



Pós-Graduação em Ciência da Computação

Uma extensão do Scrum com foco na criação de valor

Por

José Wallace Ribeiro de Macedo Junior

Dissertação de Mestrado Profissional



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

RECIFE, ABRIL/2013



Universidade Federal de Pernambuco
CENTRO DE INFORMÁTICA
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

José Wallace Ribeiro de Macedo Junior

Uma extensão do Scrum com foco na criação de valor

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

Orientador: Professor Dr. Hermano Perrelli de Moura

RECIFE, ABRIL/2013

Catálogo na fonte
Biblioteca Jane Souto Maior, CRB4-571

Macedo Junior, José Wallace Ribeiro de
Uma extensão do SCRUM com foco na criação de valor /
José Wallace Ribeiro de Macedo Junior. - Recife: O Autor,
2013.

102 f.: il., fig., tab.

Orientador: Hermano Perrelli de Moura.
Dissertação (mestrado profissional) - Universidade Federal de
Pernambuco. CIn, Ciência da Computação, 2013.

Inclui referências e apêndice.

1. Engenharia de software. 2. Gerenciamento de projetos. I.
Moura, Hermano Perrelli de (orientador). II. Título.

005.1

CDD (23. ed.)

MEI2013 – 129

Dissertação de Mestrado Profissional apresentada por **José Wallace Ribeiro de Macedo Junior** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título, “Uma extensão do Scrum com foco na criação de valor”, orientada pelo **Professor Hermano Perrelli de Moura** e aprovada pela Banca Examinadora formada pelos professores:

Prof. Alexandre Marcos Lins de Vasconcelos
Centro de Informática / UFPE

Prof. José Gilson de Almeida Teixeira Filho
Universidade de Pernambuco

Prof. Hermano Perrelli de Moura
Centro de Informática / UFPE

Visto e permitida a impressão.
Recife, 18 de abril de 2013.

Prof^a. EDNA NATIVIDADE DA SILVA BARROS
Coordenadora da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.

AGRADECIMENTOS

À minha mulher, Larissa, pela dedicação e compreensão durante os momentos em que estive ausente, muitas vezes abdicando de fazer suas próprias atividades.

À minha filha, Mariana, por alegrar nossas vidas todos os dias.

Aos meus pais, pelo o amor, dedicação e carinho que sempre tiveram por mim. Se eu consegui chegar até aqui foi graças a vocês.

À minha irmã, Kelly, por ter colaborado para que eu tivesse tempo para concluir esse trabalho.

Ao meu orientador, professor Dr. Hermano Perrelli de Moura, pela disposição, orientação e compreensão durante a elaboração deste trabalho.

Aos amigos Cézar Corado, Gleydson Vilanova e Laerte Rodrigues, parceiros desde o início. Sem vocês esta jornada teria sido bem mais difícil.

Aos colegas de trabalho, pela compreensão e apoio nos momentos em que tive que me ausentar para assistir aulas do mestrado.

E, por fim, a Deus, por ter iluminado meu caminho e dado saúde e sabedoria para que eu atingisse meus objetivos.

Livros não mudam o mundo, quem muda o mundo são as pessoas. Os livros só mudam as pessoas.

Mario Quintana

Resumo

A maioria dos estudos sobre fatores críticos de sucesso aponta que projetos de software falham por não entregar valor aos interessados. Uma das áreas de pesquisa da Engenharia de Software é a Engenharia de Software Baseada em Valor – VBSE, a qual busca incorporar atividades de criação de valor nos processos de desenvolvimento de software. Segundo ela, o sucesso de um software vai mais além do que uma simples adequação a modelos técnicos. Excelência técnica é altamente desejável, mas insuficiente. Nos últimos anos, a utilização de métodos ágeis tem crescido de forma acentuada, sendo que, segundo pesquisas recentes, Scrum é o método ágil mais utilizado. Estes métodos têm como um de seus princípios a entrega de valor ao cliente. Desta forma, este trabalho analisou a aderência do Scrum à VBSE. A análise foi realizada tendo como base os princípios e valores definidos no manifesto ágil e os elementos-chave que formam a base da VBSE. O resultado da análise mostrou que o Scrum já é aderente a alguns desses elementos-chave. Entretanto, outros elementos-chave da VBSE não são abordados. Para preencher essas lacunas, foi proposta uma extensão que consiste na criação de novos artefatos e no acréscimo de atividades ao processo do Scrum. Tal extensão visa contribuir para que os softwares entregues criem valor para os interessados, sem descaracterizar a agilidade do Scrum. Como método de avaliação, foi aplicado um questionário a profissionais com experiência em Scrum a fim de verificar o grau de concordância em relação à extensão proposta. O resultado da avaliação mostrou que maioria dos respondentes concorda com a extensão e considera que a criação de valor para os interessados deve ser o objetivo principal de um projeto de software.

Palavras-chave: Engenharia de Software Baseada em Valor - VBSE, Criação de Valor, Gerenciamento Ágil de Projetos, Scrum.

Abstract

Most of studies regarding critical success factors points that in general software projects fail because they don't deliver value to stakeholders. One of the areas of Software Engineering is the Value-Based Software Engineering – VBSE, which aims to incorporate activities that bring value into software process development. According to VBSE, the success of software goes beyond a simple adaptation to technical models. Technical excellence is highly desirable, but insufficient. In recent years, the use of agile methods has grown markedly, and, according to recent research, Scrum is the most used agile method. These methods have as one of their principles delivering value to the customer. Thus, this study examined Scrum method adherence to VBSE. The analysis was based on the principles and values defined in the Agile Manifesto and the key elements that form the basis of VBSE. The result of the analysis showed that Scrum is already compliant to some of these key elements. However, other VBSE key elements are not addressed. To fill in these gaps, it was proposed an extension that consists in the creation of new artifacts and activities in addition to the Scrum process. This extension aims to contribute to the delivered software by creating value to the stakeholders, without mischaracterizing the Scrum's agility. As evaluation method, it was applied a questionnaire to experienced professionals in Scrum to verify the concordance degree regarding the proposed extension. The evaluation result showed that most participants agree with the extension and believe that the value creation for the stakeholders should be the primary objective of a software project.

Keywords: Value-Based Software Engineering - VBSE, Value Creation, Agile Project Management, Scrum.

Lista de Figuras

Figura 1 - Métodos ágeis mais utilizados (Version One, 2011)	15
Figura 2 - Teoria da VBSE. Adaptada de (BOEHM; JAIN, 2006)	20
Figura 3 - Cadeia de Resultado. Adaptada de (BOEHM, 2006c)	22
Figura 4 - Diagrama de conflito de modelos.....	23
Figura 5 - Processo de monitoramento e controle sugerido pela VBSE.....	27
Figura 6 - Visão geral do Scrum.....	34
Figura 7 - Sprint Burndown.....	41
Figura 8 - Visão geral da extensão.....	53
Figura 9 – Sugestão de modelo para o <i>Release Backlog</i>	58

Lista de Tabelas

Tabela 1 - Estrutura da pesquisa	16
Tabela 2 - Características do Product Owner. Adaptado de (RUBIN, 2012)	35
Tabela 3 - Extensões proposta no RUP-VC	42
Tabela 4 - Aderência do Scrum ao CMMI (Marçal, 2009)	43
Tabela 5 - Resumo da análise da aderência do Scrum aos elementos-chave da VBSE.....	51
Tabela 6 - Perfil dos respondentes quanto à escolaridade e experiência com projetos de software.....	63
Tabela 7 - Perfil dos respondentes quanto à experiência em projetos Scrum	63
Tabela 8 - Perfil dos respondentes quanto à obtenção de certificação	64
Tabela 9 - Perfil dos respondentes quanto aos papéis desempenhados no Scrum..	64
Tabela 10 - Criação de valor como fator crítico de sucesso.....	65
Tabela 11 - Criação do artefato <i>Business Case</i>	67
Tabela 12 - Criação do artefato <i>Action Backlog</i>	70
Tabela 13 - Criação do artefato <i>Release Backlog</i>	73
Tabela 14 - Resultado da poposição que aborda o gerenciamento explícito de riscos	75
Tabela 15 - Resultado da poposição que aborda o monitoramento da realização de benefícios.....	77
Tabela 16 - Resultado da proposição que avalia se a extensão proposta não descaracteriza o Scrum.....	79
Tabela 17 - Avaliação geral da Extensão	82

Sumário

1. INTRODUÇÃO	13
1.1. Contexto e Motivação.....	13
1.2. Objetivos	15
1.3. Estrutura da Pesquisa	16
1.4. Organização do Trabalho.....	16
2. FUNDAMENTAÇÃO TEÓRICA.....	17
2.1. Definição de Valor	17
2.2. Engenharia de Software Baseada em Valor – VBSE.....	18
2.2.1. Teoria da VBSE	19
2.2.2. Elementos-Chave	20
2.3. Gerenciamento Ágil de Projetos.....	29
2.4. Scrum.....	32
2.4.1. Ciclo de Vida.....	33
2.4.2. Papéis.....	34
2.4.3. Artefatos	37
2.4.4. Eventos.....	39
2.5. Trabalhos relacionados e correlacionados.....	41
2.6. Considerações sobre o referencial teórico	43
3. ANÁLISE DA ADERÊNCIA DO SCRUM À VBSE.....	45
3.1. Análise da Realização de Benefícios	45
3.2. Identificação e Conciliação das Proposições de Valor	46
3.3. Análise de Caso de Negócio	47
3.4. Gerenciamento Contínuo de Riscos e Oportunidades	48
3.5. Engenharia Simultânea Sistemas e Software	49
3.6. Monitoramento e Controle Baseados em Valor.....	50
3.7. Mudança como Oportunidade	50

3.8. Considerações sobre a Análise.....	51
4. UMA PROPOSTA DE EXTENSÃO AO SCRUM COM FOCO NA CRIAÇÃO DE VALOR	53
4.1. Ponto de Extensão 1 - Criação do artefato <i>Business Case</i>	53
4.1.1. Resumo do Projeto	54
4.1.2. Contexto Estratégico	55
4.1.3. Partes Interessadas e Proposições de Valores	55
4.1.4. Riscos.....	55
4.1.5. Viabilidade Financeira	56
4.1.6. Restrições.....	56
4.2. Ponto de Extensão 2 - Criação do Artefato <i>Action Backlog</i>	56
4.3. Ponto de Extensão 3 - Criação do Artefato <i>Release Backlog</i>	57
4.4. Ponto de Extensão 4 - Monitorar Riscos	58
4.5. Ponto de Extensão 5 - Monitorar a Criação de Valor	59
5. AVALIAÇÃO DA EXTENSÃO PROPOSTA.....	60
5.1. Objetivo da Avaliação.....	60
5.2. Metodologia da avaliação.....	60
5.3. Elaboração do Questionário	60
5.3.1. Elaboração das Perguntas	61
5.3.2. Teste do Questionário	61
5.4. Execução	62
5.5. Análise e Interpretação dos Resultados.....	62
5.5.1. Perfil dos Respondentes.....	62
5.5.2. Análise das Proposições	65
5.5.3. Considerações Sobre a Análise.....	83
6. CONCLUSÃO	85
6.1. Principais Contribuições.....	85
6.2. Trabalhos Futuros	86

6.3. Considerações Finais.....	86
REFERÊNCIAS.....	88
APÊNDICE A – QUESTIONÁRIO DE AVALIAÇÃO	93
APÊNDICE B – PERFIL DOS RESPONDENTES.....	101

1. INTRODUÇÃO

Este capítulo apresenta o contexto e a motivação para o desenvolvimento deste trabalho, bem como seus objetivos, estrutura da pesquisa e organização.

1.1. Contexto e Motivação

Atualmente, organizações de todos os segmentos enfrentam uma forte concorrência e têm que lidar com clientes cada vez mais exigentes. Diante deste contexto, tais organizações buscam constantemente aumentar a eficiência de suas operações e satisfazer as necessidades de seus clientes. Para isso, o uso de recursos de Tecnologia da Informação (TI) tem se mostrado essencial. Esses recursos incluem softwares, hardwares e redes de comunicação utilizados para gerenciar informação (OLUGBODE; RICHARDS; BISS, 2007). Organizações bem sucedidas reconhecem os benefícios de investir em TI e a utiliza para direcionar valores às partes interessadas do negócio (ISACA, 2012).

Dentre os recursos de TI, o software desempenha papel fundamental na construção de sistemas de informação. Conforme a importância do software cresce, os projetos de software tornam-se cada vez mais críticos para as organizações. No entanto, muitos projetos de software ainda falham. A maioria dos estudos sobre fatores críticos de sucesso aponta que os projetos de software falham por não entregarem valor aos interessados (BOEHM, 2006a). Um estudo sobre quais caminhos a pesquisa sobre gerenciamento de projetos deve seguir concluiu que o gerenciamento de projetos deve ter como foco a criação de valor (WINTER; SMITH; MORRIS, 2006). Uma das áreas de pesquisa da Engenharia de Software é a Engenharia de Software Baseada em Valor (VBSE)¹, que tem como finalidade incorporar considerações de valor nos processos de desenvolvimento de software (BOEHM, 2006a). De acordo com a VBSE, a maioria das práticas adotadas pelos processos de desenvolvimento de software não leva em consideração a criação de valor, onde:

- Artefatos são tratados com a mesma importância;

¹ VBSE é um acrônimo para *Value-Based Software Engineering*

- Métodos de mensuração são baseados em custo e cronograma, e não no valor criado;
- Existe uma separação de conceitos, onde a responsabilidade do engenheiro de software é restrita a transformar requisitos em código, não tendo preocupação se o software entregue irá criar para os interessados.

O uso de métodos ágeis tem crescido nos últimos anos. Segundo o último relatório *CHAOS Report*² publicado em 2010, o uso desses métodos cresce a uma taxa de 22%³ (STANDISH GROUP, 2010). Esse crescimento está associado a um aumento na taxa de projetos executados com sucesso. Vários métodos ágeis têm sido propostos: *Adaptive Software Development* (ASD), *Dynamic Systems Development Method* (DSDM), *Crystal Methods* (CM), *Extreme Programming* (XP), *Feature Driven Development* (FDD), *Lean Development* (LD) e Scrum (HIGHSMITH, 2002, p. xvii). De acordo com uma pesquisa realizada em 2011 (VERSIONONE, 2011), Scrum é o método ágil mais utilizado, seguido por uma combinação de Scrum e XP. A Figura 1 apresenta o resultado desta pesquisa.

Analisando o exposto até aqui é possível observar que:

- Projetos de software falham por não entregar valor aos interessados;
- O gerenciamento de projetos com foco na criação de valor é considerado um tema relevante de pesquisa;
- A criação de valor é um fator crítico para o sucesso de um projeto. Neste sentido, a VBSE propõe uma abordagem para integrar considerações de valor em processos de desenvolvimento de software;
- O uso de métodos ágeis cresce de forma acentuada, associado ao crescimento da taxa de projetos concluídos com sucesso. Scrum é o método ágil mais utilizado, seguido por uma combinação entre Scrum e XP;

Com base nas observações acima, surgiu a motivação para analisar como o Scrum aborda a criação de valor, tendo como base os elementos-chave definidos pela VBSE. O resultado da análise será utilizado para propor uma extensão que contribua para que os softwares entregues criem valor para os interessados. Assim,

² Relatório publicado periodicamente que avalia a taxa de sucesso de projetos de software

³ Calculado através do método *Compound Annual Growth Rate*.

o tema deste trabalho é a criação de valor em projetos de software, mais especificamente, projetos que utilizam o método ágil Scrum. O contexto deste trabalho abrange organizações que desenvolvem software, incluindo organizações com e sem fins lucrativos, públicas e privadas, tendo TI com atividade-fim ou não.

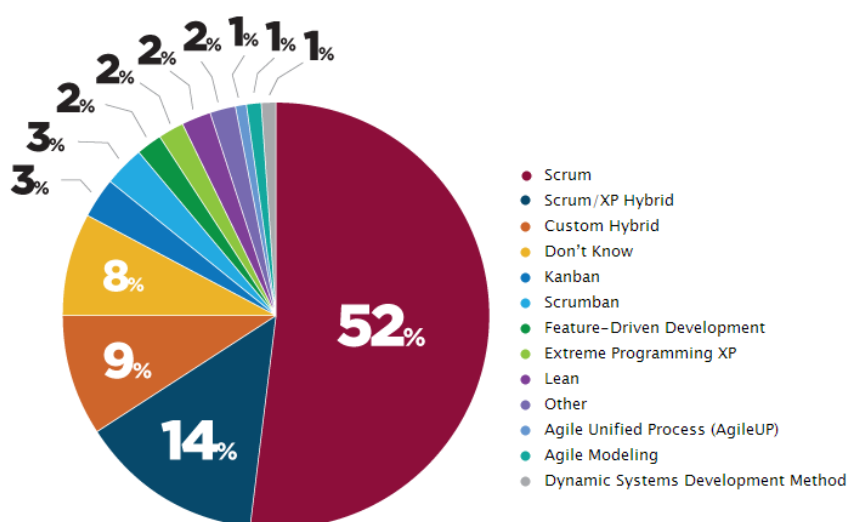


Figura 1 - Métodos ágeis mais utilizados (Version One, 2011)

1.2. Objetivos

Este trabalho tem como objetivo propor uma extensão do Scrum que contribua para que os softwares entregues criem valor para os interessados, sem tirar a característica ágil deste método e tendo como base a Engenharia de Software Baseada em Valor. Os objetivos específicos deste trabalho são:

- Estudar o método Scrum e investigar como ocorre a criação de valor em projetos de software que utilizam este método;
- Analisar a aderência do Scrum aos princípios e elementos-chave da VBSE;
- Propor uma extensão do Scrum, adicionando, quando necessário, elementos de criação de valor com base nos elementos-chave da VBSE;
- Avaliar a extensão proposta através da aplicação de um questionário a profissionais e pesquisadores com experiência em projetos Scrum.

1.3. Estrutura da Pesquisa

Este trabalho se divide em quatro fases. A Tabela 1 apresenta as atividades realizadas em cada fase da pesquisa.

Tabela 1 - Estrutura da pesquisa

Fases da pesquisa	Atividades
Fase 1: Definição do escopo do trabalho	Selecionar um tema dentro da área de interesse; Definir um tópico dentro do tema escolhido.
Fase 2: Pesquisa bibliográfica	Realizar pesquisa bibliográfica dos seguintes assuntos relacionados ao trabalho: Criação de valor em projetos; Engenharia de Software Baseada em Valor (VBSE); Métodos ágeis, mais especificamente o método Scrum.
Fase 3: Análise e Extensão	Analisar o Scrum com base nos elementos-chave da VBSE; Identificar pontos de extensão; Propor uma extensão com base nos pontos identificados.
Fase 4: Avaliação da extensão	Avaliar a extensão proposta através da aplicação de um questionário a profissionais e pesquisadores com experiência em Scrum.

1.4. Organização do Trabalho

Com relação à organização, além deste capítulo introdutório, esta dissertação está dividida em seis capítulos, descritos a seguir. No **Capítulo 2** é apresentada a fundamentação teórica e trabalhos referentes aos principais assuntos abordados neste trabalho. O **Capítulo 3** apresenta uma análise do Scrum com base nos elementos-chave da VBSE. O objetivo dessa análise é verificar a aderência do Scrum à VBSE. O **Capítulo 4** apresenta uma proposta de extensão do Scrum baseada na análise realizada no Capítulo 3. O **Capítulo 5** detalha o processo de avaliação e apresenta uma análise dos resultados obtidos. Por fim, o **Capítulo 6** apresenta a conclusão e sugestões para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta a fundamentação teórica referente aos principais assuntos abordados neste trabalho. A Seção 2.1 apresenta definições para o termo *valor* encontradas na literatura e apresenta a definição de *valor* utilizada neste trabalho. A Seção 2.2 apresenta a Engenharia de Software Baseada em Valor. A Seção 2.3 apresenta conceitos de gerenciamento de projetos e gerenciamento ágil de projetos. A Seção 2.4 apresenta o método ágil Scrum. Por fim, a Seção 2.5 apresenta trabalhos relacionados a este trabalho.

2.1. Definição de Valor

Para entender como um projeto de software cria valor, é necessário ter uma clara definição do significado do termo *valor*. Na literatura são encontradas diversas definições. Muitas definições são puramente financeiras, como o “valor monetário de algo” (BOEHM, 2006a). Outras, oriundas do marketing, são baseadas no conceito de utilidade. Kotler (2000, p. 58) define valor como “a razão entre o que o cliente recebe e o que ele dá. O cliente recebe benefícios e assume custos”. Desta forma, o cliente perceberá valor quando os benefícios forem maiores que os custos. Jan e Muahammad (2010) afirmam que o valor de um produto pode ser definido como o seu valor de mercado, relacionado ao seu preço e suas vantagens sobre os concorrentes.

A VBSE define *valor* como “valor relativo, utilidade ou importância” (BOEHM, 2006a). Essa definição é bastante abrangente e enfatiza a relatividade do termo. Em um projeto, cada parte interessada possui suas próprias percepções de valor, as quais são influenciadas pelo contexto que cada uma se encontra e podem sofrer alterações com o tempo (SHARM; AURUM; PAECH, 2008). Além disso, em um contexto organizacional, deve-se levar em consideração os objetivos estratégicos da organização. Em outras palavras, o projeto deve contribuir para que a organização atinja seus objetivos, os quais podem não ser relacionados à TI.

Neste trabalho, o termo *valor* refere-se aos benefícios que uma parte interessada espera receber e aos custos que ela está disposta a assumir. Benefícios podem ser tangíveis, fáceis de quantificar financeiramente, como também podem ser

intangíveis, os quais podem ser difíceis de serem traduzidos em valores monetários (VETSCHERA, 2005). Melhorar a imagem da organização e aumentar da satisfação dos clientes são exemplos de benefícios intangíveis. Já os custos que uma parte interessada pode estar disposta a assumir incluem, além de custos monetários, custos de tempo, energia e psicológicos. Importante observar que essa definição é relativa. Cada parte interessada pode esperar diferentes benefícios e aceitar diferentes custos.

Em projetos de software, os benefícios esperados por alguns interessados são traduzidos em um conjunto de funcionalidades a serem desenvolvidas. É importante destacar que benefícios não se confundem com funcionalidades. Os benefícios ocorrem da utilização do produto e normalmente estão ligados aos objetivos do negócio. Por exemplo, uma organização que resolve desenvolver um sistema de análise de crédito pode esperar como benefícios reduzir a inadimplência e reduzir o tempo de concessão de crédito. Engenheiros de software devem identificar funcionalidades que, ao serem entregues, gerem os benefícios desejados. Além disso, os custos que um interessado está disposto a assumir devem ser identificados. Um software pode entregar uma funcionalidade que gera um benefício desejado por um interessado. No entanto, aguardar longos períodos de processamento pode ser um custo não aceitável. Desta forma, se o software não tiver bom desempenho, não irá criar valor, mesmo entregando o benefício. Assim, podemos definir a criação de valor como a entrega dos benefícios esperados pelos interessados a custos que eles estão dispostos a assumir.

2.2. Engenharia de Software Baseada em Valor – VBSE

O objetivo da engenharia de software é criar produtos, serviços e processos que criem valor. Pessoas envolvidas neste processo tentam maximizar alguma noção simples ou complexa de valor, consciente ou inconscientemente (BIFFL; AURUM, *et al.*, 2006, p. 8). Segundo Boehm (2006a), muitos dos métodos de engenharia de software não consideram a criação de valor, onde:

- Artefatos são tratados com a mesma importância;
- O controle é baseado em custos e cronogramas, e não em valor;

- Existe uma separação de conceitos, onde a responsabilidade do engenheiro de software é transformar requisitos em código, eximindo-se da responsabilidade de criar valor.

Todas as partes interessadas esperam receber algum benefício, seja tangível ou intangível, econômico ou social, monetário ou utilitário, ou mesmo ético ou antiético (BIFFL; AURUM, *et al.*, 2006, p. 8). Normalmente, engenheiros de software não entendem os objetivos de criação de valor. Raramente existe uma mensuração e entendimento de como um projeto de software cria valor. A dificuldade em mensurar valor é que não é possível medi-lo da mesma forma que medimos massa, volume e tempo. Isso porque é um termo relativo, e cada pessoa tem suas próprias definições de valor (BERRY; AURUM, 2006).

A VBSE é uma abordagem que visa adicionar considerações de valor em todas as fases da engenharia de software. De acordo com Berry e Aurum (2006), a VBSE é um framework para melhorar os sistemas entregues aos clientes que incorpora consideração de valor em decisões técnicas e gerenciais que são tomadas durante o desenvolvimento e manutenção do software. Segundo a VBSE, projetos de software são fundados em conceitos técnicos, não em termos de valor agregado. Excelência técnica é altamente desejável, mas insuficiente. Os mais lucrativos softwares do mundo são notórios em *bugs* e falhas (SULLIVAN; CHALASAN; SAZAWAL, 1998). As próximas seções apresentam a teoria na qual a VBSE é baseada e os elementos-chave que formam sua base.

2.2.1. Teoria da VBSE

A teoria que fundamenta a VBSE é apresentada na Figura 2. Ela é baseada no Teorema de Sucesso das Empresas (BOEHM; ROSS, 1989). Tal teorema afirma que: “uma empresa será bem sucedida se, e somente se, fizer vencedoras todas as partes interessadas críticas (SCS)⁴ para o sucesso”. Isso significa que o cenário ideal é uma situação onde todas as partes saiam vencedoras.

No entanto, o Teorema acima citado nada diz sobre como conseguir e manter tal cenário. Para isso, a teoria da VBSE se baseia também no Teorema da

⁴ Tradução do acrônimo *Success Critical Stakeholders*

Realização do Ganha-Ganha⁵, o qual afirma que para fazer vencedoras as partes envolvidas é necessário:

- **Identificar os interessados:** Esse item se relaciona com a teoria da dependência, a qual tem como principal técnica a Cadeia de Resultado, apresentada posteriormente na Seção 2.2.2.1;
- **Entender como cada SCS quer ganhar:** Esse item se relaciona com teoria da utilidade, a qual visa entender quais são as necessidades e valores das pessoas;
- **Negociar com SCSs:** Esse item se relaciona com a teoria da decisão;
- **Controlar o progresso com base na realização das proposições de valor dos SCSs:** Esse item se relaciona com a teoria do controle.

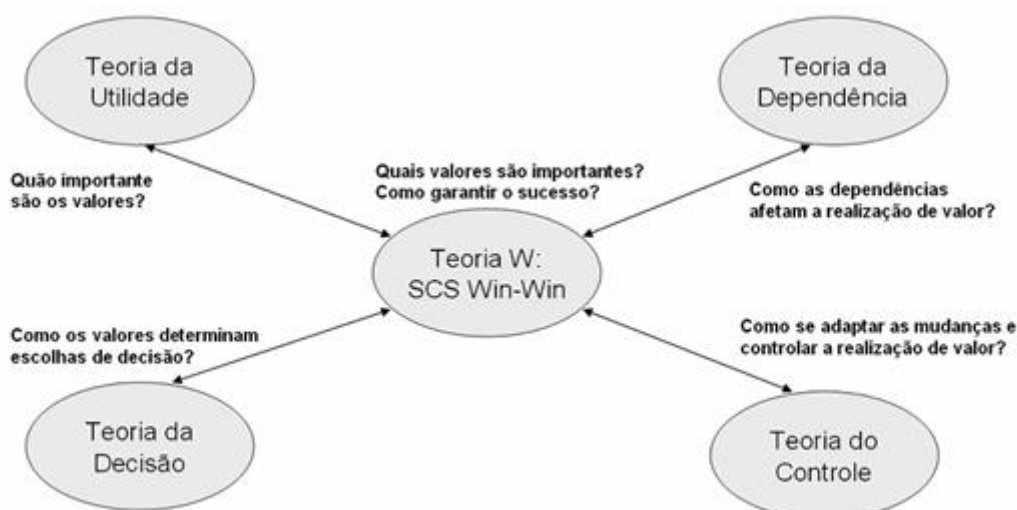


Figura 2 - Teoria da VBSE. Adaptada de (BOEHM; JAIN, 2006)

2.2.2. Elementos-Chave

Boehm (2006c) apresenta sete elementos-chave que formam a base da VBSE: Análise da Realização de Benefícios; Identificação e Conciliação das Proposições de Valor das Partes Interessadas; Análise do Caso de Negócio; Gerenciamento Contínuo de Riscos e Oportunidades; Engenharia Simultânea de Sistema e de Software; Monitoramento e Controle Baseado em Valor; Mudança como Oportunidade. As seções a seguir detalham cada um desses elementos.

⁵ Tradução livre para *Win-Win Achievement Theorem*

2.2.2.1. Análise da Realização de Benefícios

Cada vez mais, organizações investem em tecnologia da informação, o que por sua vez gera mais informações. No entanto, não existe uma clara ligação entre os investimentos em TI e os resultados do negócio. Thorp (2007) definiu esse problema como *Paradoxo da Informação*, o qual ele afirma não ser um desafio tecnológico, e sim um desafio do negócio.

Muitas empresas tratam investimentos em TI como a “bala de prata”, onde, uma vez feito o investimento, resultados serão automaticamente obtidos. Para contornar esse problema, Thorp e sua empresa desenvolveram uma abordagem chamada de Análise da Realização de Benefícios (ARB) (THORP, 2007). O objetivo dessa análise é identificar ações adicionais, de TI ou não, necessárias para que um investimento em TI gere os benefícios esperados (BOEHM, 2006c). Com a ARB, investimentos em TI passam a ser vinculados à produção de benefícios e aos resultados do negócio. A principal ferramenta da ARB é a Cadeia de Resultados⁶. Essa ferramenta vincula iniciativas que consomem recursos da organização às Contribuições e aos Resultados. A Figura 3 exemplifica o uso da cadeia de resultado em um sistema de pedidos. De acordo a VBSE, a cadeia de resultado pode ser utilizada para:

- Identificar iniciativas além do projeto de software necessárias para que o benefício do software seja realizado;
- Identificar partes interessadas que precisam ser representadas no projeto e devem compartilhar a visão do produto.

⁶ Tradução do original em inglês *Result Chain*.

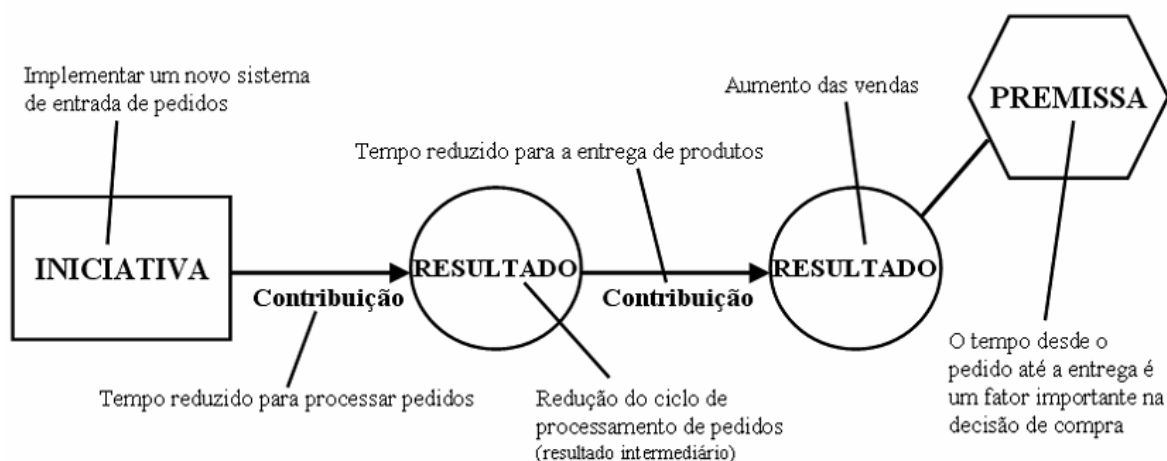


Figura 3 - Cadeia de Resultado. Adaptada de (BOEHM, 2006c)

2.2.2.2. Identificação e Conciliação das Proposições de Valores dos Interessados

Conforme a teoria apresentada na Seção 2.2.1, a VBSE é fundamentada na premissa que, para obter sucesso, um projeto deve criar valor para todas as partes interessadas críticas para o sucesso. Com base nessa premissa, nota-se que é essencial entender quais as proposições de valor de cada parte interessada do projeto. Boehm (2006c) afirma que o ideal seria se todas as proposições de valor fossem identificadas facilmente de modo que pudessem ser transformadas em requisitos do sistema.

No entanto, principalmente no início do projeto, esse cenário não é facilmente obtido. Além disso, também é necessário que as proposições de valores sejam compatíveis entre si. A Figura 4 apresenta um diagrama utilizado para identificação de conflitos. Neste diagrama, conhecido como diagrama de conflito de modelos⁷, identificamos as proposições de valores de cada grupo crítico de interessado e as confrontamos. Através desses confrontos podemos verificar quais proposições são conflitantes.

⁷ Traduzido do original em inglês *Model Clash Diagram*

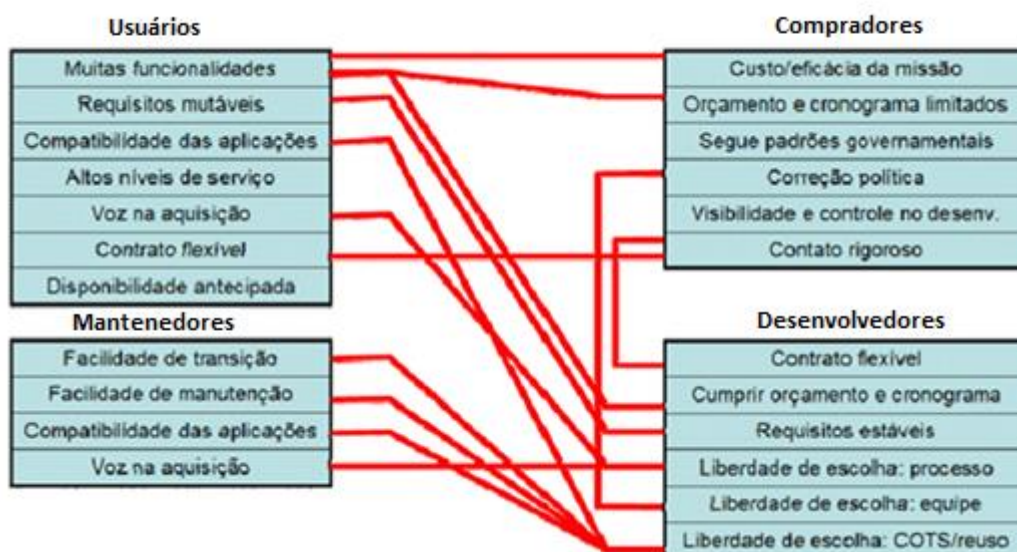


Figura 4 - Diagrama de conflito de modelos

Identificados os conflitos, a VBSE sugere algumas abordagens para conciliá-los:

- **Gerenciamento de expectativas:** essa abordagem visa divulgar os conflitos a todos as partes interessadas. Ele parte da premissa que, quando os todos os envolvidos tomam ciência dos conflitos, eles tendem a flexibilizar suas proposições;
- **Técnicas de visualização e análise de *trade-off*:** protótipos, cenários e modelos de estimativas fazem com que os envolvidos obtenham melhor entendimento de quais aspectos são mais importantes;
- **Priorização:** a priorização por parte dos interessados ajuda a identificar quais valores podem ser criados com os recursos necessários;
- **Groupware:** são ferramentas de colaboração que apoiam a realização de brainstorming, discussões e negociações de conflitos;
- **Análise de Caso de Negócio:** caso de negócio é um documento que contém justificativas para a execução do projeto, além de definir quais resultados são esperados. A apresentação dos resultados que se deseja obter ajuda na conciliação de proposições de valor conflitantes.

Lidar com proposições de valor de diferentes pessoas é fundamental para superar as limitações de abordagens que não consideram a criação de valor e,

assim, desenvolver softwares úteis às pessoas. Em um ambiente de criação de valor, a participação dos interessados é um fator crítico de sucesso do projeto. Durante as negociações, desenvolvedores aprendem mais sobre o mundo dos clientes e usuários, enquanto estes aprendem mais sobre o que é tecnicamente possível. Negociar é fundamental para conciliar conflitos. Normalmente as proposições de valores dos interessados variam consideravelmente. Aspectos políticos e econômicos podem influenciar no processo de negociação, fazendo com que algumas partes abusem de suas situações para buscarem objetivos pessoais (BOHEM, 2006d).

2.2.2.3. Análise do Caso de Negócio

De forma simplificada, a análise do Caso de Negócio envolve determinar os custos, benefícios e retorno sobre investimento (ROI)⁸ durante todo o ciclo de vida do sistema (BOEHM, 2006c). Khajavinia (2009) define um caso de negócio como um documento que contém informações relacionadas ao negócio, tais como partes interessadas e seus critérios de sucesso, escopo do produto ou serviço, análise de mercado e concorrentes, oportunidades de negócios, custos, benefícios e riscos. A VBSE afirma que o caso de negócio não pode ser restrito somente a aspectos financeiros. Dois fatores adicionais devem ser analisados: benefícios intangíveis; riscos e incertezas (BOEHM, 2006c). Um caso de negócio deve prover um mecanismo para justificar e guiar um projeto. Independente do tipo de projeto, as decisões devem ser baseadas no caso de negócio, sendo que este deve informar:

- Os objetivos do investimento;
- A quantidade de recursos (tempo e dinheiro);
- Os benefícios para o negócio;
- Os principais interessados que são fontes de requisitos.

Segundo Robertson (2004), um caso de negócio ajuda a descobrir os requisitos corretos e identificar partes interessadas, além de definir o escopo do produto ou projeto. Além disso, ele combina decisões de negócio com decisões de software. Segundo Khajavinia (2009), em muitas organizações engenheiros de software e executivos de negócio vivem em mundos separados, usando suas

⁸ Tradução do original em inglês *Return on Investment*

próprias terminologias, critérios de decisão e métodos e trabalho. Assim, muitos engenheiros de software agem como se fossem “artistas criativos”, estando interessados somente em criar novas funcionalidades. Neste cenário, um caso de negócio é uma boa ferramenta para criar uma ponte de comunicação entre o mundo dos negócios e o mundo da engenharia de software (BOEHM; SULLIVAN, 2000).

2.2.2.4. Gerenciamento Contínuo de Riscos e Oportunidades

De acordo com o guia PMBOK⁹ (PMI, 2008):

“O risco de um projeto é um evento ou condição incerta que, se ocorrer, terá um efeito positivo ou negativo sobre pelo menos um objetivo do projeto, como tempo, custo, escopo ou qualidade”.

Todo projeto tem algum risco associado, sendo o desenvolvimento relacionado a um novo software ou a uma nova versão de um software existente. O Instituto de Engenharia de Software (SEI)¹⁰ define risco como a possibilidade de sofrer perda, a qual pode ser uma diminuição da qualidade do software, aumento de custo, atraso no cronograma ou perda de *market share*. Risco por si só não é ruim. Ele é essencial para o progresso. Deve-se balancear o lado negativo do risco com os potenciais benefícios de suas oportunidades. Riscos comuns incluem:

- Novo processo de desenvolvimento;
- Requisitos técnicos;
- Restrições de clientes e mercado;
- Orçamentos e cronograma agressivos.

Segundo a VBSE, o gerenciamento de riscos deve abordar fatores relacionados às pessoas. Ela afirma que:

“Entender e abordar as proposições de valores das pessoas torna-se uma poderosa ferramenta na redução do risco de falha do projeto, ou, por uma visão complementar, aumenta a oportunidade de sucesso. Isso significa que a engenharia de software baseada em valor não é uma abordagem baseada em números, e sim um conjunto de práticas altamente orientado a pessoas,

⁹ Acrônimo para *Project Management Body of Knowledge*

¹⁰ Tradução do acrônimo *Software Engineering Institute*

e seu tratamento de incertezas balanceia considerações de risco negativas com oportunidades”.

As pessoas e os grupos adotam diferentes atitudes em relação ao risco, as quais devem ser explicitadas sempre que possível (PMI, 2008). De acordo com a VBSE, um conceito fundamental no gerenciamento de riscos é o da exposição ao risco (ER), o qual é calculado através da seguinte fórmula:

$$ER = P(p) * T(p)$$

onde $P(p)$ é a probabilidade de um resultado insatisfatório e $T(p)$ corresponde ao tamanho da perda, a qual pode ser redução de lucros, reputação, qualidade de vida ou outros atributos relacionados a valor; no caso de oportunidades, a entidade “perda” é substituída por um ganho de valor, refletido nos mesmos atributos.

2.2.2.5. Engenharia Simultânea de Sistema e Software

Segundo Sommerville (2007), “engenharia de software é uma disciplina de engenharia relacionada a todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação”. Sobre a Engenharia de Sistemas ele afirma: “a engenharia de sistemas diz respeito a todos os aspectos do desenvolvimento e da evolução de sistemas complexos, dos quais o software desempenha um papel importante”. Pelas definições acima, percebe-se que a engenharia de sistemas é uma área mais ampla que envolve várias engenharias, dentre elas a engenharia de software. Pressman afirma que, dentro de um contexto organizacional, a engenharia de sistemas é chamada de engenharia de processo de negócio. Esta última tem como objetivo definir como o trabalho será realizado dentro da organização (PRESSMAN, 2006).

De acordo com a VBSE, as constantes mudanças na área de tecnologia fazem com que organizações busquem cada vez mais agilidade nos processos de desenvolvimento (BOEHM, BARRY, 2005). Ela afirma que engenheiros de sistema e de software devem trabalhar juntos para definir conceitos do produto, requisitos, arquitetura e ciclo de vida. Assim, em um contexto organizacional, pessoas da área de negócio e engenheiros de software devem trabalhar de forma conjunta. Outra característica que a VBSE prega ser essencial é a definição de marcos de projeto

que possuem critérios para avaliar a consistência entre os diversos artefatos produzidos (D'ANUNCIAÇÃO, 2009).

2.2.2.6. Controle e Monitoramento Baseados em valor

Um dos métodos mais utilizados para monitorar e controlar projetos é o gerenciamento do valor agregado *Earned Value Management* – EVM. Esse é um método para medição de desempenho que integra as medidas custos e cronograma para auxiliar a equipe de gerenciamento a avaliar e medir o desempenho do projeto (PMI, 2008). No entanto, essa medição é feita com base em valores planejados no início do projeto. No desenvolvimento de software, mudanças são inevitáveis, o que torna esse método inadequado para este tipo de projeto. Além disso, ele não diz absolutamente nada sobre o valor que está sendo gerado (BOEHM, 2006c). Este elemento-chave afirma que o controle e monitoramento do projeto devem ser realizados com base nos valores que o projeto pretende gerar, e não somente em com base em orçamento e cronograma. Para isso, a VBSE sugere o uso do caso de negócio. Isso faz com que o caso de negócio seja constantemente atualizado para refletir mudanças ocorridas na organização. Outra forma de monitoramento sugerida pela VBSE é o monitoramento com base nas premissas, iniciativas e resultados definidos na Cadeia de Resultado. A Figura 5 ilustra o processo de monitoramento e controle sugerido na VBSE.

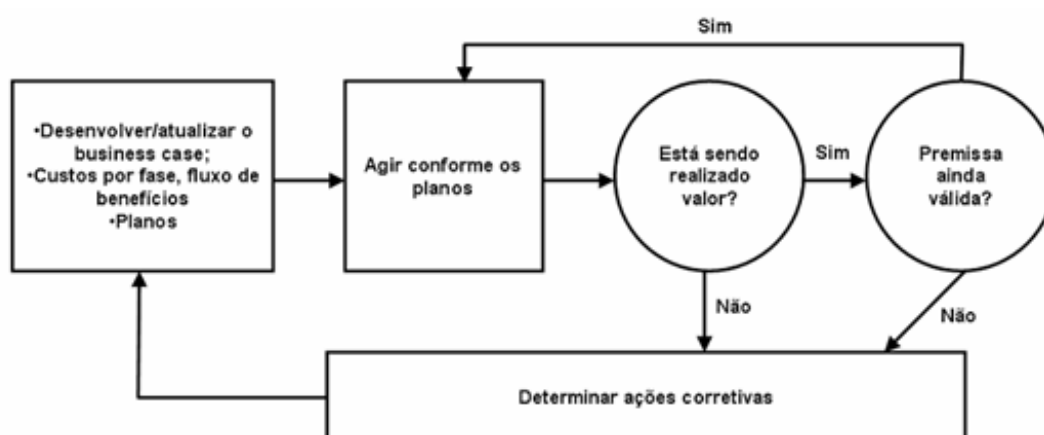


Figura 5 - Processo de monitoramento e controle sugerido pela VBSE

2.2.2.7. Mudanças como Oportunidade

Atualmente, mudanças acontecem continuamente nos ambientes internos e externos das organizações e, conseqüentemente, nas prioridades e proposições de valor dos interessados. Alguns métodos de software tratam mudanças como problemas na especificação inicial dos requisitos. Sistemas de qualidade tratam adaptações como custos de qualidade que devem ser minimizados. Esse tratamento faz com que organizações tendam a ter aversão a mudanças (BOEHM, 2006c). Organizações que conseguem se adaptar a mudanças mais rapidamente que seus concorrentes terão mais sucesso na realização da sua visão. Comparado ao hardware, o software pode ser facilmente atualizado para refletir tais mudanças. Portanto, mudanças podem criar valor para a organização (BOEHM, 2006c). Dois tipos de técnicas podem ser usados para aumentar a adaptabilidade de um projeto (D'ANUNCIAÇÃO, 2009):

- **Técnicas centradas em arquitetura:** essas técnicas têm como foco: identificar as fontes mais prováveis; usar modularização para diminuir o impacto das mudanças;
- **Técnicas centradas em refatoração:** essas técnicas focam em manter o produto o mais simples possível para acomodar mudanças.

Métodos ágeis de desenvolvimento baseiam-se em refatoração para acomodar mudanças, enquanto os processos orientados a planejamento utilizam as técnicas centradas em arquitetura.

2.2.3. Passos para incorporar os elementos-chave a um processo de desenvolvimento

A VBSE sugere alguns passos para incorporar os elementos-chave, anteriormente apresentados, no desenvolvimento de software:

- **Análise da Realização de Benefícios:** escrever o nome das iniciativas do software e suas contribuições de um lado e os resultados desejados pelos interessados do lado direito. Após isso, deve-se tentar preencher a cadeia de resultado com premissas, contribuições e resultados intermediários.

Isso fará com que iniciativas adicionais necessárias para a realização dos resultados desejados sejam identificadas;

- **Identificação e Conciliação das Proposições de Valores:** usar a cadeia de resultados para entrevistar e validar as proposições de valor dos interessados. Usar o modelo de conflitos para identificar e conciliar proposições de valor;
- **Análise do Caso de Negócio:** estimar o custo de desenvolvimento, instalação e manutenção do projeto; estimar os benefícios esperados durante o período; fazer uma lista de benefícios intangíveis que são difíceis de quantificar;
- **Gerenciamento contínuo de riscos e oportunidades:** identificar incertezas na análise do caso de negócio e na habilidade de realizar os resultados da cadeia de resultado. Riscos devem ser eliminados ou planos para gerenciar os futuros riscos devem ser criados;
- **Engenharia simultânea de sistema e software:** desenvolver paralelamente atividades de engenharia, além de definir marcos para verificação de consistências;
- **Monitoramento e controle baseados em valor:** Usar a cadeia de resultados para monitorar e validar premissas; monitorar custos e benefício do caso de negócio e utilizar marcos de verificação para atualizá-lo; monitorar riscos e oportunidades;
- **Mudança como oportunidade:** usar métodos ágeis com o acréscimo das atividades acima citadas; monitorar alterações no contexto do projeto para se antecipar às mudanças e adaptar o projeto.

2.3. Gerenciamento Ágil de Projetos

As organizações modernas iniciam projetos para criar serviços, processos, produtos novos e únicos ou então para aperfeiçoar os já existentes. De acordo com o guia PMBOK (PMI, 2008), “projeto é um esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo”. Projetos são frequentemente utilizados como meio de atingir o plano estratégico de uma organização, seja a equipe do projeto formada por funcionários da organização ou por um prestador de serviços

contratado. O objetivo fundamental de todo projeto e engenharia é maximizar a criação de valor para um dado investimento (BOEHM; SULLIVAN, 2000).

É importante fazer uma distinção entre o projeto e o gerenciamento do projeto (BOEHM; SULLIVAN, 2000). Projeto pode ser considerado como sendo a obtenção de objetivos específicos, o qual envolve uma série de atividades que consomem recursos, tendo datas de início e fim. Já o gerenciamento de um projeto pode ser definido como o processo para controlar a realização dos objetivos do projeto (MUNNS; BJEIRMI, 1996).

O sucesso e falha de um projeto são difíceis de definir e medir, uma vez que eles possuem significados diferentes para cada parte interessada. É amplamente aceito que o sucesso de projeto é uma construção multidimensional (THOMAS; FERNÁNDEZ, 2008). Um conceito importante no gerenciamento de projetos é o de parte interessada. De acordo com o PMBOK(PMI, 2008):

“Partes interessadas no projeto são pessoas e organizações ativamente envolvidas no projeto ou cujos interesses podem ser afetados com o resultado da execução ou do término do projeto. A equipe de gerenciamento de projetos precisa identificar as partes interessadas, determinar suas necessidades e expectativas e, na medida do possível, gerenciar sua influência em relação aos requisitos para garantir um projeto bem-sucedido”.

Na última década surgiram vários métodos com a promessa de dar mais agilidade ao desenvolvimento de software. De acordo com HIGHSMITH (2002), o termo “Método Ágil” tornou-se popular em 2001, quando um grupo de especialistas criou a “Agile Alliance” e estabeleceu o Manifesto Ágil para o desenvolvimento de software, o qual possui o seguinte conteúdo (BECK; BEEDLE, et al., 2001):

“Nós estamos descobrindo melhores maneiras para o desenvolvimento de software, executando e ajudando os outros a desenvolver. Por meio deste trabalho valoriza-se: Os indivíduos e suas iterações acima de processos e ferramentas; Software funcionando acima de documentação exaustiva; Colaboração do cliente acima de negociação contratual; Respostas às mudanças acima de execução de um plano. Embora haja valor nos itens à direita, nós valorizamos mais os itens à esquerda.”.

Além dos valores acima citados, foram definidos doze princípios que norteiam os métodos ágeis:

- Acomodar mudanças nos requisitos, mesmo que tardias;
- Satisfazer o cliente por meio de entrega de software funcional;
- Participação do cliente no processo de desenvolvimento;
- Entregas constantes de software funcional;
- Software funcionando é a medida primária de progresso do software;
- Foco na simplicidade;
- Atenção contínua na excelência técnica e num bom projeto melhora a agilidade;
- O time do projeto avalia seu desempenho e ajusta seu comportamento de acordo com as descobertas;
- As melhores arquiteturas, requisitos e projetos emergem de equipes auto-organizadas;
- Melhoria contínua na excelência técnica;
- Foco nas pessoas;
- Desenvolvimento sustentado.

Esses princípios possuem dois objetivos: promover um entendimento sobre o que são métodos ágeis e orientar equipes para determinar se de fato estão utilizando um método ágil. Cohen (2003) afirma que “métodos ágeis podem ser considerados uma coletânea de diferentes técnicas e métodos que compartilham os mesmos valores e princípios básicos, alguns dos quais remontando em técnicas introduzidas em meados dos anos 70 como o desenvolvimento e melhoria iterativos”. Uma das principais diferenças entre métodos ágeis e métodos tradicionais é a forma como o produto é planejado. Em métodos tradicionais, o planejamento de todo o produto é feito no início do desenvolvimento. Métodos ágeis partem da premissa de que não é possível conhecer todos os detalhes do produto no início de desenvolvimento. Assim, eles utilizam uma abordagem evolutiva que considera mudanças como necessárias.

A partir dos valores e princípios dos métodos ágeis de desenvolvimento de software, foi criado o Gerenciamento Ágil de Projetos (GAP), que, segundo Highsmith (2004), possui os seguintes objetivos:

- Inovação contínua;
- Adaptabilidade do produto;
- Tempo de entrega reduzidos;
- Capacidade de adaptação do processo e das pessoas;
- Resultados confiáveis.

Seis princípios, baseados nos métodos ágeis, guiam o GAP:

- Entregar valor ao usuário;
- Gerar entregas de forma iterativa e baseadas em funcionalidades;
- Buscar excelência técnica;
- Encorajar a exploração;
- Desenvolver equipes adaptativas (auto-organizadas e auto-disciplinadas);
- Simplicidade.

Highsmith (2004) propõe um framework para o GAP composto de cinco fases:

- **Visão:** identificar a visão e o escopo do produto, os interessados e como a equipe vai trabalhar junta;
- **Especulação:** esta fase consiste em estabelecer um conjunto inicial de requisitos, criar um plano de entregas, definir estratégias para o gerenciamento de riscos e estimar custos;
- **Exploração:** desenvolve e entrega funcionalidades aos usuários;
- **Adaptação:** revisar resultados e quando necessário, realizar adaptações;
- **Encerramento:** finalização das atividades do projeto, transferência dos aprendizados-chave e celebração.

Para cada uma das fases acima foram definidas práticas específicas que garantem o alinhamento dos princípios e valores do GAP.

2.4. Scrum

A história do Scrum teve início com o artigo “*The new product development game*” (TAKEUCHI; NONAKA, 1986). Tal artigo descreve como organizações atingiram resultados utilizando uma abordagem com foco nas equipes, enfatizando o aspecto auto-organizável (RUBIN, 2012). Scrum é um framework dentro qual

peessoas podem tratar e resolver problemas complexos e adaptativos, enquanto entregam produtos com valor aos interessados (SCHWABER; SUTHERLAND, 2011). Rubin (2012) afirma que o Scrum é uma abordagem para desenvolver produtos e serviços inovadores. O Scrum é baseado nas teorias empíricas de processo, as quais afirmam que o conhecimento vem da experiência e da tomada de decisão no que é conhecido. Ele é baseado em três pilares (SCHWABER; SUTHERLAND, 2011):

- **Transparência:** os artefatos e eventos devem ser acessíveis por pessoas responsáveis pelo resultado;
- **Inspeção:** a equipe do Scrum deve realizar inspeções para que garantir que o progresso caminha para a realização dos objetivos desejados;
- **Adaptação:** caso seja detectado algum desvio para fora dos limites aceitáveis, o processo e artefatos devem ser adaptados.

Scrum não é um processo onde é seguida uma sequência de passos para garantir a produção de produtos de alta qualidade. Ele é um framework para organizar e gerenciar o trabalho a ser realizado. Além disso, Scrum não é perfeito para todas as situações. Cada organização deve adequar o processo à sua realidade (COHN, 2009). De acordo com (LARMAN, 2003), o processo do Scrum é dividido em quatro fases:

- **Planejamento:** tem como objetivo estabelecer a visão e as expectativas dos stakeholders;
- **Preparação:** identificar requisitos e priorizá-los;
- **Desenvolvimento:** desenvolvimento do sistema em uma série de *Sprints*;
- **Entrega:** entregar o software funcionando ao cliente.

2.4.1. Ciclo de Vida

A Figura 6 apresenta uma visão geral do ciclo de vida do Scrum. Um projeto se inicia com a definição da visão do produto.

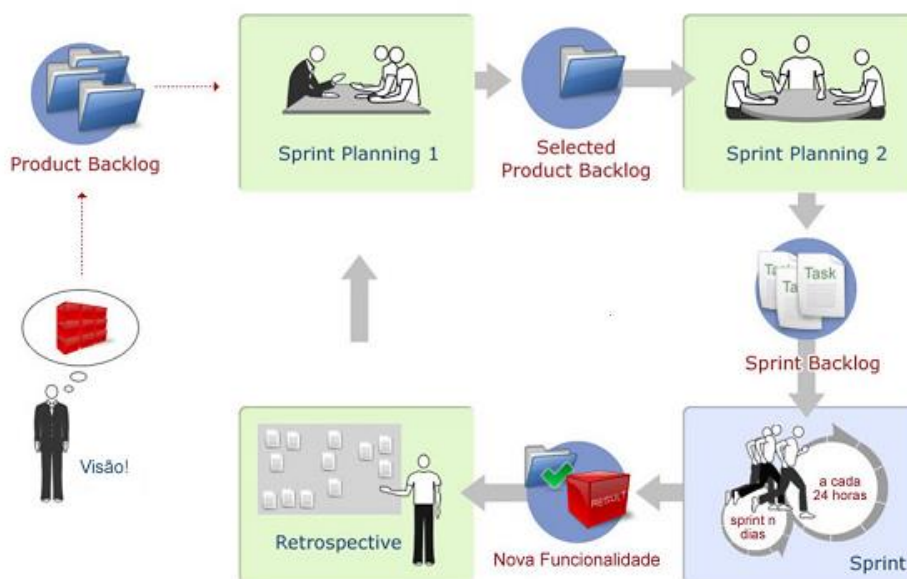


Figura 6 - Visão geral do Scrum

A partir da visão, é definido o *Product Backlog*¹¹, artefato que contém os requisitos do software que será desenvolvido. Os requisitos são desenvolvidos em uma sequência de *Sprints*. Cada *Sprint* é um ciclo de, no máximo, 30 dias, e é precedido por uma reunião de planejamento chamada *Sprint Planning*, a qual identifica os itens do *Product Backlog* que serão desenvolvidos durante a *Sprint*. Ao término da *Sprint*, espera-se que um incremento do software esteja pronto para entrega ao cliente. Antes da próxima *Sprint*, são realizadas reuniões a fim de inspecionar o produto e o processo. As próximas seções descrevem em detalhes os papéis, artefatos e eventos do método Scrum.

2.4.2. Papéis

As responsabilidades previstas no Scrum estão divididas entre três papéis: *Product Owner*, *Scrum Master* e Equipe de desenvolvimento. As seções abaixo descrevem cada um desses papéis.

¹¹ Por serem amplamente conhecidos, optou-se por manter em inglês o nome dos artefatos, eventos e papéis do Scrum.

2.4.2.1. *Product Owner*

O *Product Owner* representa os interesses de todas as partes interessadas do projeto (SCHWABER, 2004). Ele é responsável por decidir quais funcionalidades devem ser desenvolvidas, além de priorizá-las de acordo com a criação de valor.

De acordo com guia do Scrum, o *Product Owner* é a única pessoa responsável por gerenciar o *Product Backlog* e garantir o valor do trabalho da equipe (SCHWABER; SUTHERLAND, 2011). No caso de produtos comerciais, o *Product Owner* é responsável pelos lucros e perdas do produto, devendo maximizar o retorno sobre o investimento. No caso de aplicações internas, ele é responsável por maximizar o valor gerado para a organização (SUTHERLAND, 1993). De forma geral, ele é o responsável pelo sucesso no projeto.

Tabela 2 - Características do Product Owner. Adaptado de (RUBIN, 2012)

Responsabilidade	Descrição
Gerenciamento econômico	É responsável por garantir que a melhores decisões econômicas estão sendo tomadas em todos os níveis do projeto
Participar do planejamento	É um papel chave no planejamento de todas as atividades do Scrum
Gerenciar o <i>Product Backlog</i>	Todos os participantes do Scrum têm acesso ao <i>Product Backlog</i> , no entanto, o <i>Product Owner</i> é a única pessoa autorizada para alterá-lo.
Definir critérios de aceitação	É responsável por definir critérios de aceitação para cada item do <i>Product Backlog</i>
Colaborar com a equipe de desenvolvimento	Ele deve participar ativamente do dia a dia da equipe do Scrum
Interagir com os interessados	Deve representar os interessados do projeto. Por isso, ele deve gerenciar as expectativas dos interessados e verificar se o produto as atende.

Segundo Rubin (2012), o *Product Owner* atua em dois lados. De um lado ele deve entender quais as necessidades e prioridades dos interessados, de forma que possa representá-los. Do outro lado, ele deve interagir como a equipe do Scrum para comunicar o que e em que ordem o produto deve ser desenvolvido. Ele deve também garantir que critérios de aceitação sejam escritos e que testes sejam realizados a fim de garantir que o trabalho entregue crie valor aos interessados. A Tabela 2 apresenta as principais responsabilidades do *Product Owner*.

O *Product Owner* ainda é responsável por criar a visão do produto, planejar os *Releases*, gerenciar o orçamento, preparar o lançamento do produto, participar das reuniões do Scrum e colaborar com time (PICHLER, 2010).

2.4.2.2. Scrum Master

O *Scrum Master* é responsável por garantir que os princípios, valores e práticas do Scrum são compreendidos e são seguidos (SCHWABER; SUTHERLAND, 2011). Ele também é responsável por garantir que aspectos externos não interfiram nos trabalhos do time, além de remover possíveis impedimentos que apareçam durante o processo. Segundo Cohn (2009), o *Scrum Master* deve atuar como motivador para equipe. No entanto, as decisões de como executar as tarefas competem à equipe de desenvolvimento. Para ser um bom *Scrum Master* é necessário ter as seguintes características: responsabilidade; humildade; ser colaborativo; ser comprometido; ter poder de persuasão (RUBIN, 2012).

2.4.2.3. Equipe de Desenvolvimento

A equipe de desenvolvimento do Scrum é responsável por projetar, desenvolver e testar as funcionalidades do sistema. De acordo com (RUBIN, 2012), uma equipe Scrum tem as seguintes características:

- A equipe deve ser auto-organizada, tendo competência para definir como o trabalho será realizado;
- A equipe deve possuir todas as habilidades necessárias para desenvolver o produto;

- Todos possuem o título de desenvolvedor, independente da função desempenhada;
- A responsabilidade é sempre da equipe como um todo;
- Não existe subdivisão na equipe;
- O tamanho da equipe deve ser suficientemente pequeno para manter a agilidade e suficientemente grande para realizar o trabalho.

Segundo Cohn (COHN, 2009), a equipe ainda tem a responsabilidade de participar das reuniões diárias, reunião de planejamento da *Sprint* e das reuniões para inspecionar e adaptar o processo.

2.4.3. Artefatos

O Scrum define quatro artefatos: *Visão do Produto*, *Product Backlog*, *Release Backlog* e *Sprint Backlog*. As seções abaixo descrevem cada um desses artefatos.

2.4.3.1. Visão do Produto

Desenvolver a visão de onde se pretende chegar é fundamental para que objetivos sejam atingidos. Este artefato visa comunicar a essência do produto de forma concisa a todos os envolvidos no projeto, estabelecendo um panorama geral do que será construído. Segundo Rubin (2012, p. 293), algumas técnicas podem ser aplicadas:

- Sentença de elevador¹²: consiste em criar uma descrição do produto curta e concisa, de forma que possa ser entendida por uma pessoa rapidamente, como, por exemplo, dentro do trajeto de um elevador;
- Caixa de visão do produto: consiste em uma caixa para o produto, a qual deve conter as principais funcionalidades;

2.4.3.2. Product Backlog

O *Product Backlog* é uma lista ordenada com as funcionalidades desejadas, sendo a única fonte de requisitos do projeto (SCHWABER; SUTHERLAND, 2011).

¹² Tradução do termo em inglês *Elevator Statement*

Ele é composto de vários itens. Esses itens podem ser uma funcionalidade nova, correção de *bugs*, aquisição de conhecimento ou algum trabalho técnico. De acordo com Cohn (2009) ele deve ser:

- **Detalhado:** deve conter todos os detalhes necessários para o desenvolvimento;
- **Emergente:** o *Product Backlog* não é um artefato estático. Ele deve ser atualizado constantemente para refletir mudanças no negócio;
- **Estimado:** deve ser estimado para prover indicativos do esforço, tempo e custo;
- **Priorizado:** deve ser priorizado de forma a entregar primeiro itens com maior valor.

O *Product Backlog* pode ser atualizado durante os eventos de inspeção e adaptação do Scrum, como a *Daily Meeting* e a *Sprint Review* (RUBIN, 2012). Por ser dinâmico, itens podem ser adicionados, alterados ou revisados pelo *Product Owner* conforme as necessidades do negócio e dos interessados. Normalmente, no início do projeto os itens são grandes e com poucos detalhes. Conforme o projeto é desenvolvido, a equipe do Scrum adquire conhecimento e consegue detalhá-los, até que eles possam ser inseridos em uma *Sprint*, onde são projetados, desenvolvidos e testados. O Scrum não define um formato padrão para itens do *Product Backlog*. No entanto, muitos times Scrum representam itens como histórias de usuário (COHN, 2009).

2.4.3.3. *Sprint Backlog*

Segundo o guia do Scrum (SCHWABER; SUTHERLAND, 2011), o *Sprint Backlog* é uma lista de tarefas que a equipe de desenvolvimento se compromete a desenvolver durante a *Sprint*. As tarefas são definidas com base nos itens do *Product Backlog* selecionados pelo *Product Owner*. Ele é criado durante a *Sprint Planning*, na qual participa todo o time Scrum.

2.4.4. Eventos

O Scrum prevê cinco eventos. Uma importante característica é que todos possuem duração fixa. Abaixo segue uma descrição de cada um dos eventos do Scrum.

2.4.4.1. *Sprint*

De acordo com Schwaber e Sutherland (2011), a *Sprint* é o “coração” do Scrum. Este evento possui duração de, no máximo, 30 dias, tendo datas de início e fim definidas. Neste período a equipe de desenvolvimento irá executar as tarefas do *Sprint Backlog* para construir um incremento do software. Ao término da *Sprint*, os itens desenvolvidos devem estar de acordo com a definição de pronto estabelecida. O time deve ter uma visão compartilhada do que significa *pronto*. Conceitualmente, a definição de *pronto* é um *checklist* de tarefas que a equipe Scrum deve executar antes de declarar o incremento como pronto para entrega (RUBIN, 2012).

A curta duração da *Sprint* traz alguns benefícios, tais como: facilidade para planejar, rápido *feedback* e diminuição de riscos. Uma importante regra do Scrum é que, uma vez definido o objetivo da *Sprint* e a mesma tenha sido iniciada, mudanças não são permitidas. O objetivo da *Sprint* funciona como um compromisso entre os integrantes da equipe Scrum durante a *Sprint*.

2.4.4.2. *Sprint Planning*

A *Sprint Planning* determina quais itens do *Product Backlog* serão desenvolvidos na próxima *Sprint*. Essa reunião é um evento com duração de oito horas dividido em duas etapas (SCHWABER; SUTHERLAND, 2011). Na primeira etapa o *Product Owner* apresenta os itens do *Product Backlog* com mais alta prioridade. O time Scrum irá definir o objetivo da *Sprint* e escolher quais itens irão compor o *backlog* da próxima *Sprint*. A equipe de desenvolvimento é responsável por atribuir uma estimativa, normalmente em horas, para a conclusão das tarefas. A

segunda etapa da reunião é destinada à equipe decidir como o incremento será desenvolvido.

2.4.4.3. *Sprint Review*

Ao final de cada *Sprint*, o time Scrum, juntamente com os interessados, revisa o trabalho concluído em uma reunião de revisão com duração de quatro horas. Baseado nos *feedbacks*, o *Product Owner* pode revisar e atualizar o *Product Backlog* (RUBIN, 2012). O objetivo da reunião de revisão é inspecionar e adaptar o produto que está sendo desenvolvido, e não como está sendo desenvolvido (SCHWABER; SUTHERLAND, 2011). A reunião de revisão deve identificar o que está pronto e que não está pronto, além de ajudar a desenvolver uma visão compartilhada do que está sendo desenvolvido.

2.4.4.4. *Sprint Retrospective*

Segundo Schwaber e Sutherland (2011), a *Sprint Retrospective* frequentemente acontece após a reunião de revisão. O objetivo desta reunião, a qual possui duração de quatro horas, é inspecionar e adaptar o processo, e não o produto. O time do Scrum deve analisar o que ocorreu bem e quais foram os problemas na última *Sprint*. A partir dessa análise, a equipe deve criar um plano para implementar possíveis melhorias para as próximas *Sprints*.

2.4.4.5. *Daily Meeting*

Esse evento é uma reunião diária com duração de quinze minutos que acontece, preferencialmente, no mesmo horário (SCHWABER; SUTHERLAND, 2011). O objetivo é analisar o andamento da *Sprint*. Uma abordagem comum para a reunião, a qual tem como facilitador o *Scrum Master*, é cada membro da equipe responder as seguintes perguntas (RUBIN, 2012):

- O que eu fiz desde a última reunião?
- O que eu irei fazer hoje?
- Quais as dificuldades e obstáculos estão atrapalhando meu progresso?

Reuniões diárias ajudam a equipe organizar os trabalhos, melhorar a comunicação, identificar impedimentos e melhorar o nível de conhecimento da equipe. É considerada uma reunião chave de inspeção e adaptação. Durante essa reunião, o monitoramento do progresso da *Sprint* é realizado através do gráfico *Sprint Burndown*. Este gráfico permite monitorar o andamento da *Sprint* com base na execução das tarefas. O eixo vertical mostra uma estimativa do trabalho necessário, a qual é definida pela equipe de desenvolvimento durante o planejamento da *Sprint*. Esse gráfico deve ser atualizado diariamente durante a *Daily Meeting*. A Figura 7 apresenta um exemplo de um *Sprint Burndown* (SUTHERLAND, 1993).

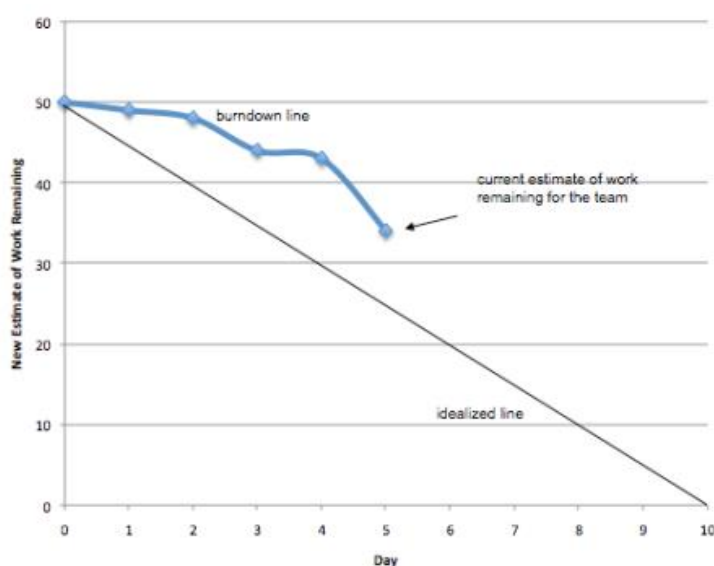


Figura 7 - Sprint Burndown

Outra ferramenta utilizada para controlar o andamento de uma *Sprint* é o quadro de tarefas. Através desse quadro é possível controlar a execução das tarefas, identificando quais tarefas precisam ser executadas, quais estão sendo executadas e quais foram concluídas (RUBIN, 2012).

2.5. Trabalhos relacionados e correlacionados

Alguns trabalhos têm sido propostos a fim de analisar aderência do Scrum ou VBSE a outros métodos existentes na Engenharia de Software. Um trabalho diretamente relacionado a esta pesquisa foi o de D'Anunciação (2009), uma vez que buscou verificar a aderência da VBSE a um processo de desenvolvimento de

software. D'Anunciação realizou uma análise a fim de verificar como a criação de valor é tratada dentro do *Rational Unified Process* - RUP (IBM, 2001). Foram identificados pontos convergentes e divergentes. Para os divergentes, foi proposta uma extensão chamada RUP-VC. A Tabela 3 apresenta as extensões propostas pelo RUP-VC.

Tabela 3 - Extensões proposta no RUP-VC

Extensão
Adição de uma Disciplina de Gerenciamento de Portfólio
Uso da Cadeia de Resultados na Definição de Programas e Portfólio
Uso de Groupware
Gerenciamento das Expectativas dos Interessados
Proposições de Valor como Critérios de Priorização de Casos de Uso
Análise de Opções considerando Riscos e Benefícios Intangíveis
Gerenciamento de Oportunidades
Avaliação de Riscos a Partir das Funções de Utilidade dos Interessados
Adaptar o Processo de Software Considerando a Exposição ao Risco
Monitoramento de Valor no Nível do Projeto
Monitoramento de Valor no Nível da Organização
Permitir Adaptações Visando Agilidade do Processo

Alguns trabalhos correlacionados a esta pesquisa foram identificados. Marçal (2009) fez uma análise da aderência do Scrum ao CMMI. A análise se restringiu ao escopo das áreas de processo de gestão de projetos. A Tabela 4 apresenta o resultado desta análise. Zanatta (2004) propôs uma extensão do Scrum visando adequá-lo às áreas de processo Gerenciamento de Requisitos e Desenvolvimento de Requisitos.

Tabela 4 - Aderência do Scrum ao CMMI (Marçal, 2009)

Práticas do CMMI	Classificação
Estimar o Escopo do Projeto	Aderente
Determinar Estimativas de Esforço e Custo	Parcialmente Aderente
Identificar Riscos do Projeto	Parcialmente Aderente
Planejar o Envolvimento dos Stakeholders	Aderente
Monitorar os Riscos do Projeto	Parcialmente Aderente
Monitorar o Envolvimento dos Stakeholders	Aderente
Conduzir Revisões em Marcos	Aderente
Tomar ações corretivas	Parcialmente Aderente
Estabelecer a Visão Compartilhada do Projeto	Aderente

2.6. Considerações sobre o referencial teórico

A VBSE é uma abordagem que visa incorporar considerações de valor nos processos de desenvolvimento de software. Para isso, define sete elementos-chave que formam a base para a criação de valor. Conforme apresentado na Seção 1.1, a utilização de métodos ágeis cresceu nos últimos, sendo que o Scrum é o método mais utilizado. O referencial teórico apresentado neste capítulo mostrou que dentre os princípios e valores definidos no manifesto ágil, o qual norteia os métodos ágeis, está a criação de valor para os interessados. Desta forma, é possível afirmar que a VBSE e o Scrum têm como premissa a criação de valor. Além disso, alguns trabalhos recentes vêm buscando analisar a aderência da VBSE e Scrum a outras metodologias. Um trabalho diretamente relacionado analisou a aderência da VBSE ao RUP, o qual teve como resultado a elaboração de uma extensão do RUP que visou torná-lo aderente à VBSE. Outros trabalhos correlacionados buscaram analisar a aderência do Scrum a outras metodologias. Os trabalhos relacionados e

correlacionados adotam abordagens semelhantes, as quais visam identificar pontos de convergência e divergência entre duas metodologias.

3. ANÁLISE DA ADERÊNCIA DO SCRUM À VBSE

Este capítulo apresenta uma análise da aderência do método ágil Scrum aos elementos-chave da VBSE apresentados na Seção 2.2.2. Nesta análise serão buscadas evidências de que os elementos-chave são de, alguma forma, abordados pelo Scrum. Para não descaracterizar a agilidade do Scrum, a análise leva em consideração os princípios e valores definidos no manifesto ágil e no gerenciamento ágil de projetos (HIGHSMITH, 2004), além dos princípios do próprio Scrum. Para a análise, foi elaborada uma escala ordenada por três categorias. São elas:

- **Não aderente:** Há pouca ou nenhuma evidência de que o elemento-chave seja abordado pelo Scrum;
- **Parcialmente aderente:** Existem evidências que mostram que o elemento-chave é abordado pelo Scrum, no entanto, existem pontos essenciais não abordados;
- **Aderente:** Existem evidências que demonstram que o Scrum satisfaz completamente o elemento-chave.

As seções abaixo apresentam a análise para cada elemento-chave proposto pela VBSE.

3.1. Análise da Realização de Benefícios

A essência deste elemento é que um investimento em TI, por si só, não garante a realização dos benefícios esperados. Desta forma, este elemento-chave tem como finalidade identificar ações adicionais, de TI ou não, necessárias para que os benefícios desejados sejam realizados, criando, assim, valor para os interessados.

Um projeto Scrum tem início com a definição da *Visão do Produto*, a qual define uma descrição geral do produto. Após a definição da visão, é criado o *Product Backlog* com as funcionalidades que irão compor o software. A partir dele, são desenvolvidos incrementos do software em uma sequência de *Sprints*. Cada incremento, de acordo com os princípios do Scrum, deve entregar valor aos interessados.

Analisando o exposto anteriormente, percebe-se que a criação de valor no Scrum tem como fundamento os itens do *Product Backlog*. Em outras palavras, o Scrum considera que, uma vez implementadas as funcionalidades referentes aos itens do *Product Backlog*, o software entregue irá criar valor aos interessados, não levando em consideração a existência de possíveis ações necessárias para que o valor seja criado. Assim, por não considerar ações adicionais necessárias para a criação de valor, o Scrum é considerado **Não Aderente** a este elemento-chave.

3.2. Identificação e Conciliação das Proposições de Valor

Este elemento aborda o ponto central da teoria da VBSE. Essa teoria é baseada na afirmação que um projeto só terá sucesso se, e somente se, entregar valor aos interessados. Em um projeto de software existem vários interessados, cada um com suas próprias proposições de valor. Portanto, para criar valor, é essencial identificar quais as proposições de valor de cada interessado e implementar funcionalidades que criem tais valores. Importante observar que valor é definido em termos dos benefícios e custos. Assim, além de identificar os benefícios, deve-se identificar os custos que cada interessado está disposto a assumir. Além disso, em um ambiente organizacional, o projeto deve contribuir para a realização dos objetivos do negócio. Portanto, deve-se ter um alinhamento entre o planejamento estratégico e o projeto.

Um dos princípios definidos no manifesto ágil e que norteia o gerenciamento ágil de projetos é a intensa participação do cliente no desenvolvimento do produto. No Scrum, essa responsabilidade é atribuída ao *Product Owner*. É sua a responsabilidade de interagir com todos os interessados do projeto e identificar quais benefícios cada um espera receber (PHAM; PHAM, 2012). De forma geral, ele é o responsável pelo sucesso do projeto. Segundo a VBSE, além de identificar as proposições de valor, deve-se garantir que elas são compatíveis entre si. Uma das características desejáveis para ser um bom *Product Owner* encontradas na literatura é a de bom negociador. Isso evidencia que o Scrum reconhece a necessidade de negociar com interessados.

Analisando o exposto acima, é possível identificar evidências de que o Scrum contempla o entendimento da teoria da VBSE e deste elemento-chave, uma vez que

confere a um de seus papéis a responsabilidade de gerenciar as proposições de valor dos interessados e entregar produtos que criem valor. Portanto, o Scrum é considerado **Aderente** a este elemento-chave.

É importante destacar que o Scrum não define como o *Product Owner* deve realizar suas tarefas, assim como a VBSE. No entanto, conforme apresentado na Seção 2.2.2.2, a VBSE sugere algumas abordagens, como uso do modelo de conflitos, gerenciamento de expectativas, técnicas de visualização e análise *trade-off*, priorização, *groupware* e análise do caso de negócio. Essas abordagens podem ser utilizadas pelo *Product Owner* para desempenhar suas tarefas.

3.3. Análise de Caso de Negócio

O Caso de Negócio tem como finalidade justificar o projeto que está sendo desenvolvido. Além disso, ele ajuda a criar uma linguagem comum e melhorar a comunicação entre executivos do negócio e profissionais de TI. A VBSE sugere a criação de um Caso de Negócio que leve em consideração benefícios tangíveis e intangíveis, bem como os riscos envolvidos. A criação de um Caso de Negócio também é sugerida com uma das práticas de gerenciamento ágil de projetos. Segundo Highsmith (2004), o Caso de Negócio deve conter um resumo do projeto e informações como prazos, custos e riscos.

O artefato do Scrum que contém algumas características de um Caso de Negócio é a Visão do Produto, uma vez que sua função é criar um entendimento compartilhado entre todos participantes para que um objetivo comum seja atingido. No entanto, conforme citado na seção 2.4.3.1, a Visão do Produto descreve somente um panorama geral do produto, normalmente usando técnicas simples, tais como caixa de visão e sentença de elevador. Estas técnicas nada dizem sobre objetivos organizacionais, benefícios, custos, prazos e riscos. Desta forma, percebemos que a *Visão do Produto*, mesmo tendo a finalidade de compartilhar uma visão comum, não contempla as características de um caso de negócio. Portanto, o Scrum é considerado **Não Aderente** a este elemento-chave.

3.4. Gerenciamento Contínuo de Riscos e Oportunidades

Este elemento-chave afirma que é necessário gerenciar os riscos e oportunidades durante todo o ciclo de desenvolvimento, levando em consideração fatores relacionados às pessoas. Incluir o gerenciamento de risco é um importante fator a considerar para obter sucesso em um projeto (NYFJORD, 2008). Arnuphaptrairong (2011) realizou uma compilação de vários trabalhos que buscaram identificar quais os principais riscos envolvidos de projetos de software. O resultado mostrou que os riscos mais citados estão relacionados à má compreensão de requisitos, à falta de comprometimento e suporte da alta administração, à falta de envolvimento e comprometimentos dos usuários, a mudanças nos requisitos e à falta de um método de gerenciamento de projetos.

Durante uma *Sprint*, a equipe de desenvolvimento mantém uma lista de impedimentos. No Scrum, um impedimento é qualquer obstáculo que diminua a produtividade da equipe. Essa lista é atualizada e discutida durante as reuniões diárias e cabe ao *Scrum Master* a remoção dos impedimentos. No entanto, impedimento não se confunde com risco. Um risco é um evento incerto que pode ou não ocorrer, tendo impacto positivo ou negativo. Já um impedimento é algo concreto e que necessariamente tem um impacto negativo. Evidente que, se ocorrer, um risco pode impactar na produtividade do time, mas não está limitado a isso.

O Scrum não aborda explicitamente o gerenciamento de riscos. Segundo Cohn (2011), um grande foco no gerenciamento explícito de riscos se torna desnecessário quando se utiliza métodos ágeis. Curtas iterações, ênfase em testes automatizados e entregas frequentes ajudam a equipe a evitar grande parte dos riscos. De fato, a maioria dos riscos comuns anteriormente citados são implicitamente abordados pelo Scrum. Devido às entregas frequentes, riscos relacionados à especificação de requisitos podem ser detectados e corrigidos rapidamente. Riscos relacionados à falta de comprometimento de usuários e interessados são minimizados pela existência do papel *Product Owner*. Ele é responsável por representar os interessados do projeto e tem uma participação ativa em todos os eventos do Scrum. A natureza dos métodos ágeis minimiza riscos relacionados a mudanças. Eles assumem que não é possível identificar todos os requisitos no início do projeto. Desta forma, conforme se obtém melhor

entendimento do projeto, mudanças serão necessárias e devem ser realizadas. Por último, o Scrum oferece eventos para controle do projeto. Nestes eventos, o time é estimulado a verificar o produto que está sendo desenvolvido e como está sendo desenvolvido, além de permitir aos membros do time Scrum compartilhar experiências e manter uma efetiva comunicação.

A VBSE sugere que a realização dos benefícios do software pode depender de outras ações, de TI ou não. Uma vez que a finalidade do software é a criação de valor através da entrega de benefícios, a não execução dessas ações representa riscos ao sucesso do projeto. Assim, faz-se necessário um gerenciamento desses riscos. Além disso, riscos não abordados implicitamente pelo Scrum poderiam tirar proveito de um gerenciamento explícito de riscos.

Com base no exposto acima, percebe-se que a natureza do Scrum trata implicitamente os principais riscos envolvidos em projetos de software. No entanto, as ações adicionais podem inserir novos riscos que precisam ser gerenciados. Desta forma, o Scrum é considerado ***Parcialmente Aderente*** a este elemento-chave.

3.5. Engenharia Simultânea Sistemas e Software

Este elemento-chave afirma que é essencial desenvolver atividades da engenharia de software paralelamente às atividades engenharia de sistema. Conforme visto na Seção 2.2.2.5, em um ambiente organizacional, a engenharia de sistemas toma a forma de engenharia de processos de negócio. Assim, sugere-se que pessoas do negócio e engenheiro de software trabalhem de forma conjunta. Além disso, a VBSE afirma que deve haver marcos onde a consistência entre os artefatos produzidos seja verificada.

No Scrum, o *Product Owner* é uma pessoa da área de negócio que participa ativamente no projeto. Ele é responsável pelo sucesso do projeto e por representar todas as partes interessadas. Assim, o Scrum prevê uma forte ligação entre pessoas do negócio e engenheiros de software. Além disso, existem marcos para verificação de consistências. Na *Sprint Review* participam o time Scrum, bem como todos os interessados no projeto. Nela é possível identificar inconsistências e realizar

adaptações com base no andamento do projeto e em processos do negócio. Portanto, o Scrum é considerado **Aderente** a este elemento-chave.

3.6. Monitoramento e Controle Baseados em Valor

Este elemento-chave afirma que é necessário monitorar e controlar o projeto com base nas proposições de valor dos interessados. Técnicas tradicionais, como *Earned Value Management* (EVM), realizam tais tarefas com base nos custos e cronograma, não levando em consideração proposições de valor. A VBSE sugere que o monitoramento e controle sejam realizados com foco no valor que os interessados esperam receber.

No Scrum, o monitoramento do projeto é realizado com o auxílio de dois gráficos: *Product Burndown* e *Sprint Burndown*. Estes gráficos mostram a quantidade de trabalho restante em função do tempo. Uma vez que esses gráficos são baseados em artefatos que contêm as funcionalidades que irão criar valor para os interessados, pode-se afirmar que no Scrum há uma prática de monitoramento baseado em uma noção de valor. No entanto, por esses gráficos, não é possível identificar se os benefícios estão de fato sendo realizados, o que só ocorre após a entrega do software. Além disso, a VBSE prevê que ações adicionais podem ser necessárias para que o valor seja criado. Essas ações também devem ser monitoradas e controladas. Desta forma, o Scrum é considerado **Parcialmente Aderente** a este elemento-chave.

3.7. Mudança como Oportunidade

De acordo com Boehm (2006c), em processos de software tradicionais mudanças normalmente são vistas como defeitos ou retrabalho. Essas mudanças podem vir do mercado, mudanças na legislação, dentre outras. Este elemento-chave tem como fundamento que mudanças podem criar valor para os interessados.

Um dos princípios dos métodos ágeis é acomodar mudanças, mesmo que tardias. Esse princípio é reflexo de uma das principais diferenças entre métodos tradicionais e métodos ágeis. Ao contrário dos tradicionais, métodos ágeis assumem que não é possível identificar todos os requisitos do produto logo no início do

desenvolvimento. Estes vão sendo descobertos e entendidos à medida que o projeto vai sendo desenvolvido. Portanto, mudanças certamente ocorrerão e devem ser contempladas no projeto.

O Scrum define eventos para inspecionar e, caso necessário, adaptar o processo e o produto. Na *Sprint Review*, o time do Scrum e os interessados discutem o incremento produzido na última *Sprint* e analisam o andamento do produto. Caso seja necessário, o *Product Backlog* deve ser alterado. Já na *Sprint Retrospective*, o time pode avaliar como está trabalhando e identificar pontos de melhoria para as próximas *Sprints*. Essas mudanças podem interferir em aspectos como tempo, custo e qualidade, os quais podem estar diretamente ligados a valores dos interessados. Analisando o exposto acima, é possível verificar que mudanças são essenciais para a utilização de métodos ágeis, sendo que o Scrum possui eventos voltados para identificação e implantação destas. Portanto, o Scrum é considerado como **Aderente** a este elemento-chave.

3.8. Considerações sobre a Análise

A análise apresentada neste capítulo mostrou que o Scrum, em sua forma original, é aderente a vários elementos da VBSE. Somente dois elementos não são, de nenhuma forma, abordados pelo Scrum.

Tabela 5 - Resumo da análise da aderência do Scrum aos elementos-chave da VBSE

Elemento-chave	Resultado da análise
Análise da Realização de Benefícios	Não Aderente
Identificação e Conciliação das Proposições de Valor	Aderente
Análise de Caso de Negócio	Não Aderente
Gerenciamento Contínuo de Riscos e Oportunidades	Parcialmente Aderente
Engenharia Simultânea Sistemas e Software	Aderente
Controle e Monitoramento Baseados em Valor	Parcialmente Aderente
Mudança como Oportunidade	Aderente

A Tabela 5 apresenta um resumo da análise realizada neste capítulo. A compatibilidade das duas abordagens é explicada pelo fato delas terem como princípio a criação de valor. Apesar de terem muitos pontos em comum, alguns elementos-chave da VBSE importantes não são abordados, o que torna necessário a inclusão de novos artefatos ao processo do Scrum

4. UMA PROPOSTA DE EXTENSÃO AO SCRUM COM FOCO NA CRIAÇÃO DE VALOR

Este capítulo apresenta uma proposta de extensão que visa tornar o Scrum aderente aos princípios e elementos-chave da VBSE, contribuindo para que os softwares entregues criem valor para os interessados, mantendo o Scrum alinhado aos princípios e valores definidos no manifesto ágil e no gerenciamento ágil de projetos. A extensão tem como base a análise realizada no capítulo 3. A Figura 8 apresenta uma visão geral da extensão. São propostos cinco pontos de extensão. As próximas seções detalham cada um desses pontos.

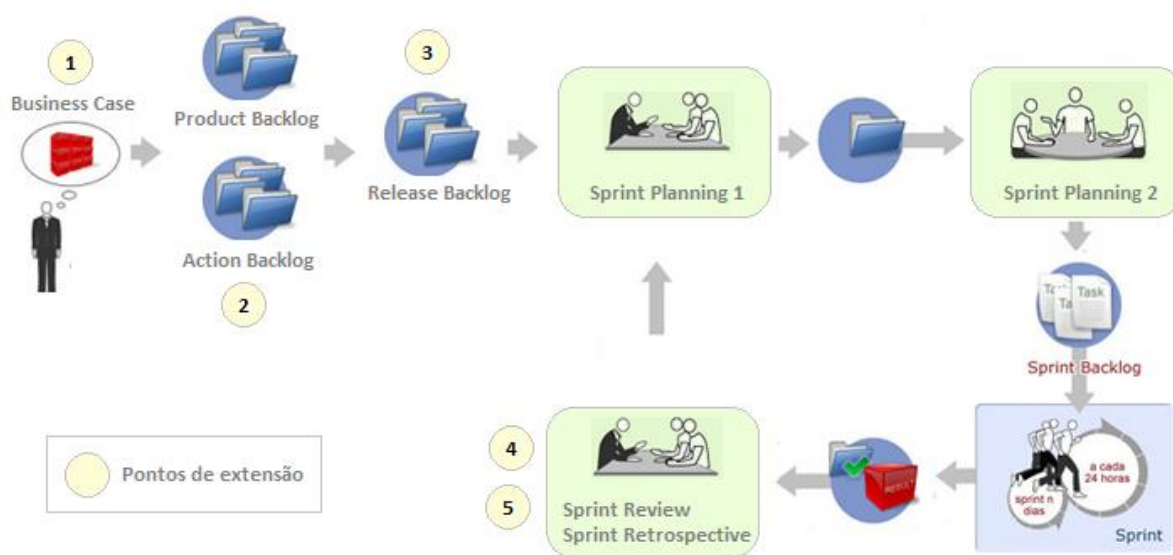


Figura 8 - Visão geral da extensão

4.1. Ponto de Extensão 1 - Criação do artefato *Business Case*

A análise apresentada na seção 3.3 concluiu que Scrum não é aderente ao elemento-chave *Análise do Caso de Negócio*. Desta forma, este ponto de extensão propõe a inclusão de um novo artefato, chamado *Business Case*¹³. A criação de um artefato semelhante está prevista em uma das práticas de gerenciamento ágil de projetos, a qual prevê a criação do artefato *Product Datasheet* (HIGHSMITH, 2004) com características e finalidade semelhantes a de um *Business Case*.

¹³ Para manter coerência com a terminologia adotada pelo Scrum, os artefatos propostos nesta extensão são escritos em inglês.

Em muitas organizações, engenheiros de software e executivos de negócio vivem em mundos separados, usando suas próprias terminologias, critérios de decisão e métodos de trabalho. Desta forma, muitos engenheiros de software agem como se fossem “artistas criativos”, estando interessados somente em criar novas funcionalidades. A inclusão deste artefato tem como finalidade criar um documento que justifique o projeto e que crie uma linguagem comum entre todos os envolvidos no projeto, contribuindo para que todos os envolvidos busquem os mesmos objetivos. Além disso, ele contribuirá para que os benefícios e custos de cada interessado fiquem explícitos a todos os envolvidos no projeto.

Na literatura existem vários modelos de Caso de Negócio. Este ponto de extensão é baseado em uma prática de gerenciamento ágil de projetos definida por Highsmith (HIGHSMITH, 2004), a qual sugere a criação de um artefato chamado *Project Data Sheet*. Sugere-se que o *Business Case* contenha os seguintes itens: Resumo do projeto; Contexto Estratégico, Partes Interessadas e Proposições de Valores, Viabilidade Financeira e Restrições. É importante observar que para cada um dos itens sugeridos existem variadas técnicas. No entanto, o objetivo deste ponto de extensão é definir quais itens são relevantes para a criação deste artefato, apresentando somente uma descrição e orientações gerais para sua elaboração.

A responsabilidade pela criação do artefato é atribuída ao *Product Owner*, uma vez que seu conteúdo está mais relacionado a aspectos do negócio do que a aspectos técnicos. No entanto, ele deve ser visível a todo o time Scrum. Ele deve ser o primeiro artefato a ser criado e deve servir como base para a criação do *Product Backlog* e o *Action Backlog*¹⁴, sendo atualizado sempre que necessário. As próximas seções descrevem cada um dos itens deste artefato.

4.1.1. Resumo do Projeto

Este item deve apresentar uma descrição geral do projeto, a qual deve conter os principais benefícios esperados pelos interessados. Uma descrição resumida com a qual todos interessados concordem é fundamental para o sucesso do projeto (IBM, 2001).

¹⁴ Artefato proposto nesta extensão, o qual será apresentado posteriormente.

4.1.2. Contexto Estratégico

Em um contexto organizacional, é necessário que todos os projetos estejam alinhados com a estratégia e objetivos da organização. Atingir esse alinhamento é um dos grandes desafios enfrentados por executivos do negócio e de TI. Assim, entender o contexto estratégico da organização e para quais objetivos o projeto contribuirá é importante para que o projeto tenha sucesso.

4.1.3. Partes Interessadas e Proposições de Valores

Este item deve descrever as partes interessadas no projeto, bem como os benefícios, tangíveis e intangíveis, que cada uma espera receber e quais custos elas estão dispostas a assumir.

4.1.4. Riscos

A análise apresentada na Seção 0 conclui que é necessário gerenciar os riscos. Na literatura encontramos várias abordagens, as quais definem atividades necessárias para o gerenciamento de riscos (IEEE STD 1540- 2001, 2001; BOEHM, 1991; P, 1999; PMI, 2008; SEI). Algumas abordagens têm sido proposta a fim de definir um gerenciamento ágil de riscos (RIBEIRO; GUSMÃO, *et al.*, 2009; NYFJORD, 2008; SOFIA). Conforme destacado em (GUSMÃO; MOURA, 2004), existe um consenso entre as atividades necessárias para o gerenciamento de riscos: Planejar a gerência de riscos; identificar riscos; analisar riscos; planejar respostas aos riscos; monitorar riscos; controlar riscos; comunicar riscos. Não é objetivo dessa extensão definir um processo de gerenciamento de riscos para o Scrum. A intenção deste item é apenas registrar o resultado das atividades de identificação, análise e planejamento de respostas aos riscos, cabendo ao time Scrum escolher a melhor forma de realizá-las. Sugere-se que este item descreva uma lista de riscos com os seguintes campos:

- **Descrição do Risco:** descrição concisa geral do risco de forma que todos os interessados compreendam;
- **Probabilidade:** representa a probabilidade de o risco acontecer;
- **Impacto da perda:** representa a perda caso o risco aconteça;

- **Exposição ao risco:** a exposição ao risco é definida em função da probabilidade e do o impacto da perda;
- **Ação resposta:** define uma ação a ser tomada caso o risco ocorra.

4.1.5. Viabilidade Financeira

Este item deve demonstrar a viabilidade financeira do projeto e fornecer dados para compará-lo a outras opções. Para isso, é necessário realizar uma estimativa dos custos monetários envolvidos durante todo o ciclo de vida do projeto e dos benefícios que serão entregues ao negócio. Um método financeiro sugerido para essa análise é Valor Presente Líquido (BOEHM, 2006a). Benefícios intangíveis, os quais são difíceis de especificar, também devem ser descritos e levados em consideração na análise.

4.1.6. Restrições

Este item deve apresentar restrições que afetam o projeto. Essas restrições podem incluir recursos humanos, conhecimento, requisitos legais, dentre outros.

4.2. Ponto de Extensão 2 - Criação do Artefato *Action Backlog*

A análise realizada na Seção 3.1 mostrou que o Scrum não é aderente ao elemento-chave *Análise da Realização de Benefícios*, uma vez que ele não leva em consideração ações adicionais necessárias para a criação de valor. Por exemplo, uma escola decide disponibilizar na WEB um software para que professores registrem suas aulas assim que elas terminem. São esperados vários benefícios, dentre eles a redução custos e a possibilidade realizar monitoramento em tempo real de alunos faltosos. No entanto, para que esses benefícios sejam realizados, os professores precisam ser capacitados para o uso de computadores e ter à disposição computadores com acesso à internet, caso contrário, o software não trará os benefícios esperados. Desta forma, percebe-se que o software, por si só, não entrega os benefícios esperados.

Este artefato tem como objetivo criar uma lista de ações adicionais, de TI ou não, necessárias para que todos os benefícios esperados sejam realizados, criando,

assim, valor para os interessados, o que, segundo a VBSE e o manifesto ágil, deve ser o objetivo do software. Cada item da lista deste artefato deve conter uma descrição e uma estimativa do tempo necessário para sua execução. Cabe ao *Product Owner*, juntamente com os interessados, planejar e garantir a execução das ações identificadas. Para a identificação das ações, sugere-se a utilização da ferramenta *Cadeia de Resultados*, apresentada na Seção 2.2.2.1. Este artefato é dinâmico, devendo ser atualizado sempre que necessário. Sugere-se que durante a *Sprint Review*, na qual o time do Scrum se reúne com os interessados, seja realizada uma revisão deste artefato para verificar a necessidade de novas ações.

4.3. Ponto de Extensão 3 - Criação do Artefato *Release Backlog*

Segundo (SUTHERLAND, 1993), o ideal é que ao final de cada *Sprint* um novo incremento do software seja entregue. No entanto, algumas organizações não conseguem liberar novas versões ao final da *Sprint*, seja por questões técnicas ou insuficiência de recursos. Rubin (2012) afirma que cada organização deve definir sua frequência de entregas, criando um artefato chamado *Release Backlog*. Este artefato tem como objetivo definir as entregas do projeto.

De acordo com o Scrum, toda entrega de software deve criar valor para os interessados. No entanto, a criação de valor pode depender de outras ações não relacionadas ao projeto de software. Portanto, ao entregar um novo *release*, deve-se garantir que as ações necessárias para criar o valor esperado foram executadas. Desta forma, esse ponto de extensão sugere que seja criado o artefato *Release Backlog*. Este artefato tem o objetivo de definir as entregas do projeto de forma integrada à execução dos itens do *Action Backlog*. A Figura 9 apresenta uma sugestão de modelo para este artefato. Ele é dividido em duas partes. A primeira contém um conjunto de itens do *Product Backlog*. Estes itens serão desenvolvidos em uma ou mais *Sprints*. A segunda parte contém os itens do *Action Backlog* necessários para que os benefícios esperados pelo novo incremento do software sejam realizados. Ao final de um *Release*, espera-se que todos os itens do *Release*, tanto do *Product Backlog* quanto do *Action Backlog*, estejam concluídos.

Release 1
Itens do Product Backlog
Itens do Action Backlog
Release 2
Itens do Product Backlog
Itens do Action Backlog
Release N
Itens do Product Backlog
Itens do Action Backlog

Figura 9 – Sugestão de modelo para o *Release Backlog*

4.4. Ponto de Extensão 4 - Monitorar Riscos

Controlar e monitorar riscos são atividades comuns às diversas abordagens de gerenciamento de riscos, sendo, inclusive, uma das práticas ágeis sugeridas (HIGHSMITH, 2004). Este ponto de extensão sugere que os riscos identificados no *Business Case* sejam priorizados com base em seus fatores de exposição e continuamente controlados e monitorados durante os pontos de controle do Scrum. A responsabilidade por essa atividade é de todo time Scrum. Além disso, mudanças no projeto podem inserir novos riscos. Desta forma, riscos identificados no *Business Case* devem ser constantemente revisados.

4.5. Ponto de Extensão 5 - Monitorar a Criação de Valor

O objetivo do projeto software é criar valor para os interessados. Nesta extensão, valor é definido em termos de benefícios e custos. Assim, para garantir o sucesso projeto, deve-se monitorar se os incrementos de software entregues estão de fato gerando os benefícios esperados dentro dos custos aceitáveis. Este ponto de extensão sugere que durante a *Sprint Review*, na qual participam o time Scrum e os interessados do projeto, seja avaliado se projeto está de fato criando valor para os interessados. Essa avaliação deve ser realizada com base no artefato *Business Case*, o qual contém os benefícios e custos aceitáveis de cada parte interessada. Além disso, as expectativas dos interessados podem sofrer mudanças. Desta forma, os artefatos criados devem ser revisados e, caso necessário, adaptados.

5. AVALIAÇÃO DA EXTENSÃO PROPOSTA

Este capítulo descreve o processo de avaliação utilizado para fazer uma avaliação da extensão proposta e apresenta uma análise dos resultados obtidos.

5.1. Objetivo da Avaliação

O objetivo desta avaliação é verificar se a extensão proposta atinge seu objetivo, que é contribuir para que os softwares entregues criem valor para os interessados, mantendo o Scrum alinhado ao manifesto ágil e às práticas ágeis de projeto.

5.2. Metodologia da avaliação

Para a avaliação deste trabalho, utilizou-se uma abordagem qualitativa, onde se buscou descrever a opinião dos participantes através de uma pesquisa-ação que, segundo Gil (1991), tem como objetivo criar uma interação entre pesquisador e participantes de modo que possam colaborar com a construção do conhecimento. A pesquisa-ação foi realizada através da aplicação de um questionário a profissionais com experiência em Scrum, tendo como objetivo identificar o grau de concordância ou discordância em relação à extensão proposta. O público-alvo da avaliação foi formado por profissionais e pesquisadores (especialistas) que tinham experiência em projetos de software, especialmente em projetos Scrum.

5.3. Elaboração do Questionário

Segundo Marconi e Lakatos (2010), o questionário é um instrumento de coleta de dados constituído de uma série ordenada de, no máximo, 30 perguntas, as quais devem ser respondidas em cerca de 30 minutos, podendo ter perguntas abertas ou fechadas. Dentre suas vantagens estão: economia de tempo; maior número de pessoas respondendo simultaneamente; rapidez na obtenção das respostas; liberdade na escolha das respostas em razão do anonimato. Ele deve conter uma nota explicando a natureza da pesquisa, sua importância e a necessidade de se obter as respostas. Além disso, é necessário um pré-teste, o qual consiste em

aplicar o questionário a uma pequena população a fim de identificar falhas e inconsistências nas perguntas.

5.3.1. Elaboração das Perguntas

O questionário deve avaliar se o objetivo do trabalho foi atingido. Desta forma, as proposições foram elaboradas tendo como base os objetivos do trabalho e visam analisar:

- a importância da criação de valor em projetos de software;
- a pertinência e a abordagem definida para cada ponto da extensão;
- se a extensão proposta contribuiu para a criação de valor;
- se a extensão proposta não descaracteriza a agilidade do Scrum.

Para cada pergunta foi elaborado um pequeno resumo explicativo e, em seguida, apresentadas proposições para verificar a concordância ou discordância do especialista. Para isso, optou-se pela utilização de uma de escala de Likert (LIKERT, 1932) com cinco proposições: Concordo Totalmente (C+), Concordo Parcialmente (C-), Discordo Parcialmente (D-), Discordo Totalmente (D+), Sem Opinião (SO).

5.3.2. Teste do Questionário

Como teste, o questionário foi enviado a um grupo de seis pessoas com o objetivo de verificar a consistência, clareza, complexidade, pertinência, tempo de respostas e ordem das perguntas. A escolha das pessoas teve como base a experiência em Scrum e a experiência em trabalhos acadêmicos. Cinco responderam o questionário e enviaram sugestões de melhorias. De forma geral, todos afirmaram que as perguntas estavam adequadas e que o questionário atendia à necessidade da avaliação. No entanto, dois pontos foram unânimes. O primeiro dizia respeito ao tamanho dos textos explicativos. Eles afirmaram que estavam muito grandes e, por isso, desmotivariam os participantes. Desta forma, sugeriram que eles fossem resumidos. Outro ponto foi a ausência de exemplos. Eles afirmaram que, devido à subjetividade de alguns pontos, a ausência de exemplos dificultava o entendimento. Com base nos *feedbacks*, foram feitos os ajustes necessários no

questionário e o mesmo foi reenviado para os respondentes, os quais aprovaram as alterações.

5.4. Execução

Depois de realizados os ajustes necessários, o questionário foi criado utilizando a ferramenta online *Survey Monkey*¹⁵. O link para o questionário foi divulgado em empresas que utilizam Scrum, lista de discussões sobre Scrum e gerenciamento ágil de projetos, fóruns e grupos de profissionais Scrum e lista de pesquisadores de universidades, ficando disponível durante vinte dias (de 18 de fevereiro de 2013 a 09 de março de 2013).

5.5. Análise e Interpretação dos Resultados

Durante o período de execução, um grupo de trinta e sete pessoas¹⁶ respondeu o questionário, o qual não inclui as pessoas que participaram da etapa de teste. As seções seguintes apresentam uma análise dos resultados obtidos.

5.5.1. Perfil dos Respondentes

Para avaliar a extensão proposta, é importante que o respondente tenha experiência com projetos Scrum. Desta forma, faz-se necessário analisar o perfil dos respondentes a fim de verificar a relevância de suas respostas. As tabelas 6, 7, 8 e 9 apresentam o perfil dos respondentes quanto à escolaridade, tempo de experiência com projetos de software, tempo de experiência em Scrum, certificação e papéis desempenhados no Scrum.

¹⁵ Ferramenta online para criação de questionários. Disponível em www.surveymonkey.com.

¹⁶ Durante a elaboração do trabalho não foram encontrados números referentes à quantidade de profissionais que utilizam Scrum para determinar a representatividade da amostragem.

Tabela 6 - Perfil dos respondentes quanto à escolaridade e experiência com projetos de software

Escolaridade	Tempo de experiência com projetos de software	Quantidade de respondentes	Percentual
Ensino Superior Incompleto	1 - 3 anos	1	2,70%
Ensino Superior Incompleto	3 - 6 anos	1	2,70%
Ensino Superior Incompleto	Mais de 6 anos	1	2,70%
Ensino Superior Completo	1 - 3 anos	2	5,41%
Ensino Superior Completo	3 - 6 anos	1	2,70%
Ensino Superior Completo	Mais de 6 anos	3	8,11%
Pós-Graduação (Especialização)	1 - 3 anos	2	5,41%
Pós-Graduação (Especialização)	3 - 6 anos	2	5,41%
Pós-Graduação (Especialização)	Mais de 6 anos	6	16,22%
Pós-Graduação (Mestrado)	Menos de 1 ano	1	2,70%
Pós-Graduação (Mestrado)	1 - 3 anos	2	5,41%
Pós-Graduação (Mestrado)	3 - 6 anos	5	13,51%
Pós-Graduação (Mestrado)	Mais de 6 anos	8	21,62%
Pós-Graduação (Doutorado)	Mais de 6 anos	2	5,41%

Tabela 7 - Perfil dos respondentes quanto à experiência em projetos Scrum

Tempo de experiência com Scrum	Número de projetos	Quantidade de respondentes	Percentual
Menos de 1 ano	Nenhum projeto	2	5,41%
Menos de 1 ano	1 - 3 projetos	6	16,20%
1 - 3 anos	1 - 3 projetos	7	18,92%
1 - 3 anos	3 - 6 projetos	8	21,62%
1 - 3 anos	Mais de 6 projetos	6	16,22%

3 - 6 anos	1 - 3 projetos	2	5,41%
3 - 6 anos	3 - 6 projetos	3	8,11%
Mais de 6 anos	Mais de 6 projetos	3	8,11%

Tabela 8 - Perfil dos respondentes quanto à obtenção de certificação

Possui Certificação Scrum	Quantidade de respondentes	Percentual
Sim	8	21,6%
Não	29	78,4%

Tabela 9 - Perfil dos respondentes quanto aos papéis desempenhados no Scrum

Papeis desempenhados no Scrum	Quantidade de respondentes	Percentual
Product Owner	16	48,5%
Scrum Master	23	69,7%
Desenvolvedor	18	54,5%

Analisando os dados das tabelas acima, percebe-se que o perfil profissional predominante é composto por pessoas pós-graduadas com mais de seis anos de experiência em projetos de software. No tocante à experiência em projetos Scrum, 78,9% dos respondentes possuem mais de um ano de experiência e participaram de pelo menos um projeto. Somente dois respondentes (5,41%) afirmaram que possuem menos de um ano de experiência e que não participaram de nenhum projeto. Quanto aos papéis desempenhados, percebe-se que os três papéis do Scrum foram representados na pesquisa. Com base nos dados acima, é possível concluir que somente dois respondentes não possuem o perfil compatível com o público-alvo da avaliação, uma vez que é difícil avaliar os impactos da extensão proposta sem ter participado de nenhum projeto. Desta forma, as respostas desses respondentes não foram levadas em consideração.

5.5.2. Análise das Proposições

As seções abaixo apresentam uma análise da concordância quantos às proposições contidas no questionário. Cada respondente foi identificado através de um número sequencial. O Apêndice B apresenta o perfil de cada um dos respondentes.

5.5.2.1. Proposição 1: A verdadeira medida de sucesso de um projeto deve ser o valor criado para os interessados.

Essa proposição buscou identificar o grau de concordância dos especialistas quanto à importância de se considerar a criação de valor como fator crítico de sucesso. A Tabela 10 apresenta um resumo das respostas para essa proposição.

Tabela 10 - Criação de valor como fator crítico de sucesso

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	18	51,43%
Concordo Parcialmente	15	42,86%
Discordo Parcialmente	2	5,71%
Discordo Totalmente	0	0%
Sem Opinião	0	0%

Sete pessoas comentaram suas respostas. Três pessoas destacaram a importância da criação de valor, mas que é necessário considerar que os recursos para o desenvolvimento do projeto são limitados. Elas apresentaram os seguintes argumentos¹⁷:

- Respondente 17: “Concordo, mas é importante que o valor criado esteja dentro dos recursos disponíveis”;
- Respondente 29: “Na minha opinião a assertiva é verdadeira, pois um projeto bem sucedido deve ter o valor agregado desejado pelos seus interessados. Se isso não ocorrer, então o projeto não foi capaz de

¹⁷ Alguns comentários sofreram pequenas alterações a fim de corrigir erros de ortografia e concordância. No entanto, a semântica dos comentários não foi alterada.

atender aos seus objetivos. Entretanto, é importante sempre lembrar sobre os valores dos custos. Pois nenhum projeto poderá ter valores de custos ilimitados”;

- Respondente 23: “Creio que ficaria melhor: A verdadeira medida de sucesso de um projeto deve ser o valor que cria para os interessados, levando-se em conta custos aceitáveis”.

A preocupação percebida nos comentários acima é pertinente. Todo projeto possui recursos limitados. Conforme explicitado na Seção 4.1, a extensão proposta sugere que o artefato *Business Case* contenha uma análise da viabilidade financeira, além de descrever as restrições do projeto. Com base nessas informações, é possível realizar um controle a fim de garantir que o projeto não extrapole os recursos disponíveis.

Três pessoas argumentaram que valores como qualidade de trabalho, satisfação do time e criação de valor para a sociedade devem ser levados em consideração. Elas apresentaram as seguintes respostas:

- Respondente 3: “Depende. Sob a ótica de quem? Para o patrocinador do projeto, certamente. Para a equipe do projeto ou para a sociedade, por exemplo, não necessariamente”;
- Respondente 14: “A medida também deve levar em conta a satisfação do time. O texto acima abordado apresenta o interessado como cliente apenas”;
- Respondente 28: “Acredito que não se pode definir sucesso para projetos sem realizar um *trade-off* entre os fatores humanos do projeto. Qualidade do trabalho e produtividade da equipe estão diretamente relacionados ao ambiente de projetos”.

A VBSE sugere que um projeto só terá sucesso se criar valor para as partes interessadas. Conforme apresentado na Seção 2.3, uma parte interessada é qualquer pessoa que seja afetada pelo projeto. O time Scrum se enquadra na definição de interessado. Caso a sociedade seja afetada pelo projeto, também deve ser considerada como parte interessada. Essa extensão propõe que as proposições de valor dos interessados sejam atendidas pelo projeto. Desta forma, podemos

considerar que a extensão proposta aborda as preocupações citadas nos comentários acima.

O Respondente 31 discordou parcialmente da proposição, afirmando que: “muito maior que o valor pura e simplesmente, é preferível usar o ROI como medida de sucesso, custos aceitáveis é muito subjetivo”. Quanto a este comentário, é importante observar que, segundo a VBSE, benefícios intangíveis devem ser considerados. Melhoria da imagem da organização, fidelização de clientes, satisfação da equipe, dentre outros, são muito difíceis de traduzir em números. Assim, o uso de técnicas orientadas a aspectos financeiros não são suficientes em uma abordagem orientada a valor.

5.5.2.2. Proposição 2: A criação do artefato Business Case contribui para a criação de valor, uma vez que torna visível a todos os envolvidos quais valores as partes interessadas esperam receber.

Essa proposição buscou identificar o grau de concordância dos especialistas quanto à pertinência da criação do artefato *Business Case*. A Tabela 11 apresenta um resumo das respostas para essa proposição.

Tabela 11 - Criação do artefato *Business Case*

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	14	40,00%
Concordo Parcialmente	15	42,86%
Discordo Parcialmente	3	8,57%
Discordo Totalmente	2	5,71%
Sem Opinião	1	2,86%

Nove pessoas comentaram suas respostas. Quatro pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 35: “O artefato, por si só, não gera valor para o cliente. Entretanto, a partir do momento que se inicia o ciclo de desenvolvimento

com base no framework, o artefato contribui indiretamente para a criação de valor”;

- Respondente 22: “A criação do artefato supre diretamente um dos princípios da VBSE, melhorando a criação de valor”;
- Respondente 29: “Artefatos que aumentem a qualidade da troca de informações entre os *stakeholders* são sempre bem-vindos em um projeto, contanto que esses artefatos não engessem o processo de desenvolvimento”;
- Respondente 14: “A criação do artefato contribui para definição dos benefícios esperados com a entrega do projeto e conseqüentemente com o sucesso deste”.

Os comentários acima reforçam os argumentos apresentados para a criação do artefato. Um dos respondentes observa que o artefato contribui indiretamente para a criação de valor. Essa é a intenção dos artefatos criados nesta extensão. Eles, por si só, não criam valor, mas contribuem para a criação de valor.

Três pessoas manifestaram preocupação com a atribuição da responsabilidade pelo artefato ao papel do Product Owner. Elas fizeram os seguintes comentários:

- Respondente 33: “Embora seja minha opinião de que o time deva saber tais informações, discordo que o *Product Owner* deva ser o responsável. No entanto, vejo valor na construção desse artefato como uma atividade do time todo”;
- Respondente 5: “Um dos grandes problemas do Scrum é o *Product Owner*. Muitas vezes ele não é ninguém da organização. Em outros momentos ele é uma pessoa que possui uma visão do negócio, mas não de forma holística. Uma adaptação mais adequada nesse ponto talvez fosse a criação de uma "junta" de *Product Owners* relacionada ao negócio. Achar que apenas um resolverá a questão é falta de experiência no uso do método”;
- Respondente 24: “Concordo, mas creio que a atuação por parte do *Product Owner* não seja tão simples. Ele pode até elaborar esse documento, mas com auxílio e supervisão do *Scrum Master*”.

A preocupação acima é pertinente, uma vez que é difícil obter uma participação ativa do *Product Owner*. Apesar do artefato ser utilizado por todos envolvidos no projeto, sua elaboração está ligada mais a aspectos do negócio do que técnicos. Por esse motivo, foi atribuída ao *Product Owner* a responsabilidade pela criação deste artefato. No entanto, cada organização pode ajustar a extensão de acordo com sua realidade.

O Respondente 23 escreveu o seguinte comentário manifestando preocupação com uma possível “engessamento” do processo: “É complicado introduzir artefatos dentro de um contexto ágil. Deve-se ter bastante cuidado para não “engessar” o processo e também para não se introduzir um esforço desnecessário. Apesar disso ainda concordo parcialmente com a esta proposta, apesar de não ser muito a favor do uso de análises de risco em projetos ágeis, pois, para mim, vai de encontro à filosofia”. O objetivo deste trabalho é propor uma extensão que contribua para a criação de valor sem descaracterizar a agilidade do Scrum. Assim, procurou-se balancear os benefícios do artefato e o esforço necessário para sua elaboração. A Seção 4.1 apresentou os motivos para a criação deste artefato, os quais foram considerados suficientes para a criação do mesmo.

O Respondente 31 discordou totalmente argumentando da seguinte forma: “Posso elencar vários motivos: se este é um artefato que vai nortear trabalho de todos os interessados, por que apenas o *Product Owner* é responsável por criá-lo? Qual é o valor que esse artefato agrega ao projeto, ao produto ou ao processo? Todas as finalidades propostas ao artefato já são consideradas no Scrum. Os riscos já são mitigados pelas suas práticas, os benefícios definidos pelas responsabilidades e a linguagem comum se for um problema será descoberta já na primeira Sprint, não será um documento que vai resolver isso”. Quanto a esse comentário, alguns argumentos, como a responsabilidade do *Product Owner* e o valor agregado pelo artefato, já foram analisados. Quanto aos outros argumentos, essa extensão está alinhada à VBSE e considera que explicitar os benefícios desejados contribui para que todos busquem um objetivo comum, criando um alinhamento entre todas as partes interessadas do projeto. A mitigação de riscos será abordada na Seção 5.5.2.5.

5.5.2.3. Proposição 3: Identificar ações adicionais é essencial para garantir que os benefícios sejam realizados. Desta forma, a inclusão do artefato *Action Backlog* contribui para a criação de valor.

Essa proposição buscou identificar o grau de concordância dos especialistas quanto à pertinência da criação do artefato *Action Backlog*.

Tabela 12 - Criação do artefato *Action Backlog*

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	18	51,43%
Concordo Parcialmente	10	28,57%
Discordo Parcialmente	5	14,29%
Discordo Totalmente	1	2,86%
Sem Opinião	1	2,86%

Doze pessoas comentaram suas respostas. Três pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 17: “Tendo em vista que a entrega dos benefícios é objetivo a ser alcançado, esse artefato é pertinente”;
- Respondente 29: “Sempre é necessário ter uma visão abrangente do projeto. Conseguir enxergar quais os limites que esse projeto alcança, evitando problemas como o relatado na descrição acima”;
- Respondente 5: “A ideia é interessante, porém, o Scrum foca muito no levantamento de necessidades, mas esquece da implantação efetiva das melhorias. Assim, talvez o artefato seja um item a mais na cadeia. Porém, acredito que o grande valor esteja em criar ações de efetiva implantação dos *releases* liberados nas *Sprints*. Embora isso tenha o viés de software, é fácil adaptar para outros contextos”.

Os comentários acima apenas reforçam os argumentos apresentados para a criação do artefato. Três pessoas escreveram os comentários abaixo considerando que o artefato contribui para a criação de valor, mas não é essencial.

- Respondente 10: “Acredito que pode contribuir, mas discordo que seja essencial para garantir”;
- Respondente 20: “Ainda assim, ninguém garante que as ações listadas serão de fato realizadas. No exemplo citado, poderiam existir controles no sistema que indiquem o que deveria ter sido feito (como o registro dos professores) e não foi feito, para que os responsáveis possam tomar as devidas providências. O artefato é importante, mas não essencial”;
- Respondente 24: “Planejar sim, garantir nem tanto”.

Se as ações identificadas forem condições necessárias para a realização dos benefícios, elas tornam-se essenciais. No entanto, percebe-se que a redação da proposição não ficou adequada, uma vez que afirma que a identificação é essencial para garantir a realização dos benefícios. Na verdade, a execução efetiva das ações é que realmente contribuirá para a criação de valor. Mas, para isso, elas devem ser previamente identificadas, objetivo deste artefato.

Quatro respondentes consideraram que as ações previstas no *Action Backlog* não são de responsabilidade do time do Scrum, apresentando os seguintes comentários:

- Respondente 35: “Do ponto de vista do negócio, faz sentido a elaboração do artefato. No entanto, não o vejo como essencial para garantir que os benefícios sejam realizados. Do ponto de vista do processo de desenvolvimento, o artefato é irrelevante”;
- Respondente 23: “Pelo exemplo utilizado, esse artefato (*Action Backlog*) parece ser mais voltado para outros interessados, não para a equipe Scrum. Por isso creio que não se enquadraria bem na sua proposta. Apesar disso eu concordo com a criação do *Action Backlog* voltado para os *stakeholders* certos”;
- Respondente 31: “Não sei o que opinar, uma vez que esse é um tipo de problema que extrapola o Scrum. O time e o *Scrum Master* não têm nada a ver com este problema. Este é um problema do negócio, onde o *Product Owner* está inserido. Isso significa que para o projeto de desenvolvimento de software isso é indiferente. Isto é relevante para o projeto institucional da empresa do *Product Owner*, então não faz sentido existir esse artefato

dentro do projeto Scrum. E quanto a pergunta, eu acho que o *Action Backlog* apenas ajudaria o *Product Owner* a identificar quais itens de backlog são mais prioritários ou dependentes, mas não trará valor direta ou indiretamente. Fora de suas atividades do Scrum, o *Product Owner* pode usar a técnica que quiser para resolver os seus problemas”;

- Respondente 32: “Não vejo vantagem em misturar as atividades do processo de desenvolvimento com as atividades da operação do sistema”.

Os comentários acima consideram que os objetivos do artefato estão fora do escopo do Scrum. Um dos argumentos da VBSE é que a responsabilidade da Engenharia de Software não pode ser limitada a transformar requisitos em código funcional. Ela deve utilizar aspectos técnicos para desenvolver softwares que criem valor, sendo que, para isso, ações adicionais podem ser necessárias. Desta forma, a não execução dessas ações pode fazer com que o software produzido não atinja seus objetivos. Importante observar que, assim como o *Scrum Master* e a equipe de desenvolvimento, o *Product Owner* é um papel do Scrum, o qual possui a responsabilidade pelo projeto. Desta forma, essa extensão considera que o artefato *Action Backlog* pode ser inserido dentro do escopo do Scrum.

Duas pessoas consideraram que os itens do *Action Backlog* devem ser inseridos no artefato *Product Backlog*. Elas escreveram os seguintes comentários:

- Respondente 32: “A função do *Product Owner* inclui a avaliação de valor das histórias (itens do *Product Backlog*) e essa avaliação precisa levar em consideração esses potenciais problemas externos. Adicionar um artefato para isso me parece desnecessário, uma tentativa de substituição da responsabilidade que já é do *Product Owner* por uma lista. Esse *backlog* à parte também potencialmente pode causar um desalinhamento entre as ações e as histórias. Falando agora como *Product Owner*, é muito mais fácil se perder tendo que gerenciar duas listas priorizadas paralelas do que simplesmente fazer anotações em histórias dizendo do que elas dependem e mostrar isso para o cliente”;
- Respondente 28: “Na minha opinião, o *Product Backlog* deve ser pensado de forma a incluir todas os produtos do projeto, inclusive os produtos que não dizem respeito ao desenvolvimento de software”.

A criação de um novo artefato tem a intenção de separar os itens que serão desenvolvidos pela equipe e as ações adicionais, uma vez que estas podem não estar relacionadas a TI. Além disso, buscou-se manter a atual estrutura do *Product Backlog*. No entanto, a essência do artefato é registrar as ações. Assim, caso necessário, cada time do Scrum pode adaptar a extensão e registrar as ações da forma que entender mais adequada.

5.5.2.4. Proposição 4: A criação do artefato Release Backlog é necessária para garantir as entregas de software gerem os benefícios esperados.

Essa proposição buscou identificar o grau de concordância dos especialistas quanto à pertinência da criação do artefato *Release Backlog*.

Tabela 13 - Criação do artefato *Release Backlog*

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	15	42,86%
Concordo Parcialmente	13	37,14%
Discordo Parcialmente	3	8,57%
Discordo Totalmente	3	8,57%
Sem Opinião	1	2,86%

Nove pessoas comentaram suas respostas. Quatro pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 20: “Concordo, porque a equipe pode não ter essa visão ou ter uma visão míope”;
- Respondente 23: “O *Product Owner* já é o responsável por gerenciar essas entregas. Caberia a ele, então, gerenciar os *Sprint Backlogs* de forma a existir o sincronismo com o *Action Backlog*”;
- Respondente 29: “O artefato proposto, provavelmente, é válido. Entretanto, não garante que o incremento será entregue de acordo com as expectativas pelo simples fato de adicionar um novo artefato ao

desenvolvimento. Os *stakeholders* poderão ignorá-lo ao final de uma *Sprint* caso se deparem com problemas críticos ou de qualquer outra natureza”.

Duas pessoas manifestaram preocupação com a atribuição da responsabilidade pelo artefato ao *Product Owner*:

- Respondente 5: “Essa está diretamente relacionada ao meu comentário anterior. Parece muito apropriada. Mas o fato de concentrar tudo no *Product Owner*, que é um único ser com limitações no conhecimento do negócio, pode inviabilizá-la”;
- Respondente 24: “Mais uma vez, creio que essa responsabilidade ficando somente a cargo do *Product Owner* possa prejudicar um pouco o projeto, dependendo do nível de envolvimento dele e da capacidade técnica”.

A atribuição ao *Product Owner* decorre dos mesmos motivos já apresentados na Seção 5.5.2.2. Três respondentes discordaram da criação do artefato com os seguintes comentários:

- Respondente 33: “Esse passo é o demonstrativo do custo operacional que essa abordagem produz, como comentado na resposta anterior. O Scrum já teve, no passado, um *Release Planning* e o gráfico *Release Burndown*. Esses itens foram removidos do framework. Fica a pergunta, isso não é um retrocesso? Do ponto de vista de desenvolvedora, sei que há inúmeras formas de tornar o *deploy* algo bem menos doloroso: automatização de build e da publicação em homologação, *one-click deploy* e outras tantas possibilidades. Colocar esse artefato no processo pode fazer com que o time considere a demora entregar incrementos como um problema, e isso é perigoso. A razão de *Sprints* serem curtos é mitigar riscos no projeto e deixar de fazer entregas frequentes e desde cedo de software de valor priva o time de diversas oportunidades de feedback”;
- Respondente 32: “Já existe *Release Planning* no Scrum, vide o livro *Essential Scrum*”;
- Respondente 31: “Igual à questão anterior. A criação do artefato é relevante apenas para o negócio do *Product Owner* e não para o time Scrum. Assim, acho que ele não se aplica ao contexto Scrum, caindo

naquela questão: O *Product Owner* fora do projeto Scrum faz o que bem entende para gerenciar suas demandas”.

Dois dos comentários acima afirmam que no Scrum já existe o artefato *Release Backlog*. De fato, é possível encontrar registro de autores que mencionam a eventual necessidade de tê-lo no processo. No entanto, conforme apresentado na Seção 4.3, o ideal é que ao final de cada *Sprint* um incremento seja entregue. O que motivou a criação do artefato foi a necessidade de sincronizar a entrega dos incrementos de software com eventuais ações necessárias que tais incrementos criem valor. Importante observar que a periodicidade das entregas é definida pelo time, podendo ser a cada trinta dias. Por último, um dos comentários argumenta que o artefato extrapola o escopo do Scrum. Importante observar que *Product Owner* é um papel do Scrum. Isso significa que todas suas atribuições estão dentro do escopo do processo.

5.5.2.5. Proposição 5: Abordar riscos explicitamente é importante para garantir a criação de valor.

Essa proposição buscou identificar o grau de concordância dos especialistas quanto à pertinência de uma abordagem explícita de riscos. A Tabela 14 apresenta um resumo das respostas para essa proposição.

Tabela 14 - Resultado da proposição que aborda o gerenciamento explícito de riscos

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	21	60,00%
Concordo Parcialmente	9	25,71%
Discordo Parcialmente	4	11,43%
Discordo Totalmente	1	2,86%
Sem Opinião	0	0,00%

Oito pessoas comentaram suas respostas. Cinco pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 35: “Entendo que tal extensão agrega valor ao uso do Scrum”;
- Respondente 22: “Creio que uma gestão explícita de riscos é essencial para maior efetividade do Scrum com relação à criação de valor”;
- Respondente 29: “Essa abordagem evitaria que alguns fatores importantes fossem esquecidos temporariamente pelos *stakeholders*. Causando problemas graves no decorrer do desenvolvimento do projeto”;
- Respondente 5: “Em muitas empresas de desenvolvimento, riscos são continuamente conhecidos, mas a alta direção não faz nada para mitigá-los ou resolvê-los. Conhecer sempre é bom, mas é fundamental criar mecanismos para expô-los a alta direção”;
- Respondente 20: “Todos esses artefatos ajudam bastante na definição dos requisitos de *backlog* das *Sprints* e, conseqüentemente, na entrega dos benefícios esperados pelo cliente?”.

Os comentários acima reforçam os argumentos apresentados para a criação deste artefato. Três pessoas comentaram afirmando que o Scrum já aborda implicitamente o gerenciamento de riscos.

- Respondente 33: “Isso já deve acontecer em projetos que usam Scrum. É papel do time (*Product Owner*, desenvolvedores e *Scrum Master*) e do cliente atentar para esses aspectos. Monitoramento de riscos sempre esteve presente e por isso não a considero uma extensão”;
- Respondente 17: “Gerenciar riscos é importante. No entanto, acredito que o Scrum já gerencie implicitamente”;
- Respondente 31: “Se grande parte dos riscos do projeto já foram mitigados, então por que gerenciar a minoria? Qual ganho se tem com isso? Como esse valor se transfere do projeto para o produto final?”.

Não há discordância quanto ao fato das práticas do Scrum gerenciarem implicitamente os riscos. No entanto, conforme apresentado na Seção 0, a abordagem explícita foi incluída devido a eventual necessidade de ações adicionais contidas no artefato *Action Backlog*. Essas ações, de TI ou não, são necessárias para que os benefícios sejam realizados. Desta forma, riscos associados a essas ações precisam ser gerenciados.

5.5.2.6. Proposição 6: Monitorar se os incrementos de software entregues estão de fato realizando os benefícios esperados é importante para realizar adaptações no produto

Essa proposição buscou identificar o grau de concordância dos respondentes quanto à pertinência de se monitorar a realização de benefícios. A Tabela 15 apresenta um resumo das respostas para essa proposição.

Tabela 15 - Resultado da proposição que aborda o monitoramento da realização de benefícios

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	24	68,57%
Concordo Parcialmente	9	25,71%
Discordo Parcialmente	1	2,86%
Discordo Totalmente	1	2,86%
Sem Opinião	0	0,00%

Oito pessoas comentaram suas respostas. Cinco pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 33: “Concordo plenamente com a frase. A escolha por *Concordo Parcialmente* foi, novamente, porque isso não se trata de uma extensão. Esse é precisamente o papel da *Sprint Review Meeting* no Scrum”;
- Respondente 13: “Só tem que ter cuidado para que essas adaptações não tenham *Gold Plating*”;
- Respondente 17: “Controlar os recursos também é fundamental”;
- Respondente 29: “Tarefa extremamente relevante. Tenta sempre manter o projeto alinhado de acordo com as necessidades mutáveis dos clientes”;
- Respondente 35: “Concordo com a afirmação, mas tenho dúvidas se o momento para realizar esse monitoramento é a *Sprint Review*. Talvez uma outra reunião entre o *Product Owner* e os interessados seja mais adequada”.

Duas pessoas escreveram os seguintes comentários afirmando a *Sprint Review* já teria essa finalidade:

- Respondente 22: “*Sprint Review* já é um ponto de controle”;
- Respondente 23: “Não consegui identificar qual extensão foi feita neste caso. Talvez seja apenas o *Business Case*. Se for eu acho interessante este artefato, apesar das restrições que citei em outra resposta”.

Não há discordância a cerca dos comentários acima. De fato, a *Sprint Review* é um ponto de controle do Scrum. O que se busca nessa extensão é monitorar se os benefícios estão de fato sendo realizados, e não somente verificar as entregas de software. Para isso, é importante o uso do artefato *Business Case*, também proposto nesta extensão. A criação de uma nova reunião, conforme sugerido por um dos respondentes, seria de difícil operacionalização, devido à dificuldade de reunir todos os interessados.

O Respondente 31 discordou afirmando que não é papel do time do Scrum realizar esta tarefa. Ele afirma que: “Novamente está existindo uma confusão nas responsabilidades. O time (e consequentemente a *Sprint Review*) não tem como objetivo monitorar a criação de valor. Isto tem que ser feito antes da *Sprint* começar, na priorização do *backlog*. O *Product Owner* tem que fazer uma análise do ROI e descobrir quais funcionalidades aumentarão esse ROI antes dela ser implementada, não depois”. Há uma discordância quanto ao comentário acima. Se o objetivo do projeto é entregar software que gere os benefícios esperados pelos interessados, deve haver um monitoramento da realização dos benefícios, sendo que este só pode ser feito após a entrega do software. Novamente, os benefícios esperados não são limitados a aspectos financeiros. Assim, o uso exclusivo de técnicas financeiras se torna insuficiente em abordagens orientadas a valor.

5.5.2.7. Proposição 7: A extensão proposta neste trabalho não descaracteriza a agilidade do Scrum.

Essa proposição buscou identificar o grau de concordância dos especialistas quanto à manutenção da agilidade do Scrum com implementação das extensões propostas. A Tabela 16 apresenta um resumo das respostas para essa proposição.

Tabela 16 - Resultado da proposição que avalia se a extensão proposta não descaracteriza o Scrum

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	14	40,00%
Concordo Parcialmente	17	48,57%
Discordo Parcialmente	3	8,57%
Discordo Totalmente	1	2,86%
Sem Opinião	0	0,00%

Dez pessoas comentaram suas respostas. Três pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 22: “Os artefatos aparentemente intervêm na qualidade do desenvolvimento do produto e processo de negócio, mantendo o processo de desenvolvimento do software ágil”;
- Respondente 15: “Acredito que a geração, atualização e monitoração de mais artefatos geram um *overhead* inicial que, a longo prazo, podem ou não comprometer a agilidade. De forma geral, eu concordo. Porém vale ressaltar o esforço a mais para manter esses artefatos”;
- Respondente 20: “Acho que os artefatos não atrapalham a dinâmica das *Sprints* e só ajudam os engenheiros a atacarem os requisitos certos, que de fato irão gerar valor para o cliente. A criação desses artefatos fará com que o *Product Owner* deixe registrado o se supõe que ele implicitamente faz ao priorizar os requisitos, que era definir as entregas de acordo com os benefícios esperados”.

Os comentários realçam os argumentos utilizados para criar essa extensão. Duas pessoas observaram que, como qualquer framework, as extensões propostas devem ser customizadas:

- Respondente 17: “Acredito que, como qualquer *framework*, ele deve ser customizado para as necessidades de cada organização”;

- Respondente 29: “É importante considerar que a proposta é uma extensão de um framework para a criação de processos para desenvolvimento. Dessa forma, cada ambiente de desenvolvimento deverá adaptar os artefatos propostos de acordo com suas necessidades. Considerando adicionar ou remover alguns artefatos ao Scrum. Com isso o Scrum não será afetado negativamente pela extensão”.

Há total concordância com os comentários acima. A essência da extensão é contribuir para que os projetos criem valor para os interessados. No entanto, os artefatos sugeridos podem ser alterados para adequá-los ao contexto em questão. Duas pessoas argumentam que os novos artefatos geram a necessidade de mais gestão, o que pode impactar na agilidade do método. Elas afirmam que:

- Respondente 28: “A inserção de novas fases no processo, como o *Action Backlog*, pode produzir mais reuniões, mais documentação e consequentemente necessidade de mais gestão”;
- Respondente 1: “Do meu ponto de vista, as extensões propostas trazem práticas de gestão de projetos que tiram parte da agilidade do Scrum”;
- Respondente 10: “Acredito que estas mudanças podem ajudar. Porém acredito que o custo para *Sprints* com duas semanas pode ser muito elevado”;
- Respondente 33: “Estritamente pelo manifesto e pelos princípios ágeis, a agilidade não é descaracterizada por esse modelo. De qualquer forma, volto a dizer que qualquer forma de burocratização a mais torna o processo mais moroso e mais pesado, podendo, sim, prejudicar a agilidade ao não ver como problema a entrega mais esparsa de valor e ao sobrecarregar o *Product Owner*, que já tem um bocado a fazer em um ciclo padrão do Scrum. Mais ainda, essa extensão parece tentar eliminar a necessidade de entendimento da visão de negócios do restante do time. Eu vejo isso de uma forma extremamente negativa. Ao dar tanta reponsabilidade ao *Product Owner*, o modelo permite que os desenvolvedores e *Scrum Master* não deem atenção à razão pela qual o software está sendo construído e ao seu valor para o cliente. Vejo essa

mentalidade em fábricas de software e acredito que reintroduzi-la a times supostamente ágeis é dar um grande passo para trás”;

- O Respondente 31: “Este trabalho para dentro do Scrum uma responsabilidade que não é dele, que é gerenciar o negócio do *Product Owner*, que nada tem a ver com o negócio do time”.

Conforme destacou um dos respondentes, levando em consideração os princípios e valores do manifesto ágil, a extensão proposta não descaracteriza o Scrum. Evidente que a inserção de novos artefatos gera um esforço a mais. No entanto, esse esforço contribui para que os projetos atinjam seus objetivos, que é criar valor para os interessados. Quanto a eliminar a necessidade de o time entender do negócio, esse não é um objetivo da extensão. Pelo contrário, o artefato *Business Case* tem a finalidade de definir quais benefícios são esperados pelos interessados e compartilhar um objetivo comum entre todos os envolvidos.

O Respondente 35 sugere uma validação da extensão em uma fábrica de software: “Aparentemente não, mas não é por meio desta pesquisa que você irá chegar à resposta para a questão. Recomendo que sejam feitas validações do framework estendido em fábricas de software”. A recomendação contida no comentário acima é pertinente. No entanto, dentro do prazo disponível, não foi possível realizar essa validação.

5.5.2.8. Proposição 8: De forma geral, a extensão proposta neste trabalho contribui para que projetos que utilizam o Scrum criem valor para os interessados.

Essa proposição buscou identificar o grau de concordância dos respondentes quanto à manutenção da agilidade do Scrum com implementação das extensões propostas. A Tabela 17 apresenta um resumo das respostas para essa proposição.

Seis pessoas comentaram suas respostas. Três pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 23: “Concordo por achar que essa extensão vai deixar ainda mais o foco a criação de valor”;

- Respondente 22: “A extensão aparentemente abrange os princípios da VBSE, que visam criar valor aos interessados”;
- Respondente 4: “A criação de um *Business Case*, juntamente com um *Action Backlog*, me pareceram as ações mais efetivas para caminhar nestes sentido, bem como a proposta de monitoramento de valor entregue ao cliente. Em suma, o *Business Case* me pareceu um termo de abertura do projeto, o *Action Backlog* soou como um conjunto de ações necessárias para que o software atinja os objetivos estratégicos - entrega de valor - para os quais foi concebido, e o monitoramento é uma prática essencial para que os dois artefatos anteriores funcionem”.

Tabela 17 - Avaliação geral da Extensão

Grau de Concordância	Respostas	Percentual
Concordo Totalmente	23	65,71%
Concordo Parcialmente	9	25,71%
Discordo Parcialmente	2	5,71%
Discordo Totalmente	1	2,86%
Sem Opinião	0	0,00%

Seis pessoas comentaram suas respostas. Três pessoas fizeram os seguintes comentários concordando com a proposição:

- Respondente 23: “Concordo por achar que essa extensão vai deixar ainda mais o foco a criação de valor”;
- Respondente 22: “A extensão aparentemente abrange os princípios da VBSE, que visam criar valor aos interessados”;
- Respondente 4: “A criação de um *Business Case*, juntamente com um *Action Backlog*, me pareceram as ações mais efetivas para caminhar nestes sentido, bem como a proposta de monitoramento de valor entregue ao cliente. Em suma, o *Business Case* me pareceu um termo de abertura do projeto, o *Action Backlog* soou como um conjunto de ações necessárias para que o software atinja os objetivos estratégicos - entrega de valor -

para os quais foi concebido, e o monitoramento é uma prática essencial para que os dois artefatos anteriores funcionem”.

Três respondentes acreditam que benefícios gerados pela extensão não compensam os esforços destinados à criação dos novos artefatos.

- Respondente 33: “Há ambientes em que tal modelo poderia ser utilizado, mas acredito fortemente que através de retrospectivas bem executadas qualquer time que realmente precise dessa burocratização chegaria nela por suas próprias pernas. E, nesse caminho, ainda poderia se fortalecer como time e descobrir formas melhores de ajustar seus comportamentos para sequer precisar de tantos novos artefatos”;
- Respondente 10: “Acredito que não contribui o suficiente para justificar o esforço. Na minha opinião, existem outras formas menos custosas e mais ágeis de contribuir para o projeto”;
- Respondente 31: “Pode auxiliar o *Product Owner* a identificar o valor das funcionalidades e ajudar a priorizar, mas para o time como um todo só gera interrupções no trabalho”.

5.5.3. Considerações Sobre a Análise

Analisando as respostas obtidas, é possível inferir que a maior parte dos respondentes concorda com a proposta de extensão apresentada neste trabalho. No entanto, é importante observar as limitações dessa avaliação. Em um questionário não é possível explicar em detalhes todos os conceitos e motivações envolvidos na elaboração da extensão. Além disso, para que houvesse número suficiente de respondentes, o único requisito para responder o questionário foi ter conhecimento no método Scrum. Dessa forma, os conceitos e elementos-chave da VBSE, os quais serviram de base para a extensão, não foram abordados. As análises apresentadas na seção anterior mostraram que muitos dos questionamentos são abordados pela extensão, no entanto, não foram compreendidos pela omissão dos conceitos da VBSE.

Outro fator a observar é que as respostas de cada participante são influenciadas pelas experiências que cada um teve com o método Scrum, as quais

podem ser totalmente diferentes. Além disso, os valores definidos no manifesto ágil podem ter diferentes interpretações, o que pode levar a formação de opiniões diferentes sobre a extensão.

6. CONCLUSÃO

Este capítulo apresenta as principais contribuições desta dissertação, sugestões para trabalhos futuros e apresenta as considerações finais.

6.1. Principais Contribuições

A criação de valor em projetos de software é um tema atual e que vem ganhando atenção de profissionais e pesquisadores. Nos últimos anos, houve um aumento na taxa de sucesso em projetos de software. Esse aumento foi associado à utilização de métodos ágeis. No entanto a taxa de falha permanece elevada. Segundo Boehm (2006a), a maior parte dos projetos falha por não criar valor para os interessados. Desta forma, entender como projetos de software criam valor para os interessados é fundamental para aumentar a taxa de sucesso. Neste contexto, as principais contribuições deste trabalho foram:

- **Explicitar a criação de valor em projetos de software ágeis:** métodos ágeis possuem como princípio a entrega de valor. No entanto, não há uma definição clara sobre como esse valor é criado e entregue. Essa extensão apresenta uma definição clara de valor e define como ele é criado e entregue aos interessados, apresentando meios para contribuir com esse objetivo;
- **Avaliar a aderência de um método ágil à VBSE:** a VBSE apresenta princípios e elementos-chave que visam adicionar considerações de valor no desenvolvimento de software. Este trabalho analisou com um método ágil aborda a VBSE;
- **Propor uma extensão como foco na criação de valor:** a principal contribuição deste trabalho foi propor uma extensão do Scrum que o alinha aos princípios defendidos pela VBSE, possibilitando melhorar criação de softwares que gerem os benefícios esperados pelos interessados.

6.2. Trabalhos Futuros

A criação de valor em projetos de desenvolvimento de software é um tema emergente que pode ser aplicado em todas as atividades da Engenharia de Software. Desta forma, algumas possibilidades de pesquisa podem ser derivadas deste trabalho, tais como:

- aplicar a extensão proposta neste trabalho em uma organização real, a fim de comparar os resultados de um projeto que utiliza a extensão com projetos que utilizam o Scrum em sua forma original;
- existem diversos métodos ágeis na literatura (HIGHSMITH, 2002). Uma possível pesquisa seria investigar como outros métodos ágeis abordam a criação de valor;
- investigar como as práticas atuais da Engenharia de Software criam valor para os interessados;
- mensurar o valor entregue aos interessados é uma tarefa complexa, uma vez que envolve tanto benefícios tangíveis quanto intangíveis. Uma possibilidade de trabalho seria investigar métodos para quantificar o valor entregue.

6.3. Considerações Finais

O objetivo deste trabalho foi avaliar a aderência do método ágil Scrum aos princípios e elementos-chave da VBSE e, com base nessa análise, propor uma extensão que contribuísse para a criação de valor, levando em consideração os princípios e valores definidos no manifesto ágil. A análise realizada no Capítulo 3 mostrou que alguns princípios e elementos-chave da VBSE já são contemplados pelo Scrum em sua forma original. De fato, o Scrum tem como princípio a entrega de valor. A extensão proposta neste trabalho não tem o objetivo de alterar o Scrum para que ele crie valor, e sim contribuir para que esse objetivo seja atingido.

A avaliação conduzida mostrou que a maioria dos participantes manifestou concordância com a extensão proposta e que o foco do projeto deve ser a criação de valor. No entanto, as seguintes preocupações foram constantemente observadas nas respostas:

- necessidade de controlar os recursos, uma vez que nenhum projeto possui recursos ilimitados;
- aumento do número de atribuições ao papel *Product Owner*;
- possível burocratização e morosidade do processo; e
- atribuição de responsabilidades que não estão dentro do escopo do Scrum;

As preocupações acima são pertinentes e foram discutidas na Seção 5.5. É importante observar que a essência da extensão é incorporar considerações de valor ao processo do Scrum. Como em qualquer método de desenvolvimento de software, os artefatos e atividades sugeridos podem ser customizados para o contexto ao qual serão aplicados. Infelizmente, devido ao prazo disponível, não foi possível realizar uma avaliação da extensão em projetos reais, comparando os resultados de projetos com e sem a extensão. No entanto, o método de avaliação realizado contou com um número significativo de pessoas com experiência em Scrum. As respostas e comentários contribuíram para demonstrar a relevância e pertinência do trabalho. Por fim, espera-se que este trabalho contribua para que times Scrum possam desenvolver software que entreguem os benefícios esperados pelos interessados.

REFERÊNCIAS

ARNUPHAPTRAIRONG , T. **Top Ten Lists of Software Project Risks:** Evidence from the Literature Survey. Proceedings of the International MultiConference of Engineers and Computer Scientists 2011. Hong Kong: 3. 2011. p. 732-737.

BECK, K. et al. **Manifesto for Agile Software Development**, 2001. Disponível em: <<http://agilemanifesto.org>>. Acesso em: 13 abr. 2012.

BERRY, M.; AURUM, A. Measurement and Decision Making. In: BIFFL, S., et al. **Value-Based Software Engineering**. Berlim: Springer, 2006. p. 156-177.

BIFFL, S. et al. **Value-Based Software Engineering**. Berlim: Springer, 2006.

BOEHM, B. Software Risk Management: Principles and Practices. **Software, IEEE**, v. 8, p. 902-916, 1991.

BOEHM, B. Value-Based Software Engineering: Overview and Agenda. In: BIFFL, S., et al. **Value-Based Software Engineering**. Berlim: Springer, 2006a. p. 3-14.

BOEHM, B. Value-Based Software Engineering: Seven Key Elements and Ethical Considerations. In: BIFFL, S., et al. **Value-Based Software Engineering**. Berlim: Springer, 2006c. p. 109-132.

BOEHM, B.; JAIN, A. An Initial Theory of Value-Based Software Engineering. In: BIFFL, S., et al. **Value-Based Software Engineering**. Berlim: Springer, 2006b. p. 16-37.

BOEHM, B.; ROSS, R. Theory-W Software Project Management: Principles and Examples. **IEEE Transactions on Software Engineering**, 7, 1989. 902 - 916.

BOEHM, B.; SULLIVAN, K. Software Economics: A Roadmap. **Proceedings of the Conference on The Future of Software Engineering**, p. 319-343, 2000.

BOEHM, B. Stakeholder Value Proposition Elicitation and Reconciliation. In: BIFFL, S., et al. **Value-Based Software Engineering**. Berlim: Springer, 2006d. p. 122-154.

COHEN, D. E. A. **Agile software development: a DACS state of art report**. NY: **Data Analysis Center for Software**. Fraunhofer Center for Experimental Software Engineering Maryland and The University of Maryland. Rome, p. 71. 2003.

COHN, M. **Succeeding with Agile - Software Development using Scrum**. Boston: Addison-Wesley, 2009.

COHN, M. Managing Risk on Agile Projects with the Risk Burndown Chart. **Mountain Got Software**, 2011. Disponível em: <<http://www.mountaingotsoftware.com/blog/managing-risk-on-agile-projects-with-the-risk-burndown-chart>>. Acesso em: 03 jan. 2013.

D'ANUNCIAÇÃO, Gustavo Tibério. **Uma Extensão do Rational Unified Process baseada na Criação de Valor**. 2009. 139 f. Dissertação (Mestrado) - Departamento de Centro de Informática, Universidade Federal de Pernambuco, Recife, 2009.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 1991.

GUSMÃO, C. M. G.; MOURA, H. P. **Gerência de Risco em Processos de Qualidade de Software**: uma Análise Comparativa. III Simpósio Brasileiro de Qualidade de Software. Brasília: [s.n.]. 2004. p. 1-398.

HIGHSMITH, J. **Agile Software Development Ecosystems**. Boston: Addison Wesley, 2002.

HIGHSMITH, J. **Agile Project Managemet: Creating Innovative Products**. Redwood City: Addison-Wesley, 2004.

IBM. **Rational Unified Process**, 2001. Disponível em: <<http://www.wthreex.com/rup/portugues/index.htm>>. Acesso em: 03 nov. 2012.

IBM. Atividade: Desenvolver Caso de Negócio. **Rational Unified Process**, 2001. Disponível em: <[http://www.wthreex.com/rup/process/activity/ac_dbzcs.htm#Describe the Product](http://www.wthreex.com/rup/process/activity/ac_dbzcs.htm#Describe%20the%20Product)>. Acesso em: 1 nov. 2012.

IEEE STD 1540- 2001. **IEEE Standard for Software Life Cycle**. IEEE. New York, p. 30. 2001.

ISACA. **Cobit 5 - A Business Framework for the Governance and Management of Enterprise IT**. [S.l.]: [s.n.], 2012.

JAN, N.; MUHAMMAD, I. Master Thesis Report_MSE 2010-40, 2010.

KHAJAVINIA, R. **The basis for building a business case in software development, a case study**. Eurocon 2009, IEEE. St.-Petersburg: . 2009. p. 379-385.

KOTLER, P. **Administração de Marketing**. 10. ed. São Paulo: Prentice Hall, 2000.

LARMAN, C. **Agile & Iterative Development - A Manager's Guide**. [S.l.]: Addison-Wesley, 2003.

LIKERT, Rensis. **A technique for the measurement of attitudes**. Archives of psychology, 1932.

MARÇAL, Ana Sofia Cysneiros. **SCRUMMI: Um processo de gestão ágil baseado no SCRUM e aderente ao CMMI**. 2009. 205 f. Dissertação (Mestrado) - Departamento de Centro de Ciências Tecnológicas, Universidade de Fortaleza, Fortaleza, 2009.

MARCONI, M. D. A.; LAKATOS, M. **Fundamentos da Metodologia Científica**. 7. ed. São Paulo: Atlas, 2010.

MUNNS, A. K.; BJEIRMI, B. F. **The role of project management in achieving project success**, 1996.

NYFJORD, J. **Outlining a Model Integrating Risk Management and Agile Software Development**. Software Engineering and Advanced Applications, 2008. SEAA '08. 34th Euromicro Conference. Parma: . Setembro 2008. p. 476-483.

OLUGBODE, M.; RICHARDS, R.; BISS, T. The role of information technology in achieving the organisation's strategic development goals: A case study. **Information Systems**, Oxford, 32, n. 5, Julho 2007. 641 - 648.

P, J. **CMM in Practice: Processes For Executing Software Projects At Infosys**. : Addison-Wesley, 1999.

PHAM, A.; PHAM, P. **Scrum in Action - Agile Software Project Management and Development**. Boston: Cengage, 2012.

PICHLER, R. **Agile Product Management with Scrum - Creating Products that Customers Love**. Boston: Addison-Wesley, 2010.

PMI. **Guia PMBOK - Um Guia do Conhecimento em Gerenciamento de Projetos**. 4. ed. Newtown Square: Project Management Institute, 2008.

PRESSMAN, R. **Engenharia de Software**. 6. ed. São Paulo: Mcgraw-hill Interamericana, 2006.

RIBEIRO, L. et al. **A case study for the implementation of an agile risk management process in multiple projects environments**. Management of Engineering & Technology, 2009. PICMET 2009. Portland International Conference on. Portland: . Agosto 2009. p. 1396-1404.

ROBERTSON, S. Requirements and the business case, v. 21, p. 93-95, Outubro 2004.

RUBIN, K. S. **Scrum Essentials**. Arbor: Addison Wesley, 2012.

SCHWABER, K. **Agile Project Management with Scrum**. : Microsoft Press, 2004.

SCHWABER, K.; SUTHERLAND, J. **Scrum Guide - Um guia definitivo para o Scrum**. 2011.

SEI. Software Risk Evaluation (SRE) Method Description (Version 2.0).

SHARM, D.; AURUM, A.; PAECH, B. Business Value through Product Line Engineering – A Case Study, 2008.

SOMMERVILLE, I. **Engenharia de Software**. 8. ed. São Paulo: Pearson Addison-Wesley, 2007.

STANDISH GROUP. **Chaos Report**. Standish Group. 2010.

SULLIVAN, K.; CHALASAN, P.; SAZAWAL, V. Software Design as an Investment Activity: A Real Options Perspective. **University of Virginia Department of Computer Science**, 1998.

SUTHERLAND, J. **The Scrum Papers: Nut, Bolts, and Origins of an Agile Framework.** 1993.

TAKEUCHI, H.; NONAKA, I. The new new product development game. **Havard Business Review**, 1986.

THOMAS, G.; FERNÁNDEZ, W. Success in IT projects A matter of definition ? **International Journal of Project Management**, 2008.

THORP, J. **The Information Paradox: Realizing the Business Benefits.** : Fujitsu Consulting, 2007.

VERSIONONE. **State of Agile Survey.** p. 9. 2011.

VETSCHERA, R. Preference-Based Decision Support in Software Engineering. In: BIFFL, S., et al. **Value-Based Software Engineering.** Berlim: Springer, 2005. p. 67-89.

WINTER, M.; SMITH, C.; MORRIS, P. Directions for future research in project management: The main findings of a UK government-funded research network. **International Journal of Project Management**, p. 638-649, 2006.

ZANATTA, A. L. **xScrum: uma proposta de extensão de um Método Ágil para Gerência e Desenvolvimento de Requisitos visando adequação ao CMMI.** 180 f. Dissertação (Mestrado) - Programa de Pós-Graduação em Ciência da Computação, Universidade Federal de Santa Catarina, Florianópolis, 2009.

APÊNDICE A – QUESTIONÁRIO DE AVALIAÇÃO

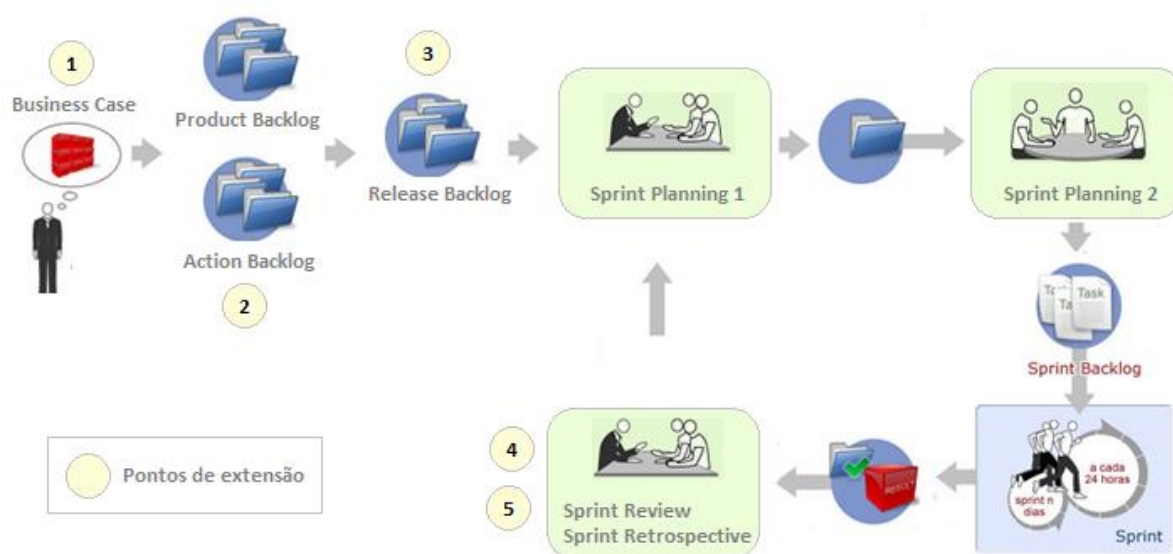
Apresentação da pesquisa

Este questionário visa avaliar uma proposta de extensão do Scrum que tem como foco a criação de valor. Essa proposta é parte do trabalho de curso de mestrado profissional do Centro de Informática da Universidade Federal de Pernambuco – UFPE.

Tempo estimado para responder o questionário: 12 minutos;

Resumo

A maioria dos estudos sobre fatores críticos de sucesso aponta que projetos de software falham por não entregar valor aos interessados. A Engenharia de Software Baseada em Valor - VBSE é uma abordagem que visa incorporar atividades de criação de valor nos processos de desenvolvimento de software. Segundo ela, o sucesso de um software vai mais além do que uma simples adequação a modelos técnicos. Excelência técnica é altamente desejável, mas insuficiente. Muitos dos mais lucrativos softwares do mundo são notórios em bugs e falhas. Nos últimos anos, tem crescido a utilização de métodos ágeis, sendo que, segundo pesquisas recentes, Scrum é o método ágil mais utilizado. Desta forma, este trabalho realizou uma análise da aderência do Scrum aos princípios da VBSE, e, com base nessa análise, foi proposta uma extensão que visa contribuir para que os softwares desenvolvidos criem valor às partes interessadas. A figura abaixo apresenta uma visão geral da extensão. Ela será detalhada nos próximos passos.



Perfil do participante

1) Nome (opcional): _____

2) Email (opcional): _____

3) Escolaridade (maior grau):

- ☐ Ensino Fundamental
- ☐ Ensino Médio
- ☐ Ensino Superior Completo
- ☐ Pós-Graduação (Especialização)
- ☐ Pós-Graduação (Mestrado)
- ☐ Pós-Graduação (Doutorado)

4) Instituição: _____

5) Qual seu tempo de experiência com projetos de software?

- ☐ Menos de um ano
- ☐ 1- 3 anos
- ☐ 3 – 6 anos
- ☐ Mais de 6 anos

6) Qual seu tempo de experiência com Scrum?

- ☐ Menos de um ano
- ☐ 1- 3 anos
- ☐ 3 – 6 anos
- ☐ Mais de 6 anos

7) Quantos projetos Scrum você participou?

- ☐ Nenhum projeto
- ☐ 1- 3 projetos
- ☐ 3 – 6 projetos
- ☐ Mais de 6 projetos

8) Possui certificação Scrum?

- ☐ Sim
- ☐ Não

9) Se possuir, especifique suas certificações:

10) Quais papéis você já desempenhou em projetos Scrum?

- ☐ Product Owner
- ☐ Scrum Master
- ☐ Desenvolvedor

Criação de valor como fator crítico de sucesso

No contexto deste trabalho, valor é definido em função dos benefícios esperados por um interessado e os custos que ele está disposto a assumir. Importante diferenciar benefícios de funcionalidades. Normalmente, benefícios estão ligados a objetivos do negócio. Por exemplo, um interessado pode esperar como benefício um aumento de 20% no número de vendas. O software deve possuir funcionalidades que gerem esse benefício. Já os custos incluem, além do custo monetário, todos os esforços necessários para a utilização do benefício. Portanto, a criação de valor se dá através da entrega funcionalidades que geram os benefícios esperados a custos aceitáveis.

Um projeto poder ser um sucesso em termos de custos e cronograma, mas um fracasso do ponto de vista da criação de valor para os interessados.

11) A verdadeira medida de sucesso de um projeto deve ser o valor criado para os interessados.

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente
- ☐ Sem Opinião

Caso deseje, comente sua resposta:

Ponto de Extensão 1 - Criação do artefato Business Case

Em muitas organizações, engenheiros de software e executivos de negócio Em muitas organizações, engenheiros de software e executivos de negócio vivem em mundos separados, usando suas próprias terminologias, critérios de decisão e métodos de trabalho. Desta forma, muitos engenheiros de software agem como se fossem “artistas criativos”, estando interessados somente em criar novas funcionalidades. Este artefato tem como finalidade criar uma linguagem comum entre todos os envolvidos no projeto e tornar visível todos os benefícios esperados pelos interessados, além de identificar os principais riscos do projeto. Sugere-se que o Business Case contenha os seguintes itens: Resumo do Projeto; Contexto Estratégico; Partes Interessadas e suas Proposições de Valores; Riscos; Viabilidade Financeira; Restrições. O Product Owner é responsável pela criação e atualização deste artefato. Ele deve ser o primeiro artefato a ser criado e deve servir como base para a criação do Product Backlog e Action Backlog, sendo atualizado sempre que necessário.

12) A criação do artefato Business Case contribui para a criação de valor, uma vez que torna visível a todos os envolvidos quais valores as partes interessadas espera receber.

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente
- ☐ Sem Opinião

Caso deseje, comente sua resposta:

Ponto de Extensão 2 - Criação do artefato Action Backlog

O objetivo de um projeto de software deve ser entregar funcionalidades que gerem os benefícios esperados pelos interessados. No entanto, a realização de alguns benefícios pode depender de outras ações, de TI ou não. Em outras palavras, o software, por si só, pode não ser condição suficiente para a realização de alguns benefícios. Por exemplo, uma escola decide disponibilizar na WEB um software para que professores registrem suas aulas assim que elas terminem. São esperados vários benefícios, dentre eles a redução custos e a possibilidade realizar monitoramento em tempo real de alunos faltosos. No entanto, para que esses benefícios sejam realizados, os professores precisam ser capacitados para o uso de computadores e ter à disposição computadores com acesso à internet, caso contrário, o software não trará os benefícios esperados. Desta forma, percebe-se que o software, por si só, não entrega os benefícios esperados. A criação deste artefato tem como objetivo criar uma lista de ações adicionais necessárias para que todos os benefícios sejam realizados. Cabe ao Product Owner, juntamente com os interessados, planejar e garantir a execução das ações identificadas.

13) Identificar ações adicionais é essencial para garantir que os benefícios sejam realizados. Desta forma, a inclusão do artefato Action Backlog contribui para a criação de valor.

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente

☐ Sem Opinião

Caso deseje, comente sua resposta:

Ponto de Extensão 3 - Criação do artefato Release Backlog

No Scrum, ao término de cada Sprint, espera-se que um incremento do software esteja pronto para ser liberado. No entanto, algumas organizações, por questões técnicas ou de indisponibilidade de recursos, não conseguem liberar um incremento por Sprint. Além disso, a realização de alguns benefícios pode depender de ações adicionais identificadas no Action Backlog. Desta forma, sugere-se a criação de um artefato chamado Release Backlog. Este artefato deve definir as entregas de software de forma sincronizada com a execução das ações identificadas no Action Backlog. O Product Owner é o responsável por definir as entregas do software.

14. A criação do artefato Release Backlog é necessária para garantir as entregas de software gerem os benefícios esperados

- ☐ Concordo Totalmente
☐ Concordo Parcialmente
☐ Discordo Parcialmente
☐ Discordo Totalmente
☐ Sem Opinião

Caso deseje, comente sua resposta:

Ponto de Extensão 4 - Monitorar riscos

As curtas iterações e entregas frequentes ajudam o time do Scrum a evitar grande parte dos riscos do projeto. No entanto, a realização dos benefícios pode depender das ações do Action Backlog. Uma vez que a finalidade do software é a

entrega de benefícios, a não execução dessas ações representa riscos ao sucesso do projeto. Assim, faz-se necessário um gerenciamento desses riscos. Além disso, riscos não abordados implicitamente pelo Scrum poderiam tirar proveito desse gerenciamento. Esta extensão sugere que os riscos sejam identificados no Business Case e continuamente monitorados durante os eventos do Scrum.

15) Abordar riscos explicitamente é importante para garantir a criação de valor.

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente
- ☐ Sem Opinião

Caso deseje, comente sua resposta:

Ponto de Extensão 5 - Monitorar a realização dos benefícios

O objetivo do projeto software é criar valor para os interessados. Dessa forma, deve-se monitorar se os incrementos de software entregues estão de fato gerando os benefícios esperados. Assim, esta extensão sugere que durante a Sprint Review, na qual participam o time Scrum e os interessados do projeto, seja feito um monitoramento para saber se os benefícios estão sendo realizados, tendo como base o artefato Business Case. Além disso, as expectativas dos interessados podem sofrer mudanças. Desta forma, os artefatos criados devem ser revisados e, caso necessário, adaptados.

16) Monitorar se os incrementos de software entregues estão de fato realizando os benefícios esperados é importante para realizar adaptações no produto

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente
- ☐ Sem Opinião

Caso deseje, comente sua resposta:

17) A extensão proposta neste trabalho não descaracteriza a agilidade do Scrum

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente
- ☐ Sem Opinião

Caso deseje, comente sua resposta:

18) De forma geral, a extensão proposta neste trabalho contribui para que projetos que utilizam o Scrum criem valor para os interessados.

- ☐ Concordo Totalmente
- ☐ Concordo Parcialmente
- ☐ Discordo Parcialmente
- ☐ Discordo Totalmente
- ☐ Sem Opinião

Caso deseje, comente sua resposta:

APÊNDICE B – PERFIL DOS RESPONDENTES

Respondente	Titulação	Instituição/ Organizaçã o	Tempo de experiência em projetos de software	Tempo de experiência em projetos Scrum	Número de projetos Scrum	Certificações	Papéis desempenhados em projetos Scrum
1	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	1 - 3 anos	1 - 3 projetos	Certified Scrum Master	Product Owner
2	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	1 - 3 anos	Mais de 6 projetos	Certified Scrum Master	Product Owner Scrum Master
3	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	3 - 6 anos	1 - 3 projetos	Certified Scrum Master Certified Product Owner	Product Owner Scrum Master
4	Pós-Graduação (Mestrado)	UFPE	1 - 3 anos	1 - 3 anos	3 - 6 projetos	Certified Product Owner	Product Owner Scrum Master
5	Pós-Graduação (Doutorado)	UFPE	Mais de 6 anos	Mais de 6 anos	Mais de 6 projetos	Certified Scrum Master Certified Product Owner Certified Scrum Professional	Product Owner Scrum Master
6	Ensino Superior Completo	Caelum	Mais de 6 anos	Mais de 6 anos	Mais de 6 projetos	Certified Scrum Master Certified Scrum Professional	Product Owner Scrum Master Desenvolvedor
7	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	3 - 6 anos	3 - 6 projetos	Certified Scrum Master	Product Owner

						Certified Product Owner	Scrum Master
8	Pós-Graduação (Mestrado)	UFPB	3 - 6 anos	1 - 3 anos	3 - 6 projetos		Product Owner Scrum Master
9	Ensino Superior Completo	In Forma Software	Mais de 6 anos	1 - 3 anos	Mais de 6 projetos		Product Owner
10	Pós-Graduação (Mestrado)	UFAL	3 - 6 anos	1 - 3 anos	3 - 6 projetos		Scrum Master Desenvolvedor
11	Pós-Graduação (Especialização)	FGV	Mais de 6 anos	Menos de 1 ano	1 - 3 projetos		Scrum Master
12	Pós-Graduação (Especialização)	Infoway Tecnologia/PI	3 - 6 anos	1 - 3 anos	Mais de 6 projetos		Product Owner Scrum Master Desenvolvedor
13	Pós-Graduação (Doutorado)	UFPI	Mais de 6 anos	3 - 6 anos	3 - 6 projetos		Product Owner Scrum Master
14	Ensino Superior Completo	UNICAP	1 - 3 anos	1 - 3 anos	3 - 6 projetos		Product Owner
15	Pós-Graduação (Mestrado)	UNICAP	1 - 3 anos	1 - 3 anos	1 - 3 projetos		Scrum Master Desenvolvedor
16	Pós-Graduação (Especialização)	FGV	Mais de 6 anos	1 - 3 anos	1 - 3 projetos		Product Owner Scrum Master

17	Ensino Superior Incompleto	In Forma Software	Mais de 6 anos	1 - 3 anos	Mais de 6 projetos		Scrum Master
18	Pós-Graduação (Especialização)	Faculdade dos Guararapes	1 - 3 anos	1 - 3 anos	3 - 6 projetos		Scrum Master Desenvolvedor
19	Pós-Graduação (Mestrado)	UFPE	1 - 3 anos	Menos de 1 ano	Nenhum projeto		
20	Ensino Superior Incompleto	Faculdade Joaquim Nabuco	3 - 6 anos	1 - 3 anos	1 - 3 projetos		Desenvolvedor
21	Ensino Superior Incompleto	IFPI	1 - 3 anos	1 - 3 anos	1 - 3 projetos		Product Owner Desenvolvedor
22	Pós-Graduação (Mestrado)	IFPI	Mais de 6 anos	1 - 3 anos	1 - 3 projetos		Scrum Master
23	Ensino Superior Completo	UNIBRATEC	3 - 6 anos	1 - 3 anos	Mais de 6 projetos		Scrum Master Desenvolvedor
24	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	1 - 3 anos	1 - 3 projetos		Scrum Master Desenvolvedor
25	Pós-Graduação (Especialização)	UFPE	Mais de 6 anos	1 - 3 anos	3 - 6 projetos		Product Owner Scrum Master
26	Pós-Graduação (Especialização)	UFPE	Mais de 6 anos	1 - 3 anos	Mais de 6 projetos		
27	Pós-Graduação	UFPE	3 - 6 anos	3 - 6 anos	1 - 3 projetos		Scrum Master

	(Mestrado)						Desenvolvedor
28	Pós-Graduação (Mestrado)	TJ/PI	3 - 6 anos	1 - 3 anos	3 - 6 projetos		Scrum Master Desenvolvedor
29	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	Menos de 1 ano	1 - 3 projetos		Desenvolvedor
30	Ensino Superior Completo	UFRN	1 - 3 anos	Menos de 1 ano	1 - 3 projetos		Desenvolvedor
31	Pós-Graduação (Especialização)	IFRN	Mais de 6 anos	Menos de 1 ano	1 - 3 projetos		Desenvolvedor
32	Ensino Superior Completo	UFAL	Mais de 6 anos	3 - 6 anos	3 - 6 projetos		Scrum Master Desenvolvedor
33	Pós-Graduação (Mestrado)	UFRN	3 - 6 anos	1 - 3 anos	3 - 6 projetos		Scrum Master Desenvolvedor
34	Pós-Graduação (Mestrado)	UFPE	Mais de 6 anos	Mais de 6 anos	Mais de 6 projetos		Product Owner Scrum Master Desenvolvedor
35	Pós-Graduação (Mestrado)	UFPE	Menos de 1 ano	Menos de 1 ano	1 - 3 projetos		Desenvolvedor