

DANILO GARCIA LOPES

**PROPOSTA DE INCORPORAÇÃO DE ANÁLISE DE VALOR
AGREGADO AO PROCESSO SCRUM**

Monografia apresentada ao PECE –
Programa de Educação Continuada em
Engenharia da Escola Politécnica da
Universidade de São Paulo como parte
dos requisitos para conclusão do curso de
MBA em Tecnologia de Software.

São Paulo
2014

DANILO GARCIA LOPES

**PROPOSTA DE INCORPORAÇÃO DE ANÁLISE DE VALOR
AGREGADO AO PROCESSO SCRUM**

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para a conclusão do curso de MBA em Tecnologia de Software.

Área de Concentração: Tecnologia de Software

Orientador: Prof. Dr. Sergio Martins Fernandes

São Paulo
2014

DEDICATÓRIA

Aos meus queridos pais.

AGRADECIMENTOS

Sou grato a Deus, pela vida, saúde e oportunidades.

Aos meus pais Jacob e Lydia e a minha esposa Fernanda, pelo apoio incondicional.

Ao Prof. Dr. Sergio Martins Fernandes, por toda a orientação no desenvolvimento deste trabalho.

Aos meus colegas de turma, pela troca de experiências.

E, por fim, à Universidade de São Paulo – USP, à Escola Politécnica da Universidade de São Paulo – EPUSP e ao PECE – Programa de Educação Continuada em Engenharia que me possibilitaram a realização deste curso de Pós-Graduação.

RESUMO

O objetivo deste trabalho é propor a incorporação da análise de valor agregado ao processo SCRUM, de modo aplicável a um ambiente determinado, que poderia ser beneficiado pela utilização de métodos ágeis, porém utilizava o ciclo cascata como processo de desenvolvimento de *software* e, como ferramenta de monitoramento e controle do progresso físico e orçamentário de projetos, a técnica de gerenciamento de valor agregado. Dada a natureza sequencial e linear do ciclo cascata, alterações de escopo acabam gerando a necessidade de replanejamento e problemas são identificados tardiamente no decorrer do processo. As metodologias ágeis vêm em resposta a estes problemas, já que são baseadas em desenvolvimento e entrega incrementais. Porém, métodos ágeis não definem como monitorar os custos. Para suprir esta necessidade, alguns autores apresentam a técnica de gerenciamento de valor agregado. O trabalho utiliza como referencial teórico autores que descrevem métodos ágeis para o desenvolvimento de *software*, conceitos sobre análise de valor agregado e abordagens distintas de aplicação da técnica de valor agregado a métodos ágeis. A partir de uma análise comparativa das abordagens e dos requisitos identificados no contexto, foi proposta uma abordagem de incorporação da análise de valor agregado ao processo SCRUM específica para este ambiente e realizou-se a exemplificação de sua aplicação. Desta forma as conclusões do trabalho podem auxiliar organizações que conduzem projetos de desenvolvimento de *software* interessadas no emprego de métodos ágeis em conjunto com uma técnica de monitoramento e controle.

Palavras-chave: Métodos Ágeis; SCRUM; Análise de Valor Agregado; Monitoramento e Controle.

ABSTRACT

The objective of this study is to propose the incorporation of earned value analysis to SCRUM process, applicable to a particular environment, which would get benefits from agile methods, but used to use the waterfall cycle as software development process, and earned value management as a tool for monitoring and controlling physical progress and budget of projects. Given the sequential and linear nature of the waterfall cycle, scope changes have been generating the need for re-planning and problems are identified later in the process. Agile methodologies have been used in response to these problems, since they are based on incremental development and delivery. However, agile methods do not define how to monitor costs. To meet this need, some authors present the earned value management technique. This work uses as a referential theory authors that describe agile software development methods, concepts of earned value analysis and approaches for applying earned value to agile methods. From a comparative analysis of the approaches and from requirements identified in the context, an approach for incorporating earned value analysis to SCRUM process, specific to this environment, was proposed and an exemplification of the application was made. Thus, the conclusions of the study can help organizations that conduct software development projects and are interested in using agile methods in together with a monitoring and control technique.

Keywords: Agile; SCRUM; Earned Value Analysis; Monitoring and Control.

LISTA DE FIGURAS

Figura 1 – Processo SCRUM	22
Figura 2 – Equipe SCRUM	23
Figura 3 – Técnica de Gerenciamento do Valor Agregado.....	33
Figura 4 – Gráfico Burn-Down e Índices de Desempenho (CPI e SPI)	44
Figura 5 – Variação de Prazos (SV) em um Gráfico Burn-up.....	46
Figura 6 – Gráfico Burn-up com Escala de Custos	47
Figura 7 – Triângulos de Ferro nas Perspectivas de EVM e Ágil	49
Figura 8 – Gráfico Burn-down de Release	53
Figura 9 – Gráfico Burn-down de Sprint	54
Figura 10 – Gráfico de CPI e SPI	55
Figura 11 – Gráfico de EVM em percentuais.....	59
Figura 12 – Resumo da proposta de incorporação de EVM a SCRUM.....	85
Figura 13 – Exemplo de Gráfico Burn-down com CPI e SPI	89
Figura 14 – Exemplo de Gráfico Burn-up com Escala Monetária.....	90

LISTA DE TABELAS

Tabela 1 – Exemplo das métricas, variâncias e índices de desempenho	35
Tabela 2 – Interpretações de variações e índices de desempenho	35
Tabela 3 – EVM e questões comuns de gerenciamento de projetos	39
Tabela 4 – Backlog da Sprint Estimado em Pontos de História	50
Tabela 5 – Valores de Pontos de História por Sprint.....	50
Tabela 6 – Backlog da Sprint com Valor Agregado	51
Tabela 7 – Valor Agregado, CPI e SPI por Sprint	51
Tabela 8 – Comparativo entre as abordagens	65
Tabela 9 – Comparativo entre as abordagens do ponto de vista de EVM.....	66
Tabela 10 – Exemplo de Backlog do Produto	86
Tabela 11 – Exemplo de cálculo de custo médio por hora da equipe	87
Tabela 12 – Exemplo de Backlog de Sprint com PV	87
Tabela 13 – Exemplo de Backlog de Sprint com EV e AC	88
Tabela 14 – Exemplo de CPI e SPI por Sprint	88

LISTA DE ABREVIATURAS E SIGLAS

AC	Custo Real (<i>Actual Cost</i>)
ACWP	Custo Real do Trabalho Realizado (<i>Actual Cost of Work Performed</i>)
AgileEVM	Gerenciamento de Valor Agregado Ágil (<i>Agile Earned Value Management</i>)
ASD	Desenvolvimento de Software Adaptativo
BAC	Orçamento no Término (<i>Budget at Completion</i>)
BCWP	Custo Orçado do Trabalho Realizado (<i>Budget Cost of Work Performed</i>)
BCWS	Custo Orçado do Trabalho Agendado (<i>Budget Cost of Work Scheduled</i>)
CPI	Índice de Desempenho de Custos (<i>Cost Performance Index</i>)
CV	Variação de Custos (<i>Cost Variance</i>)
DOD	Departamento de Defesa dos Estados Unidos da América
DOE	Departamento de Energia dos Estados Unidos da América
DSDM	Método de Desenvolvimento de Sistemas Dinâmicos
EAC	Estimativa no Término (<i>Estimated at Completion</i>)
EACt	Estimativa de Prazo no Término (<i>Time Estimate at Completion</i>)
ETC	Estimativa para Terminar (<i>Estimated to Complete</i>)
EV	Valor Agregado (<i>Earned Value</i>)
EVM	Gerenciamento de Valor Agregado (<i>Earned Value Management</i>)
FDD	Desenvolvimento Dirigido à Funcionalidades
PMI	<i>Project Management Institute</i>
PV	Valor Planejado
ROI	Retorno sobre o investimento (<i>Return on Investment</i>)
SPI	Índice de Desempenho de Prazos (<i>Schedule Performance Index</i>)

SV	Variação de Prazos (<i>Schedule Variance</i>)
TCPI	Índice de desempenho para término (<i>To-Complete Performance Index</i>)
VAC	Variação no Término (<i>Variation at Completion</i>)
XP	<i>Extreme Programming</i>

SUMÁRIO

1. INTRODUÇÃO.....	14
1.1. Motivação	14
1.2. Objetivo.....	15
1.3. Justificativa	15
1.4. Metodologia	17
1.5. Estrutura do Trabalho	18
2. REVISÃO BIBLIOGRÁFICA	19
2.1. Métodos Ágeis	19
2.1.1. Apresentação	19
2.1.2. SCRUM	21
2.1.2.1. Papéis	22
2.1.2.2. Eventos	24
2.1.2.3. Artefatos	26
2.1.2.4. Monitoramento	28
2.1.2.5. Benefícios e Dificuldades	29
2.2. Técnica de Gerenciamento de Valor Agregado	30
2.2.1. Apresentação	30
2.2.2. Índices de Desempenho	34
2.2.3. Previsões.....	35
2.2.4. Benefícios e Dificuldades	38
2.3. Análise de abordagens de EVM em projetos ágeis	40
2.3.1. Diferenças entre projetos ágeis e tradicionais	40
2.3.2. Abordagens de EVM em projetos ágeis	41
2.3.2.1. Abordagem 1: AgileEVM - Gerenciamento de Valor Agregado em Projetos SCRUM	41

2.3.2.2.	Abordagem 2: Valor Agregado e Reporte Ágil	45
2.3.2.3.	Abordagem 3: Valor Agregado de Forma Ágil.....	47
2.3.2.4.	Abordagem 4: Medição de Progresso de Projetos de Software baseados em SCRUM.....	52
2.3.2.5.	Abordagem 5: Ágil em um Ambiente de Contrato do Governo.	55
2.3.2.6.	Abordagem 6: Valor Agregado para Desenvolvimento Ágil	57
2.4.	Considerações do Capítulo.....	60
3.	PROPOSTA DE INCORPORAÇÃO DE EVM A SCRUM.....	61
3.1.	Introdução.....	61
3.2.	Ambiente.....	61
3.3.	Análise comparativa das abordagens.....	63
3.4.	Processo.....	67
3.4.1.	Papéis.....	67
3.4.1.1.	Dono do Produto	67
3.4.1.2.	Scrum Master	68
3.4.1.3.	Equipe de Desenvolvimento.....	70
3.4.2.	Artefatos	72
3.4.2.1.	<i>Backlog</i> do Produto	72
3.4.2.2.	<i>Backlog</i> da <i>Sprint</i>	73
3.4.2.3.	Incremento de Produto Potencialmente Entregável	75
3.4.3.	Eventos.....	76
3.4.3.1.	Sprint.....	76
3.4.3.2.	Planejamento da Sprint	78
3.4.3.3.	Execução da Sprint	80
3.4.3.4.	Reunião Diária SCRUM	81
3.4.3.5.	Revisão da Sprint.....	82
3.4.3.6.	Retrospectiva da Sprint.....	83

3.5.	Resumo da proposta	84
3.6.	Aplicação	85
3.7.	Considerações do Capítulo.....	90
4.	CONSIDERAÇÕES FINAIS.....	92
4.1.	Contribuições do Trabalho.....	92
4.2.	Trabalhos Futuros.....	93
	REFERÊNCIAS.....	95

1. INTRODUÇÃO

No ambiente de uma empresa multinacional de consultoria em tecnologia da informação, onde a maioria de seus projetos inclui desenvolvimento de *software* customizado, diversos requisitos são necessários para que a gestão de prazos, custos, reporte à gerência executiva e clientes sejam realizados de forma correta, precisa e frequente.

O fornecimento de informações sobre os custos e o andamento dos projetos semanalmente aos clientes é, muitas vezes, uma cláusula contratual. A utilização da técnica de gerenciamento de valor agregado torna-se uma premissa, já que todo o portfólio de projetos é controlado utilizando suas métricas e índices de desempenho.

Neste ambiente os projetos são tipicamente executados utilizando-se o ciclo cascata. Porém, muitas vezes clientes solicitam mudanças no escopo e/ou na priorização de requisitos que acabam gerando a necessidade de replanejamento e, em casos extremos, até a paralisação do projeto. Além disso, dada a natureza sequencial e linear do ciclo cascata, muitas vezes problemas são identificados tardiamente aumentando o custo para a correção.

As metodologias ágeis vêm em resposta a estes problemas, já que são baseadas em desenvolvimento e entrega incrementais, assumem que requisitos costumam mudar constantemente durante o processo de desenvolvimento e buscam entregar *software* funcionando rapidamente aos clientes. Porém, métodos ágeis não definem como gerenciar e controlar os custos. Como, então, usufruir das características e benefícios das metodologias ágeis, porém mantendo os requisitos de monitoramento e controle de custos do projeto?

1.1. Motivação

Alguns dos problemas derivados da utilização do ciclo cascata observados no ambiente citado são definidos por Pressman (2011) como sendo as principais desvantagens da utilização do ciclo cascata em projetos de *software*: (i) Projetos de *software* raramente seguem o fluxo sequencial que o ciclo cascata propõe; (ii) O cliente dificilmente consegue estabelecer explicitamente todas as necessidades; (iii) O cliente deve ser paciente, já que uma versão do sistema só estará disponível próximo ao final do projeto; (iv) Erros não detectados nas fases iniciais, podem causar impactos desastrosos.

Passar a utilizar metodologias ágeis em projetos neste ambiente poderia solucionar, ou ao menos minimizar, as dificuldades observadas. Por outro lado, métodos ágeis não definem como gerenciar e controlar os custos, nem fornecem métricas de custo para suportar a tomada de decisão de negócios. Sendo assim, é possível utilizar-se dos benefícios da técnica de gerenciamento de valor agregado para suprir essas necessidades e, ainda, atender à demanda da organização quanto à utilização das métricas de valor agregado para o controle e monitoramento de custos e prazos dos projetos para reporte à gerência executiva e clientes.

1.2. Objetivo

Dados os problemas relativos à utilização do ciclo cascata e os requisitos de reporte do progresso físico e orçamentário abordados, o objetivo deste trabalho é propor a incorporação da análise de valor agregado ao processo SCRUM, de modo aplicável ao ambiente analisado.

1.3. Justificativa

O ambiente analisado apresenta a necessidade da resolução de problemas derivados do uso do ciclo cascata em seu contexto de desenvolvimento de sistemas. Para tanto, sugere-se a utilização de SCRUM como método para o processo de desenvolvimento de *software*. Porém, como SCRUM não define

como monitorar os custos do projeto e o uso da técnica de valor agregado é estabelecido pela gerência executiva da organização analisada, foram estudadas diferentes abordagens de aplicação de métodos ágeis e análise de valor agregado em conjunto.

Segundo Christoph (2011) as métricas da análise de valor agregado podem fornecer medidas de produtividade de custo que não existem no contexto ágil. Medidas de produtividade de custo permitem que o gerente do projeto apresente o quão eficientemente estão sendo geridos os recursos das partes interessadas, mesmo quando problemas com a disponibilidade de recursos estão impactando o desenvolvimento de funcionalidades do projeto.

Da mesma forma, Sulaiman; Barton e Blackburn (2006) destacam em seu artigo que SCRUM não define uma forma para gerenciar os custos e apresentam a análise de valor agregado como uma técnica capaz de suprir esta necessidade.

Além de possibilitar a inclusão da análise dos custos em projetos ágeis, a técnica de gerenciamento de valor agregado é, muitas vezes, requerida em uma cláusula contratual estabelecida com o cliente antes mesmo do início do projeto. Os autores Alleman; Henderson e Seggelke (2003) afirmam em sua pesquisa que no ambiente de desenvolvimento de *software* de uma empresa subcontratada pelo governo americano são necessários artefatos formais por diversas razões, muitas vezes por exigência contratual, e que EVM é uma das formas de cumprir as exigências de reporte de progresso contra o plano nesse ambiente.

Similarmente, Christoph (2011) destaca duas situações comuns em que um gerente de projeto poderia gerar medidas de valor agregado para um projeto ágil: quando EVM é um requisito estabelecido em contrato ou, ainda, quando a composição da equipe e, em menor escala, as durações de *sprint* não podem ser mantidas constantes.

A maioria dos praticantes de EVM e de métodos ágeis é familiarizada com apenas uma ou outra dessas abordagens, já que essas abordagens não são, muitas vezes, vistas como complementos naturais, segundo Christoph (2011). Porém, para Rusk (2009), métodos ágeis e a técnica de gerenciamento de valor agregado possuem um encaixe natural um para o outro e implementações EVM podem ser simplificadas para aplicação em projetos ágeis.

1.4. Metodologia

Para o desenvolvimento deste trabalho foi realizada uma revisão bibliográfica sobre métodos ágeis para o desenvolvimento de *software*, sobre os conceitos da técnica de gerenciamento de valor agregado, e também a respeito de abordagens de aplicação de métodos ágeis e análise de valor agregado em conjunto.

Em seguida, foi elaborada uma análise comparativa dos artigos que estudam a aplicação de EVM em processos ágeis, em função de critérios semelhantes apresentados nas diferentes abordagens.

Posteriormente, foi identificado um ambiente que originalmente utiliza o ciclo cascata como processo de desenvolvimento de *software* e a técnica de gerenciamento de valor agregado como ferramenta de monitoramento e controle do progresso físico e orçamentário de projetos, que seria beneficiado pela utilização de métodos ágeis. Este ambiente foi analisado e dele foram extraídos requisitos que devem ser atendidos com a integração de EVM ao processo SCRUM.

Baseando-se no levantamento realizado na revisão bibliográfica e na análise comparativa e do ambiente, foi proposta uma abordagem de incorporação da análise de valor agregado ao processo SCRUM específica para este ambiente e realizou-se a exemplificação de sua aplicação.

1.5. Estrutura do Trabalho

O Capítulo 1 - INTRODUÇÃO apresenta as motivações, o objetivo, as justificativas, a metodologia e a estrutura do trabalho.

O Capítulo 2 - REVISÃO BIBLIOGRÁFICA apresenta a revisão bibliográfica sobre métodos ágeis, os conceitos da técnica de gerenciamento de valor agregado e seis diferentes abordagens de aplicação de EVM a métodos ágeis.

O Capítulo 3 - PROPOSTA DE INCORPORAÇÃO DE EVM A SCRUM apresenta uma análise comparativa entre as abordagens de aplicação de análise de valor agregado a métodos ágeis, uma análise do ambiente organizacional sob o ponto de vista do processo de desenvolvimento de *software* e uma proposta de incorporação da técnica de gerenciamento de valor agregado a SCRUM para o ambiente analisado.

O Capítulo 4 - CONSIDERAÇÕES FINAIS descreve as conclusões obtidas no desenvolvimento deste trabalho, as contribuições do autor e sugestões de estudos posteriores sobre o mesmo assunto.

REFERÊNCIAS relaciona todas as referências utilizadas na revisão bibliográfica deste trabalho.

2. REVISÃO BIBLIOGRÁFICA

2.1. Métodos Ágeis

2.1.1. Apresentação

Os métodos ágeis de desenvolvimento de *software* tiveram início nos anos 80 como um movimento contrário aos métodos burocráticos de desenvolvimento de *software*. A motivação para a criação desses métodos foi a necessidade de superar problemas dos métodos tradicionais, como o modelo cascata, considerado pelos precursores dos métodos ágeis como sendo lento e contraditório à eficiência do trabalho de engenheiros de *software* (SALGADO et al., 2010).

Métodos ágeis apresentam uma abordagem incremental para especificação, desenvolvimento e entrega de *software*. São mais adequados para o desenvolvimento de aplicações onde os requisitos costumam mudar rapidamente durante o processo de desenvolvimento. Destinam-se a entregar *software* funcionando rapidamente aos clientes, que podem, em seguida, propor alterações e requisitos novos a serem incluídos nas iterações posteriores do sistema. Têm como objetivo reduzir a burocracia do processo, evitando o trabalho que tem valor duvidoso a longo prazo e eliminando documentação que provavelmente nunca será utilizada (SOMMERVILLE, 2011).

Apesar de algumas metodologias ágeis já existirem desde os anos 80 foi em fevereiro de 2001 que um grupo de desenvolvedores de *software* reuniu-se nos Estados Unidos para dividir experiências e discutir alternativas ao desenvolvimento de *software* tradicional, culminando na assinatura do chamado “Manifesto para o Desenvolvimento Ágil de *Software*”. Este grupo passou a ser conhecido como “Aliança Ágil” e possui os seguintes valores (BECK et al., 2001):

Estamos descobrindo maneiras melhores de desenvolver *software*, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através deste trabalho, passamos a valorizar:

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

Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda.

O manifesto ágil não rejeita processos, ferramentas, documentação, negociação de contratos ou planejamento, mas simplesmente mostra que eles têm importância secundária quando comparados a indivíduos e interações, ao software em funcionamento, à colaboração do cliente e às respostas rápidas a mudanças e alterações (SOARES, 2004).

Complementado os valores estabelecidos no manifesto, foram definidos doze princípios que norteiam o que é ser ágil (BECK et al., 2001):

1. Nossa maior prioridade é satisfazer o cliente através da entrega contínua e adiantada de *software* com valor agregado.
2. Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente.
3. Entregar frequentemente *software* funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.
4. Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
5. Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e o suporte necessário e confie neles para fazer o trabalho.
6. O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face.
7. *Software* funcionando é a medida primária de progresso.
8. Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
9. Contínua atenção à excelência técnica e bom design aumenta a agilidade.
10. Simplicidade – a arte de maximizar a quantidade de trabalho não realizado – é essencial.
11. As melhores arquiteturas, requisitos e designs emergem de equipes auto organizáveis.
12. Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo.

Os métodos ágeis mais utilizados e difundidos atualmente são *Extreme Programming* (XP) e SCRUM. Porém, muitos outros têm sido propostos, como por exemplo: Crystal, Desenvolvimento de Software Adaptativo (*Adaptive Software Development* - ASD), Método de Desenvolvimento de Sistemas Dinâmicos (*Dynamic Systems Development Method* - DSDM), Desenvolvimento Dirigido à Funcionalidades (*Feature Driven Development* - FDD), Desenvolvimento de Software Enxuto (*Lean Software Development* - LSD). Embora os métodos ágeis sejam todos baseados em desenvolvimento e entrega incrementais, propõem diferentes formas para atingir estes objetivos. Nem todo processo ágil aplica os doze princípios do manifesto ágil atribuindo-lhes o mesmo peso, e alguns modelos relevam a importância de um ou mais princípios. Entretanto, estes princípios definem a essência da agilidade mantida nos processos ágeis (PRESSMAN, 2011).

Neste capítulo é apresentado SCRUM, um dos métodos ágeis mais difundidos atualmente.

2.1.2. SCRUM

SCRUM é um processo ágil desenvolvido no início dos anos 90 por Ken Schwaber e Jeff Sutherland. Seus criadores definem SCRUM com sendo “um *framework* no qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível” (SCHWABER; SUTHERLAND, 2013, p. 3).

SCRUM é um termo emprestado do esporte *rugby*. Refere-se a uma forma de reiniciar o jogo após uma infração acidental ou quando a bola saiu do jogo. Apesar de SCRUM ser mais comumente utilizado para desenvolver produtos de *software*, os valores fundamentais e os princípios do SCRUM podem e estão sendo utilizados para desenvolver diferentes tipos de produtos ou para organizar fluxos de vários tipos de trabalho. Enquanto o foco de XP está nas práticas de programação, o SCRUM está focado nas práticas de gerenciamento de projetos (RUBIN, 2012).

O processo, suas práticas, artefatos e regras são simples. Por outro lado, SCRUM não é um processo prescritivo, ou seja, não descreve o que fazer em cada circunstância. SCRUM é utilizado para trabalhos complexos em que, muitas vezes, é impossível prever tudo o que vai ocorrer. SCRUM provê uma estrutura e um conjunto de práticas que permite manter tudo visível. Isso permite que os praticantes de SCRUM saibam o que está acontecendo e quais ajustes fazer para manter o projeto no rumo desejado (SCHWABER, 2004).

A figura 1 a seguir apresenta os papéis, eventos e artefatos de SCRUM que serão detalhados neste capítulo:

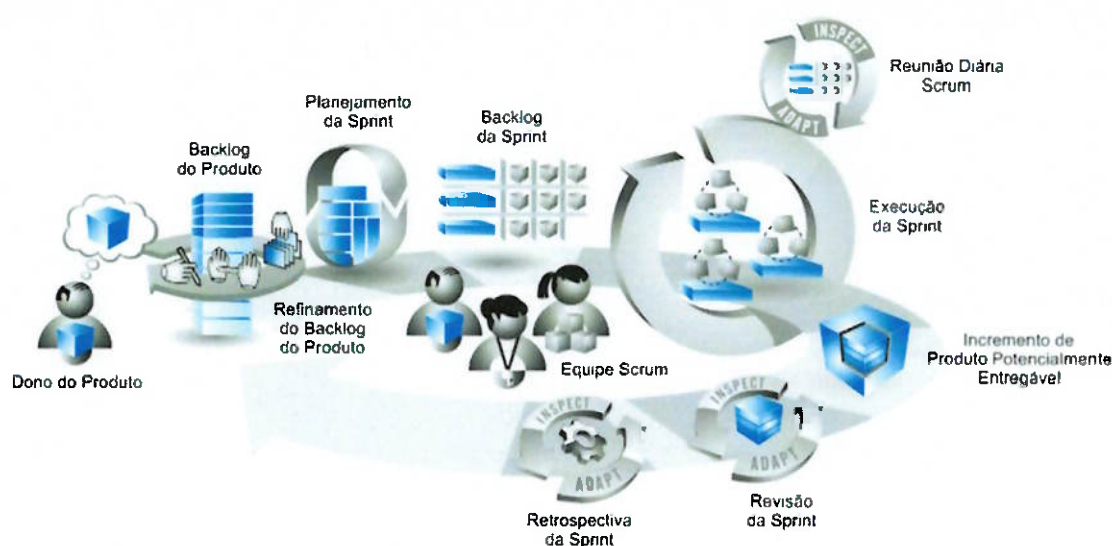


Figura 1 – Processo SCRUM
Fonte: Rubin (2012)

2.1.2.1. Papéis

A equipe SCRUM é composta por três papéis: o dono do produto, a equipe de desenvolvimento, e o *Scrum Master*. As equipes SCRUM são auto organizadas (escolhem a melhor forma de realizar o seu trabalho, em vez de serem dirigidas por pessoas de fora da equipe) e multifuncionais (possuem as competências necessárias para realizar o trabalho sem depender de agentes externos). São equipes projetadas para otimizar a flexibilidade,

criatividade e produtividade (SCHWABER; SUTHERLAND, 2013). A seguir são exemplificados na figura 2 os papéis que formam a equipe SCRUM:

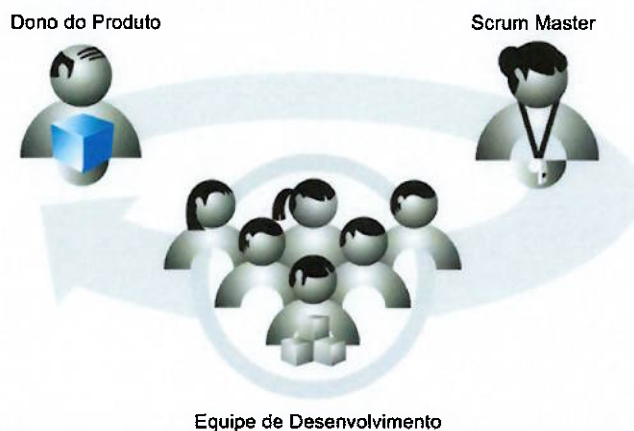


Figura 2 – Equipe SCRUM
Fonte: Rubin (2012)

- **Dono do Produto:** (do inglês *Product Owner*) é a autoridade única responsável por decidir quais recursos e funcionalidades devem ser construídas e em qual ordem isto deve acontecer. O dono do produto mantém e comunica a todos os outros participantes uma visão clara do que a equipe SCRUM está tentando alcançar. Deve entender as necessidades e prioridades das partes interessadas, clientes e usuários bem o suficiente para agir como sua voz e garantir que a solução correta está sendo desenvolvida. Também deve garantir que os critérios para aceitação de funcionalidades sejam especificados e os testes que verificam tais critérios sejam executados para determinar se os recursos estão completos. É responsável pelo sucesso global da solução que está sendo desenvolvida ou mantida (RUBIN, 2012).
- **Scrum Master:** é um facilitador que organiza reuniões diárias, acompanha o *backlog* de trabalho a ser realizado, registra decisões, avalia o progresso contra o atraso e se comunica com clientes e gestão de fora da equipe (SOMMERVILLE, 2011). Além disto, Smith e Sidky (2009) descrevem outras funções do *Scrum Master*: Auxiliar a equipe a desenvolver práticas que sustentam os princípios ágeis, atuar

como um guia no treinamento sobre como ser ágil e usar SCRUM, proteger a equipe de burocracia corporativa e de atividades que não agregam valor para o desenvolvimento de *software* (SMITH; SIDKY, 2009).

- **Equipe de Desenvolvimento:** A equipe de desenvolvimento é o grupo de profissionais que constroem os incrementos do produto em cada iteração, ou *sprint* (GRIFFITHS, 2012). Schwaber (2004, p. 7) define algumas características da equipe de desenvolvimento: “A equipe é auto gerenciada, auto organizada e multifuncional.” Os membros da equipe são responsáveis coletivamente pelo sucesso de cada iteração e do projeto como um todo (SCHWABER, 2004).

2.1.2.2. Eventos

SCRUM utiliza eventos para estruturar os vários fluxos de trabalho de desenvolvimento de *software* incremental. Cada evento possui um período fixo de tempo para sejam executadas as atividades dentro dele. Este espaço de tempo pré-definido garante que uma quantidade adequada de tempo seja dispendida no planejamento, sem que haja desperdício no processo de planejamento. Os eventos SCRUM são destinados a estabelecer regularidade e uma cadência. Destinam-se também a minimizar a necessidade de reuniões de improviso, que causam desperdício de tempo e que não são parte integrante de SCRUM. Todos os eventos são uma oportunidade formal para inspecionar e adaptar algo. A inspeção permite que a equipe identifique qualquer variação no plano atual. Se forem identificados quaisquer desvios inaceitáveis, devem ser feitos ajustes no produto ou processo. Esses ajustes devem ser feitos o mais rápido possível para minimizar ainda mais o desvio (HUNDHAUSEN, 2012).

A seguir são detalhados os eventos de SCRUM: *Sprint*, Reunião Diária SCRUM, Planejamento, Execução, Revisão e Retrospectiva da *Sprint*.

- ***Sprint***: em SCRUM o trabalho é realizado em iterações, ou ciclos, denominadas *sprints*. As *sprints* têm duração fixa, com datas de início e fim determinadas, com duração entre uma a quatro semanas tipicamente. Dentro deste espaço de tempo definido, a equipe deverá trabalhar em um ritmo constante para completar um conjunto escolhido de trabalho, alinhado ao objetivo da *sprint*. Como regra, não são permitidas alterações de objetivo, de escopo ou de pessoal durante a execução de uma *sprint*. Cada *sprint* inicia-se com o planejamento, inclui o trabalho de desenvolvimento (ou Execução da *Sprint*), e termina com a revisão e a retrospectiva (RUBIN, 2012).
- **Reunião Diária SCRUM**: (do inglês *Daily SCRUM*) diariamente a equipe SCRUM realiza reuniões curtas, de aproximadamente quinze minutos. Cada membro da equipe responde a três questões: O que você realizou desde a última reunião? Quais obstáculos está encontrando? O que planeja realizar até a próxima reunião? O *Scrum Master* conduz a reunião e avalia as respostas de cada membro da equipe. Estas reuniões auxiliam a equipe a identificar potenciais problemas o mais cedo possível e promovem uma estrutura de equipe auto organizada.
(PRESSMAN, 2011).
- **Planejamento da *Sprint***: Reunião realizada no início de cada *sprint* entre o dono do produto, a equipe de desenvolvimento e o *Scrum Master* com objetivo de determinar um subconjunto de itens do *backlog* do produto (uma lista ordenada de tudo o que pode ser necessário para o produto) a ser construído na próxima *sprint*. Durante o planejamento, o dono do produto e a equipe de desenvolvimento acordam um objetivo para a *sprint*. A partir do objetivo definido, a equipe de desenvolvimento determina quais dos itens poderão ser efetivamente concluídos na próxima *sprint*. Cada item é subdividido em tarefas e estimado tipicamente em horas (RUBIN, 2012).

- **Execução da *Sprint*:** Após o planejamento da *sprint*, a equipe de desenvolvimento, orientada pelo *Scrum Master*, realiza todas as tarefas necessárias para que as funcionalidades definidas para a *sprint* sejam concluídas. É responsabilidade da equipe de desenvolvimento gerenciar o fluxo de trabalho durante a execução da *sprint*. Ninguém diz à equipe em que ordem executar ou como concluir as tarefas do *backlog* da *sprint* (um subconjunto de itens do *backlog* do produto que foram selecionados para uma *sprint* específica). Em vez disso, os membros da equipe definem seu próprio trabalho e se auto organizam para alcançar o objetivo da *sprint* (RUBIN, 2012).
- **Revisão da *Sprint*:** Ao final de cada *sprint*, é realizada uma reunião de revisão. A equipe de desenvolvimento apresenta ao dono do produto e a outras partes interessadas que participarem da reunião tudo o que foi desenvolvido durante a *sprint* (SCHWABER, 2004). “Esta reunião informal em que a funcionalidade é apresentada se destina a unir as pessoas e de forma colaborativa determinar o que a equipe deve fazer em seguida” (SCHWABER, 2004, p. 9).
- **Retrospectiva da *Sprint*:** Ocorre após a revisão de *sprint* e antes da próxima reunião de planejamento da *sprint*. O objetivo da retrospectiva é promover uma reflexão sobre o processo e identificar oportunidades de melhoria que possam ser implementadas na próxima *sprint*. Durante este período o *Scrum Master* incentiva a equipe de desenvolvimento a inspecionar os processos, ferramentas, pessoas e relações visando torná-los mais efetivos (GRIFFITHS, 2012).

2.1.2.3. Artefatos

Artefatos do SCRUM representam trabalho ou valor para assegurar a transparência e oportunidades para inspeção e adaptação. Artefatos definidos pelo SCRUM são especificamente concebidos para maximizar a

transparência de informações-chave para que todos tenham o mesmo entendimento do artefato (SCHWABER; SUTHERLAND, 2013).

- **Backlog do Produto:** é definido por Griffiths (2012, p. 38) como sendo “uma lista ordenada de tudo o que pode ser necessário para o produto”. O Dono do Produto é responsável por seu conteúdo e priorização. Requisitos com maior prioridade são mais detalhados e têm estimativas mais precisas (GRIFFITHS, 2012). Schwaber (2004, p. 7) afirma que “o *Backlog* do Produto é dinâmico; a gestão o altera constantemente para identificar o que o produto necessita para ser apropriado, competitivo e útil”. O *Backlog* do Produto evolui à medida que o produto e o ambiente em que ele será utilizado evoluem (SCHWABER, 2004). Refinamento do *Backlog* do Produto (do inglês *Product Backlog Grooming*) é uma atividade de detalhamento, estimativa e priorização dos itens do *backlog* do produto. A atividade é liderada pelo dono do produto com colaboração do *Scrum Master*, da equipe de desenvolvimento e de outras partes interessadas. Se houver mudanças de prioridades, os itens são reordenados. Itens podem ser criados, subdivididos ou excluídos do *backlog* do produto à medida que mais informações são obtidas (RUBIN, 2012).
- **Backlog da Sprint:** é um conjunto de itens do *backlog* do produto que foram selecionados para uma *sprint* específica. Somente pode ser alterado pela equipe de desenvolvimento. Contém um plano de como alcançar o objetivo da *sprint* e serve como previsão para a equipe de desenvolvimento da funcionalidade que fará parte da *sprint* (GRIFFITHS, 2012).
- **Incremento de Produto Potencialmente Entregável:** (do inglês *Potentially Shippable Product Increment*) é o resultado de uma *sprint*. Significa que tudo o que a equipe SCRUM concordou em fazer foi realmente concluído de acordo com a definição consensual de “concluído”. Esta definição especifica o grau de confiança do trabalho

realizado, que é de boa qualidade e é potencialmente entregável (RUBIN, 2012).

2.1.2.4. Monitoramento

A maneira mais comum de efetuar o monitoramento de projetos ágeis (não somente SCRUM) é através de gráficos *burn* (ou gráficos de queima). Estes gráficos dão a visibilidade do progresso de um projeto, revelam a estratégia utilizada, apresentam o progresso realizado contra as previsões, e abrem a porta para discussões sobre a melhor forma de proceder, incluindo as difíceis discussões sobre a possibilidade redução do escopo ou aumento do prazo (COCKBURN, 2004).

Comumente, são utilizados três tipos de gráficos *burn*:

- **Gráfico *Burn-Down de Release*:** segundo Mahnic e Zabkar (2012) o gráfico *burn-down* de *Release* apresenta a quantidade de trabalho restante no início de cada *sprint*, plotando a soma de pontos da história de todas as histórias inacabadas no *backlog* do produto. Este gráfico torna visível a correlação entre a quantidade de trabalho restante e o progresso da equipe SCRUM em reduzir este trabalho.
- **Gráfico *Burn-Down de Sprint*** é semelhante ao gráfico *burn-down* de *release*, mas ao invés de apresentar o panorama de toda a *release*, representa a quantidade de trabalho restante que precisa ser concluído até o final da *sprint* (MAHNIC; ZABKAR, 2012).
- **Gráfico *Burn-Up*:** mostra o aumento da quantidade de funcionalidades completas em função do tempo. O benefício desse gráfico é a representação de status e taxa de requisitos entregues de forma simples e de fácil entendimento (CABRI; GRIFFITHS, 2006).

Segundo Cockburn (2004), gráficos *burn* possuem um mapeamento natural para os gráficos de valor agregado utilizados em projetos militares / governamentais norte-americanos.

2.1.2.5. Benefícios e Dificuldades

Um estudo realizado por Mah apud Cohn (2012) comparou 26 projetos ágeis com uma base de dados de 7500 projetos tradicionais. Os projetos ágeis tiveram produtividade 16% maior do que os projetos tradicionais. Este aumento foi classificado por Mah apud Cohn (2012) como sendo estatisticamente significativo. Rico apud Cohn (2012) aponta uma redução de custos média de 26% em sua pesquisa que avaliou 51 estudos publicados a respeito de projetos ágeis.

Além do aumento de produtividade e a redução de custos, Cohn (2012) aponta um outro benefício significativo para ser ágil. As equipes ágeis são menos propensas a construir funcionalidades que não são mais necessárias. Devido ao *feedback* frequente, *sprints* com tempo fixo e capacidade de priorizar cada *sprint*, uma equipe SCRUM é mais propensa a trabalhar apenas em recursos que os usuários realmente precisam, diferentemente um processo de desenvolvimento sequencial, que no momento em que o *software* é entregue, muitas vezes os usuários não precisam mais da funcionalidade fornecida (COHN, 2012).

Mann e Maurer apud Dybå e Dingsøyr (2008) estudaram o impacto de SCRUM em horas extras e na satisfação dos clientes e afirmam que os clientes acreditam que as reuniões diárias os mantêm atualizados e que as reuniões de planejamento são úteis para reduzir a “confusão” sobre o que deve ser desenvolvido. Adicionalmente, SCRUM levou a uma redução das horas extras, e os desenvolvedores perceberam que o processo SCRUM promoveu maior envolvimento dos clientes e melhoria na comunicação.

Segundo Cohn (2012) o sucesso é possível para todas as organizações. Porém as mudanças necessárias para colher todos os frutos que se tornar ágil pode trazer são profundas. Essas mudanças demandam muito não só

dos desenvolvedores, mas do resto da organização também. É necessário mudar não só as práticas, mas também a mentalidade. Como a transição para SCRUM envolve solicitar às pessoas que trabalhem de maneiras que não são familiares e em parte são contrárias à formação e experiência, as pessoas ficam muitas vezes hesitantes, se não totalmente resistentes, à mudança (COHN, 2012).

Outras dificuldades de SCRUM são apresentadas por Smith e Sidky (2009):

- SCRUM não deseja especialistas. Pode ser difícil converter rapidamente uma equipe já existente tornando um grupo de especialistas em um grupo onde qualquer pessoa pode realizar qualquer tarefa;
- Uma equipe SCRUM não pode ser bem sucedida sem um *Scrum Master* forte, o que torna o processo altamente dependente de um indivíduo;
- Porque SCRUM é principalmente um *framework*, a equipe ainda precisa identificar as práticas e métodos a serem utilizados.

Christoph (2011) destaca uma outra desvantagem: SCRUM sacrifica a eficiência global de conclusão, a fim de fornecer capacidade imediata. Porém, essa desvantagem pode passar despercebida, já que SCRUM não requer, na maioria das vezes, que a completeza seja definida. Todo retrabalho, refatoração, remodelagem, etc. é somente um novo item do *backlog* do produto a ser incluído em uma *sprint* futura.

2.2. Técnica de Gerenciamento de Valor Agregado

2.2.1. Apresentação

Gerenciamento de Valor Agregado (do inglês *Earned Value Management* - EVM) é uma técnica de monitoramento e controle de projetos capaz de integrar informações de escopo, prazo e custo, permitindo ainda a projeção

de custos e prazos para o fim do projeto. EVM foi desenvolvida por engenheiros industriais norte-americanos para melhor controlar custos de produção há mais de cem anos. No entanto, a técnica foi publicada pela primeira vez somente nos anos 60 pelo Departamento de Defesa dos Estados Unidos da América (do inglês *United States of America Department of Defense* - DOD). Valor agregado tem foco na relação entre os custos reais incorridos e no trabalho realizado no projeto dentro de um determinado período de tempo. O foco está no desempenho obtido em comparação ao que foi gasto para obtê-lo (FLEMING; KOPPELMAN, 2000).

Vargas (2013) define EVM como sendo a avaliação entre o que foi obtido em relação ao que foi realmente gasto e ao que se planejava gastar, onde se propõe que o valor a ser agregado inicialmente por uma atividade é o valor orçado para ela. Na medida em que cada atividade de um projeto é concluída, o valor inicialmente orçado para a atividade passa a constituir o valor agregado do projeto (VARGAS, 2013).

O Guia do Conhecimento em Gerenciamento de Projetos do *Project Management Institute* (PMI) afirma que a técnica de gerenciamento de valor agregado integra as medidas de escopo, custos e cronograma para auxiliar a equipe de gerenciamento a avaliar e medir o desempenho e progresso do projeto. EVM requer as informações de uma linha de base integrada contra a qual o desempenho pode ser medido na duração do projeto. Além disso, seus princípios podem ser aplicados a todos os projetos em qualquer setor (PMI, 2013).

A técnica de gerenciamento de valor agregado desenvolve e monitora três dimensões chave para avaliar o desempenho de um projeto: valor planejado, valor agregado e custo real:

- **Valor Planejado:** (do inglês *Planned Value* - PV) é o valor que indica a parcela do orçamento que deveria ser gasta, considerando o custo de linha da base da atividade, atribuição ou recurso (VARGAS, 2013). Também conhecido como Custo Orçado do Trabalho Agendado (do

inglês *Budget Cost of Work Scheduled* - BCWS), o valor planejado descreve o quão longe o trabalho do projeto deve estar em qualquer ponto do cronograma do projeto. É um reflexo numérico do trabalho orçado que está programado para ser realizado e é a linha de base estabelecida. O valor total planejado para o projeto é conhecido como Orçamento no Término (do inglês *Budget at Completion* - BAC) (PMI, 2005).

- **Valor Agregado:** (do inglês *Earned Value* - EV) é o valor que indica a parcela do orçamento que deveria ser gasta, considerando-se o trabalho realizado até o momento e o custo de linha de base para a atividade, atribuição ou recurso (VARGAS, 2013). Conhecido também como Custo Orçado do Trabalho Realizado (do inglês *Budget Cost of Work Performed* - BCWP), é um retrato instantâneo do progresso do trabalho em um determinado ponto no tempo. Reflete a quantidade de trabalho que foi realmente realizado até a data (ou até um determinado período de tempo), expresso como o valor previsto para esse trabalho (PMI, 2005).
- **Custo Real:** (do inglês *Actual Cost* - AC) apresenta os custos reais decorrentes do trabalho já realizado por um recurso ou atividade, até a data de referência, ou data atual do projeto (VARGAS, 2013). Também conhecido como Custo Real do Trabalho Realizado (do inglês *Actual Cost of Work Performed* - ACWP), é uma indicação do nível de recursos que têm sido despendidos para conseguir o trabalho real realizado até o momento, ou em um determinado período de tempo (PMI, 2005).

A partir destas três dimensões é possível obter as seguintes variações:

- **Variação de Prazos:** (do inglês *Schedule Variance* - SV) é a diferença, em termos de custo, entre o valor agregado (EV) e a

agenda de linha de base (PV). Se SV for positiva o projeto estará adiantado; se for negativa, o projeto estará atrasado (VARGAS, 2013).

$$SV = EV - PV$$

- **Varição de Custos:** (do inglês *Cost Variance* - CV) é a diferença entre o custo previsto para atingir o nível atual de conclusão (EV) e o custo real (AC), até a data de referência. Se CV for positiva, o custo do trabalho agregado está aquém do valor realmente gasto; se for negativa, a atividade está agregando um valor inferior ao que se gastou no trabalho (VARGAS, 2013).

$$CV = EV - AC$$

A figura 3 exemplifica as principais dimensões e variações utilizadas na técnica de gerenciamento de valor agregado:

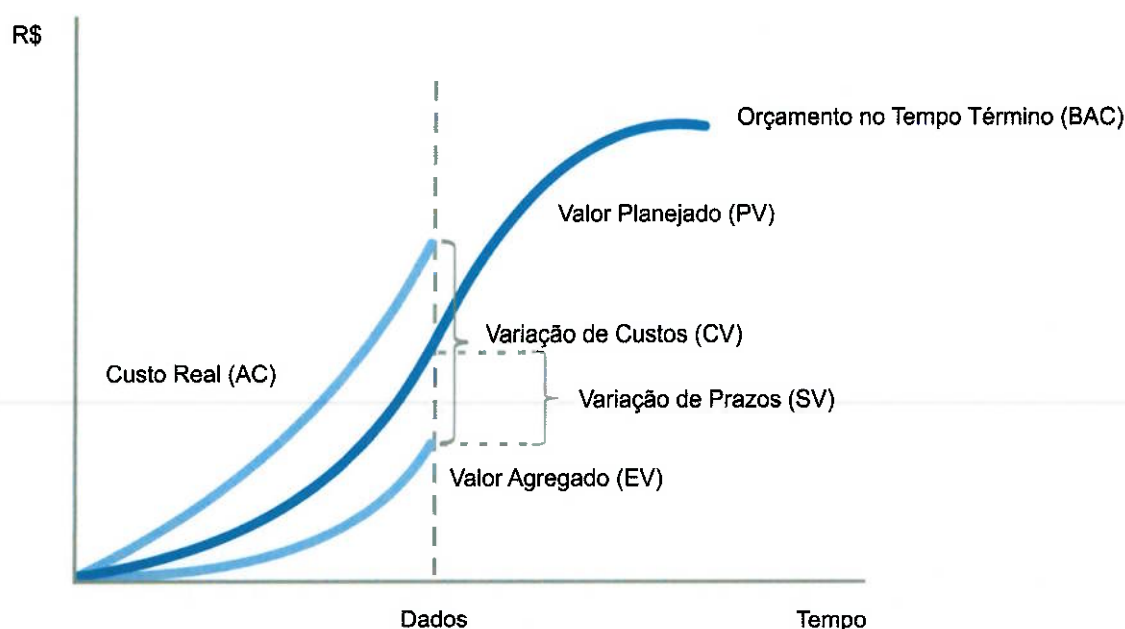


Figura 3 – Técnica de Gerenciamento do Valor Agregado
Fonte: Cabri e Griffiths (2006)

2.2.2. Índices de Desempenho

Os valores da variação de prazos e da variação de custos podem ser convertidos em indicadores de eficiência: Índice de desempenho de prazos e índice de desempenho de custos. Estes índices refletem o desempenho dos custos e dos prazos de qualquer projeto, permitindo uma comparação a outros projetos ou num portfólio de projetos.

- **Índice de Desempenho de Prazos:** (do inglês *Schedule Performance Index* - SPI) é a medida do progresso alcançado comparado ao progresso planejado num projeto. Um valor de SPI menor que 1 indica que menos trabalho foi executado do que o planejado. Já um valor de SPI maior que 1 indica que foi executado mais trabalho do que o planejado. O SPI é igual a razão entre o valor agregado e o valor planejado (PMI, 2013).

$$SPI = \frac{EV}{PV}$$

- **Índice de Desempenho de Custos:** (do inglês *Cost Performance Index* - CPI) é a medida do valor do trabalho executado comparado ao custo real ou progresso feito no projeto. Um valor de CPI menor que 1 indica um excesso de custo para o trabalho executado. Um valor de CPI maior que 1 indica um desempenho de custo abaixo do limite até a data presente. O CPI é igual a razão entre o valor agregado e o custo real (PMI, 2013).

$$CPI = \frac{EV}{AC}$$

A tabela 1 a seguir exemplifica as métricas e os cálculos de variâncias e índices de desempenho de EVM:

Elementos da WBS	Valores			Variação		Índice de Desempenho	
	Valor Planejado (PV)	Valor Agregado (EV)	Custo Real (AC)	Prazo EV - PV	Custo EV - AC	Prazo EV / PV	Custo EV / AC
1.0 Plano Pré-Piloto	63.000	58.000	62.500	(5.000)	(4.500)	0,92	0,93
2.0 Checklists	64.000	48.000	46.800	(16.000)	1.200	0,75	1,03
3.0 Currículo	23.000	20.000	23.500	(3.000)	(3.500)	0,87	0,85
4.0 Avaliação Mid-Term	68.000	68.000	72.500	-	(4.500)	1,00	0,94
5.0 Suporte à Implementação	12.000	10.000	10.000	(2.000)	-	0,83	1,00
6.0 Manual Prático	7.000	6.200	6.000	(800)	(200)	0,89	1,03
7.0 Plano de Roll-Out	20.000	13.500	18.100	(6.500)	(4.600)	0,68	0,75
Totais	257.000	223.700	239.400	(33.300)	(15.700)	0,87	0,93

Tabela 1 – Exemplo das métricas, variâncias e índices de desempenho
Fonte: PMI (2013)

Já a tabela 2 abaixo mostra o que as variações e índices de desempenho EVM indicam sobre um projeto em relação aos prazos e custos:

Variações e Índices de Desempenho		Prazos		
		SV > 0 & SPI > 1.0	SV = 0 & SPI = 1.0	SV < 0 & SPI < 1.0
Custos	CV > 0 & CPI > 1.0	Adiantado no prazo Abaixo do orçamento	Dentro do prazo Abaixo do orçamento	Atrasado no prazo Abaixo do orçamento
	CV = 0 & CPI = 1.0	Adiantado no prazo Dentro do orçamento	Dentro do prazo Dentro do orçamento	Atrasado no prazo Dentro do orçamento
	CV < 0 & CPI < 1.0	Adiantado no prazo Acima do orçamento	Dentro do prazo Acima do orçamento	Atrasado no prazo Acima do orçamento

Tabela 2 – Interpretações de variações e índices de desempenho
Fonte: PMI (2005)

2.2.3. Previsões

Segundo Vargas (2013), um dos principais objetivos da utilização da análise de valor agregado é, a partir do desempenho obtido pelo projeto até o momento, projetar os custos e os prazos finais para o projeto, dentro de diferentes cenários.

- **Estimativa para Terminar:** (do inglês *Estimated to Complete* - ETC) é o valor financeiro necessário para se completar o projeto (VARGAS, 2013). A Estimativa para Terminar é definida pelo PMI (2013) como

sendo “o custo previsto para terminar todo o trabalho restante do projeto”.

- **Estimativa de Prazo no Término:** (do inglês *Time Estimate at Completion* - EACt) utilizando o Índice de Desempenho de Prazos (SPI) e o Valor Planejado (PV) médio por unidade de tempo, a equipe do projeto pode gerar uma estimativa aproximada de quando o projeto será concluído, se as tendências atuais continuarem, em comparação a quando foi originalmente suposto ser concluído (PMI, 2005).

$$EAC_t = \frac{(BAC/SPI)}{(BAC/\{meses\})}$$

- **Estimativa no Término:** (do inglês *Estimated at Completion* - EAC) é o valor financeiro que representa o custo final do projeto quando concluído. Inclui os custos reais incorridos (AC) e os valores restantes estimados (ETC) (VARGAS, 2013). A estimativa no término pode ser calculada de diversas formas, porém serão apresentadas as três formas mais comuns segundo PMI (2013):

- **Previsão de EAC para o trabalho ETC executado no ritmo orçado:** este método de cálculo de EAC aceita o desempenho do projeto real até a data (se favorável ou desfavorável) como representado pelos custos reais e prevê que todo o trabalho ETC futuro será executado no ritmo orçado (PMI, 2013).

$$EAC = AC + (BAC - EV)$$

- **Previsão de EAC para o trabalho ETC executado ao CPI presente:** este método assume que o que têm acontecido até o momento no projeto tende a continuar no futuro. Assume-se que o trabalho ETC a ser executado terá o mesmo CPI incorrido pelo projeto até a data (PMI, 2013).

$$EAC = \frac{BAC}{CPI}$$

- **Previsão de EAC para o trabalho ETC considerando fatores SPI e CPI:** nesta previsão, o trabalho ETC será executado numa taxa de eficiência que considera os índices de desempenho de prazo e de custos. Este método é mais útil quando o cronograma do projeto é um fator de impacto no esforço de ETC (PMI, 2013).

$$EAC = AC + \left[\frac{(BAC - EV)}{(CPI \times SPI)} \right]$$

- **Varição no Término:** (do inglês *Variation at Completion* - VAC) é a diferença entre o custo orçado (BAC) e o custo projetado final (EAC). Mostra à equipe se o projeto vai terminar abaixo ou acima do orçamento (VARGAS, 2013).

$$VAC = BAC - EAC$$

- **Índice de desempenho para término:** (do inglês *To-Complete Performance Index* - TCPI) é a projeção calculada do desempenho de custos que deve ser atingido no trabalho restante para alcançar um objetivo de gerenciamento especificado, como o BAC ou a ETC (PMI, 2013). Uma outra forma de previsão é a utilização do TCPI como o índice de desempenho de custo futuro que deve ser produzido para recuperar o CPI realizado até o momento, de modo a projetar um índice de desempenho de custos final igual a 1 (VARGAS, 2013).

$$TCPI = \frac{(BAC - EV)}{(BAC - AC)}$$

2.2.4. Benefícios e Dificuldades

Os autores Fleming e Koppelman (2005) listaram uma série de benefícios obtidos com a aplicação de EVM, alguns deles citados a seguir:

- O emprego de um único sistema de controle gerencial que fornece dados precisos, consistentes, confiáveis e oportunos para a gestão em todos os níveis, permitindo o acompanhamento do desempenho de todos os projetos ou trabalhos de produção de uma empresa;
- Evidência empírica coletada e documentada em mais de 700 contratos do Departamento de Defesa dos Estados Unidos que empregaram gerenciamento de valor agregado, refletindo um padrão de histórico de desempenho consistente e previsível;
- A utilidade e a estabilidade do Índice de Desempenho de Custo (CPI) acumulado – que relaciona o trabalho realizado contra os custos gastos para efetuar esse trabalho – para monitorar continuamente as tendências de desempenho do projeto;
- A utilidade do Índice de Desempenho de Custos (CPI) para fazer previsões estatisticamente otimistas ou pessimistas para o fim do projeto, e para monitorar a eficiência de gastos;
- A utilidade do Índice de Desempenho de Prazos (SPI), que relaciona o trabalho físico realmente realizado e o trabalho inicialmente previsto, a fim de monitorar e gerenciar o cronograma para conclusão do projeto.

Além dos benefícios citados, PMI (2005) evidencia que algumas questões comuns de gerência de projetos podem ser respondidas através do uso da análise de valor agregado, como pode ser visto na tabela 3 a seguir:

Questões de Gerenciamento de Projetos	EVM - Medidas de Desempenho
Como estamos utilizando o tempo?	Análise de prazos e previsões
-Estamos adiantados ou atrasados no prazo?	-Variação de Prazos (SV)
-Quão eficientemente estamos utilizando o tempo?	-Índice de Desempenho de Prazos (SPI)
-Quando terminaremos o trabalho?	-Time Estimate at Completion (EACt)
Como estamos utilizando os recursos?	Análise de custos e previsões
-Estamos abaixo ou acima do orçamento?	-Variação de Custos (CV)
-Quão eficientemente estamos utilizando os recursos?	-Índice de Desempenho de Custos (CPI)
-Quão eficientemente devemos utilizar os recursos restantes?	-Índice de Desempenho para Término (TCPI)
-Qual é o provável custo do projeto?	-Estimativa no Término (EAC)
-Estaremos abaixo ou acima do orçamento?	-Variação no Término (VAC)
-Qual será o custo do trabalho restante?	-Estimativa para Terminar (ETC)

Tabela 3 – EVM e questões comuns de gerenciamento de projetos
Fonte: PMI (2005)

Por outro lado, Ray Stratton (2006) apud Lacerda (2009) cita alguns impedimentos para aplicação de EVM:

- EVM requer planejamento de tempo e recursos: EVM consiste na comparação de planejamento com realização. Se não há planejamento de recursos ao longo do tempo, não existe parâmetro de comparação à medida que houver realização;
- EVM requer uma mudança de cultura: as pessoas normalmente confundem tempo percorrido, horas trabalhadas e custo realizado com avanço físico real para os projetos. Como exemplo, o autor cita que ter trabalhado 50 horas de 100 planejadas não significa necessariamente ter completado 50% do trabalho prometido. EVM requer uma avaliação do que realmente foi alcançado, independente do consumo dos recursos;
- Executivos devem ter interesse em EVM: Assim como em qualquer iniciativa, a alta gestão de uma organização deve ter interesse em EVM para que ela se sustente. Os executivos devem entender EVM se pretendem que seus gerentes de projetos a utilizem. Na perspectiva dos executivos, EVM pode ser especialmente útil na priorização, cancelamento, adiantamento e postergação de projetos de um portfólio.

Outras limitações são destacadas por (GHOSH, 2011):

- EVM não mede a qualidade dos entregáveis, apenas os considera como completos;
- EVM não considera o caminho crítico do projeto. É possível estar à frente no prazo, porém atrasado nas atividades do caminho crítico;
- EVM utiliza o percentual de conclusão de um pacote de trabalho para mensurar o progresso do projeto. A porcentagem completa pode ser uma medida subjetiva. Por exemplo, não é incomum encontrar atividades que levaram muito mais tempo do que o que foi planejado para concluir os 10% restantes do pacote de trabalho.

2.3. Análise de abordagens de EVM em projetos ágeis

Antes de serem analisadas abordagens de aplicação de EVM em projetos ágeis de *software*, algumas diferenças entre projetos ágeis e tradicionais precisam ser observadas.

2.3.1. Diferenças entre projetos ágeis e tradicionais

Em um projeto ágil, a aplicação é dividida em recursos que são elaborados em várias iterações com base em suas prioridades. Análise, projeto, codificação, testes e implementação são feitos apenas para os recursos que estão na iteração atual. O cliente colabora no projeto ágil e solicita tanto mudanças de escopo quanto mudança de prioridades. Projetos tradicionais definem o escopo à frente e é custoso efetuar mudanças no escopo durante o projeto, dado que retrabalhos são necessários nos estágios de desenvolvimento anteriores. Projetos ágeis utilizam elaboração progressiva, enquanto projetos tradicionais definem o plano de base no início do projeto (GHOSH, 2011).

Projetos tradicionais assumem algumas premissas: o escopo é bem compreendido e pode ser totalmente definido no início de um projeto; o trabalho é finalizado em uma natureza sequencial e linear, onde as taxas de

progresso atuais são indicadores de taxas futuras. Já em projetos ágeis de *software* o trabalho é concluído de forma iterativa, e não sequencial e linear. Por exemplo, o *feedback* de cada iteração afeta a próxima iteração. O escopo é definido em alto nível no início do projeto e vai sendo detalhado a cada iteração. Mudanças de escopo são esperadas e frequentes ao longo do projeto, dado que os usuários do *software* entendem melhor o que querem à medida que veem algo tangível, e ágil procura ser responsivo a essas mudanças (CABRI; GRIFFITHS, 2006).

Muitos métodos ágeis, principalmente SCRUM, enfatizam planejamento incremental e desencorajam o planejamento total de projetos de *software* até o último nível de decomposição nas fases iniciais. A base para desencorajar estes grandes planos iniciais é a quantidade de ambiguidade presente em sistemas complexos de *software* o que torna a definição completa dos requisitos antecipadamente muito difícil (SULAIMAN; BARTON; BLACKBURN, 2006).

Sulaiman e Smits (2007) afirmam que métodos ágeis não definem como gerenciar e controlar os custos para avaliar o retorno esperado sobre o investimento. Os gráficos *burn-down* e *burn-up* não fornecem informações sobre o custo do projeto de forma direta e clara. Métricas ágeis não fornecem estimativas de custos na conclusão da *release*, nem métricas de custo para suportar a tomada de decisão de negócios.

2.3.2. Abordagens de EVM em projetos ágeis

São apresentadas neste capítulo, seis abordagens distintas, porém com características semelhantes, de aplicação da técnica de gerenciamento de valor agregado a projetos ágeis de *software*.

2.3.2.1. Abordagem 1: AgileEVM - Gerenciamento de Valor Agregado em Projetos SCRUM

Em seu artigo, Sulaiman; Barton e Blackburn (2006) afirmam que SCRUM concentra-se em priorizar requisitos frequentemente de modo a maximizar o retorno sobre o investimento (ROI). A fim de avaliar o ROI real, os custos do projeto devem ser avaliados para determinar se o planejamento das *releases* tem valor de negócio suficiente ou se algum replanejamento é necessário. O *framework* SCRUM, porém, não define uma forma para gerenciar os custos. Os autores, então, apresentam EVM como uma técnica capaz de suprir esta necessidade.

A técnica EVM foi adaptada a partir de práticas e valores de SCRUM. Esta adaptação foi nomeada AgileEVM (*Agile Earned Value Management* ou Gerenciamento de Valor Agregado Ágil) e definida como sendo “um conjunto simplificado de cálculos de valor agregado adaptado do EVM tradicional utilizando métricas SCRUM” (SULAIMAN; BARTON; BLACKBURN, 2006).

AgileEVM requer um conjunto mínimo de parâmetros de entrada: o custo real de um projeto, um *backlog* do produto estimado, o plano que fornece informações sobre o número de iterações na *release*, e a velocidade assumida. Todas as estimativas podem ser em horas, pontos de história, dias da equipe ou qualquer outra estimativa consistente de tamanho. O fator crítico é que seja uma estimativa numérica de algum tipo (SULAIMAN; SMITS, 2007).

Os autores afirmam que existe uma correlação entre as previsões de prazo realizadas utilizando SCRUM em comparação à Estimativa no Término (EAC) prevista utilizando EVM. Assim sendo, é derivada uma equação que possibilita estimar a data de entrega a partir da velocidade média das *sprints* (*velocity*) de SCRUM denominada *Release Date Velocity* (RD_V). A partir da equação RD_V , é derivada uma outra equação equivalente para EVM denominada RD_{EV} . Desta forma, os autores definem a seguinte hipótese: “Se a data de entrega projetada baseada na velocidade média (RD_V) for igual à data de entrega projetada baseada em EVM (RD_{EV}) então as técnicas de EVM podem ser utilizadas em projetos SCRUM para calcular o EAC” (SULAIMAN; BARTON; BLACKBURN, 2006).

A equação RD_V foi definida conforme abaixo:

$$RD_V = SD + L \cdot \left(n + \frac{W'}{v} \right)$$

onde SD é a data de início de uma *release*, L é o tamanho da *sprint* em dias, n é o número de *sprints*, W' é o trabalho necessário para completar uma *release*, v é média de trabalho completo por *sprint* (também conhecido como velocidade média ou *velocity*).

Já a equação RD_{EV} foi definida conforme abaixo:

$$RD_{EV} = SD + L \cdot \left(n \cdot \frac{EAC}{AC_n} \right)$$

onde SD é a data de início de uma *release*, L é o tamanho da *sprint* em dias, n é o número de *sprints*, EAC é a estimativa de custo total quando o projeto for concluído e AC_n é o custo real até o momento da medição.

Os autores testaram a aplicação de AgileEVM em dois projetos distintos, tendo como objetivos: validar a correspondência das datas de entrega projetadas comparando velocidade média e análise EVM, determinar se as métricas EVM podem fornecer valor adicional além das métricas SCRUM estabelecidas para a tomada de decisões de projeto e analisar se o impacto da análise de custo agrega valor às equipes técnicas para a tomada de decisões. Os dois projetos tinham adaptações diferentes ao framework SCRUM, um utilizava *sprints* curtas com uma pequena equipe (5 a 7 pessoas), enquanto o outro possuía uma grande equipe (15 a 18 pessoas) e *sprints* mensais. Ambos os projetos utilizavam pontos da história para fins de estimativa e foram monitorados utilizando ambas as técnicas AgileEVM e *burn-down* (SULAIMAN; BARTON; BLACKBURN, 2006).

A análise dos dados obtidos nos dois projetos monitorados por Sulaiman; Barton e Blackburn (2006) mostraram que as previsões das datas de entrega foram idênticas tanto para o cálculo através de velocidade média, quanto pela análise de valor agregado. Estes resultados se mostraram dentro do esperado pelos autores, dada a equivalência matemática das equações RD_V e RD_{EV} (SULAIMAN; BARTON; BLACKBURN, 2006).

Além da obtenção da data de entrega prevista, os autores integraram os índices CPI e SPI a um gráfico *burn-down*. Essa integração evidencia outro benefício de AgileEVM, apresentar claramente a eficiência de utilização do orçamento e dos prazos, que um gráfico *burn-down* não pode demonstrar (QUINTELLA; LACERDA, 2011).

Como exemplo, destacam que um dos projetos analisados teve uma reserva orçamento de 30% e no gráfico *burn-down* com o índice CPI integrado (figura 4) é possível observar claramente a erosão desta reserva. Esta erosão não seria reconhecida em um gráfico *burn-down* isolado (SULAIMAN; BARTON; BLACKBURN, 2006).

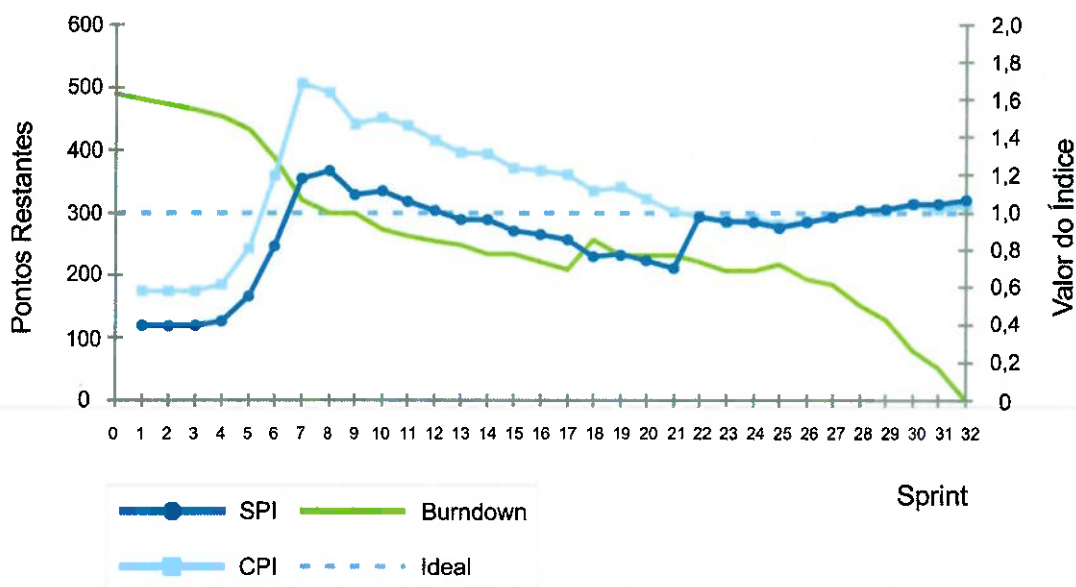


Figura 4 – Gráfico Burn-Down e Índices de Desempenho (CPI e SPI)
Fonte: Sulaiman; Barton e Blackburn (2006)

Segundo Sulaiman; Barton e Blackburn (2006) a gerência executiva se mostrou confortável com as métricas AgileEVM e afirmou que foram úteis para a tomada de decisões sobre a direção do produto, e que a análise

desses dados auxiliou a validação de decisões e da necessidade de mudanças no processo. As métricas CPI, SPI e EAC foram solicitadas não só pela gerência executiva, mas também pelos membros da equipe SCRUM, que se mostraram interessados na porcentagem concluída, esforço a ser concluído e data de entrega prevista. Não foram percebidos impactos na velocidade média da equipe SCRUM gerados pela implementação de AgileEVM.

2.3.2.2. Abordagem 2: Valor Agregado e Reporte Ágil

Os autores Cabri e Griffiths (2006) afirmam que projetos ágeis de *software* assumem algumas premissas que são fundamentalmente diferentes de projetos tradicionais, e isso afeta se, e como a técnica de valor agregado é aplicada. Projetos tradicionais determinam o escopo no início e mudanças de escopo são pouco frequentes. Já em projetos ágeis de *software*, o escopo inicial não é considerado completo. O escopo é detalhado quando iterações do projeto são concluídos, com base em *feedback* dos usuários e das partes interessadas ao final de cada iteração. Essas questões tornam a aplicação de EVM diretamente a um projeto ágil um desafio.

Por outro lado, os autores destacam que o conceito de EVM que relaciona progresso físico real aos custos reais é tão relevante para projetos ágeis quanto para qualquer projeto e que os gerentes de projeto estão sempre buscando responder questões tais como: "Quanto já foi feito?", "Quanto mais ainda deve ser feito?", "Quanto temos gasto?" e "Quanto será o custo final?". Segundo Cabri e Griffiths (2006) a chave para responder essas questões é utilizar os mecanismos de monitoramento de projetos ágeis e aprimorá-los.

Uma das principais técnicas de acompanhamento de projetos ágeis são os gráficos *burn-down* e *burn-up*. Um gráfico *burn-up* mostra o aumento da quantidade de funcionalidade completa em função do tempo. O benefício desse gráfico é a representação de status e taxa de recursos entregues de forma simples e de fácil entendimento. Segundo Cabri e Griffiths (2006) isto é

conceitualmente equivalente ao Valor Agregado (EV) acumulado em uma data específica. Um gráfico *burn-down* apresenta informações semelhantes, e indica claramente quantos requisitos ainda precisam ser concluídos (CABRI; GRIFFITHS, 2006).

Cabri e Griffiths (2006) afirmam que enquanto métricas de valor agregado não são diretamente aplicáveis a projetos ágeis, os conceitos são. O conceito de variação de prazos (SV) pode ser facilmente observado em gráficos *burn-up* (figura 5). Porém, em um gráfico *burn-up* a variação de prazos é uma medida em quantidade de requisitos (ou *features*), diferentemente de EVM tradicional, onde é medida em valores monetários.

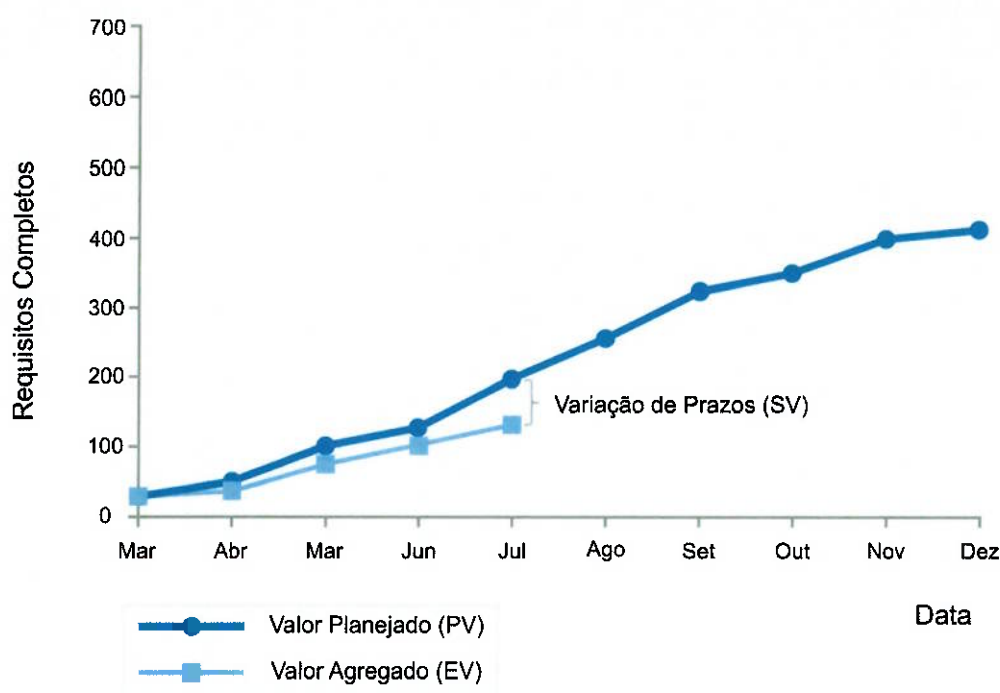


Figura 5 – Variação de Prazos (SV) em um Gráfico Burn-up
Fonte: Cabri e Griffiths (2006)

Gráficos *burn-down* e *burn-up* tipicamente não monitoram custos, porém, os autores Cabri e Griffiths (2006) propõem a inclusão das despesas em uma outra escala em um gráfico *burn-up* e também a incorporação do Custo Planejado, Custo Real, e Requisitos Completos, como pode ser observado no figura 6, onde é possível avaliar as três dimensões fundamentais de EVM, sendo o Valor Planejado (PV) obtido através da curva Custo Planejado,

Custo Real (AC) observado na curva de mesmo nome, e os gastos orçados para o trabalho real completo (ou Valor Agregado - EV) derivados da curva Requisitos Completos.

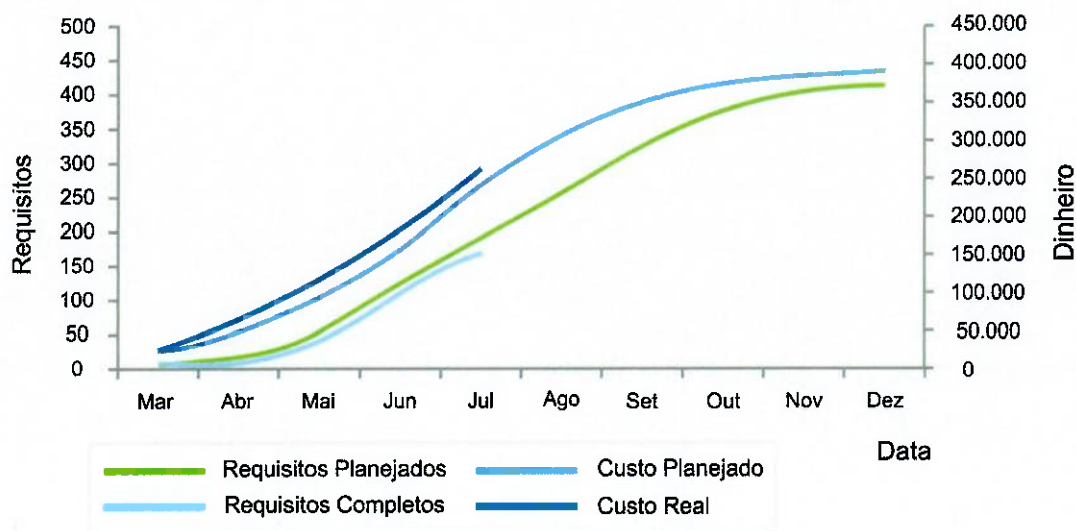


Figura 6 – Gráfico Burn-up com Escala de Custos
Fonte: Cabri e Griffiths (2006)

Além disso, os autores destacam que é possível extrair um valor equivalente de SPI dividindo a quantidade de Requisitos Completos por Requisitos Planejados e também obter um valor similar ao CPI dividindo o Custo Planejado pelo Custo Real. No entanto, na análise de Quintella e Lacerda (2011), este índice indica apenas a relação entre o dinheiro planejado e o dinheiro gasto, enquanto a abordagem tradicional de EVM para CPI é a divisão entre Valor Agregado (EV) e Custo Real (AC), que revela a eficiência do uso do dinheiro.

Os autores concluem que, embora esses indicadores sejam úteis para o status instantâneo, a utilização dos mesmos para previsão dos custos e prazos finais pode não levar a resultados corretos, dado o aspecto de progressão não-linear de projetos ágeis de *software*. No entanto, é possível identificar no início problemas para os quais as ações corretivas precisam ser tomadas, o principal objetivo de EVM (CABRI; GRIFFITHS, 2006).

2.3.2.3. Abordagem 3: Valor Agregado de Forma Ágil

Uma outra abordagem proposta por Christoph (2011) possibilita a geração das métricas de EVM para projetos ágeis, sem sacrificar as vantagens dos métodos ágeis. O autor apresenta características da técnica de valor agregado e das metodologias ágeis e destaca alguns contrastes entre elas.

Christoph (2011) afirma que em EVM todo o valor é determinado pelo plano, o que torna a definição e controle do escopo críticos para que os benefícios da abordagem sejam perceptíveis. Se for possível definir e gerenciar o escopo adequadamente, então é possível identificar variações de desempenho antecipadamente, tomar ações corretivas e prever com precisão a conclusão do projeto. O autor também destaca que quando usado corretamente, EVM pode suportar de forma eficiente e efetiva qualquer tipo de projeto.

Segundo Christoph (2011) os métodos ágeis com iterações de comprimento muito curtos ou variáveis geralmente são menos adequados para adaptar-se à técnica de valor agregado. As abordagens ágeis com iterações fixas, como SCRUM e *Feature-Driven Development*, podem ser adaptadas mais facilmente porque o tamanho da *sprint* pode ser ajustada de forma coordenada com ciclos de reporte de status, usualmente necessários em projetos.

Ao ser analisado o “triângulo de ferro” de gerenciamento de projetos (do inglês *iron triangle*) pelas perspectivas ágil e EVM, conforme a figura 7 a seguir, nota-se que funcionam de maneira oposta. Projetos ágeis procuram manter prazos e custos constantes, porém o escopo é variável. Já EVM auxilia o gerenciamento de prazos e custos mantendo o escopo constante (ou pelo menos controlado) (CHRISTOPH, 2011).

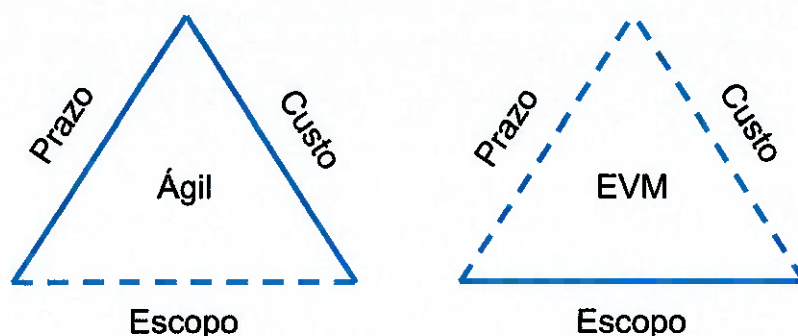


Figura 7 – Triângulos de Ferro nas Perspectivas de EVM e Ágil
Fonte: Christoph (2011)

Enquanto escopo é gerenciado, métricas EVM fornecem medidas de produtividade real em relação ao plano, o que permitem ao gerente de projeto tomar decisões informadas sobre os recursos e a sequência de atividades a serem executadas para manter um projeto no prazo e dentro do cronograma. Os índices de desempenho de custos e prazos são utilizados apenas previsão de resultados futuros, se o escopo do trabalho é bem compreendido (CHRISTOPH, 2011).

Christoph (2011) define velocidade (do inglês *velocity*) como sendo a medida de produtividade no universo ágil. Velocidade é a soma das estimativas das funcionalidades entregues e aceitas por iteração, e é medida nas mesmas unidades que as estimativas de funcionalidades, sejam em pontos da história, quantidade de requisitos, ou qualquer outra unidade apropriada.

Como métodos ágeis tipicamente assumem um período de tempo fixo para cada iteração e uma equipe constante para a realização do trabalho, a medida de velocidade (ou *velocity*) fornece um indicador valioso de quanto tempo pode levar a conclusão de um determinado *backlog* do produto. Por outro lado, se a duração das *sprints* e a composição da equipe mudarem constantemente durante o projeto, a medida de velocidade não fornecerá um valor preditivo corretamente (CHRISTOPH, 2011).

A geração de dados de EVM requer que em alguma fase do projeto ocorra o planejamento do trabalho e, em seguida, o trabalho completo seja mensurado contra esse plano. Para Christoph (2011), no contexto ágil isto deve ocorrer na *sprint*. No início de cada *sprint*, funcionalidades ou histórias de usuários do *backlog* do produto são avaliadas e priorizadas pela equipe ágil. Os recursos que oferecem o mais alto valor de negócio são movidos para o *backlog* da *sprint*, e estimados em pontos de história, conforme tabela 4 a seguir:

Data	Planejado na Sprint	Req nº	Requisito	Pontos de História	Completo	Valor Agregado
01-Jan-11	1	25	Simplificar o processo atual de e-mail	1	Não	0
16-Dez-10	1	16	Permitir escolha do formato do email	1,15	Não	0
30-Nov-10	1	9	Criar uma nova página inicial	1,05	Não	0
14-Nov-10	1	12	Permitir seleção da prioridade do e-mail	1	Não	0
29-Out-10	1	3	Popular dados relevantes automaticamente	1,1	Não	0
13-Out-10	1	5	Permitir impressão do documento	1,15	Não	0
27-Set-10	1	7	Uma cópia do e-mail deve ser armazenada	1,05	Não	0
27-Set-10	1	8	Um log de auditoria deve ser impresso	1,15	Não	0
27-Set-10	1	9	Permitir buscas por conteúdo	1,05	Não	0
27-Set-10	1	10	Mensagem de erro para leitura	1,05	Não	0
29-Out-10	1	11	Popular assinatura automaticamente	1,1	Não	0
27-Set-10	1	11	Habilitar numeração de página por e-mail	1,15	Não	0
Total				13		0

Tabela 4 – Backlog da Sprint Estimado em Pontos de História
Fonte: Christoph (2011)

O necessário para gerar métricas de EVM é que os custos totais planejados para a *sprint* e os custos totais para a *sprint* sejam divididos pelo total de pontos da história planejados, gerando assim o valor de um ponto história. A tabela 5 a seguir exemplifica este cálculo:

Sprints	Pontos de História	Horas	Horas / Pontos de História	Custo da Sprint	Valor por ponto de história
Sprint 1	13	580,00	44,62	\$ 35.090	\$ 2.699
Sprint 2	16	720,00	45,00	\$ 43.560	\$ 2.723
Sprint 3	12	580,00	48,33	\$ 35.090	\$ 2.924
		1.880,00		\$ 113.740	

Tabela 5 – Valores de Pontos de História por Sprint
Fonte: Christoph (2011)

À medida que cada requisito é concluído, ele "agrega" valor de acordo com o valor determinado de cada ponto história para a respectiva *sprint* (2.699,00 dólares por ponto de história concluída na *sprint* 1, conforme exemplo nas tabelas 5 e 6).

Data	Planejado na Sprint	Req nº	Requisito	Pontos de História	Completo	Valor Agregado
01-Jan-11	1	25	Simplificar o processo atual de e-mail	1	Sim	\$ 2.699
16-Dez-10	1	16	Permitir escolha do formato do e-mail	1,15	Sim	\$ 3.104
30-Nov-10	1	9	Criar uma nova página inicial	1,05	Sim	\$ 2.834
14-Nov-10	1	12	Permitir seleção da prioridade do e-mail	1	Não	-
29-Out-10	1	3	Popular dados relevantes automaticamente	1,1	Sim	\$ 2.969
13-Out-10	1	5	Permitir impressão do documento	1,15	Sim	\$ 3.104
27-Set-10	1	7	Uma cópia do e-mail deve ser armazenada	1,05	Não	-
27-Set-10	1	8	Um log de auditoria deve ser impresso	1,15	Sim	\$ 3.104
27-Set-10	1	9	Permitir buscas por conteúdo	1,05	Sim	\$ 2.834
27-Set-10	1	10	Mensagem de erro para leitura	1,05	Sim	\$ 2.834
29-Out-10	1	11	Popular assinatura automaticamente	1,1	Sim	\$ 2.969
27-Set-10	1	11	Habilitar numeração de página por e-mail	1,15	Não	-
5-Fev-11		1	Interface para servidor existente	1,15	Não	-
7-Fev-11		2	Login de usuário pelo O/s	1,15	Não	-
15-Fev-11		6	Substituir O/s subjacente	1,15	Sim	\$ 3.104
18-Fev-11		13	Habilitar fluxo de trabalho	1,05	Sim	\$ 2.834
Total				17,5		\$ 32.388

Tabela 6 – Backlog da Sprint com Valor Agregado
Fonte: Christoph (2011)

Desta forma, os dados necessários para obtenção de CPI e SPI ficam disponíveis. O exemplo a seguir (tabela 7) mostra os dois indicadores sendo acompanhados em várias iterações. Estas informações apresentam não só o que foi concluído, mas quão eficientemente foi concluído, e podem ser utilizadas para identificar o impacto das mudanças de requisitos na performance.

Sprints	Velocidade Planejada	PV	Valor por ponto de história	AC	Velocidade Atual	EV	CPI (EV/AC)	SPI (EV/PV)
Sprint 1	13	\$ 35.090	\$ 2.699	\$ 31.250	12	\$ 32.388	1,04	0,92
Sprint 2	16	\$ 43.560	\$ 2.723	\$ 45.887	17	\$ 46.291	1,01	1,06
Sprint 3	12	\$ 35.090	\$ 2.924	\$ 33.006	11	\$ 32.164	0,97	0,92
		\$ 113.740		\$ 110.143		\$ 110.843	1,01	0,97

Tabela 7 – Valor Agregado, CPI e SPI por Sprint
Fonte: Christoph (2011)

O autor destaca que neste exemplo o valor agregado é baseado em um valor de ponto da história que muda a cada nova iteração, mas não se altera durante a iteração. A fim de manter um valor de ponto de história constante para todo o projeto, todo o *backlog* do produto deveria ser mensurado em função do custo planejado total para o projeto (CHRISTOPH, 2011).

Christoph (2011) afirma que o índice de desempenho de custos (CPI) fornece uma medida de produtividade que flutua de acordo com os recursos disponíveis e que combinado com as medidas para controlar a quantidade de *churn* (requisitos modificados) no backlog da *sprint*, o projeto possuirá um conjunto de dados rico que pode mostrar o impacto de mudanças nos requisitos de uma forma clara a todas as partes interessadas.

2.3.2.4. Abordagem 4: Medição de Progresso de Projetos de Software baseados em SCRUM

Mahnic e Zabkar (2012) afirmam que a possível perda de controle gerencial é uma das grandes preocupações quando da adoção de métodos ágeis de desenvolvimento de *software*, e que o monitoramento contínuo do processo de desenvolvimento por meio de conjunto adequado de medidas é fundamental para assegurar a visibilidade, inspeção e adaptação.

Os autores propõem um conjunto de medidas de progresso a serem observadas em estudo de caso: velocidade (a quantidade de trabalho realizado em cada *sprint*, expressos em pontos da história), quantidade de trabalho restante (representada pelos gráficos de *burn-down* de *release* e de *sprint*), e índices de desempenho de prazos e custos. As duas primeiras medidas são já conhecidas e estabelecidas como medidas de progresso em projetos ágeis. Com o objetivo de monitorar os custos de desenvolvimento, os índices de valor agregado foram adicionados pelos autores, uma vez que estes não estão incluídos em outras medidas de projetos ágeis (MAHNIC; ZABKAR, 2012).

O projeto que serviu como um estudo de caso para Mahnic e Zabkar (2012) durou 7 meses e consistiu de 9 sprints de 3 semanas cada. O método de desenvolvimento escolhido foi SCRUM devido a requisitos vagos e mutáveis. O tamanho da equipe que trabalhou no projeto variou ligeiramente entre 6 e 8 pessoas. O tamanho de cada história de usuário foi estimado em pontos de história na reunião de planejamento da *sprint*.

A velocidade planejada foi estimada pela equipe SCRUM no início de cada *sprint*. A velocidade real foi calculada no final da *sprint* somando-se os pontos de história de todas as histórias aceitas pelo dono do produto. A análise dos resultados revelou que a velocidade real foi menor que velocidade planejada para a maioria das *sprints*. Isto se deu por dois erros comuns, cometidos no projeto analisado, mas que devem ser evitados em projetos SCRUM: a velocidade planejada deve ser estimada considerando-se a velocidade real de sprints anteriores e não deve haver mudanças na equipe de desenvolvimento durante o projeto (MAHNIC; ZABKAR, 2012).

No estudo de caso apresentado pelos autores, foi possível observar através do gráfico de *burn-down* de *release* (figura 8) que a equipe não foi capaz de reduzir a quantidade de trabalho restante com em tempo suficiente para concluir o projeto em sete *sprints*, como era esperado no início do projeto (MAHNIC; ZABKAR, 2012).

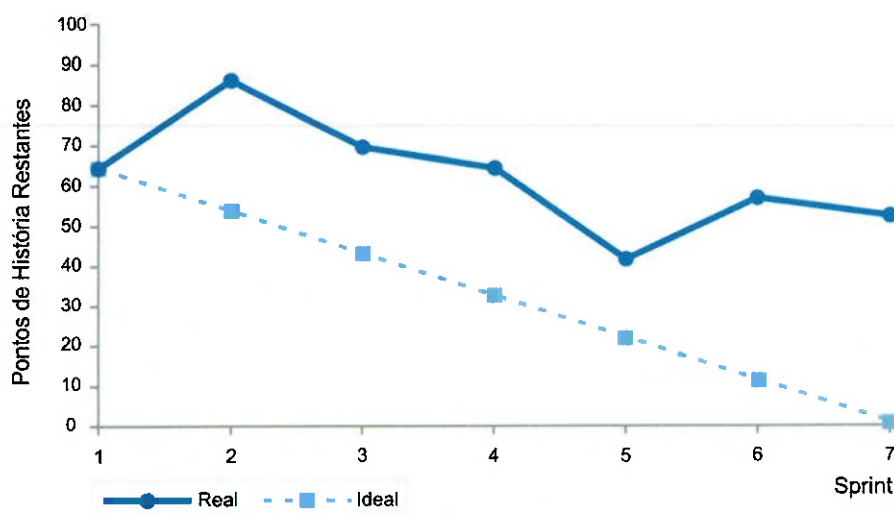


Figura 8 – Gráfico Burn-down de Release
Fonte: Mahnic e Zabkar (2012)

Já em contraste à figura 8 anterior, o gráfico de *burn-down* da *sprint* número dois (figura 9) apresenta uma tendência de queda mais evidente, indicando que a equipe de desenvolvimento concluiu quase todas as histórias de usuários previstas para a *sprint* número dois (MAHNIC; ZABKAR, 2012).

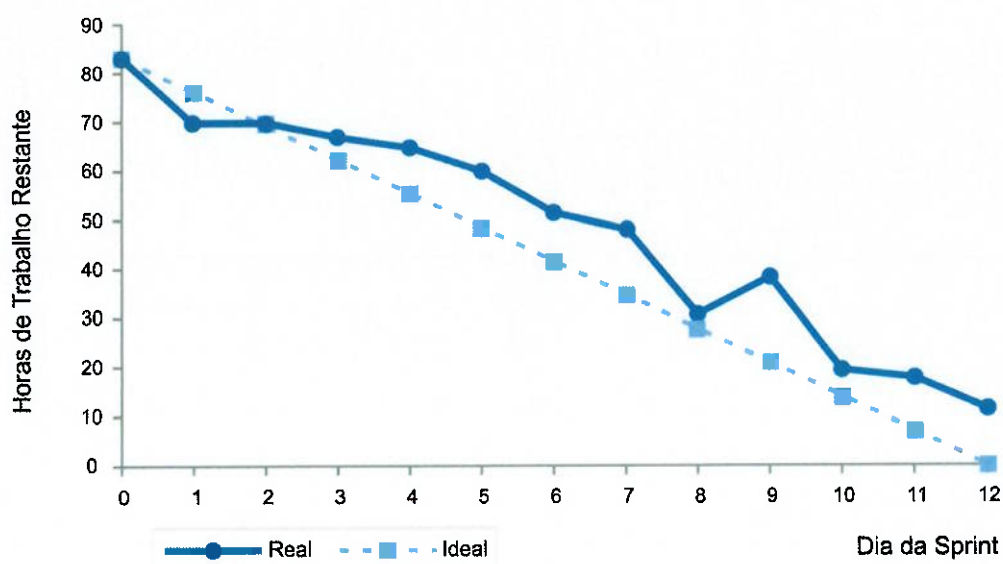


Figura 9 – Gráfico Burn-down de Sprint
Fonte: Mahnic e Zabkar (2012)

Além da velocidade e dos gráficos *burn-down* de *release* e *sprint*, Mahnic e Zabkar (2012) introduziram o cálculo dos índices de EVM no nível de *sprint*. Os autores destacam que EVM não é parte de SCRUM, mas é uma boa prática muitas vezes requerida (por exemplo, nos projetos do governo dos Estados Unidos). A abordagem proposta fornece os valores de índice de desempenho de prazos (SPI) e índice de desempenho de custos (CPI) em uma base diária, permitindo respostas imediatas em casos de desvios do plano, o que pode ser especialmente útil quando *sprints* mais longas são utilizadas. Para o cálculo de SPI e CPI foi necessário a coleta do número de horas gastas em cada tarefa entre duas reuniões diárias SCRUM consecutivas e o custo por hora foi considerado o mesmo para todos os membros da equipe SCRUM.

A figura 10 a seguir apresenta os valores de SPI e CPI para a segunda *sprint* do projeto analisado no estudo de Mahnic e Zabkar (2012).

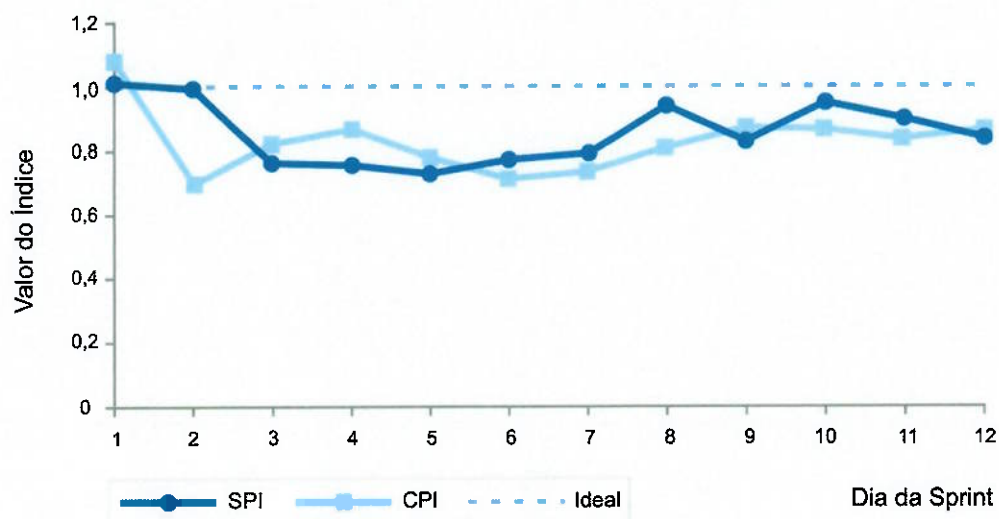


Figura 10 – Gráfico de CPI e SPI
Fonte: Mahnic e Zabkar (2012)

Embora o SPI forneça informações semelhantes ao gráfico *burn-down* da *sprint*, o CPI revela informações sobre a eficiência de custos, o que não está incluído em outras medidas de projetos ágeis (MAHNIC; ZABKAR, 2012).

2.3.2.5. Abordagem 5: Ágil em um Ambiente de Contrato do Governo

Em seu artigo, Alleman; Henderson e Seggelke (2003) afirmam que uma das insuficiências de um método de desenvolvimento ágil é a sua incapacidade de previsão dos custos e prazos futuros sem o embasamento em métricas anteriores.

Os autores propõem a utilização de XP como método de desenvolvimento de *software* para um projeto do Departamento de Energia (DOE) dos Estados Unidos. Introduzir processos de desenvolvimento ágeis nesse ambiente é um desafio, não por causa dos processos em si, mas por causa do reporte financeiro (ALLEMAN; HENDERSON; SEGSELKE, 2003).

Segundo Alleman, Henderson e Seggelke (2003), apresentar um relatório de velocidade (ou *velocity*) para o Departamento de Energia não era uma opção quando XP foi introduzido para o processo de desenvolvimento. Além disso, os pagamentos definidos em contrato eram baseados no valor agregado para o período de contabilização.

O desejo de utilizar métodos ágeis de desenvolvimento de *software*, mantendo a conformidade com as obrigações contratuais de relatórios pode parecer um conflito, mas, segundo os autores, não é. Alleman; Henderson e Seggelke (2003) propuseram algumas adaptações a XP de forma que pudesse ser combinada com a utilização de EVM.

Segundo os autores, uma abordagem para mensurar o valor agregado (EV) de um componente de *software* é a utilização de "requisitos testáveis" como critério de conclusão. Para Alleman; Henderson e Seggelke (2003) um requisito testável é "aquele que pode ser decomposto em um conjunto preciso, inequívoco, e indivisível de requisitos menores". Esses critérios só podem ser cumpridos se for possível escrever um caso de teste que possa validar se o requisito foi ou não implementado corretamente. Os autores propõem que os requisitos testáveis sejam verificados através de testes de unidade e testes funcionais.

Além da utilização de requisitos testáveis, foi proposta como abordagem de medição de EV que o progresso de uma determinada tarefa fosse definido somente como 0% ou 100% completo e que as durações das tarefas fossem mantidas suficientemente pequenas. Segundo os autores, seguir essa simples diretriz, pode fazer com que a "geração de valor" da técnica de gerenciamento de valor agregado torne-se semelhante, em muitos aspectos, a um método de desenvolvimento ágil de *software*. Definir tarefas em alto nível de granularidade mantém todos os princípios EVM, porém fará o comportamento do sistema de gestão possuir muitos dos atributos de um processo ágil (ALLEMAN; HENDERSON; SEGSELKE, 2003).

Alleman; Henderson e Seggelke (2003) afirmam que em XP a velocidade (ou *velocity*) é a medida do esforço investido para produzir *software* e que a técnica de gerenciamento de valor agregado é muito similar a esta medida. Os autores propõem que, para cada iteração, o valor planejado (PV) e o custo real (AC) sejam obtidos a partir de um sistema de controle de horas. O valor agregado (EV) pode ser definido por meio de histórias de usuários, tarefas e requisitos testáveis. Também a partir de histórias de usuários, tarefas e velocidade é possível obter as medidas de Estimativa para Terminar (ETC) e Estimativa de Prazo no Término (EACt) de EVM.

A abordagem proposta por Alleman; Henderson e Seggelke (2003) resume-se nos seguintes itens:

- Substituir a velocidade de XP por métricas EVM;
- Criar medidas de valor agregado (EV) de granularidade fina utilizando requisitos testáveis;
- Estabelecer o a linha de base de valor agregado (EV) no início de cada iteração;
- Capturar o custo real (AC) a partir de um sistema de controle de horas;
- Calcular a variância de custos (CV) e a variância de prazos (SV) a partir das três métricas base de EVM;
- Calcular a estimativa para terminar (ETC) e a estimativa de prazo no término (EACt) também partir das três métricas base de EVM.

Segundo Alleman; Henderson e Seggelke (2003) se essa abordagem for seguida é possível que uma equipe XP cumpra as exigências de EVM no planejamento, gerenciamento de custos e prazos e relatórios com facilidade.

2.3.2.6. Abordagem 6: Valor Agregado para Desenvolvimento Ágil

John Rusk (2009) inicia seu artigo citando um mito que pretende desmistificar: “projetos ágeis não podem definir escopo adiante”. Segundo o autor, *Feature Driven Development* (FDD) sempre define o escopo completo

no início do projeto, *Crystal Clear*, uma outra abordagem ágil, dá opção de definição de escopo no início e SCRUM registra as funcionalidades a serem desenvolvidas em uma lista evolutiva chamada *Backlog* do Produto.

Segundo Rusk (2009), métodos ágeis e a técnica de gerenciamento de valor agregado possuem um encaixe natural um para o outro e implementações EVM podem ser simplificadas para aplicação em projetos ágeis. O autor destaca que métodos ágeis provêm a opção de gerar uma definição razoavelmente completa do escopo no início do projeto e que isso é suficiente para EVM. A definição de metas globais para escopo, tempo e custo é necessária para que seja possível acompanhar o progresso em direção a tais metas.

A curva de um gráfico *burn-up*, muito utilizado no contexto ágil, é equivalente ao valor agregado (EV), segundo o autor. Portanto, a maioria das equipes ágeis já monitora o valor agregado, embora não utilizem este termo (RUSK, 2009).

Rusk (2009) considerou que o gráfico *burn-up* por si só não era suficiente para os projetos ágeis executados na empresa em que trabalhava, e que era necessário entender também o contexto financeiro do progresso da equipe. O autor propõe então um gráfico com três curvas:

- o progresso esperado, apresentado como uma porcentagem do total do projeto;
- quanto do produto já foi construído, como uma porcentagem do tamanho total do produto;
- os custos incorridos até o momento, como um percentual do orçamento total do projeto.

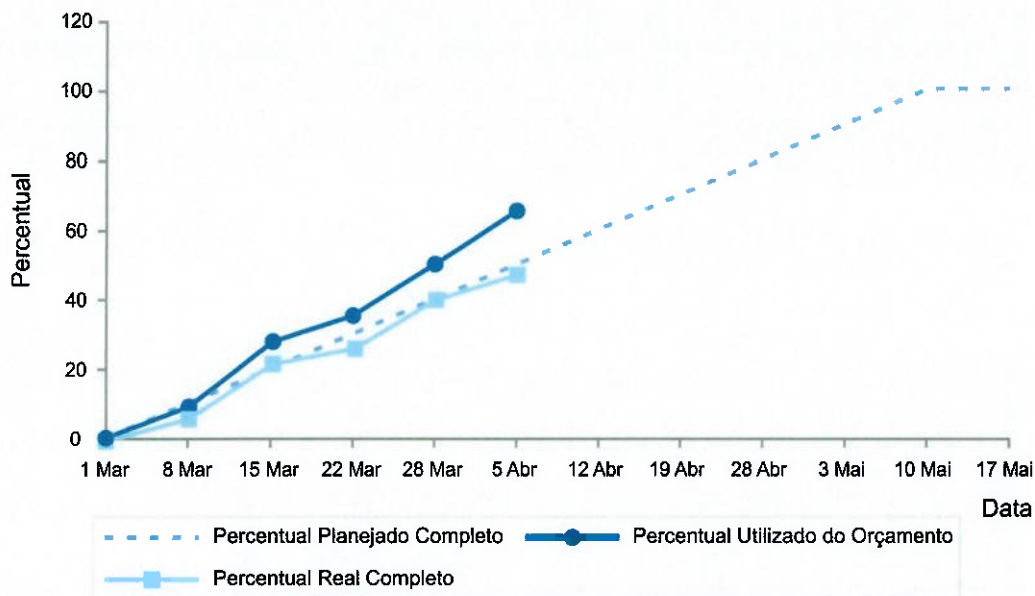


Figura 11 – Gráfico de EVM em percentuais
Fonte: Rusk (2009)

O autor não utiliza os acrônimos já conhecidos de EVM. Além disso, tudo é medido em porcentagens, ao invés da prática habitual de EVM que apresenta as medidas em valores monetários. Para Rusk (2009), essa abordagem facilita a comunicação com todos os envolvidos no projeto, desde os desenvolvedores até os executivos. Os percentuais podem ser mais facilmente entendidos para quem não está habituado com os termos de EVM, e evitam a necessidade de treinamento. Se necessário, é possível obter os valores em unidades monetárias apenas multiplicando os percentuais pelo orçamento total do projeto (RUSK, 2009).

A abordagem proposta por Rusk (2009) se resume nos seguintes pontos:

- Simplificação da distribuição de valor planejado ao longo do tempo, dado que os projetos ágeis entregam aproximadamente a mesma quantidade de trabalho em cada iteração;
- Simplificação da obtenção de custo real. Não há integração entre os sistemas financeiros ou de controle de ponto. A cada semana é registrada a quantidade de horas trabalhadas no projeto por cada membro da equipe;

- O valor agregado para uma tarefa acumula quando a tarefa for concluída e não é agregado nenhum valor para a conclusão parcial de uma tarefa;
- Não são utilizadas tabelas numéricas. O gráfico é o único artefato para a comunicação de valor agregado, e funciona melhor se analisado em períodos de tempo maiores, para uma *release* inteira, ou até mesmo para todo o projeto.

Rusk (2009) conclui que se for possível definir o escopo em alto nível no início do projeto e estabelecer um plano de progresso para esse escopo, o uso de EVM se torna viável e útil em projetos ágeis. No entanto, o autor destaca que, se o projeto for de caráter exploratório e não for possível estabelecer objetivos globais para o fim do projeto, o uso de EVM pode se tornar ineficaz.

2.4. Considerações do Capítulo

Este capítulo apresenta os conceitos de métodos ágeis, descreve suas motivações e objetivos, detalha seus valores e princípios e contextualiza historicamente seu surgimento. Apresenta também um dos mais difundidos métodos ágeis atualmente, SCRUM, detalhando seus papéis, artefatos e eventos, descreve o processo e aponta benefícios e dificuldades em sua adoção. Além dos conceitos de métodos ágeis e SCRUM, este capítulo discorre sobre a técnica de gerenciamento de valor agregado, suas métricas, variações, índices de desempenho e previsões. Semelhantemente, apresenta benefícios e dificuldades da aplicação de EVM e as diferenças que devem ser consideradas ao aplicá-la em projetos tradicionais ou ágeis de *software*. Posteriormente são apresentadas seis abordagens distintas da utilização da técnica de gerenciamento de valor agregado em conjunto com métodos ágeis, são detalhados os conceitos, metodologia utilizada e motivações de cada abordagem.

3. PROPOSTA DE INCORPORAÇÃO DE EVM A SCRUM

3.1. Introdução

Neste capítulo é apresentado um ambiente de desenvolvimento de software com algumas questões que precisam ser resolvidas e alguns requisitos que devem ser cumpridos. Para a resolução destes problemas, propõe-se a incorporação da técnica de gerenciamento valor ao método ágil SCRUM. Para isto, o ambiente foi analisado, foi realizado um comparativo de abordagens que já utilizam EVM com processos ágeis, o processo SCRUM foi analisado sob a perspectiva da incorporação de EVM ao processo e sob o contexto do ambiente. Posteriormente realiza-se a exemplificação da abordagem proposta.

3.2. Ambiente

Na filial brasileira de uma empresa multinacional de consultoria em tecnologia da informação atualmente são realizados diversos projetos de desenvolvimento de *software* customizados às necessidades de seus clientes. A grande maioria desses clientes está inserida no mercado financeiro, como por exemplo, bancos e seguradoras.

Dado que os projetos são executados utilizando o ciclo cascata como processo de desenvolvimento de *software*, os problemas são identificados tardiamente o que tende a aumentar o custo para a correção. Além disso, muitas vezes mudanças no escopo e na priorização de requisitos são solicitadas pelos clientes, gerando constantes replanejamentos.

Neste ambiente o escopo de trabalho de um projeto é definido através de uma lista de inventário, que contém os requisitos ou funcionalidades que devem ser construídos. Tipicamente as equipes possuem entre 10 a 15 profissionais de diferentes especialidades e um gerente de projetos. Um membro da equipe tem o papel de analista funcional ou analista de negócios.

Este profissional é responsável por identificar os requisitos do produto, entender o funcionamento do negócio e atuar como o representante do cliente perante a equipe.

Semanalmente são requeridos pela gerência executiva da organização reportes do andamento dos projetos com relação ao progresso e aos custos. Essas informações são providas pelos índices de desempenho de prazos e custos (SPI e CPI), obtidos aplicando-se a técnica de gerenciamento de valor agregado.

Cada gerente deve fornecer as medidas de EVM de seus projetos a um escritório de gerenciamento de projetos (do inglês *Project Management Office* - PMO) que recebe essas informações e as consolida para reportar os status de todos os projetos do portfólio. Após a consolidação dos dados, o escritório de projetos apresenta os resultados à gerência executiva da empresa. Além dos reportes internos à organização, as quantidades de horas planejadas e incorridas, bem como as métricas e índices de desempenho de cada projeto são enviados às partes interessadas de cada cliente semanalmente.

A partir do contexto atual desse ambiente, foi identificada uma lista de requisitos que precisa ser atendida:

- O processo de desenvolvimento precisa ser mais flexível, podendo contemplar mudanças de escopo constantes e entregas mais frequentes e mais rapidamente aos clientes;
- O reporte do avanço físico e orçamentário dos projetos ao escritório de projetos deve ser feito através das métricas e dos índices de desempenho de prazos e custos de EVM;
- O escritório de projetos deve reportar semanalmente o status de cada projeto a seu respectivo cliente;
- O escritório de projetos deve reportar semanalmente o progresso e custo de cada projeto do portfólio à gerência executiva da organização.

3.3. Análise comparativa das abordagens

As abordagens de utilização de EVM com ágil descritas neste trabalho foram analisadas e comparadas em função de alguns critérios que apareceram em várias das abordagens. Apesar de serem abordagens independentes, é possível observar características semelhantes entre elas.

O método ágil SCRUM é o mais utilizado entre as abordagens analisadas, tendo sido escolhido por 1, 3 e 4. Já XP aparece somente em um dos trabalhos, na abordagem 5. Os outros dois trabalhos (2 e 6) não definiram a utilização de nenhum método ágil específico, considerando que as características centrais da aplicação da técnica não se prendem a um método específico.

As abordagens 3, 4 aplicam a técnica de gerenciamento de valor agregado no âmbito de *sprint*, da mesma forma que a abordagem 5 aplica no âmbito da iteração. Já a abordagem 1 concentra-se em mensurar o progresso no nível de *release*. A abordagem 2 é a que garante o mais alto nível de monitoramento, aplicando sua técnica a todo o projeto. Os autores da abordagem 6 afirmam que sua proposta pode ser aplicada tanto a *releases*, quanto ao projeto como um todo.

Para a abordagem 1, o plano de trabalho estabelecido para aplicação da técnica é o *backlog* do produto. Para as abordagens 3 e 4, que aplicam a técnica no âmbito de *sprint*, o plano definido é o próprio *backlog* da *sprint*. Uma lista de requisitos é utilizada pelas abordagens 2 e 6, enquanto histórias de usuários e tarefas são utilizadas na abordagem 5.

Quantidade de pontos de história é a unidade de estimativa utilizada nas abordagens 1 e 3. Já as abordagens 4 e 6 realizam suas estimativas em horas de trabalho. A abordagem 2 utiliza a quantidade de requisitos (ou

features) para a aplicação da técnica e a abordagem 5 não cita qual unidade utiliza.

As abordagens 2 e 5 propõe a inclusão de um eixo na perspectiva monetária, possibilitando assim a representação do valor planejado, custo real e valor agregado em um gráfico *burn-up*. A abordagem 6 atua de forma semelhante, porém a representação de PV, AC e EV são realizadas de forma simplificada em percentuais. Já a abordagem 1 incorpora os índices de desempenho de prazos e custos (CPI e SPI) a um gráfico *burn-down*. Um gráfico contendo os valores de CPI e SPI em cada dia da *sprint* é proposto pela abordagem 4. A abordagem 3 não utiliza gráficos em sua proposta. A proposta de inclusão de duas escalas simultâneas das abordagens 1 e 2 não permite a comparação direta entre todos os elementos plotados no gráfico, já que estes elementos não estão na mesma perspectiva.

Não houve aplicação prática das abordagens 2 e 3 em projetos. Por outro lado a abordagem 1 foi aplicada a dois projetos, e as abordagens 4, 5 e 6 foram experimentadas em um projeto cada.

A tabela 8 a seguir resume as características centrais analisadas no comparativo:

Abordagem	Método ágil utilizado	Escopo da aplicação da técnica	Plano de trabalho estabelecido	Unidade de medida das estimativas	Gráficos utilizados	Aplicação em projetos
Abordagem 1: AgileEVM - Gerenciamento de Valor Agregado em Projetos Scrum (SULAIMAN; BARTON; BLACKBURN, 2006)	Scrum	Release	Backlog do Produto	Quantidade de pontos de história	Burn-down de release com CPI e SPI integrados	Aplicação em 2 projetos
Abordagem 2: Valor Agregado e Reporte Ágil (CABRI; GRIFFITHS, 2006)	Não definido, trata ágil em geral	Todo o projeto	Lista de requisitos	Quantidade de requisitos	Burn-up do projeto com PV, AC e EV integrados	-
Abordagem 3: Agregando Valor da forma Ágil (CHRISTOPH, 2011)	Scrum	Sprint	Backlog da Sprint	Quantidade de pontos de história	-	-
Abordagem 4: Mensurando Progresso de Projetos de Software baseados em Scrum (MAHNIC; ZABKAR, 2012)	Scrum	Sprint	Backlog da Sprint	Horas de trabalho	CPI e SPI por dias da sprint	Aplicação em 1 projeto
Abordagem 5: Fazendo Ágil Funcionar em um Ambiente de Contrato do Governo (ALLEMAN; HENDERSON; SEGELKE, 2003)	XP	Iteração	Histórias de usuário e tarefas	-	Burn-up do projeto com PV, AC e EV integrados	Aplicação em 1 projeto
Abordagem 6: Valor Agregado para Desenvolvimento Ágil (RUSK, 2009)	Não definido, trata ágil em geral	Release ou todo o projeto	Lista de requisitos	Horas de trabalho	Burn-up do projeto com PV, AC e EV integrados (porém em percentuais)	Aplicação em 1 projeto

Tabela 8 – Comparativo entre as abordagens
Fonte: Elaboração Própria

Do ponto de vista da aplicação da técnica de valor agregado, as abordagens 1, 3 e 4 foram consideradas as mais fiéis quanto à utilização das métricas, variâncias e índices de desempenho. Não foram necessárias grandes modificações para obtenção das métricas. Além disso, essas abordagens aproveitam-se de práticas já instituídas em projetos ágeis para a aplicação dos conceitos de EVM.

A abordagem 2 obtém SPI e CPI através do gráfico, porém estas medidas são similares e não equivalentes ao proposto por EVM. O foco da abordagem 5 é definir como utilizar um método ágil (no caso XP) em um ambiente altamente regulado no governo americano e que já utiliza a análise de valor agregado. Por isso não é tão detalhada no quesito de aplicação da técnica. A abordagem 6 não utiliza o vocabulário de EVM e as medidas de PV, AC e EV são realizadas em percentuais. Estas simplificações tornam a aplicação da

técnica mais didática e auxilia iniciantes ou partes interessadas não acostumados à EVM.

A tabela 9 a seguir apresenta um comparativo do ponto de vista das métricas, variâncias e índices de desempenho da técnica de valor agregado, analisando como cada uma delas é tratada nas diferentes abordagens.

Abordagem	Valor Planejado (PV)	Custo Real (AC)	Valor Agregado (EV)	Variância de Custos (CV)	Variância de Prazos (SV)	Índice de Desempenho de Custos (CPI)	Índice de Desempenho de Prazos (SPI)
Abordagem 1: AgileEVM - Gerenciamento de Valor Agregado em Projetos Scrum (SULAIMAN; BARTON; BLACKBURN, 2006)	PV = [Percentual planejado completo] * [Orçamento no término da release] ou PV = PPC * BAC	Custo orçado em PV para a release	EV = [Percentual completo da release] * [Orçamento no término da release] ou EV = APC * BAC	CV = EV - AC	SV = EV - PV	CPI = EV / AC	SPI = EV / PV
Abordagem 2: Valor Agregado e Reporte Ágil (CABRI; GRIFFITHS, 2006)	Custo planejado para o projeto	Custo real do projeto	Obtido no gráfico (derivado da quantidade de requisitos completos)	-	Obtido no gráfico (em quantidade de requisitos)	CPI = [Custo planejado] / [Custo Real]	SPI = [Requisitos completos] / [Requisitos planejados]
Abordagem 3: Agregando Valor da forma Ágil (CHRISTOPH, 2011)	Custo planejado para a sprint	Custo real da sprint	EV = PV / [Velocidade planejada] * [Pontos de história completos na sprint]	-	-	CPI = EV / AC	SPI = EV / PV
Abordagem 4: Mensurando Progresso de Projetos de Software baseados em Scrum (MAHNIC; ZABKAR, 2012)	PV = [Quantidade de horas previstas] * [Custo por hora]	AC = [Quantidade de horas incorridas] * [Custo por hora]	EV = [Quantidade de horas incorridas para tarefas concluídas] * [Custo por hora]	-	-	CPI = EV / AC	SPI = EV / PV
Abordagem 5: Fazendo Ágil Funcionar em um Ambiente de Contrato do Governo (ALLEMAN; HENDERSON; SEGGELE, 2003)	Obtido através de um sistema de controle de horas	Obtido através de um sistema de controle de horas	Substitui a velocidade de XP pelo valor agregado	CV = EV - AC	SV = EV - PV	-	-
Abordagem 6: Valor Agregado para Desenvolvimento Ágil (RUSK, 2009)	Definido por uma curva simplificada num gráfico em percentuais.	Definido por uma curva simplificada num gráfico em percentuais.	Definido por uma curva simplificada num gráfico em percentuais.	Obtido no gráfico (se a curva do EV está acima ou abaixo da curva do AC)	Obtido no gráfico (se a curva do EV está acima ou abaixo da curva do PV)	-	-

Tabela 9 – Comparativo entre as abordagens do ponto de vista de EVM
Fonte: Elaboração Própria

Embora as abordagens tenham sido apresentadas de forma independente, suas propostas não se mostraram necessariamente excludentes e permitem que características sejam combinadas.

3.4. Processo

A partir do ambiente apresentado, dos requisitos elencados e da análise comparativa das abordagens, o processo e os artefatos, papéis e eventos de SCRUM foram analisados. O contexto atual para cada um dos elementos foi observado. Foram identificadas questões centrais que precisam de modificações, sob a perspectiva da utilização de SCRUM e EVM neste ambiente. Com base nos conceitos obtidos na revisão bibliográfica, são propostas soluções ou alternativas para as questões identificadas e são apresentadas as justificativas que nortearam essas decisões. Foi definida então uma abordagem de aplicação da técnica de gerenciamento de valor agregado juntamente com SCRUM no ambiente definido.

3.4.1. Papéis

Cada papel desempenhado no processo SCRUM foi analisado sob a perspectiva do ambiente atual, seus requisitos e com foco na incorporação da análise de valor agregado ao processo SCRUM.

3.4.1.1. Dono do Produto

- **Contexto atual:**

Atualmente, um membro da equipe tem o papel de analista funcional. Esta pessoa é a responsável por obter as necessidades do cliente, entender o funcionamento de seu negócio e repassar essas informações ao restante da equipe. O colaborador que exerce este papel tem interação constante com as partes interessadas do cliente e as representa no dia a dia do projeto. Possui controle do inventário e a responsabilidade de priorizar os requisitos a serem construídos.

- **Questões chave:**

Muitas vezes este profissional tem conhecimento limitado, por não ser um especialista da área de negócios do cliente. O tempo de análise dos

requisitos e definição da solução, em grande parte dos projetos, é reduzido e, falhas de comunicação podem comprometer a solução adotada.

Sendo somente um representante do cliente, o analista funcional não tem o poder para tomada de decisão. Decisões importantes precisam ser avaliadas e discutidas com as partes interessadas dos clientes, o que acaba reduzindo a agilidade da equipe.

Além disso, por não fazer parte do quadro de colaboradores do cliente, sua visão tende a favorecer a sua empresa ou a equipe da qual faz parte, principalmente quando há questões contratuais envolvidas.

- **Solução proposta:**

Propõe-se que o papel de dono do produto seja exercido por um colaborador do cliente, dado que sua experiência no negócio poderá maximizar o valor do produto e agregar mais conhecimento à equipe. Por ter alçada para tomada de decisões, o dono do produto do cliente pode potencializar a agilidade da equipe.

- **Justificativa:**

Para Rubin (2012), o dono do produto é considerado o responsável pela entrega de bons resultados de negócios, deve ser competente para tomar decisões difíceis e a organização deve respeitar suas decisões. Por estes fatores o autor afirma que em um modelo de desenvolvimento terceirizado, onde uma empresa contrata outra para o desenvolvimento da solução, o ideal é que o dono do produto seja um representante de quem está contratando os serviços.

3.4.1.2. Scrum Master

- **Contexto atual:**

Todo projeto é acompanhado por um gerente de projetos. Este profissional é responsável tanto por gerenciar o andamento, quanto por reportar precisa e frequentemente o status do projeto.

Atualmente não há qualquer projeto que utilize SCRUM, ou outro método ágil, na empresa. A figura do *Scrum Master* não é conhecida em detalhes pela maioria dos profissionais e gerência.

- **Questões chave:**

O foco do *Scrum Master* é auxiliar todos os envolvidos a compreender e abraçar os valores, princípios e práticas de SCRUM. Do ponto de vista de SCRUM, não é atividade do *Scrum Master* coletar as métricas e calcular os índices de desempenho e variâncias da técnica de valor agregado.

- **Solução proposta:**

Para o papel de *Scrum Master*, sugere-se que haja treinamentos específicos para os membros da equipe, de forma a aprofundar o conhecimento e capacitar os profissionais a exercerem tal papel. Propõe-se que o papel do *Scrum Master* seja exercido por um membro da equipe que já tenha sido devidamente treinado.

Tendo em vista que não seria atividade do *Scrum Master* obter as métricas e calcular as variâncias e índices de desempenho da técnica de gerenciamento de valor agregado, propõe-se que o papel exercido atualmente pelo gerente de projetos seja mantido no processo.

- **Justificativa:**

Segundo Rubin (2012), o *Scrum Master* é o líder do processo SCRUM, age como um treinador tanto para a equipe de desenvolvimento, quanto para o dono do produto. Atua frequentemente na remoção de impedimentos e auxilia a equipe SCRUM e o restante da organização a desenvolver sua alta-performance.

Porém, o *Scrum Master* não é o líder do projeto e não tem responsabilidade sobre a gestão do avanço físico e orçamentário dos projetos, nem quanto a reportes de status a alta gerência ou a clientes. A manutenção do gerente de projetos como um papel do processo visa atender os requisitos de reportes estabelecidos pelo ambiente, sem interferir no papel exercido pelo *Scrum Master*.

3.4.1.3. Equipe de Desenvolvimento

- **Contexto atual:**

O tamanho das equipes é definido a partir da quantidade total de horas estimadas para o projeto e do prazo de entrega desejado pelas partes interessadas do cliente. A partir dessas duas variáveis, são calculadas aproximadamente com quantas pessoas é possível entregar o volume de horas dentro do prazo definido. Tipicamente as equipes de desenvolvimento possuem entre 10 a 15 pessoas e um gerente que pode gerenciar mais de uma equipe paralelamente.

As equipes são formadas por diferentes especialistas: Programadores, Analistas Programadores, Analistas de Testes, Analistas Funcionais e Analistas de Sistemas. O custo destes profissionais varia de acordo com sua especialidade e também com sua faixa salarial. O custo do projeto é calculado pela quantidade total de horas estimadas para o projeto multiplicada pela média do custo por hora dos profissionais que compõe a equipe.

- **Questões chave:**

Algumas vezes as equipes mudam durante o andamento do projeto. Isto pode trazer impacto nos prazos, na divisão das tarefas, e, além disto, como o custo médio da equipe por hora é utilizado para o cálculo do custo do projeto, pode haver impactos diretos no custo total do projeto. Além disso, a medida de velocidade pode ser utilizada como base para projeções de data de

entrega, e se houver modificações na equipe, este indicador poderá não refletir a realidade.

Por definição, SCRUM não trabalha com especialistas na equipe de desenvolvimento. O papel de todos os membros é o mesmo e cada um está apto a desenvolver qualquer tarefa.

- **Solução proposta:**

Propõe-se que a equipe seja mantida constante dentro da mesma *sprint*, dado que as métricas EVM serão obtidas por *sprint*. No início de cada *sprint*, deve ser calculado o custo médio por hora dos profissionais da equipe.

Deve-se haver uma mudança da cultura organizacional quanto à formação de equipes de especialistas.

- **Justificativa:**

Segundo Christoph (2011), métodos ágeis tipicamente assumem um período de tempo fixo para cada iteração e uma equipe constante para a realização do trabalho. Sendo assim, a medida de velocidade pode fornecer um indicador importante para obter quanto tempo levará a conclusão de um determinado *backlog* do produto. Porém, se durante o projeto a composição da equipe e a duração das *sprints* mudarem constantemente, a medida de velocidade poderá não fornecer um valor preditivo correto.

Em SCRUM, a equipe de desenvolvimento deve fazer todo o trabalho para produzir uma ou mais fatias verticais de funcionalidade do produto em cada *sprint*, incluindo a concepção, desenvolvimento, integração e testes dessa funcionalidade. Para isto, é necessário que a equipe seja hábil e competente em todas essas tarefas. Diluir o trabalho entre diferentes equipes com especialidades específicas pode se tornar um sério impedimento para o sucesso do uso de SCRUM (RUBIN, 2012).

Na aplicação da técnica de gerenciamento de valor agregado em projetos SCRUM realizada por Sulaiman; Barton e Blackburn (2006) os resultados obtidos não foram influenciados pelo tamanho da equipe. Mesmo em equipes pequenas (de 5 a 7 pessoas) ou grandes (de 15 a 18 pessoas) a aplicação da técnica foi eficiente. Porém estas equipes se mantiveram constantes durante a análise dos autores.

3.4.2. Artefatos

Os artefatos gerados durante o processo foram analisados para que os requisitos do ambiente sejam cumpridos e que a incorporação do valor agregado ao processo SCRUM seja possível.

3.4.2.1. *Backlog* do Produto

- **Contexto atual:**

No ambiente descrito, o escopo de trabalho é definido através de uma lista de inventário que contém os requisitos, funcionalidades e/ou componentes que devem ser desenvolvidos ou alterados. O cliente ou a equipe podem incluir novos itens no inventário durante o andamento do projeto. O inventário é controlado pelo analista funcional e é priorizado com apoio do cliente, de acordo com o valor que cada item pode agregar ao negócio.

A partir do inventário definido, alguns membros mais experientes da equipe de desenvolvimento realizam estimativas em horas para elaboração de cada item. As estimativas são feitas com base na quantidade de artefatos a serem construídos (por exemplo: telas do sistema, relatórios, etc.) e em complexidades previamente definidas como: muito simples, simples, média, complexa e muito complexa. Para cada complexidade há um esforço médio pré-definido e baseado em projetos anteriores.

- **Questões chave:**

Muitas vezes não é possível quantificar exatamente o número de artefatos a partir de um requisito, ou ainda classificar de forma precisa a complexidade dos artefatos. Isto torna as estimativas subjetivas e dependentes da experiência prévia do profissional que as executa. Além disto, a granularidade dos itens do inventário pode ser diferente. Por exemplo, no inventário podem conter requisitos, funcionalidades ou ainda tarefas.

- **Solução proposta:**

Propõe-se a utilização do inventário como um *backlog* do produto, porém cada item definido como uma história de usuário, mantendo-se, assim, o mesmo nível de granularidade para todos os itens. Para a estimativa, é proposto que cada item do *backlog* do produto seja estimado em pontos de história, possibilitando que toda a equipe atue coletivamente nas estimativas.

- **Justificativa:**

O fator crítico para aplicação de EVM a métodos ágeis segundo Sulaiman e Smits (2007) é que haja um *backlog* do produto, um plano de *release* definido para que seja possível obter a quantidade de *sprints* para a *release*, uma maneira para identificar o custo real do projeto e que sejam feitas estimativas numéricas de algum tipo, sendo em horas, pontos de história, dias da equipe ou qualquer outra estimativa consistente de tamanho.

Já para Rusk (2009), se for possível definir o escopo em alto nível no início do projeto e estabelecer um plano de progresso para esse escopo, o uso de EVM se torna viável e útil em projetos ágeis. O autor afirma também que métodos ágeis provêm a opção de gerar uma definição razoavelmente completa do escopo no início do projeto e que isso é suficiente para EVM.

3.4.2.2. *Backlog da Sprint*

- **Contexto atual:**

Tipicamente os projetos são planejados de forma completa logo em seu início. Porém, atualmente alguns projetos são subdivididos em entregas menores, visando possibilitar o paralelismo de atividades, ganho de produtividade, e redução de riscos, trabalhando-se antecipadamente em atividades que fazem parte do caminho crítico do projeto, por exemplo. Para isto, a lista de inventário é subdividida em trechos de funcionalidade que façam sentido ser entregues em conjunto, e que sejam viáveis do ponto de vista da elaboração em paralelo.

As estimativas dos itens constantes no inventário da entrega são realizadas em horas. O custo total da entrega é calculado multiplicando-se a quantidade total de horas da entrega, pelo custo médio por hora dos profissionais da equipe. A quantidade de horas planejadas e incorridas são semanalmente reportadas aos clientes.

- **Questões chave:**

Apesar de existirem entregas parciais de trechos de funcionalidade em alguns projetos, estes itens não são priorizados sob o ponto de vista do maior valor agregado ao negócio do cliente, mas sim da perspectiva do ganho de produtividade e velocidade da equipe, redução de riscos e planejamento do cronograma.

As estimativas em horas, além de já serem utilizadas em todos os projetos da organização, facilitam o cálculo dos custos em função do custo médio por hora da equipe. Além disso, o volume de horas totais da entrega ou do projeto também é apresentado aos clientes como indicador do progresso do projeto e dos custos incorridos.

- **Solução proposta:**

É proposto que os itens do *backlog* do produto que agregarão maior valor ao negócio do cliente, de acordo com a visão do dono do produto, sejam refinados e subdivididos em tarefas menores, estimadas em horas de trabalho para formar o *backlog* da *sprint*.

Propõe-se que o *backlog* da *sprint* contenha o valor planejado (PV) para cada tarefa, a indicação se a tarefa foi ou não concluída, o valor agregado (EV) para as tarefas concluídas e o custo real (AC) de cada tarefa.

- **Justificativa:**

Os itens do *backlog* do produto são colaborativamente avaliados e priorizados pela equipe Ágil. As funcionalidades que são as determinadas a oferecer o mais alto valor de negócio são movidas para o *backlog* da *sprint*, onde são avaliadas por meio de um conjunto de critérios baseados em complexidade, experiência com um requisito semelhante, dificuldade planejada, conhecimentos necessários, ou outros fatores apropriados (CHRISTOPH, 2011).

Os valores planejados das tarefas podem ser obtidos através da quantidade de horas estimadas para a execução da tarefa multiplicada pelo custo médio por hora da equipe. Para o custo real, cada membro da equipe deve apontar quantas horas foram incorridas em cada atividade em um sistema de controle de horas. Já o valor agregado é obtido na conclusão de cada item. A conclusão dos itens poderá ser feita durante a reunião diária SCRUM.

3.4.2.3. Incremento de Produto Potencialmente Entregável

- **Contexto atual:**

As entregas atuais, na maioria vezes, não representam os requisitos com maior valor agregado ao negócio do cliente. Os testes integrados são executados somente no final do projeto e erros são tardiamente identificados.

- **Questões chave:**

O foco da subdivisão atual dos projetos em entregas menores não está no cliente. A perspectiva principal é a produtividade da equipe. Também são consideradas as funcionalidades reutilizáveis e possíveis questões técnicas que influenciariam a ordem de execução.

- **Solução proposta:**

Da perspectiva da inclusão das métricas de valor agregado ao processo, não são necessárias modificações no artefato Incremento de Produto Potencialmente Entregável. Sugere-se a utilização do mesmo seguindo a definição padrão de SCRUM.

- **Justificativa:**

SCRUM requer que as equipes construam um incremento de funcionalidade do produto a cada *sprint*. Este incremento deve ser "potencialmente entregável" porque o dono do produto pode escolher a implementar a funcionalidade entregue imediatamente. Isto requer que este incremento tenha sido consistentemente testado, bem estruturado e documentado, e esta é a definição de conclusão de deste incremento (SCHWABER, 2004).

3.4.3. Eventos

Foram analisados os eventos de SCRUM de modo a possibilitar a incorporação da análise de valor agregado ao processo e a atender as necessidades específicas do ambiente apresentado.

3.4.3.1. Sprint

- **Contexto atual:**

No ambiente descrito, alguns projetos podem ser subdivididos em entregas menores, possibilitando paralelizar o desenvolvimento, ou, por exemplo, construir blocos de funcionalidade reutilizáveis no início do projeto. Um projeto estimado em dez mil horas de esforço pode ser subdividido em cinco entregas de duas mil horas, por exemplo. Em média, cada entrega dura entre um e dois meses. Nos projetos atuais, as atividades do cronograma podem ser consideradas parcialmente completas.

- **Questões chave:**

As entregas parciais, na maioria das vezes, não têm foco no valor agregado ao negócio do cliente, mas na produtividade da equipe. As sprints devem ser planejadas de forma a maximizar o retorno sobre o investimento do cliente e possibilitar entregas mais frequentes.

Embora as atividades possam ser consideradas parcialmente completas, muitas vezes após uma inspeção de qualidade, essas atividades podem não ter atingido o nível esperado de qualidade e consequentemente necessitarão de correções ou retrabalho. O percentual de completude muitas vezes é subjetivo e, além disso, é comum que os 10% restantes para a conclusão de uma tarefa tomem muito mais do que o previsto inicialmente.

- **Solução proposta:**

Sugere-se a utilização de *sprints* de duração fixa com um mês de duração. Do ponto de vista da incorporação da análise de valor agregado a SCRUM, propõe-se que os itens do *backlog* da *sprint* que não forem completamente finalizados não tenham nenhum valor agregado. Somente as tarefas que, após realizadas as inspeções de qualidade e consideradas 100% concluídas pela equipe e dono do produto devem ter seu valor agregado creditado.

- **Justificativa:**

Segundo Rubin (2012), sprints de curta duração são mais fáceis de planejar, geram *feedback* mais rapidamente, permitem entregas antecipadas e mais frequentes, e também provêm vários e significativos pontos de checagem e controle.

Um fator relevante para a que possibilita a incorporação da análise de valor agregado a SCRUM é que as *sprints* possuem tamanho fixo, e isto, de acordo com Christoph (2011), torna-se um facilitador para a aplicação de EVM dado que o tamanho da *sprint* pode ser ajustado de forma coordenada com os ciclos de reporte de status.

Não há agregação de valor para requisitos parcialmente concluídos, conforme abordagem proposta por Alleman; Henderson e Seggelke (2003). Semelhantemente na abordagem de Sulaiman; Barton e Blackburn (2006) não há crédito de valor para itens do *backlog* parcialmente completos. Os itens do *backlog* estão concluídos ou não (0% ou 100%). O valor planejado para a tarefa só é agregado quando o cliente ou dono do produto aceitam o item como concluído.

3.4.3.2. Planejamento da Sprint

- **Contexto atual:**

Os projetos, na maioria das vezes, são planejados por completo no início. Há um cronograma detalhado para as atividades granulares. Apenas alguns membros da equipe mais experientes participam do planejamento e realizam estimativas. Mudanças no planejamento, ou alterações de escopo durante o projeto são vistos como problemas no ambiente atual. Replanejamentos de cronograma são, na maioria das vezes, trabalhosos e ineficientes.

Do ponto de vista da técnica de gerenciamento de valor agregado, a partir do momento que estão definidas a linha de base e os esforços necessários para a conclusão de cada atividade, o valor planejado do projeto como um todo pode ser calculado.

- **Questões chave:**

Com base na velocidade das *sprints* anteriores, é possível projetar a capacidade da equipe para a próxima *sprint*. A partir dos itens presentes no *backlog* do produto, pode-se avaliar quais agregam maior valor ao negócio, e com base em suas estimativas em pontos de história, definem-se quantos itens poderão compor o *backlog* da *sprint*.

Nesta reunião de planejamento, além da previsão da capacidade da equipe e da definição de quais itens do *backlog* do produto serão trabalhados no *backlog* da próxima *sprint*, é necessário que todos os itens sejam detalhados

em tarefas, e suas estimativas sejam refinadas pela equipe. Com as estimativas definidas, é possível obter o valor planejado para cada tarefa, e consequentemente o valor planejado para toda a *sprint*.

- **Solução proposta:**

Para compor o *backlog* da *sprint* propõe-se que os itens do *backlog* do produto selecionados sejam detalhados em tarefas e suas estimativas sejam refinadas em horas de trabalho da equipe. As horas de trabalho definidas para cada tarefa devem ser multiplicadas pelo custo médio por hora da equipe, gerando-se assim o valor planejado (PV) de cada tarefa que compõe *backlog* da *sprint*, e consequentemente o valor planejado total da *sprint*.

- **Justificativa:**

Para Christoph (2011) a geração de dados de EVM requer que em alguma fase do projeto ocorra o planejamento do trabalho e, em seguida, o trabalho completo seja mensurado contra esse plano. O autor afirma que no contexto ágil isto deve ocorrer no planejamento da *sprint*, onde funcionalidades ou histórias de usuários do *backlog* do produto são avaliadas e priorizadas pela equipe ágil e os requisitos que oferecem o mais alto valor de negócio são movidos para o *backlog* da *sprint*.

Na abordagem proposta por Mahnic e Zabkar (2012) o tamanho de cada história de usuário foi estimado na reunião de planejamento da *sprint*, enquanto que a quantidade de trabalho executado e restante foram registrados todos os dias nas reuniões diária da equipe. Estas medidas, juntamente com alguns parâmetros básicos do projeto, como a duração da *sprint*, e o custo de horas de engenharia de cada desenvolvedor, permitem o cálculo das medidas derivadas da técnica de valor agregado que indicam o andamento do projeto.

Segundo Sulaiman; Barton e Blackburn (2006) se não houve nenhuma *sprint* concluída anteriormente, uma estimativa de uma velocidade inicial deve ser

feita. Uma vez em curso, as medidas de velocidade real podem ser utilizadas para estimar as velocidades futuras da equipe.

3.4.3.3. Execução da Sprint

- **Contexto atual:**

No ambiente citado, atualmente os projetos são realizados utilizando o ciclo cascata. Mesmo projetos que são subdivididos em entregas menores, cada entrega executa o ciclo cascata em menor escala como processo de desenvolvimento. A definição de quais itens serão construídos em qual entrega e qual a ordem de execução leva em consideração os pré-requisitos técnicos de arquitetura e de reuso de funcionalidades, bem como a possibilidade de avanço em paralelo, para o ganho de produtividade da equipe.

- **Questões chave:**

Assumindo que os itens do *backlog* da *sprint* não iniciam simultaneamente, a equipe deve determinar qual é o próximo item a ser construído. Uma forma é selecionar o item com a maior prioridade definida no *backlog* da *sprint*, o que de garantiria que quaisquer itens não concluídos durante a *sprint* têm menor prioridade do que os concluídos. Porém, cabe à equipe avaliar se há alguma dependência técnica, ou de habilidades, para definir a ordem de execução.

- **Solução proposta:**

Sugere-se a que o evento de execução da *sprint* seja realizado conforme a definição do processo SCRUM. Não são necessárias alterações no ponto de vista da incorporação da técnica de valor agregado ao processo.

- **Justificativa:**

Tratar a execução da *sprint* como um mini projeto em cascata, segundo Rubin (2012), é uma abordagem perigosa. Utilizar a estratégia de “mini cascata” apesar de parecer lógico, pode ser arriscado. Se o time ficar sem tempo para finalizar os testes do produto, ao final da *sprint* é possível que

90% de cada característica tenha sido concluído, mas que nenhuma característica esteja 100% finalizada. O dono do produto, e consequentemente o cliente, não receberá nenhum valor econômico de um trabalho parcialmente feito.

3.4.3.4. Reunião Diária SCRUM

- **Contexto atual:**

As equipes tipicamente têm uma reunião semanal. O gerente do projeto participa desta reunião e realiza o acompanhamento das atividades de cada membro da equipe. As tarefas da semana seguinte são distribuídas para cada responsável. Caso haja algum impedimento ou dificuldade, estes podem ser discutidos na reunião semanal, ou, se necessário, realizadas outras reuniões sob demanda.

- **Questões chave:**

A visão do progresso das atividades do projeto só é obtida semanalmente. Algumas vezes os membros da equipe concluem suas atividades antes da reunião semanal, e não há agilidade na redistribuição de tarefas. Alguns impedimentos poderiam ser resolvidos de forma mais rápida, aumentando a produtividade da equipe. A equipe poderia ser auto gerenciada, sem que houvesse a necessidade da participação do gerente do projeto.

- **Solução proposta:**

Propõe-se a realização da reunião diária SCRUM, onde além de cada membro da equipe deverá apontar quantas horas dispendeu em cada tarefa, e atualizar o avanço físico de cada uma de suas atividades. Isto permitirá que sejam obtidos o valor agregado (EV) e custo real (AC) por tarefa. A partir desses dados será possível obter o valor agregado total e o custo real total para a *sprint*. Estas informações poderão alimentar gráficos *burn-down* e *burn-up*, com o objetivo de identificar previamente problemas que necessitam de ações corretivas.

- **Justificativa:**

Através de gráficos *burn-up* e *burn-down*, é possível obter status instantâneo do avanço do projeto. Tais gráficos podem apontar, logo no início, problemas que podem precisar de medidas corretivas a ser tomadas, a principal vantagem da análise valor agregado. Isto só pode ser feito por meio de atualizações regulares dos dados do gráfico, o que pode ser realizado através das reuniões diárias, bem como através da prática ágil de iterações de curta duração (CABRI; GRIFFITHS, 2006).

Além da dos gráficos *burn-down* a abordagem de Mahnic e Zabkar (2012) introduz o cálculo dos índices de EVM no nível de *sprint*. Os autores propõem que sejam obtidos os valores dos índices de desempenho de prazos e custos (SPI e CPI) em uma base diária. Para isto, foi necessário a coleta da quantidade de horas gastas em cada tarefa entre duas reuniões diárias SCRUM consecutivas e o custo por hora foi considerado o mesmo para todos os membros da equipe. Acompanhar SPI e CPI diariamente permite respostas imediatas em casos de desvios do plano.

3.4.3.5. Revisão da Sprint

- **Contexto atual:**

No ambiente atual não são realizadas reuniões ao término de entregas ou ao final de projetos. Após a conclusão dos testes integrados, uma versão do sistema é disponibilizada ao cliente que efetua os testes funcionais ou de aceitação. Não há uma reunião de apresentação ou de inspeção realizada pela equipe em conjunto com as partes interessadas do cliente.

- **Questões chave:**

Dado que o foco da revisão da *sprint* é a inspeção e adaptação do produto, não são necessárias modificações no evento do ponto de vista do ambiente ou da inclusão de valor agregado ao processo.

- **Solução proposta:**

Não há propostas de modificações para o evento de revisão da *sprint*. Sugere-se a execução do evento conforme definido pelo processo SCRUM.

- **Justificativa:**

O objetivo da revisão da *sprint* é inspecionar o produto que está sendo construído e modificar o *backlog* do produto se necessário. A equipe de desenvolvimento apresenta o que foi concluído na *sprint*, e o dono do produto confirma o que está concluído ou não em sua visão (GRIFFITHS, 2012).

3.4.3.6. Retrospectiva da Sprint

- **Contexto atual:**

A gerência executiva avalia se os prazos e custos foram mantidos dentro do esperado para cada entrega ou ao final de um projeto. Caso negativo atuam na mitigação dos problemas nas entregas ou projetos posteriores. Por outro lado, não há qualquer reunião ou alinhamento com a equipe após a conclusão de uma entrega ou mesmo do projeto como um todo.

- **Questões chave:**

Não só a alta gestão deve estar ciente dos desvios de custos ou de progresso dos projetos. Na retrospectiva da *sprint* a equipe deve ser incentivada a inspecionar o processo, as pessoas e ferramentas de forma a refletir e identificar maneiras de torná-los mais efetivos. Isto deve ser feito também nas questões de avanço físico e orçamentário dos projetos.

Nos métodos ágeis, são utilizados os gráficos *burn-up* e *burn-down* para acompanhamento do progresso físico do projeto, porém não há outra medida utilizada para o controle de custos.

- **Solução proposta:**

No evento de retrospectiva da *sprint* sugere-se a inclusão da análise do valor agregado e a geração dos gráficos de *burn-down* e *burn-up* com as métricas

e indicadores de EVM incorporados, com o objetivo de comparar os custos e o progresso orçados ao realizado, e caso haja discrepâncias, ações corretivas possam ser tomadas para que os problemas sejam mitigados nas *sprints* subsequentes.

- **Justificativa:**

Enquanto a revisão de *sprint* é um momento para inspecionar e adaptar o produto, a retrospectiva da *sprint* é uma oportunidade de inspecionar e adaptar o processo (RUBIN, 2012).

Os autores Mahnic e Zabkar (2012) afirmam que embora o SPI possa fornecer informações semelhantes às de um gráfico *burn-down* de *sprint*, o CPI revela informações sobre a eficiência de custos, e isto não está incluído em outras medidas comumente utilizadas em projetos ágeis.

Além disso, em seu artigo, Sulaiman; Barton e Blackburn (2006) afirmam que a maioria dos desenvolvedores reagiu positivamente com a disponibilidade dos dados da análise de valor agregado e que sentiam necessidade de uma análise a fim de determinar a causa de um certo problema, como por exemplo, uma queda repentina de velocidade da equipe, e que após a aplicação da técnica existem dados disponíveis para que se possa avaliar. Na análise dos resultados dos autores, foi possível observar que a equipe foi capaz de identificar rapidamente mudanças nas métricas chave de EVM, e isso ajudou na melhoria contínua da equipe.

3.5. Resumo da proposta

Considerando as propostas realizadas na seção anterior para a utilização da análise de valor agregado e SCRUM no ambiente descrito, resume-se a abordagem sugerida na figura 12 a seguir:

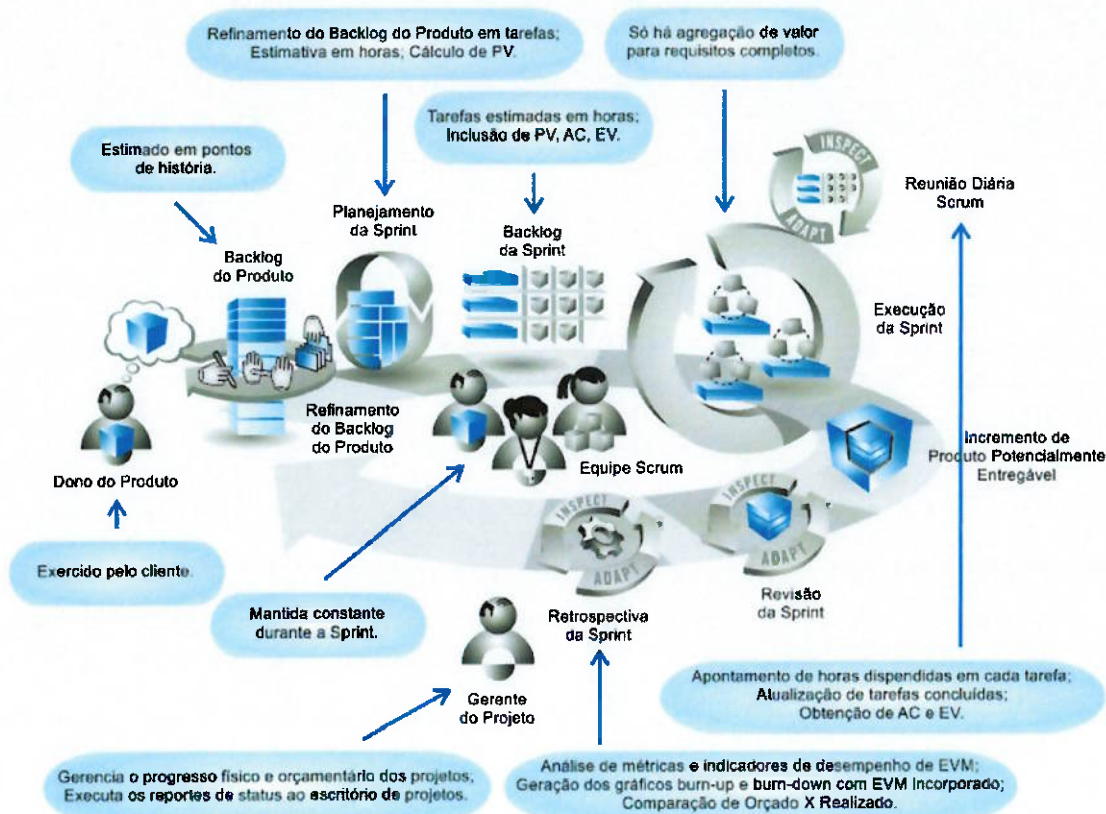


Figura 12 – Resumo da proposta de incorporação de EVM a SCRUM
Fonte: Adaptado de Rubin (2012)

3.6. Aplicação

A partir da análise do processo SCRUM sob a perspectiva da incorporação da técnica de valor agregado e com base no contexto do ambiente apresentado, exemplifica-se a aplicação da abordagem definida neste trabalho.

Inicia-se a partir do *backlog* do produto onde cada requisito foi estimado em pontos de história pela equipe de desenvolvimento. A tabela 10 a seguir exemplifica o *backlog* do produto estimado em pontos de história.

Req. N°	Requisito	Estimativa em pontos de história
1	Consulta inicial por número de cartão	2,5
2	Consulta inicial por dados do titular	3,7
3	Visualizar propostas existentes	1,8
4	Pré-análise de cliente pessoa física	2,6
5	Preenchimento de nova proposta	2,4
6	Checklist de proposta	2,6
7	Formalização de proposta	4,0
8	Acompanhamento de proposta	3,4
9	Fechamento de proposta	2,8
10	Cancelamento de proposta	1,6
11	Fluxo de cliente VIP	3,0
12	Impressão do termo	2,4
13	Relatório analítico	2,7
14	Relatório sintético	3,2
Total		38,7

Tabela 10 – Exemplo de Backlog do Produto
Fonte: Elaboração Própria

No planejamento da *sprint*, os itens do *backlog* do produto que agregam mais valor ao negócio são priorizados e refinados em tarefas no *backlog* da *sprint*. As tarefas são, então, estimadas em horas pela equipe de desenvolvimento.

Para a geração das métricas de EVM é necessário calcular o custo total planejado para a *sprint*. O valor planejado para cada tarefa é calculado a partir da quantidade de horas de cada atividade multiplicado pela média do custo por hora dos profissionais que compõe a equipe. O custo médio por hora é exemplificado na tabela 11 a seguir.

Recursos	Custo por hora	
Programador 1	R\$	37,80
Programador 2	R\$	40,50
Programador 3	R\$	39,15
Programador 4	R\$	45,90
Programador 5	R\$	44,55
Analista Programador 1	R\$	56,70
Analista Programador 2	R\$	55,35
Analista de Testes 1	R\$	51,30
Analista de Testes 2	R\$	48,60
Analista Funcional 1	R\$	66,15
Analista de Sistemas 1	R\$	71,55
Analista de Sistemas 2	R\$	79,65
Custo médio por hora	R\$	53,10

Tabela 11 – Exemplo de cálculo de custo médio por hora da equipe
Fonte: Elaboração Própria

A soma dos valores planejados para cada tarefa fornece o valor planejado para a *sprint* conforme abordagem proposta por Christoph (2011). A tabela 12 a seguir apresenta o *backlog* de uma *sprint* com cada tarefa estimada em horas, e o respectivo valor planejado, calculado a partir do custo médio da equipe por hora.

Req. N°	Tarefa	Estimativa em horas	Valor Planejado (PV)	Completo (Sim/Não)
1	Codificar a interface de usuário	104,0	R\$ 5.519,75	Não
1	Codificar a camada de acesso a dados	151,2	R\$ 8.028,72	Não
1	Testar a camada de acesso a dados	132,3	R\$ 7.025,13	Não
1	Testar a interface de usuário	85,1	R\$ 4.516,16	Não
2	Codificar a interface de usuário	113,4	R\$ 6.021,54	Não
2	Codificar a camada de acesso a dados	47,3	R\$ 2.508,98	Não
2	Codificar a camada de serviços	56,7	R\$ 3.010,77	Não
2	Implementar o tratamento de erros	122,9	R\$ 6.523,34	Não
2	Testar a camada de serviços	113,4	R\$ 6.021,54	Não
2	Testar a camada de acesso a dados	141,8	R\$ 7.526,93	Não
2	Testar a interface de usuário	104,0	R\$ 5.519,75	Não
3	Codificar a interface de usuário	132,3	R\$ 7.025,13	Não
3	Implementar o tratamento de erros	94,5	R\$ 5.017,95	Não
3	Testar a interface de usuário	113,4	R\$ 6.021,54	Não
		1512	R\$ 80.287,20	

Tabela 12 – Exemplo de Backlog de Sprint com PV
Fonte: Elaboração Própria

Durante a execução da *sprint*, à medida que cada tarefa é 100% concluída, ela agrega o valor correspondente ao seu valor planejado conforme proposta de Alleman; Henderson e Seggelke (2003). Os dados são obtidos na reunião diária SCRUM, onde os membros da equipe atualizam o andamento das tarefas, e apontam quais tarefas do *backlog* da *sprint* foram concluídas. O custo real é obtido através de um sistema diário de lançamento de horas trabalhadas, onde cada membro da equipe aponta quantas horas trabalhou em cada atividade do *backlog* da *sprint*. Na tabela 13 é possível observar as tarefas, bem como seus respectivos valores agregados e custos reais.

Req. N°	Tarefa	Estimativa em horas	Valor Planejado (PV)	Completo (Sim/Não)	Valor Agregado (EV)	Custo Real (AC)
1	Codificar a interface de usuário	104,0	R\$ 5.519,75	Sim	R\$ 5.519,75	R\$ 5.310,00
1	Codificar a camada de acesso a dados	151,2	R\$ 8.028,72	Sim	R\$ 8.028,72	R\$ 7.965,00
1	Testar a camada de acesso a dados	132,3	R\$ 7.025,13	Não	R\$ -	R\$ 3.982,50
1	Testar a interface de usuário	85,1	R\$ 4.516,16	Sim	R\$ 4.516,16	R\$ 4.779,00
2	Codificar a interface de usuário	113,4	R\$ 6.021,54	Sim	R\$ 6.021,54	R\$ 6.265,80
2	Codificar a camada de acesso a dados	47,3	R\$ 2.508,98	Sim	R\$ 2.508,98	R\$ 2.124,00
2	Codificar a camada de serviços	56,7	R\$ 3.010,77	Sim	R\$ 3.010,77	R\$ 3.186,00
2	Implementar o tratamento de erros	122,9	R\$ 6.523,34	Sim	R\$ 6.523,34	R\$ 6.372,00
2	Testar a camada de serviços	113,4	R\$ 6.021,54	Sim	R\$ 6.021,54	R\$ 5.841,00
2	Testar a camada de acesso a dados	141,8	R\$ 7.526,93	Sim	R\$ 7.526,93	R\$ 7.965,00
2	Testar a interface de usuário	104,0	R\$ 5.519,75	Não	R\$ -	R\$ 4.779,00
3	Codificar a interface de usuário	132,3	R\$ 7.025,13	Sim	R\$ 7.025,13	R\$ 6.903,00
3	Implementar o tratamento de erros	94,5	R\$ 5.017,95	Sim	R\$ 5.017,95	R\$ 6.372,00
3	Testar a interface de usuário	113,4	R\$ 6.021,54	Sim	R\$ 6.021,54	R\$ 5.841,00
		1512	R\$ 80.287,20		R\$ 67.742,33	R\$ 77.685,30

Tabela 13 – Exemplo de Backlog de Sprint com EV e AC
Fonte: Elaboração Própria

Desta forma, ficam disponíveis todas as métricas de EVM, possibilitando a obtenção dos índices de desempenho de custos e prazos (CPI e SPI). A tabela 14 a seguir exemplifica o cálculo de CPI e SPI para cada *sprint*, após a conclusão da segunda *sprint*.

Sprint	Velocidade Planejada	Valor Planejado (PV)	Valor por ponto de história	Custo Real (AC)	Velocidade Real	Valor Agregado (EV)	CPI (EV/AC)	SPI (EV/PV)
Sprint 1	8	R\$ 80.287,20	R\$ 10.035,90	R\$ 77.685,30	6,75	R\$ 67.742,33	0,87	0,84
Sprint 2	7,6	R\$ 75.402,00	R\$ 9.921,32	R\$ 82.995,30	8,2	R\$ 81.354,79	0,98	1,08
Sprint 3	10,2							
Sprint 4	7							
Sprint 5	5,9							
Total	38,7	R\$ 155.689,20		R\$ 160.680,60		R\$ 149.097,11	0,93	0,96

Tabela 14 – Exemplo de CPI e SPI por Sprint
Fonte: Elaboração Própria

Após a obtenção das métricas de EVM e dos índices de desempenho de prazos e custos, é possível gerar um gráfico *burn-down* de *release* seguindo a proposta de Sulaiman; Barton e Blackburn (2006), que apresenta a quantidade de pontos de história efetivamente agregados em cada *sprint*, e incluindo no gráfico os valores obtidos de CPI e SPI para cada *sprint*.

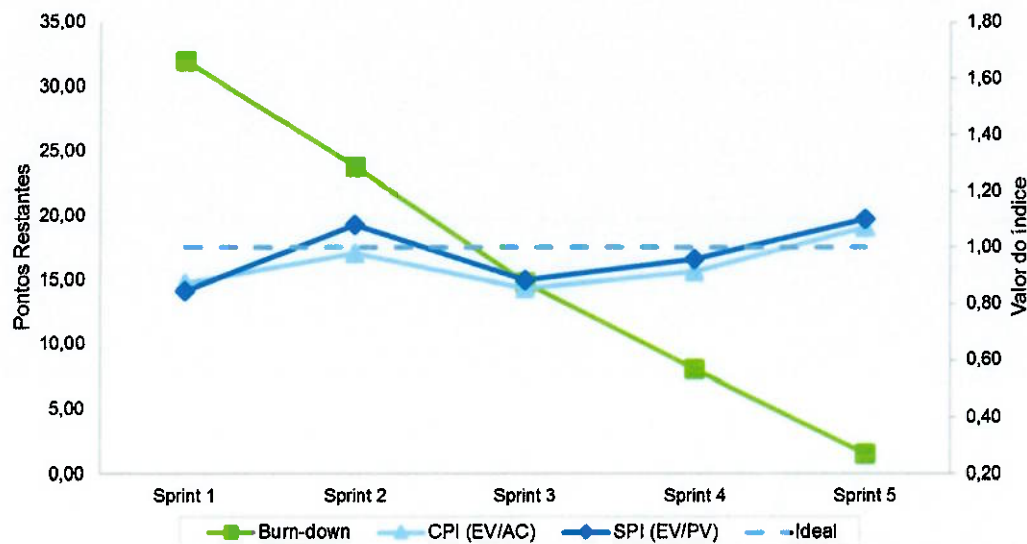


Figura 13 – Exemplo de Gráfico Burn-down com CPI e SPI
Fonte: Elaboração Própria

Semelhantemente à abordagem de Cabri e Griffiths (2006) propõe-se a inclusão de uma outra escala em um gráfico *burn-up* na perspectiva financeira e a incorporação do valor planejado, custo real, pontos de história planejados e pontos de história completos, como pode ser observado no gráfico (figura 14):

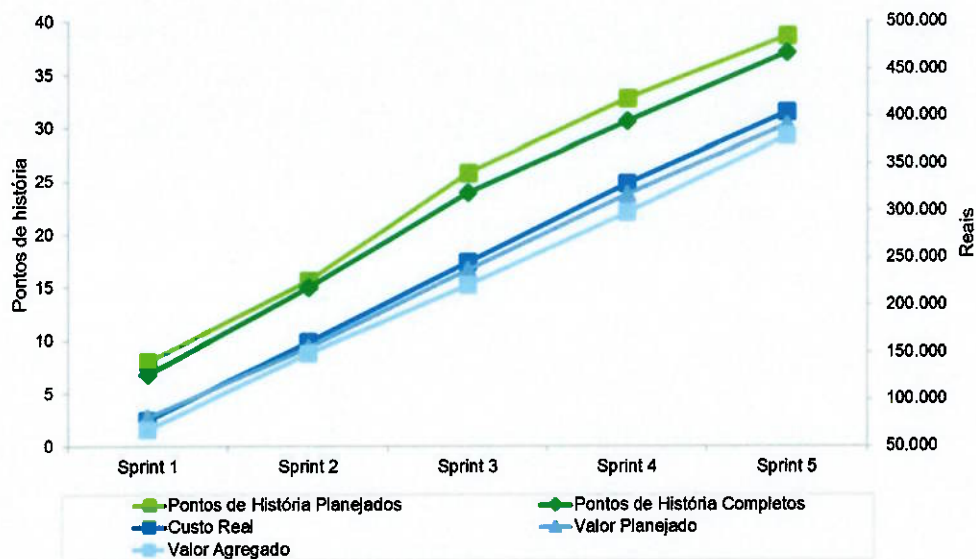


Figura 14 – Exemplo de Gráfico Burn-up com Escala Monetária
Fonte: Elaboração Própria

O controle e monitoramento podem ser feito diariamente, atualizando-se as tarefas completas, consequentemente o valor agregado, e o custo real através do apontamento diário das horas dispendidas em cada atividade. A visão diária do progresso e da saúde financeira do projeto pode auxiliar a equipe a tomar decisões e efetuar correções mais rapidamente, e assim, mitigar problemas maiores no futuro. Além disso, obtendo-se os índices de desempenho de prazos e custos é possível cumprir as exigências do escritório de projetos, a expectativa da gerência executiva e das partes interessadas quanto ao status do projeto.

3.7. Considerações do Capítulo

É apresentado neste capítulo um ambiente de desenvolvimento de *software* que apresenta dificuldades por utilizar o ciclo cascata, que poderia se beneficiar da utilização de um método ágil em seu processo de desenvolvimento. Porém este ambiente apresenta alguns requisitos de reporte de avanço físico e orçamentário de projetos através de EVM. Propõe-se então, a incorporação da técnica de valor agregado ao processo SCRUM de forma aplicável neste ambiente específico. Para tal, é realizada uma análise comparativa das abordagens de aplicação de EVM com métodos

ágeis estudadas, e SCRUM é analisado sob a perspectiva da incorporação de EVM ao processo. São analisados o contexto e as questões chave dos papéis, artefatos e eventos de SCRUM, são propostas soluções para as questões, unificando as características centrais das diferentes abordagens descritas. Posteriormente é apresentado um exemplo da aplicação da abordagem proposta.

4. CONSIDERAÇÕES FINAIS

4.1. Contribuições do Trabalho

Com a verificação e análise do comparativo realizado, do estudo do ambiente e seus requisitos, e a análise do processo SCRUM sob a perspectiva da incorporação da análise de valor agregado, pode-se observar que com pequenas alterações no processo SCRUM, é possível incorporar a análise de valor agregado como ferramenta de monitoramento de custos que o SCRUM não contempla.

O estudo permitiu observar que é possível utilizar um processo de desenvolvimento ágil, mais flexível, que se adapte a mudanças constantes de escopo e que permita a entrega frequente de funcionalidades completas aos clientes, sem deixar de cumprir os requisitos e premissas de reporte de status que a organização necessita. Além disso, o escritório de projetos pode comparar todos os projetos do portfólio em uma mesma base, independentemente do processo de desenvolvimento que tenha sido adotado.

A introdução do cálculo dos índices de desempenho de prazos e custos em uma base diária permite o acompanhamento e respostas imediatas a desvios do plano e a mitigação de problemas futuros. Proporcionar à equipe e às partes interessadas dados úteis e compreensíveis é essencial para o direcionamento da equipe SCRUM rumo a uma melhoria contínua dos processos.

Através de gráficos *burn-up* e *burn-down* com as métricas EVM incorporadas, é possível obter o status instantâneo do avanço do projeto e identificar no início problemas que necessitem de medidas corretivas, a principal vantagem da técnica de gerenciamento de valor agregado. Embora o SPI apresente informações semelhantes às existentes no gráfico *burn-down*, o CPI completa

todo o quadro com as informações sobre a eficiência de custos, que não está incluída em nenhuma outra medida no universo ágil.

Este trabalho mostrou ainda que a abordagem proposta de incorporação de análise de valor agregado ao processo SCRUM não exigiu trabalho administrativo adicional que pudesse prejudicar a agilidade do processo de desenvolvimento, e puderam ser aproveitadas práticas já instituídas no ambiente, como o apontamento de horas incorridas diariamente nas atividades do plano.

Foi possível observar que as diferentes abordagens, apesar de distintas, possuem características semelhantes, que, ao combinadas, geram uma abordagem robusta e que pode ser aplicada não somente ao ambiente descrito, mas também a outros ambientes semelhantes de desenvolvimento de *software*. Sendo assim, as considerações deste trabalho podem auxiliar organizações que realizam projetos de desenvolvimento de *software* e tenham interesse na utilização de métodos ágeis em conjunto com uma técnica de monitoramento e controle, tal como a análise de valor agregado.

4.2. Trabalhos Futuros

O estudo da utilização de EVM em conjunto com métodos ágeis não se esgota neste trabalho. Outros aspectos sobre o tema podem ser analisados e aprofundados. Entre as possibilidades, sugere-se:

- Aplicação prática da abordagem proposta neste trabalho no próprio ambiente descrito, a observação e análise dos resultados obtidos ao longo do tempo;
- Aplicação da abordagem em ambientes semelhantes, que poderiam se beneficiar do controle e monitoramento de custos de projetos ágeis de *software* providos pela abordagem sugerida;

- Incorporação de EVM a outros métodos ágeis, tais como XP, Desenvolvimento Dirigido à Funcionalidades, Desenvolvimento de Software Enxuto, entre outros;
- Análise da necessidade de adequação da técnica de gerenciamento de valor agregado dependendo do método ágil utilizado no processo de desenvolvimento.

REFERÊNCIAS

ALLEMAN, G.; HENDERSON, M.; SEGELKE, R. **Making Agile Development Work in a Government Contracting Environment: Measuring velocity with Earned Value**. IEEE Proceedings of the Agile Development Conference, 2003. p. 114-119.

BECK, K.; ANDRES, C. **Extreme Programming Explained: Embrace Change**. 2nd ed. Boston: Addison-Wesley, 2004.

BECK, K.; BEEDLE, M.; VAN BENNEKUM, A.; COCKBURN, A.; CUNNINGHAM, W.; FOWLER, M.; GRENING, J.; HIGHSMITH, J.; HUNT, A.; JEFFRIES, R.; KERN, J.; MARICK, B.; MARTIN, R.; MELLOR, S.; SCHWABER, K.; SUTHERLAND, J.; THOMAS, D. **Manifesto for Agile Software Development**. Disponível em: <<http://www.agilemanifesto.org>>. Acesso em: 14 dez. 2013.

BECK, K. **Extreme Programming Explained: Embrace Change**. 1st ed. Boston: Addison-Wesley, 1999.

CABRI, A.; GRIFFITHS, M. **Earned Value and Agile Reporting**. IEEE Proceedings of Agile 2006 Conference, 2006. p. 23-28.

CHRISTOPH, E. Earning Value the Agile Way: Using Story Points to Generate EV Metrics. **The Measurable News - PMI Earned Value Management Community of Practice**. n. 2, 2011.

COCKBURN, A. **Crystal Clear: A Human-Powered Methodology for Small Teams**. Addison-Wesley, 2004.

COHN, M. **Agile Estimating and Planning**. 1st ed. New Jersey: Pearson Education, 2005.

COHN, M. **Succeeding with Agile: Software Development Using Scrum**. 1st ed. Boston: Addison-Wesley, 2012.

DYBÅ, T.; DINGSØYR, T. Empirical Studies of Agile Software Development: A Systematic Review. **Information and Software Technology**, v. 50, n. 9-10, 833-859, 2008.

FLEMING, Q.; KOPPELMAN, J. **Earned Value Project Management**. 2nd ed., Newton Square: Project Management Institute, 2000.

GHOSH, S. **Systemic Comparison of the Application of EVM in Traditional and Agile Software Project**. University of Maryland, 2011. Disponível em: <<http://pm.umd.edu/files/public/documents/student-papers/2011/EVM%20in%20Waterfall%20and%20Agile%20Software%20Project%20by%20Sam%20Ghosh.pdf>>. Acesso em: 12 jul. 2014.

GRIFFITHS, M. **PMI-ACP Exam Prep: A Course in a Book for Passing the PMI Agile Certified Practitioner Exam**. RMC Publications, 2012.

HUNDHAUSEN, R. **Professional Scrum Development with Microsoft Visual Studio 2012**. 1st ed. Redmond: Microsoft Press, 2012.

LACERDA, I. M. **Fatores Críticos de Sucesso no uso de Earned Value Management em Projetos de Desenvolvimento de Software e a Relação com a Qualidade Percebida**. 207 p. Dissertação (Mestrado) - Universidade Federal Fluminense, Niterói, 2009.

MAHNIC, V.; ZABKAR, N. Measuring Progress of Scrum-based Software Projects. **Elektronika Ir Elektrotechnika Journal**. v. 18, n. 8, 2012.

PMI - Project Management Institute. **PMBOK® Guide: A Guide to the Project Management Body of Knowledge**. 5th ed., Newton Square: PMI Publications, 2013.

PMI - Project Management Institute. **Practice Standard for Earned Value Management**. Newton Square: Project Management Institute, 2005.

PRESSMAN, R. S. **Engenharia de Software: Uma Abordagem Profissional**. 7. ed. São Paulo: McGraw Hill, 2011.

QUINTELLA, H.; LACERDA, I. Earned Value Management em Projetos Ágeis de Software: Abordagens De Aplicação. **Relatórios De Pesquisa Em Engenharia De Produção**. v. 11, n. 9, 2011.

RUBIN, K. **Essential Scrum: A Practical Guide to the Most Popular Agile Process**. 1st ed. Michigan: Addison-Wesley, 2012.

RUSK, J. Earned Value for Agile Development. **DoD Software Tech News**. v. 12, n. 3, 2009.

SALGADO, A.; MELCOP, T.; ACCHAR, J.; REGO, P. A.; FERREIRA, A. I. F.; KATSURAYAMA, A. E.; MONTONI, M.; ZANETTI, D. **Aplicação de um Processo Ágil para Implantação de Processos de Software baseado em Scrum na Chemtech**. In: SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE, 9., 2010, Belém. Anais... Belém: SBQS, 2010. p. 351-258.

SCHWABER, K. **Agile Project Management with Scrum**. 1st ed. Redmond: Microsoft Press, 2004.

SCHWABER, K.; SUTHERLAND, J. **The Scrum Guide: The Definitive Guide to Scrum**. 2013. 16 p. Disponível em: <<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide.pdf>>. Acesso em: 22 fev. 2014.

SHORE, J.; WARDEN, S. **The Art of Agile Development**. 1st ed. Sebastopol: O'Reilly Media, 2007.

SMITH, G.; SIDKY, A. **Becoming Agile: In An Imperfect World**. 1st ed. Greenwich: Manning Publications, 2009.

SOARES, M. S. Metodologias Ágeis Extreme Programming e Scrum para o Desenvolvimento de Software. **Revista Eletrônica de Sistemas de Informação**, v. 3, n. 1, 2004.

SOMMERVILLE, I. **Software Engineering**. 9th ed. Boston: Addison-Wesley, 2011.

SULAIMAN, T.; BARTON, B.; BLACKBURN, T. **AgileEVM - Earned Value Management in Scrum Projects**. IEEE Proceedings of Agile 2006 Conference, 2006. p. 7-16.

SULAIMAN, T.; SMITS, H. Measuring Integrated Progress on Agile Software Development Projects. **Methods & Tools Software Development Magazine**, v. 15, n. 3, p. 2-9, 2007. Disponível em: <<http://www.methodsandtools.com/PDF/mt200703.pdf>>. Acesso em: 22 jun. 2014.

TELES, V. M. **Extreme Programming: Aprenda Como Encantar Seus Usuários Desenvolvendo Software Com Agilidade e Alta Qualidade**. 1. ed. São Paulo: Novatec, 2004.

VARGAS, R. V. **Análise de Valor Agregado: Revolucionando o Gerenciamento de Prazos e Custos**. 6. ed. Rio de Janeiro: Brasport, 2013.