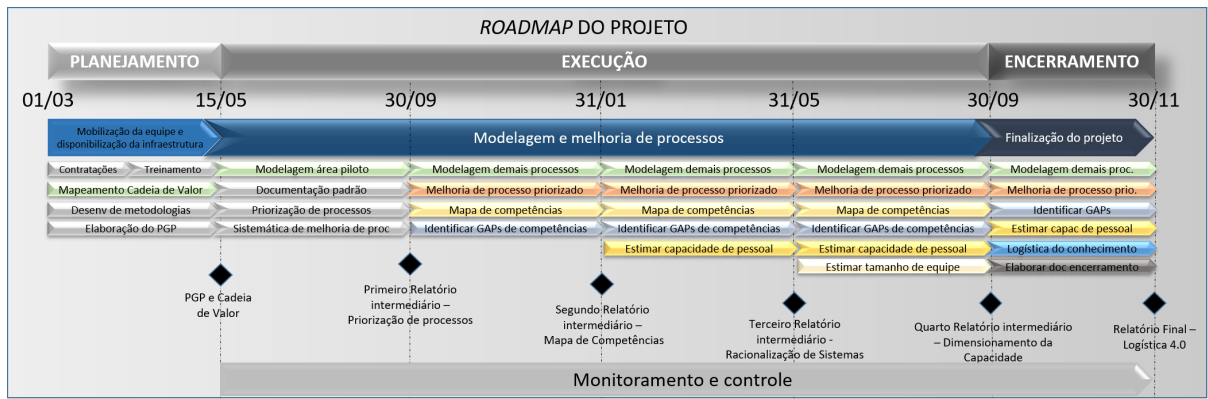


Figura 6.5: Roadmap do projeto.

Por meio da Figura 6.5 pode ser vista a divisão prevista para que os objetivos fossem alcançados. Percebe-se que, na primeira etapa está contemplado o desenvolvimento das metodologias necessárias para o atingimento dos objetivos do projeto. A execução de fato com toda a equipe de projeto alocada se deu com um pequeno atraso com relação ao previsto no *roadmap*, ao invés de iniciar no dia 15/05, a execução teve seu início efetivo em meados de junho. Assim, apenas nessa época foi possível iniciar a aplicação efetiva da metodologia proposta. Dessa forma o quadro para identificação dos riscos foi aberto e foi solicitado que os assistentes preenchessem os riscos com o que achavam que poderia ser nocivo, tanto a nível de projeto como a nível de Sprint.

Dentre os diversos riscos identificados pelos assistentes, um dos mais impactantes e mais votados está em destaque na Figura 6.6.

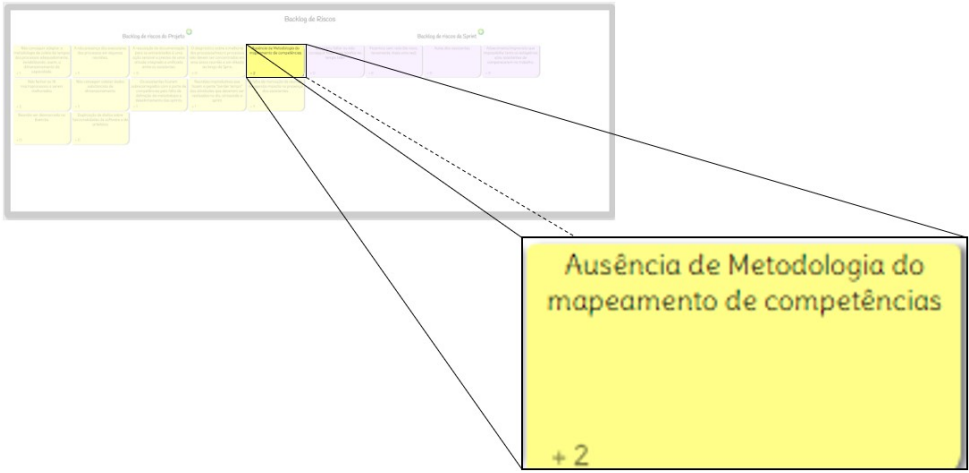


Figura 6.6: Risco de projeto identificado.

Esse risco era obviamente um dos mais graves, tendo em vista que a metodologia é o pilar para a coleta de dados e execução das atividades para que se chegue a uma entrega.

Dessa forma, esse foi um dos primeiros itens a serem votados na *Overall Risk Meeting* e, quando solicitado que os assistentes o analisassem confrontando com a tabela de Probabilidade x Impacto x Detecção, o resultado foi unânime. Todos entenderam que o risco tinha alta probabilidade de ocorrência, afinal a metodologia ainda não estava pronta, alto impacto, pois provocaria o não cumprimento de uma entrega contratual, no entanto não era de tão fácil detecção, afinal era sabido que estava sendo desenvolvido um trabalho no sentido de elaborar a metodologia, mas não estava visível em que ponto o grupo responsável se encontrava.

Definido grau de risco, foi preenchido o *post-it* referente a esse risco e colocado no quadro de riscos iniciais de projeto. Passou-se então para a etapa seguinte, a definição de um plano de contenção. Nessa etapa a pergunta que se busca responder é: o que pode ser feito para evitar que o risco se concretize?

Para que fosse gerado o plano de contenção foi feito um *brainstorming* com o time, onde sugestões foram colocadas na mesa para que fossem analisadas e discutidas. Apesar de ser um momento de discussão, essa etapa não se estendeu muito, pois a equipe de projeto discutiu e estabeleceu rapidamente que o plano de contenção para o referido risco seria uma maior integração entre os alunos e professores na elaboração da metodologia. Nessa altura o time de assistentes já conhecia o contexto do cliente e poderia sugerir algo que fosse atingível dentro das limitações do cliente e de tempo.

Acordado então mais uma etapa, o plano de contenção para o risco identificado, foi preenchido um *post-it* com o plano de contenção definido e sinalizado com o mesmo identificador do risco, pois assim é possível rastrear qual plano se refere a qual risco, e colocado no quadro de plano de contenção, na parte referente ao *Doing*, pois dessa maneira se demonstra que o item deverá entrar prontamente em andamento.

Com o preenchimento do cartão e colocado no seu devido lugar encerra-se a fase de definição do plano de contenção. Após essa etapa passamos para a definição de um plano de contingência, desse modo, caso o plano de contenção não surta efeito e mesmo assim o risco ocorra, existe uma solução de contorno. Nesta fase buscamos respostas para as seguintes perguntas: se o risco ocorrer, ou se o plano de contenção não surtir efeito, que pode ser feito para minimizar seu impacto? Temos uma solução de contorno?

Similarmente ao método de definição do plano de contenção, é feito também um *brainstorming* para a definição de um plano de contingência. Novos cenários são colocados em questão e debatidos novamente com o time. Assim como na etapa anterior o consenso não demorou a ser atingido. O time entendeu nesse caso que, a equipe composta por alunos e professores deveriam se dedicar integralmente para alinhar a metodologia que seria

aplicada aos processos modelados, com o objetivo de identificar os gaps de competências para a execução desses processos, dessa forma houve uma capacitação de toda a equipe a fim de facilitar o entendimento e finalizar a metodologia proposta.

Um ponto importante que deve ser levado em consideração na execução do método é que todo *brainstorming* tende a sair do controle caso não haja um moderador com autoridade para organizar as ideias levantadas e a discussão. A atuação deve ser constante visando controlar os ímpetos e conseguir que o time colabore e a reunião não vire um conflito.

Adicionalmente a metodologia aqui discutida, propõe-se a utilização de práticas e medições, estes não foram utilizados em sua totalidade no projeto em questão devido a pesquisa não estar pronta a época, no entanto sugere-se sua utilização para que a probabilidade de sucesso do projeto seja ampliada. Essas práticas e medições são descritas na seção seguinte.

6.3 Proposta de medições

Complementarmente a metodologia sugerida nessa pesquisa são propostas algumas práticas e medições para que sejam aplicadas em conjunto com o que é apresentado nesse estudo e que possibilitem o favorecimento do sucesso do projeto.

Essas práticas e medições têm como base a pesquisa elaborada e descrita no Capítulo 4. Como resultado da pesquisa foram obtidas as percepções acerca do que especialistas na área acreditam ser fundamental para o sucesso do projeto, dessa forma as variáveis que mais influenciam no sucesso de um projeto devem ser monitoradas com maior diligência.

Conforme apresentado no Capítulo 4, os constructos que têm maior influência no sucesso do projeto são:

- Para o constructo Técnico-Fatores:
 - FIT9 - Não existe uma estrutura da organização do projeto; e
 - FIT8 - Plano de projeto não é atualizado regularmente;
- Para o constructo Pessoas-Fatores:
 - FIPe17 - Gerente de projeto incompetente; e
 - FIPe24 - Falta de apoio da alta administração.

Como são estes os constructos com maior influência, serão definidas ações que visam suprir as faltas decorrentes desses critérios. Como forma de minimizar o impacto do fator FIT9 (Falta de estrutura da organização do projeto), sugere-se que seja criado um *checklist*

geral, tendo em vista que é difícil efetuar a medição desse parâmetro. Dessa forma todos os projetos iniciados por uma instituição devem seguir tal *checklist* visando elaborar uma estrutura que atenda ao projeto. Além desse produto, recomenda-se também a utilização de *templates* com padrão de documentação a ser utilizada, dessa forma a documentação padrão já contará com todos os dados necessários.

Para o fator FIT8 (Plano de projeto não é atualizado regularmente) sugere-se que sejam verificados de forma periódica a documentação, visando garantir, por meio do versionamento do documento, que o plano do projeto esteja regularmente atualizado. Essa verificação deverá ocorrer de 15 em 15 dias ou até um dia após a ocorrência de alguma mudança.

Para o fator FIPe17 (Gerente de projeto incompetente) deve ser considerado que sejam feitos testes regulares acerca do conhecimento do profissional sobre o projeto e sobre gestão de projetos. Esses testes devem ocorrer duas vezes por ano visando assegurar que o profissional tenha os conhecimentos mais atuais. Como forma de mensuração o profissional deve atingir uma proficiência de 90% nas disciplinas.

Por fim, para o último fator verificado FIPe24 (Falta de apoio da alta administração) deve ser considerado um plano de comunicação consistente, além de serem feitos *workshops* duas vezes por ano para integração da alta administração junto com o time de projeto. Dessa forma a mensuração deveria ter como base a quantidade de *workshops* realizados no decorrer do projeto, bem como avaliada a frequência da comunicação do Gerente do Projeto com a alta administração, sugere-se relatórios semanais presencialmente para engajar os envolvidos.

Das sugestões acima descritas, no projeto utilizado foi possível utilizar apenas as duas primeiras tendo em vista o tempo de aplicação curto. No entanto acredita-se que estas práticas garantirão que os fatores mais críticos indicados pelos especialistas sejam devidamente evitados. Sendo assim, é apresentado na seção seguinte a conclusão desse estudo.

Capítulo 7

Conclusão

Este trabalho de pesquisa promoveu um estudo acerca dos métodos de gestão de riscos mais atuais do mercado. Os riscos são tratados como eventos incertos que podem afetar um ou mais objetivos do projeto, sendo tratados tanto como ameaças como oportunidade e estão presentes em todos os projetos de qualquer indústria. Dessa forma existe uma clara necessidade de um modelo de gestão de riscos estruturado visando facilitar a identificação e visualização das ameaças e oportunidades pelos gestores para uma rápida ação.

Adaptabilidade e rápida tomada de ação são características de métodos ágeis de gestão. Os agilistas defendem que métodos ágeis por si só já garantem são inertes aos riscos pela sua alta adaptabilidade. Porém este estudo comprova que apesar da utilização desses métodos aumentarem a taxa de sucesso em projetos, ainda há uma alta taxa de insucesso, o que comprova que ainda existe a necessidade latente da elaboração de um método eficaz de gestão de riscos para métodos ágeis.

Dentre todos os métodos apresentados, aquele mais difundido e utilizado atualmente é o Scrum, atingindo aproximadamente 70% do mercado de acordo com a pesquisa da companhia Version One. Esse número bastante expressivo guiou também essa pesquisa, fazendo com que o foco de aplicação fosse voltado para o Scrum.

A metodologia proposta adiciona duas novas cerimônias ao Scrum, ambas com o intuito de levantamento de riscos, sejam eles de projetos ou referentes a demanda planejada para a Sprint, define uma matriz de avaliação de riscos adaptada do FMEA para classificação dos riscos com relação a sua probabilidade, impacto e facilidade de detecção. A metodologia apresentada também aproveita-se da utilização comum do *planning poker* por times ágeis para introduzir uma nova forma de avaliação de riscos, aqui chamada de *Risk Evaluation Poker*. Essa nova maneira de avaliação visa a obtenção do consenso por meio da exposição de motivos por parte dos integrantes do time. Por fim e um dos pontos mais importante, foi estabelecido um quadro baseado nos princípios da metodologia Kanban para que os riscos possam ser incluídos e visualizados constantemente.

Adicionalmente a metodologia, foram sugeridas práticas e medições para evitar aqueles fatores que mais impactam negativamente o sucesso de um projeto. Essas variáveis são fruto de uma pesquisa desenvolvida com 82 gestores de todo o Brasil. Com base em pesquisas científicas foi desenvolvido um questionário com 120 perguntas segregadas entre fatores e critérios de insucesso em projetos. Ao resultado dessa pesquisa foi aplicado um modelo de Equações Estruturais para que fosse possível identificar quais são os fatores e critérios que mais influenciam o sucesso de um projeto.

No decorrer da aplicação do estudo foram encontradas algumas limitações. Um dos obstáculos enfrentados na pesquisa reside no problema da percepção dos especialistas com a validade preditiva, pois as percepções devem ser coletadas em um tempo presente e associadas a um resultados e o que foi obtido na pesquisa foram percepções de insucesso sobre percepções de sucesso, o que poderia gerar um resultado desfavorável, no entanto, ainda assim foi possível obter a confiabilidade no modelo, conforme demonstrado no Capítulo 4.

Ainda com relação a pesquisa e a aplicação das equações estruturais, ocorre outra limitação, mais especificamente com relação ao modelo utilizado, pois a época foi aplicado o melhor modelo de equações estruturais para a situação, o modelo de fator comum, no entanto, estudos recentes apontam que para dar conta de artefatos com caráter construtivistas, o mais indicado é o modelo composto utilizado juntamente com a análise confirmatória composta, pois este modelo é formado por uma combinação linear ponderada de indicadores observáveis e, portanto, em contraste com o modelo de fator comum, os indicadores não compartilham necessariamente uma causa comum [102]. Este modelo mais recente não foi utilizado devido ao período de aplicação da pesquisa ser anterior a publicação de estudos comprobatórios de sua eficiência.

Outro fator limitador foi a quantidade de respondentes a pesquisa, pois o questionário foi enviado para aproximadamente 700 pessoas e foi obtida uma baixa taxa de resposta, atingindo apenas 88% de confiabilidade, segundo o método utilizado, idealmente deveriam ser atingidos minimamente 95% de confiabilidade, no entanto, acredita-se que refinando o questionário podem ser obtidas mais respostas e atingir o percentual de confiabilidade ideal.

Por fim a última limitação encontrada foi o tempo de aplicação do método, pois o mesmo só pôde ser aplicado durante 03 Sprints em virtude da paralisação temporária do projeto onde o mesmo estava sendo validado. Esse período curto de aplicação inviabilizou a identificação de oportunidades para o projeto e, dessa forma, o quadro não pode ser utilizado em sua totalidade.

Apesar das limitações encontradas, a metodologia se mostrou bastante efetiva e os riscos identificados tiveram ações de contenção executadas e, em muitos dos casos, os

riscos deixaram de existir. Além disso a forma visual de expor o risco fez com que o gestor sempre verificasse o andamento da ação de contenção e identificasse possíveis gargalos que impediam determinadas ações de serem executadas. Durante o período de execução a pesquisa sobre fatores e critérios de insucesso em projetos ainda estava em andamento, o que impediu que os indicadores propostos fossem aplicados em conjunto com o método.

Espera-se que a metodologia proposta possa agregar ainda mais valor a projetos que trabalham com o Scrum como modelo de gestão e que os riscos possam ser mais facilmente identificados, avaliados e tratados durante as Sprints e o projeto como um todo. A intenção é que os riscos sejam identificados e tratados de forma constante evitando assim que as ameaças se concretizem e impactem o projeto, bem como potencialize o aproveitamento das oportunidades a fim de que os projetos alcancem o sucesso de forma mais assertiva.

Devido as limitações de tempo para aplicação da metodologia e das práticas e medições, sugere-se como trabalhos futuros a aplicação conjunta destes com a metodologia durante um prazo mais longo para que seja possível uma comprovação mais assertiva. Outra recomendação para aplicações futuras é que, para a elaboração da matriz de Probabilidade x Impacto x Detecção seja feito um estudo com a alta gestão acerca do apetite de risco da organização, por meio da aplicação de cenários de risco e sua avaliação com a utilização do *Analytic Hierarchy Process* (AHP) como método de suporte a tomada de decisão. Além dessas também é sugerido que seja gerado um modelo composto de equações estruturais para avaliar as possíveis diferenças de resultados com relação a este estudo.

Referências

- [1] Group, The Standish: *CHAOS Manifesto 2014: Value versus Success & the Orthogonals*. Relatório Técnico, 2014. xii, 1, 45
- [2] Ferreira, E C: *Proposta de metodologia de gestão de riscos para projetos ágeis de software no Instituto Nacional de Estudos e Pesquisas Anísio Teixeira (INEP)*. 2017. <http://www.repositorio.unb.br/handle/10482/24903>. xii, 75
- [3] Hastie, Shane e Stéphane Wojewoda: *Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch*, 2015. <https://www.infoq.com/articles/standish-chaos-2015>. 1, 2
- [4] Project Management Institute: *Guia do Conhecimento em Gerenciamento de Projetos*. 6ed edição, 2017, ISBN 9789720940018. 1, 2, 10, 12, 13, 14, 36
- [5] Brasileira, Norma: *ABNT NBR ISO 31000:2018*, 2018. 2, 3, 36, 37, 39, 40
- [6] COSO: *Gerenciamento de Riscos Corporativos - Estrutura Integrada*. página 135, 2007. 2
- [7] OGC: *Management of Risk: Guidance for Practitioners*. 2010, ISBN 0113312741. <http://books.google.com/books?id=BcM-b4DTttcC&pgis=1>. 3, 38
- [8] PMI: *Practice Standard Project Risk Management.pdf*, 2009, ISBN 13: 978-1-930699-47-2. 3
- [9] Gold, Benjamin e Clive Vassell: *Using risk management to balance agile methods*. 2015. 3
- [10] Moran, Alan: *Agile Risk Management*. (Cockburn 2007):87–94, 2014. 3, 43
- [11] Aken, J. E. van: *Management research on the basis of the design paradigm: The quest for field-tested and grounded technological rules*. Journal of Management Studies, 41(2):219–246, 2004, ISSN 00222380. 5, 55
- [12] Lacerda, Daniel Pacheco, Aline Dresch, Adriano Proença e José Antonio Valle Antunes Júnior: *Design Science Research: método de pesquisa para a engenharia de produção*. Gestão & Produção, 20(4):741–761, 2013, ISSN 0104-530X. 5
- [13] Bayazit, Nigan: *Investigating Design: A Review of Forty Years of Design Research*. Design Issues, 20(1):16–29, 2004, ISSN 0747-9360. 5

- [14] Çağdaş, Volkan e Erik Stubkjær: *Design research for cadastral systems*. Computers, Environment and Urban Systems, 35(1):77–87, 2011, ISSN 01989715. 5
- [15] Peffers, Ken, Tuure Tuunanen, Marcus A. Rothenberger e Samir Chatterjee: *A Design Science Research Methodology for Information Systems Research*. Journal of Management Information Systems, 24(3):45–78, 2007, ISSN 0742-1222. 5, 6, 7, 9
- [16] Järvinen, Pertti: *Action research is similar to design science*. Quality and Quantity, 41(1):37–54, 2007, ISSN 00335177. 7
- [17] Peng, Liyu: *Quality , Passenger Satisfaction and Loyalty Based on the Structural Equation Model*. (71273023):0–4, 2016. 7
- [18] Kaplan, David.: *Structural Equation Modeling*. Sage Publications, Inc., Thousand Oaks, California, 1st edição, 2000, ISBN 9781623962449. 7, 45
- [19] Chiu, Y. C.: *An introduction to the history of project management : from the earliest times to A.D. 1900*. Eburon, 2010, ISBN 9789059724372. 10
- [20] Kwak, Young Hoon: *A brief history of project management*. The Story of Managing Projects, 2005. <https://pdfs.semanticscholar.org/b926/08d4f03e9ef4bb688851e6eee47886a60be5.pdf>. 10
- [21] Wsocki, Robert K: *Effective project management: traditional, agile, extreme*. John Wiley & Sons, 2014, ISBN 9781118729168. 11, 12, 17, 18, 20
- [22] Špundak, Mario: *Mixed Agile/Traditional Project Management Methodology – Reality or Illusion?* Procedia - Social and Behavioral Sciences, 119:939–948, 2014, ISSN 18770428. <http://www.sciencedirect.com/science/article/pii/S187704281402196X>. 11, 17
- [23] Barry Boehm: *Get ready for agile methods, with care*. <https://pdfs.semanticscholar.org/25bc/573a4c8b9ea9314f82797bbfafb2ffbd2d3a.pdf>. 11, 19
- [24] Commerce, Office of Government: *Gereciando projetos de sucesso com PRINCE2*. The Stationery Office, 2011, ISBN 9780113313471. 15, 16
- [25] Collyer, Simon, Clive Warren, Bronwyn Hemsley e Chris Stevens: *Aim, fire, aim-Project planning styles in dynamic environments*. Project Management Journal, 41(4):108–121, setembro 2010, ISSN 87569728. <http://doi.wiley.com/10.1002/pmj.20199>. 17
- [26] Middleton, Peter e David Joyce: *Lean software management: BBC worldwide case study*. IEEE Transactions on Engineering Management, 59(1):20–32, 2012, ISSN 00189391. 17
- [27] Project Management Institute e Agile Alliance: *Agile Practice Guide*. Project Management Institute, Newtown Square, Pennsylvania, 2017. 17

- [28] Holweg, Matthias: *The genealogy of lean production*. Journal of Operations Management, 25(6):420–438, 2007. 18
- [29] Krafifcik, John F.: *Triumph of the Lean Production System*. Sloan Management Review, 30(1):41–52, 1988. 18
- [30] Ohno, Taiichi: *Toyota Production System: Beyond Large-Scale Production*. <http://www.amazon.com/Toyota-Production-System-Beyond-Large-Scale/dp/0915299143>. 18
- [31] Poppendieck, Mary, Tom Poppendieck e Addison Wesley: *Lean Software Development: An Agile Toolkit*. 2003, ISBN 0-321-15078-3. <http://200.17.137.109:8081/novobsi/Members/teresa/optativa-fabrica-de-sw-organizacoes-ageis/artigos/AddisonWesley-LeanSoftwareDevelopment-AnAgileToolkit.pdf>. 18
- [32] Alahyari, Hiva, Richard Berntsson Svensson e Tony Gorschek: *A study of value in agile software development organizations*. Journal of Systems and Software, 125:271–288, 2017, ISSN 0164-1212. <http://www.sciencedirect.com/science/article/pii/S0164121216302539>. 18
- [33] Serrador, Pedro e Jeffrey K. Pinto: *Does Agile work? - A quantitative analysis of agile project success*. International Journal of Project Management, 33(5):1040–1051, 2015, ISSN 02637863. <http://dx.doi.org/10.1016/j.ijproman.2015.01.006>. 18, 19
- [34] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland e Dave Thomas: *Manifesto for Agile Software Development*, 2001. <http://agilemanifesto.org/>. 19
- [35] Sutherland, Jeff: *SCRUM : A arte de fazer o dobro de trabalho na metade do tempo*. 2016, ISBN 9788544100882. <https://ler.amazon.com.br/?asin=B000EI3TKM>. 20, 23, 25
- [36] Takeuchi, Hirotaka e Ikujiro Nonaka: *The New New Product Development Game*. Harvard Business Review, 1986. 20
- [37] Schwaber, Ken: *Agile Project Management with Scrum*. 2004, ISBN 073561993X. 20, 21, 23, 24
- [38] Sabbagh, Rafael: *Scrum Gestão Ágil para projetos de Sucesso*. Número 1. Casa do código, 2014, ISBN 9780874216561. 24, 43
- [39] Beck, Kent: *Embracing Change with Extreme Programming*. IEEE Computer Society, 92:F83–F88, 1999, ISSN 1359-2998. 25
- [40] Beck, Kent e Cynthia Andres: *Extreme Programming Explained: Embrace Change*. Addison Wesley Professional, 2004, ISBN 0321278658. 25, 26

- [41] Sugimori, Y., K. Kusunoki, F. Cho e S. Uchikawa: *Toyota production system and kanban system materialization of just-in-time and respect-for-human system*. International Journal of Production Research, 15(6):553–564, 1977, ISSN 1366588X. 26
- [42] Anderson, David J: *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press, 2010, ISBN 978-0-9845214-2-9. <http://www.c4u-trade.com/documenten/Presentation20141030IOACGlobalC4ChainDynamics.pdf>. 27
- [43] Anderson, David J e Andy Carmichael: *Essential Kanban Condensed*. Blue Hole Press, 1st edição, 2016, ISBN 9780984521425. 27, 28, 29
- [44] STAPLETON, Jennifer: *DSDM Business Focused Development*. 2003, ISBN 0-321-11224-5. 30, 31, 32
- [45] Abrahamsson, P, J Warsta, M T Siponen e J Ronkainen: *New Directions on Agile Methods: A Comparative Analysis*. Software Engineering, 2003. Proceedings. 25th International Conference on, 6:244–254, 2003, ISSN 0270-5257. 30, 32
- [46] Agile Business Consortium: *DSDM Atern Handbook*, 2008. <https://www.agilebusiness.org/content/foreword-0>. 31
- [47] Palmer, Steve R. e Mac Felsing: *A Practical Guide to Feature Driven Development*. 2002, ISBN 0130676152. 32, 33
- [48] Scott W. Ambler: *The Agile Unified Process (AUP) Home Page*. <http://www.amblysoft.com/unifiedprocess/agileUP.html>. 33, 34
- [49] Christou, Ioannis, Stavros Ponis e Eleni Palaiologou: *Using the Agile Unified Process in Banking*. IEEE Software, 27(3):72–79, maio 2010, ISSN 0740-7459. <http://ieeexplore.ieee.org/document/5232801/>. 34
- [50] Cockburn, Alistair: *Crystal Clear A Human-Powered Methodology for Small Teams*. Addison Wesley Professional, 2004, ISBN 0201699478. 35, 36
- [51] Elbanna, A e S Sarker: *The Risks of Agile Software Development: Learning from Adopters*. IEEE Software, 33(5):72–79, 2016, ISSN 0740-7459. 36, 43
- [52] Brasiliano, Ribeiro e Antonio Celso: *Gestão e Análise de Riscos Corporativos*. ISBN 9788587297198. 39
- [53] Berner, Christine Louise e Roger Flage: *Creating risk management strategies based on uncertain assumptions and aspects from assumption-based planning*. Reliability Engineering & System Safety, 167:10–19, 2017, ISSN 09518320. <http://www.sciencedirect.com/science/article/pii/S0951832016302460>. 42
- [54] Cerpa, Narciso, Matthew Bardeen, Barbara Kitchenham e June Verner: *Evaluating logistic regression models to estimate software project outcomes*. Information and Software Technology, 52(9):934–944, 2010, ISSN 09505849. <http://dx.doi.org/10.1016/j.infsof.2010.03.011>. 42, 44

- [55] Boehm, B.W. Barry: *Software risk management: principles and practices*. IEEE Software, 8(1):32–41, 1991, ISSN 0740-7459. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=62930>. 42
- [56] Büyüközkan, Gülçin e Da Ruan: *Choquet integral based aggregation approach to software development risk assessment*. Information Sciences, 180(3):441–451, 2010, ISSN 00200255. 42
- [57] ISACA: *Cobit 5 for Risk*. 2013, ISBN 9781604204582. 42
- [58] Software Engineering Institute: *CMMI for Development, Version 1.3*. Carnegie Mellon University, (November):482, 2010, ISSN CMU/SEI-2010-TR-033. <http://repository.cmu.edu/sei/287/>. 42
- [59] Li, Jianping, Minglu Li, Dengsheng Wu e Hao Song: *An integrated risk measurement and optimization model for trustworthy software process management*. Information Sciences, 191(February):47–60, 2012, ISSN 00200255. <http://dx.doi.org/10.1016/j.ins.2011.09.040>. 42
- [60] SCRUMstudy: *Um Guia para o Conhecimento em Scrum (guia sbok)*. 2013, ISBN 9780989925204. 43
- [61] Pinto, Jeffrey K e Dennis P Slevin: *Project success definitions and measurement techniques*. Project Management Journal, (February), 1988. <http://www.pmi.org>. 44
- [62] Rodríguez-Segura, Enrique, Isabel Ortiz-Marcos, José Javier Romero e Javier Tafur-Segura: *Critical success factors in large projects in the aerospace and defense sectors*. Journal of Business Research, 69(11):5419–5425, 2016, ISSN 01482963. <http://www-sciencedirect-com.ez54.periodicos.capes.gov.br/science/article/pii/S0148296316303526>. 44
- [63] Cooke-Davies, Terry: *The "real" success factors on projects*. International Journal of Project Management, 20(3):185–190, 2002, ISSN 02637863. 44
- [64] Joslin, Robert e Ralf Müller: *The relationship between project governance and project success*. JPMA, 34:613–626, 2016. http://ac-els-cdn-com.ez54.periodicos.capes.gov.br/S0263786316000090/1-s2.0-S0263786316000090-main.pdf?_tid=95339578-4ee7-11e7-b1d7-00000aacb360&acdnat=1497214400_069e8a93301ab2ebc0c555cc5d730e88. 44
- [65] Natasha K. Bowen e Shenyang Guo: *Structural Equation Modeling*. 45, 46
- [66] Hair, Joseph F. Jr., G. Tomas M. Hult, Christian Ringle e Marko Sarstedt: *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)*, volume 46. 2014, ISBN 9781452217444. <http://linkinghub.elsevier.com/retrieve/pii/S0024630113000034>. 46, 48, 49, 50, 51

- [67] Hair, Joe F., Marko Sarstedt, Christian M. Ringle e Jeannette A. Mena: *An assessment of the use of partial least squares structural equation modeling in marketing research*. Journal of the Academy of Marketing Science, 40(3):414–433, 2012, ISSN 00920703. 46
- [68] Sarstedt, Marko, Christian M Ringle, Donna Smith, Russell Reams e Joseph F Hair: *Partial least squares structural equation modeling (PLS-SEM): A useful tool for family business researchers*. Journal of Family Business Strategy, 5:105–115, 2014. http://ac-els-cdn-com.ez54.periodicos.capes.gov.br/S1877858514000060/1-s2.0-S1877858514000060-main.pdf?_tid=d708916c-433c-11e7-8130-00000aab0f26&acdnat=1495931604_25de4b6a3a2ce9ebf6aebf287ef51f73. 46
- [69] Vinzi, Vincenzo Esposito, Laura Trinchera e Silvano Amato: *Handbook of Partial Least Squares*. 2010, ISBN 978-3-540-32825-4. <http://link.springer.com/10.1007/978-3-540-32827-8>. 47, 48, 49, 50, 51, 52
- [70] Kwong, Ken e Kay Wong: *Partial Least Squares Structural Equation Modeling (PLS-SEM) Techniques Using SmartPLS*. Marketing Bulletin Technical Note, 24(1), 2013. 47
- [71] Oliver Gotz, Kerstin Liehr Gobbers e Manfred Krafft: *Evaluation of Structural Equation Models Using the Partial Least Squares (PLS) Approach*. Handbook of Partial Least Squares: Concepts, Methods and Applications, 2010. 48, 49, 50
- [72] Pinto, Inês Cância Reis: *Implementação do algoritmo PLS-SEM em R*, 2017. 48, 49
- [73] Costigliola, Francesco: *Partial Least Square – Path Modeling: metodologia, software e aplicação*. 2010. <http://run.unl.pt//handle/10362/8818>. 49
- [74] Cronbach, Lee J.: *Coefficient Alpha and the Internal Structure of Tests*. Psychometrika, 16(3):297–334, 1951, ISSN 00333123. 49
- [75] Joseph F. Hair, William C. Black, Barry J. Babin Rolph E. Anderson: *Multivariate Data Analysis*, volume 7th Editio. 2009. 52, 67
- [76] ABPMP: *Guia para o Gerenciamento de Processos de Negócio BPM CBOOK 3.0*, volume 53. Association of Business Process Professionals, 2013, ISBN 9788578110796. 52, 53
- [77] ISO/IEC/IEEE: *ISO / IEC / IEEE 15939*. 2017, 2017. 53
- [78] Staron, Mirosław e Wilhelm Meding: *MeSRAM – A method for assessing robustness of measurement programs in large software development organizations and its industrial evaluation*. Journal of Systems and Software, 113:76–100, março 2016, ISSN 01641212. <http://linkinghub.elsevier.com/retrieve/pii/S0164121215002368>. 53
- [79] Staron, Mirosław, Wilhelm Meding, Kent Niesel e Alain Abran: *A Key Performance Indicator Quality Model and Its Industrial Evaluation*. 2016. 53

- [80] Camargo, Leonidas Lopes de: *USO DE INDICADORES DA QUALIDADE PARA O GERENCIAMENTO ESTRATÉGICO DE EMPRESAS DO RAMO COMERCIAL*. 2000. <https://repositorio.ufsc.br/bitstream/handle/123456789/78947/177321.pdf?sequence=1>. 54
- [81] OGC: *Continual Service Improvement*. 2011, ISBN 9780113310494. http://link.springer.com/10.1007/978-0-387-77393-3_6. 54
- [82] Gil, Antonio Carlos: *Métodos e técnicas de pesquisa social*, volume 264. 1999, ISBN 9788522451425. 55, 56, 58
- [83] Lakatos, Eva M. e Marina A. Marconi: *Técnicas de pesquisa: planejamento e execução de pesquisas, amostragens e técnicas de pesquisas, elaboração, análise e interpretação de dados*. 2002, ISBN 9788562894145. 55
- [84] Simon, Herbert a: *The sciences of the artificial, (third edition)*, volume 33. 1997, ISBN 9780262691918. <http://linkinghub.elsevier.com/retrieve/pii/S0898122197829410>. 55
- [85] Jones, S R: *An introduction to power and sample size estimation*. Emergency Medicine Journal, 20(5):453–458, 2003, ISSN 1472-0205. <http://emj.bmj.com/cgi/doi/10.1136/emj.20.5.453>. 56
- [86] Cohen, Jacob: *Statistical Power Analysis*. Current Directions in Psychological Science, 1(3):98–101, 1992, ISSN 14678721. 56, 57
- [87] Bussab, Wilton de Oliveira e Pedro Alberto Morettin: *Estatística Básica*. Editora Saraiva, São Paulo, 8ª edição, 2013, ISBN 9788502207998. 56
- [88] 5.1.1 - Erros cometidos nos testes de hipóteses - Inferência / Portal Action. <http://www.portalaction.com.br/inferencia/511-erros-cometidos-nos-testes-de-hipoteses>. 57
- [89] American Statistical Association: *G Power 3.1 manual*. American Statistical Association, 76:27–32, 2017, ISSN 19134126. http://www.gpower.hhu.de/fileadmin/redaktion/Fakultaeten/Mathematisch-Naturwissenschaftliche_Fakultaet/Psychologie/AAP/gpower/GPowerManual.pdf. 57
- [90] Chow, Tsun e Dac Buu Cao: *A survey study of critical success factors in agile software projects*. Journal of Systems and Software, 81(6):961–971, 2008, ISSN 01641212. <http://linkinghub.elsevier.com/retrieve/pii/S0164121207002208>. 59
- [91] Joslin, Robert e Ralf Müller: *Relationships between a project management methodology and project success in different project governance contexts*. JPMA, 33:1377–1392, 2015. http://ac-els-cdn-com.ez54.periodicos.capes.gov.br/S0263786315000551/1-s2.0-S0263786315000551-main.pdf?_tid=7ed5f5ba-5373-11e7-9830-00000aacb35d&acdnat=1497714297_747c0d8d3cbf9393a84f329fa3ba0af8. 60

- [92] John Hulland: *Use of partial least squares (PLS) in strategic management research: A review of four recent studies*. Strategic Management Journal, 20(2):195, 1999, ISSN 01432095. 63
- [93] Joseph F. Hair Jr, William C. Black, Barry J Babin Rolph E. Anderson: *Multivariate Data Analysis*, 2010. 63
- [94] Theo K. Dijkstra: *Latent Variables and Indices: Herman Wold's Basic Design and Partial Least Squares*. Em *Handbook of Partial Least Squares Concepts, Methods and Applications*, páginas 23–46. 2010. 64
- [95] Chin, Wynne W.: *The Partial Least Squares Approach to Structural Equation Modeling*. https://www.researchgate.net/profile/Wynne_Chin/publication/232569511_The_Partial_Least_Squares_Approach_to_Structural_Equation_Modeling/links/0deec533e0f7c00f59000000/The-Partial-Least-Squares-Approach-to-Structural-Equation-Modeling.pdf. 66
- [96] VersionOne: *12th Annual State of Agile Report*, 2018. <https://www.versionone.com/about/press-releases/12th-annual-state-of-agile-survey-open/>. 69
- [97] Stamatis, D. H.: *Failure Mode and Effect Analysis: FMEA From Theory to Execution*. 2ª edição, 2003, ISBN 0873895983. <http://amstat.tandfonline.com/doi/abs/10.1080/00401706.1996.10484424>. 76
- [98] Mahnič, Viljan e Tomaž Hovelja: *On using planning poker for estimating user stories*. Journal of Systems and Software, 85(9):2086–2095, 2012, ISSN 01641212. 77
- [99] Dalkey, Norman: *The Delphi Method: An Experimental Study of Group Opinion.*, 1969, ISBN 0016-3287. ISSN 00163287. <http://linkinghub.elsevier.com/retrieve/pii/S001632876980025X>. 77
- [100] Moløkken-Østvold, Kjetil, Nils Christian Haugen e Hans Christian Benestad: *Using planning poker for combining expert estimates in software projects*. Journal of Systems and Software, 81(12):2106–2117, 2008, ISSN 01641212. <http://dx.doi.org/10.1016/j.jss.2008.03.058>. 77
- [101] Grenning, James: *Planning poker or how to avoid analysis paralysis while release planning*. Hawthorn Woods: Renaissance Software Consulting, (April):1–3, 2002. <http://www.renaissancesoftware.net/files/articles/PlanningPoker-v1.1.pdf>. 77
- [102] Schubert, Florian, Jörg Henseler e Theo K. Dijkstra: *Confirmatory Composite Analysis*. European Journal of Information Systems, 9(December):1–14, 2018, ISSN 1664-1078. 93

Apêndice A

Questionários acerca de fatores e critérios de insucesso em projetos

Os decisores foram questionados sobre cada um dos itens indicados nas Tabelas abaixo e, para cada um deles deveriam responder de acordo com uma escala de Likert de 05 intens, que iam desde Não tem impacto algum (refletindo o o menor valor da escala, ou seja, 1) a Está diretamente relacionado (é indissociável)(refletindo o maior valor da escala, ou seja, 5). Dessa forma foi possível coletar o sentimento de cada decisor acerca da influência de cada fator e critério de insucesso em projetos.

Tabela A.1: Fatores de insucesso - Organizacional

ID	Fator
FIO1	Falta de patrocínio dos executivos
FIO2	Falta de comprometimento da gestão
FIO3	Cultura organizacional muito tradicional
FIO4	Cultura organizacional muito política
FIO5	Capacidades locais adequadas não estarem disponíveis
FIO6	Ambiente social desfavorável
FIO7	Organograma não é claro e as tarefas não são bem definidas
FIO8	Ambiente organizacional desfavorável
FIO9	Urgência da necessidade
FIO10	Influências ambientais
FIO11	A missão do projeto (propósito) não está claramente indicada
FIO12	Organização muito grande
FIO13	Falta de compensação para funcionários
FIO14	Ambiente político
FIO15	Dono do projeto não está claramente definido
FIO16	Falta de logística

Tabela A.2: Fatores de insucesso - Pessoas

ID	Fator
FIPe1	Falta das habilidades necessárias dos colaboradores
FIPe2	Falta de competência em gestão de projetos
FIPe3	Falta de esforço do time
FIPe4	Resistência de grupos individuais
FIPe5	Mau relacionamento com cliente
FIPe6	Falta na qualidade de liderança do Gerente de Projeto
FIPe7	Equipe de projeto não qualificada
FIPe8	Falta de construção / motivação efetiva da equipe
FIPe9	Falta de consulta regular com os principais interessados
FIPe10	Falta de apoio das partes interessadas
FIPe11	Falta de competências de designers de projetos
FIPe12	Falta de compromisso do usuário final
FIPe13	Falta de envolvimento do usuário / cliente
FIPe14	Desempenho ruim de consultores
FIPe15	Pessoal insuficiente
FIPe16	Falta de apoio da alta gerência ao Gerente de Projetos e equipe de projeto no site
FIPe17	Gerente de projeto incompetente
FIPe18	Falta de uma administração equitativa de pessoal
FIPe19	O líder do projeto não possui habilidades administrativas adequadas
FIPe20	O líder do projeto não possui habilidades interpessoais adequadas
FIPe21	Não há desenvolvimento de funcionários (treinamento)
FIPe22	Não há comprometimento de todas as partes no projeto
FIPe23	Falta de apoio ao patrocinador do projeto
FIPe24	Falta de apoio da alta administração

Tabela A.3: Fatores de insucesso - Processos

ID	Fator
FIPr1	Falta de monitoramento e controle efetivo
FIPr2	Procedimentos não formalizados
FIPr3	Falta de uma adequada gestão de riscos (identificação / avaliação / monitoramento e controle)
FIPr4	Falta de flexibilidade na gestão
FIPr5	Canais de comunicação não são claros
FIPr6	Falta de boa comunicação / feedback
FIPr7	Falta de entendimento e aceitação dos planos de projetos
FIPr8	Falta de autonomia do Gerente de Projeto
FIPr9	Âmbito e natureza do trabalho não são bem definidos
FIPr10	Falta de capacidade técnica do Gerente de Projetos
FIPr11	Desvios do plano não são claramente e periodicamente tratados
FIPr12	As reuniões de projetos não garantem o fluxo eficiente de informações
FIPr13	Não há resolução efetiva de conflitos
FIPr14	Não há gerenciamento efetivo de mudanças
FIPr15	Não há delegação de autoridade
FIPr16	Não existe orçamento detalhado do projeto
FIPr17	Cronograma não é realista
FIPr18	Os objetivos e os termos do projeto não podem ser alterados se as condições o fizerem necessário
FIPr19	Políticas não são claras para sustentar as atividades e resultados do projeto
FIPr20	A data final da conclusão do projeto não está claramente definida
FIPr21	Não existe unicidade das atividades do projeto
FIPr22	Escopo mal definido
FIPr23	Requisitos mal definidos
FIPr24	Planejamento mal feito
FIPr25	Falta de mecanismo de monitoramento de processos ágeis
FIPr26	Falta da presença do cliente
FIPr27	Papel do cliente mal definido

Tabela A.4: Fatores de insucesso - Técnico

ID	Fator
FIT1	Desconhecimento de práticas / metodologia de gestão ágil de projetos
FIT2	Ferramentas e tecnologia inapropriadas
FIT3	Escolha correta da metodologia / ferramenta
FIT4	Experiência passada não relevante
FIT5	Objetivos e finalidade não aceitos por todos os envolvidos
FIT6	Falta de administração de orçamento
FIT7	A execução do projeto não está em conformidade com o plano acordado
FIT8	Plano de projeto não é atualizado regularmente
FIT9	Não existe uma estrutura da organização do projeto
FIT10	Mal desempenho dos contratados
FIT11	Duração do projeto (3 anos ou mais)
FIT13	Não há um report periódico com patrocinador
FIT14	Não há atualização regular do orçamento
FIT15	Decisões do proprietário ou do gestor não são tomadas de forma tempestiva
FIT16	Os dados necessários para as operações do sistema não estão disponíveis
FIT17	Falta de competências da equipe de gerenciamento de projetos

Tabela A.5: Critérios de insucesso - Organizacional

ID	Fator
CSO1	Não produzir benefícios organizacionais
CSO2	Gerar uma mudança mínima na organização
CSO3	Não afetar a melhoria da capacidade organizacional
CSO4	Não alcançar objetivos organizacionais
CSO5	Os impactos do projeto nos beneficiários não são visíveis

Tabela A.6: Critérios de insucesso - Pessoas

ID	Fator
CSPe1	Não propiciar satisfação do cliente
CSPe2	Não propiciar satisfação da equipe
CSPe3	Não propiciar satisfação do patrocinador
CSPe4	Não propiciar satisfação do usuário
CSPe5	Não proporciona ganho de conhecimento a equipe
CSPe6	Os recursos não foram mobilizados e usados conforme planejado
CSPe7	O projeto não satisfaz as necessidades dos usuários
CSPe8	Não há satisfação da direção da organização

Tabela A.7: Critérios de insucesso - Processos

ID	Fator
CSPr1	Terminador fora do custo estimado
CSPr2	Completado em desacordo com as especificações
CSPr3	Terminar fora no prazo
CSPr4	Projeto não atingir seus propósitos
CSPr5	Não atingir os requisitos do cliente
CSPr6	Produto não é utilizado conforme planejado
CSPr7	Handover complexo dos produtos
CSPr9	Não aderente aos procedimentos definidos
CSPr10	Não produz recompensas financeiras pessoais
CSPr11	Não atinge dos critérios de qualidade

Tabela A.8: Critérios de insucesso - Técnico

ID	Fator
CST1	Não proporcionar projetos futuros
CST2	Não ser motivado por projetos futuros
CST3	O projeto não ter boa aceitação
CST4	Atividades não foram realizadas como agendadas
CST5	Não está em conformidade com as normas ambientais
CST6	Projeto ter alto número de mudanças de escopo

Tabela A.9: Critérios de sucesso em projetos

ID	Fator
SP1	Qualidade (entregar o produto ou projeto de acordo com os critérios de qualidade requeridos pelo cliente)
SP2	Escopo (alcançar todos os requisitos e objetivos propostos)
SP3	Escopo (o produto se comporta da forma como o cliente espera)
SP4	Tempo (entregar o projeto no prazo estabelecido)
SP5	Custo (entregar o projeto dentro do orçamento previsto)
SP6	A adoção de uma metodologia e processos de gestão bem definidos possibilitam um gerenciamento adequado do portfólio e do programa, e aumenta a eficiência na realização do plano de metas integrado na estratégia da organização através dos projetos.
SP7	A existência de uma metodologia de gestão de projetos na organização favorece a redução no número de projetos malsucedidos ou cancelados por não adequação estratégica.
SP8	O desempenho e os resultados dos projetos são perceptíveis, considerados satisfatórios e melhorados com a adoção de metodologias bem definidas em projetos