

# Uma análise do método ágil Scrum conforme abordagem nas áreas de processo Gerenciamento e Desenvolvimento de Requisitos do CMMI

Alexandre Lazaretti Zanatta<sup>1</sup>, Patrícia Vilain<sup>2</sup>

Universidade de Passo Fundo - UPF  
ICEG - Ciência da Computação  
Caixa Postal 611 - CEP 99001-970  
Passo Fundo - RS - Brasil  
zanatta@upf.br

Universidade Federal de Santa Catarina - UFSC  
CTC-INE - Campus Universitário  
Caixa Postal 476 - CEP 88040-900  
Florianópolis - SC - Brasil  
vilain@inf.ufsc.br

**Abstract:** *In this article we analyze the agile method Scrum in relation to the process areas Requirements Management and Requirements Development of the CMMI model. The results obtained from this analysis indicate that Scrum does not meet all the required practices present in such process areas. We point out what is missing in Scrum to fully comply with these areas and propose solutions for it.*

**Resumo:** *Este artigo tem como objetivo realizar uma análise do método ágil Scrum em relação às áreas de processo Gerenciamento de Requisitos e Desenvolvimento de Requisitos do modelo CMMI. Os resultados da análise realizada apresentam dados que apontam que o Scrum não atende totalmente as exigências presentes nestas áreas do modelo CMMI. As exigências não atendidas pelo Scrum são destacadas e algumas soluções são propostas.*

## 1. Introdução

O recente surgimento dos métodos ágeis no cenário do desenvolvimento de software pode requerer que a atual área da engenharia de requisitos repense alguns de seus procedimentos. Isto se deve principalmente porque estes métodos abdicam, em parte, de controles e documentação tão presentes nesta área. Os métodos ágeis em geral não mencionam como realizam, por exemplo, a documentação da especificação de requisitos, ou como mantêm a rastreabilidade dos requisitos. Os princípios apresentados pelo manifesto ágil e discutidos por Fowler (2003, p.1) mostram que certos valores relacionados com a área de engenharia de requisitos continuam sendo importantes, como o entendimento dos requisitos, porém preocupam-se em não gerar muita documentação que, justificam, provavelmente nunca será lida.

As discussões em torno da compatibilidade dos métodos ágeis com modelos de qualidade de software, que possuem documentação em todos os processos, principalmente na área de requisitos, têm aumentado significativamente. Turner e Jain (2002, p.161) comentam que, apesar da existência de características distintas entre os métodos ágeis e o modelo CMMI, ambos possuem planos específicos para o desenvolvimento de software e buscam o melhor para que a organização produza software com qualidade.

---

<sup>1</sup> Professor da Universidade de Passo Fundo (UPF) e doutorando na Universidade Pontifícia de Salamanca

<sup>2</sup> Professora da Universidade Federal de Santa Catarina. (UFSC)

Outros autores, como Paulk (2001), Boehm (2002) e Paetsch (2003), em seus respectivos trabalhos, apontam também para uma discussão recente: Desenvolvimento de software ágil é compatível com o modelo CMMI? Neste contexto, apresentam diferenças e semelhanças nas duas abordagens, acreditando que a área da engenharia de software está passando por mais uma nova fase denominada Desenvolvimento tradicional de software *versus* Desenvolvimento de software ágil. Na verdade, atualmente não existe consenso se os métodos ágeis são compatíveis com modelos de qualidade de software, como, por exemplo, o CMMI. Este tema ainda é considerado bastante polêmico.

Esse trabalho tem como principal objetivo, auxiliar as organizações que trabalham com o método ágil Scrum e, que desejam, por algum motivo, estar de acordo com as áreas de processo Gerenciamento de Requisitos e Desenvolvimento de Requisitos do modelo CMMI. Para isto, o Scrum é analisado e os pontos onde ele não atende às exigências destas áreas de processo do CMMI são identificados.

Esse artigo está organizado da seguinte maneira. Inicialmente são apresentados o modelo CMMI e o método ágil Scrum. Na sequência, apresenta-se uma análise dos dados colhidos. E, por fim, são apresentadas as conclusões e soluções para os problemas encontrados.

## 2. CMMI (*Capability Maturity Model Integration*)<sup>3</sup>

Segundo Chrissis *et al.* (2003, p. 18) “o propósito do CMMI é estabelecer um guia para melhorar o processo da organização e sua capacidade para gerenciar o desenvolvimento, aquisição e manutenção de produtos e serviços”. Ahern *et al.* (2004, p.46) destacam que um dos principais objetivos do modelo CMMI é assegurar a compatibilidade com a norma ISO/IEC 15504 permitindo a análise de áreas independentes do nível de maturidade. O modelo CMMI possui 4 (quatro) disciplinas: Engenharia de Sistemas, Engenharia de Software, Produto Integrado e Desenvolvimento de Processo e, finalmente, a Aquisição. Estas disciplinas, também conhecidas como áreas do conhecimento, auxiliam no planejamento da melhoria do processo de toda organização.

A implementação de uma ou mais destas disciplinas ao mesmo tempo com uma única terminologia e infra-estrutura de treinamento e avaliação é considerada uma grande vantagem do modelo CMMI, pois a organização determina em quais disciplinas deseja melhorar seu processo. Estas disciplinas são compostas por áreas de processo que, quando executadas, determinam a melhoria do processo na disciplina escolhida. Segundo (Fiorini, 1998, p.29), área de processo é um conjunto de práticas relacionadas em uma área que quando executadas coletivamente satisfazem um conjunto de objetivos importantes para a melhoria significativa daquela área.

O modelo CMMI, segundo o SEI (*Software Engineering Institute*), Chrissis *et al.* (2003, p. 18) e Ahern *et al.* (2004, p.46), é dividido em duas representações: Estágio e Contínuo. A representação Estágio possui 5 níveis de maturidade (Inicial, Gerenciado, Definido, Gerenciado Quantitativamente e De Otimização). Cada nível (estágio) possui diversas áreas de processos onde cada uma se encontra em um único nível. A representação Contínuo tem 6 níveis para dimensão da capacitação (Incompleto, Executado, Gerenciado, Definido, Gerenciado Quantitativamente, De Otimização). Diferentemente na representação Estágio, as áreas de processo na representação Contínuo são independentes dos níveis de maturidade, ficando relacionadas apenas com a capacidade<sup>4</sup> do processo, ou seja, uma determinada área de processo em particular poderá ter sua capacidade avaliada independente das outras áreas de processo. O modelo CMMI possui vinte e cinco áreas de processo. As áreas de processo do modelo CMMI possuem práticas específicas e genéricas. A prática específica (SP - *Specific Practice*) é uma descrição detalhada das atividades

---

<sup>3</sup> CMM and Capability Maturity Model são marcas registradas no *U.S. Patent and Trademark Office*. CMM Integration, CMMI, SCAMPI, são marcas de serviço da Carnegie Mellon University

<sup>4</sup> Como o termo “capabilidade” não existe no vocabulário português, deverá ser entendido como capacidade.

que são consideradas fundamentais para alcançar os objetivos específicos. As práticas genéricas se posicionam ao final de cada área de processo e são chamadas de genéricas pois estão relacionadas com várias áreas de processo. Uma prática genérica é a descrição de uma atividade fundamental para alcançar os objetivos genéricos.

Esse trabalho dedicará atenção especial somente em duas áreas de processo: Gerenciamento de Requisitos (REQM *Requirement Management*) e Desenvolvimento de Requisitos (RD *Requirement Development*) da representação Contínuo. A área de processo REQM tem como objetivo específico controlar todas as mudanças ocorridas nos requisitos, e a área de processo RD é identificar as necessidades do cliente e as traduzir em requisitos do produto. E, dentro destas áreas, serão analisadas as suas práticas específicas.

### 3. O método ágil Scrum

O método ágil Scrum tem como objetivo, segundo Schwaber (2002, p.2), definir um processo para projeto e desenvolvimento de software orientado a objeto, que seja focado nas pessoas e que seja indicado para ambientes em que os requisitos surgem e mudam rapidamente. O Scrum também é considerado um método específico para o gerenciamento do processo de desenvolvimento de software.

O método baseia-se ainda, conforme Schwaber (2002, p.7), em princípios como: equipes pequenas de, no máximo, 7 pessoas; requisitos que são pouco estáveis ou desconhecidos; e iterações curtas. Divide o desenvolvimento em intervalos de tempos de, no máximo 30 dias, também chamadas de *Sprints*. Este método não requer ou fornece qualquer técnica ou método específico para a fase de desenvolvimento de software, apenas estabelece conjuntos de regras e práticas gerenciais que devem ser adotadas para o sucesso de um projeto. As práticas gerenciais do Scrum são: *Product Backlog*, *Daily Scrum*, *Sprint*, *Sprint Planning Meeting*, *Sprint Backlog* e *Sprint Review Meeting*. A seguir uma breve descrição do *Product Backlog* e do *Sprint*, que são as práticas relacionadas à área de requisitos no método ágil Scrum.

O *Product Backlog* é o ponto inicial do Scrum, sendo considerada a prática responsável pela coleta dos requisitos, conforme aponta Schwaber (2002, p.33). Nesta prática, através de reuniões com todos *stakeholders* no projeto, são apontados os itens com todas as necessidades do negócio e os requisitos técnicos a serem desenvolvidos, ou seja, o *Product Backlog* é uma lista de atividades que provavelmente serão desenvolvidas durante o projeto. A *Sprint* é considerada a principal prática do Scrum, onde são implementados os itens de trabalho definidos no *Product Backlog* pela equipe Scrum, que pode durar de uma a quatro semanas. Conforme Abrahamsson (2002, p.30), o *Sprint* inclui as fases tradicionais do desenvolvimento de software: requisitos, análise, projeto e entrega.

### 4. Análise Realizada

O método ágil Scrum foi avaliado segundo as perspectivas do modelo CMMI, somente nas áreas de processo REQM e RD pertencentes à categoria Engenharia deste mesmo modelo. Devido à significativa importância das práticas para entendimento do modelo CMMI e alcance das metas, cada produto de trabalho resultante das práticas específicas das áreas de processo REQM e RD do modelo CMMI foi analisado e verificou-se se o Scrum atende este produto de trabalho conforme o modelo CMMI. Para isso, elaborou-se uma escala ordenada de três categorias, com a determinação de classificação do atendimento ao Scrum. As três categorias são:

**NA: Não atendido** - Há pouca evidência de que a SP/CMMI esteja presente no Scrum

**PA: Parcialmente atendido** - Existem evidências de que a SP/ CMMI esteja presente no Scrum

**A: Atendido** - Há evidências significativas de que a SP/CMMI esteja presente no Scrum

Após a análise individual dos típicos produtos de trabalho de uma prática específica, pode-se concluir se esta prática específica encontra-se em uma das três categorias, **NA**, **PA** e **A**.

A seguir apresentam-se os resultados dos dados pesquisados e as interpretações das análises. Inicialmente, descreve-se a análise da área de processo REQM e, então, apresenta-se a análise da área de processo RD.

#### 4.1. Análise da área de processo Gerenciamento de Requisitos

A seguir, faz-se uma análise do Scrum conforme as práticas específicas da área de processo REQM da categoria Engenharia do modelo CMMI.

##### SP 1.1-1 Obter o entendimento dos requisitos

No Scrum, os requisitos são levantados diretamente com os clientes, e os critérios de seleção dos requisitos são alocados para cada *Sprint* para que sejam utilizados durante a divisão de tarefas. Os requisitos são entendidos pelos participantes do projeto e colocados numa lista de itens denominada de *Product Backlog*, que representa “o que” deverá ser analisado e desenvolvido pela equipe. Depois do entendimento dos requisitos, o *Product Backlog* será dividido em conjuntos de tarefas que serão designadas por itens e que apresentam descrições, tempo e a definição de quem a realizará. Cada conjunto de tarefas será analisado e desenvolvido em um *Sprint* e é chamado de *Sprint Backlog List*. A seguir, a Figura 1 apresenta os produtos de trabalho resultantes da prática específica REQM SP1.1-1 do modelo CMMI, com a determinação da escala de classificação do atendimento pelo Scrum.

Produtos de Trabalho Resultantes	NA	PA	A
1) Lista de critérios para distinguir fornecedores de requisitos	-	-	X
2) Critério para avaliação e aceitação dos requisitos	-	-	X
3) Análise dos resultados conforme lista de critérios estabelecidos	-	-	X
4) Elaboração de um conjunto de requisitos aceitos	-	-	X

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 1 Produtos de trabalho da REQM SP1.1-1

Na reunião inicial no Scrum, o *Sprint Backlog Planning* são apresentados os critérios para a elaboração do *Product Backlog*. Percebe-se, então, que os itens 1 e 2 da Figura 1 são Atendidos pelo Scrum. Como a elaboração de um conjunto de requisitos aceitos e a análise dos resultados, itens 3 e 4 da Figura 1, respectivamente, são justamente os resultados finais da análise do *Product Backlog*, conhecido como *Sprint Backlog*, todos os produtos de trabalho resultantes da REQM SP1.1-1 do modelo CMMI estão *Atendidos* no Scrum. Assim, também pode-se dizer que esta prática é considerada *Atendida*.

##### SP 1.2-2 Obter o compromisso com os requisitos

O *Scrum Master* e o *Product Owner* trabalham para que todos os participantes do projeto entrem num acordo comum sobre o entendimento dos requisitos que serão analisados para o projeto. Os produtos de trabalho resultantes desta SP são: a) avaliação dos impactos dos requisitos; b) compromissos documentados dos requisitos e suas mudanças. A avaliação dos prováveis impactos que os requisitos terão sobre o projeto é realizada pelo *Product Owner*, e as documentações sobre o impacto e as mudanças são registradas entre as atividades realizadas durante a definição do *Product Backlog* e o *Sprint Backlog*. Com isso, tanto a avaliação dos impactos e o compromisso na documentação são atendidos na prática específica REQM SP1.2-2 do modelo CMMI no Scrum, e, portanto, considera-se essa prática *Atendida*.

##### SP 1.3-1 Gerenciar as mudanças dos requisitos

Todo processo de Gerenciamento de Requisitos no Scrum é controlado durante o *Sprint* que, ao final do desenvolvimento, no *Sprint Review Meeting*, resulta em um incremento que é analisado pelos *stakeholders* no projeto, desde clientes até fornecedores. Caso novos requisitos apareçam, ou alguma mudança é solicitada, estes são acrescentados ao *Product Backlog*, que reiniciará o processo Scrum. Vale ressaltar que um projeto Scrum encerra quando o desenvolvimento de todos os itens da

lista das funcionalidades que estão presentes do *Product Backlog* são finalizados. A seguir, a Figura 2 apresenta os produtos de trabalho resultantes da prática específica REQM SP 1.3-1 do modelo CMMI, com a determinação da escala de classificação do atendimento pelo Scrum.

Produtos de Trabalho Resultantes	NA	PA	A
1) Status dos requisitos	-	-	X
2) Banco de Dados dos requisitos	X	-	-
3) Banco de Dados para Tomada de decisão dos requisitos	X	-	-

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 2 Produtos de trabalho da REQM SP1.3-1

Durante o *Sprint*, mantêm-se, através da *Daily Scrum*, um controle (*status*) de qual requisito está sendo desenvolvido, o que satisfaz o item 1 desta prática específica. O Scrum não determina explicitamente que utiliza um banco de dados dos requisitos, ou outra forma de armazenamento destes requisitos, muito menos que o utiliza para auxílio nas tomadas de decisões. Portanto, os itens 2 e 3 da prática específica REQM SP 1.3-1 do modelo CMMI não estão atendidos no Scrum. Como não existe, neste caso, um consenso sobre o *Não Atendimento* ou o *Atendimento* de todos os produtos de trabalho desta prática, então ela será considerada como **Parcialmente Atendida**.

#### SP 1.4-2 Manter a rastreabilidade bidirecional dos requisitos

Para o Scrum a rastreabilidade não é importante, pois o que interessa é o resultado do produto final e não a técnica envolvida no desenvolvimento do produto. Os integrantes da equipe, em suas reuniões diárias, chegam a um consenso se devem seguir ou não o desenvolvimento dos requisitos. O Scrum não relaciona os requisitos aos *stakeholders*, nem menciona a relação dos requisitos entre si ou os requisitos aos modelos resultantes do projeto. Os produtos de trabalho resultantes desta SP são: a) matriz de rastreabilidade; b) um sistema de acompanhamento de requisitos.

O Scrum não possui uma matriz de rastreabilidade e não implementa um sistema de acompanhamento de requisitos, apesar de ser indicado para ambientes em que os requisitos são poucos estáveis ou em muitas vezes desconhecidos. Então, como os produtos de trabalho da prática específica REQM SP 1.4-2 do modelo CMMI não são atendidos, esta prática é considerada **Não Atendida** no método Scrum.

#### SP 1.5-1 Identificar inconsistências entre Projeto de Trabalho e Req.

Como o método Scrum baseia-se em princípios onde os requisitos são poucos estáveis e o projeto não está totalmente definido, as prováveis inconsistências são verificadas pela equipe Scrum somente durante o *Sprint*, através das reuniões diárias e do *Sprint Review*. O projeto somente chegará ao final quando todas as funcionalidades descritas no *Product Backlog* forem realizadas. As ações corretivas são realizadas no *Sprint Review* em consonância com o Scrum Master e o Cliente. Desta forma, o Scrum minimiza a existência de inconsistências entre o Projeto de Trabalho e os requisitos apresentados. Os produtos de trabalho resultantes desta SP são: a) documentação das inconsistências do projeto; b) ações corretivas.

O Scrum não determina explicitamente a geração de documentação das inconsistências do projeto. Por outro lado, as ações corretivas, executadas após a entrega do incremento que é o resultado final do *Sprint*, também conhecido como *Product Increment*. Como não existe, neste caso, um consenso sobre o *Não Atendimento* ou o *Atendimento* de todos os produtos de trabalho desta prática, então ela será considerada como **Parcialmente Atendida**.

Finalizando a análise da área de processo Gerenciamento de Requisitos, a Figura 3 ilustra a classificação do atendimento do Scrum em relação as práticas específicas da área de processo REQM do modelo CMMI.



Figura 3 Escala de classificação do atendimento do Scrum em relação às práticas específicas da área de processo REQM do CMMI.

Da análise realizada, apenas 40% das práticas específicas encontram-se na categoria *Atendida*, 40% na categoria de *Parcialmente Atendida* e 20% na categoria *Não Atendida*. Percebe-se que o Scrum não está em total conformidade com as exigências das práticas específicas da área de processo REQM do modelo CMMI.

#### 4.2. Análise da área de processo Desenvolvimento de Requisitos

A seguir, será apresentada uma análise do Scrum conforme as práticas específicas da área de processo RD da categoria Engenharia do modelo CMMI.

##### SP 1.1-1 Coletar as necessidades dos *stakeholders*

No *Product Backlog* são colocadas todas as necessidades dos *stakeholders*, sendo o *Product Owner* o responsável pela criação e execução da lista no *Sprint Backlog*. Assim, como todas as necessidades dos *stakeholders* no projeto são coletadas e identificadas, a prática específica RD SP 1.1-1 do modelo CMMI está *Atendida* pelo Scrum. Cabe ressaltar que esta prática específica não define produtos de trabalhos e é aplicada somente na Representação Contínuo do modelo CMMI.

##### SP 1.1-2 Eliciar as necessidades

O Scrum não define técnicas para elicitar os requisitos, como casos de uso, protótipos, JAD (*Joint Application Development*), etc. Portanto, o único produto de trabalho (*utilizar métodos para eliciação das necessidades, expectativas, restrições e interfaces externas*) resultante da prática específica RD SP 1.1-2 do modelo CMMI não é atendido no Scrum, esta prática é considerada *Não Atendida* no método Scrum.

##### SP 1.2-1 Desenvolver os Requisitos dos clientes

O resultado do *Sprint Planning Meeting*, conhecido como *Sprint Backlog*, define e identifica o que será desenvolvido durante os 30 dias do Sprint. A Figura 4 apresenta os produtos de trabalho resultantes da prática específica RD SP 1.2-1 do modelo CMMI, com a determinação da escala de classificação do atendimento pelo Scrum.

Produtos de Trabalho Resultantes	NA	PA	A
1) Requisitos dos clientes	-	-	X
2) Restrições dos clientes na condução da verificação	-	-	X
3) Restrições dos clientes na condução da validação	-	-	X

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 4 Produtos de trabalho da RD SP1.2-1

Todos os requisitos do cliente, item 1 da Figura 4, são acompanhados durante o desenvolvimento pelo *Product Owner*, pois é sua função determinar se os requisitos estão sendo desenvolvidos de acordo com o solicitado pelo cliente, comparando-os ainda com a lista das funcionalidades presente no *Product Backlog*. Restrições dos clientes na condução de verificação e

validação, conforme itens 2 e 3 da Figura 4, são acompanhadas pelo *Product Owner* e o *Scrum Master*, respectivamente. Desta forma, como os produtos de trabalho da prática específica RD SP 1.2-1 do modelo CMMI são atendidos, esta prática é considerada *Atendida* no Scrum.

#### SP 2.1-1 Estabelecer os requisitos dos produtos e seus componentes

O estabelecimento dos requisitos dos produtos e dos componentes é realizado no Scrum através do *Product Backlog*, onde são detalhadas todas as funcionalidades, características, infraestrutura, arquitetura e tecnologia que o produto deverá possuir. A seguir, a Figura 5 apresenta os produtos de trabalho resultantes da prática específica RD SP 2.1-1 do modelo CMMI, com a determinação da escala de classificação do atendimento pelo Scrum.

Produtos de Trabalho Resultantes	NA	PA	A
1) Requisitos derivados (custos, performance, etc.)	-	-	X
2) Requisitos dos produtos	-	-	X
3) Requisitos dos componentes dos produtos	-	-	X

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 5 Produtos de trabalho da RD SP2.1-1

Schwaber (2002, p. 63), aponta a existência da definição de custos e orçamentos no projeto que será desenvolvido no Scrum, indicando que o produto de trabalho do item 1 também é atendido. Assim, os produtos de trabalho da prática específica RD SP 2.1-1 do modelo CMMI são atendidos e a prática é considerada *Atendida* no Scrum.

#### SP 2.2-1 Alocar os requisitos dos componentes dos produtos

Após a definição da lista das funcionalidades do produto que estão no *Sprint Backlog List* e que serão desenvolvidas durante o *Sprint*, a equipe Scrum, o *Scrum Master* e o *Product Owner* atuam na transformação das necessidades dos *stakeholders* em requisitos de produto, decidindo como alocar ou distribuir os requisitos para os componentes de produto. A seguir, a Figura 6 apresenta os produtos de trabalho resultantes da prática específica RD SP 2.2-1 do modelo CMMI, com a determinação da escala de classificação do atendimento pelo Scrum.

Produtos de Trabalho Resultantes	NA	PA	A
1) Planilhas com alocação dos requisitos	X	-	-
2) Alocação dos requisitos temporários	X	-	-
3) Projeto das restrições	X	-	-
4) Requisitos derivados	X	-	-
5) Relacionamentos entre os requisitos derivados	X	-	-

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 6 Produtos de trabalho da RD SP2.2-1

A alocação dos requisitos aos componentes de produto não é um processo detalhado durante o *Sprint*, pois o Scrum não define técnicas para esta fase. Portanto, todos os itens da Figura 6 podem estar presentes, porém não estão especificados no Scrum. Diante disto, a prática específica RD SP 2.2-1 do modelo CMMI é considerada *Não Atendida* no Scrum.

#### SP 2.3-1 Identificar os requisitos de interfaces

No Scrum, os requisitos de interface com outros sistemas não são detalhados, seja durante a elaboração do *Product Backlog* ou na execução do *Sprint*, descrevendo superficialmente a ligação com outros sistemas. Desta forma, o único produto de trabalho (*elaboração dos requisitos de*

interface) da prática específica RD SP 2.3-1, do modelo CMMI, não é atendido no Scrum e esta prática é considerada *Não Atendida*.

### SP 3.1-1 Estabelecer conceitos operacionais e cenários

Com a criação do *Product Backlog*, que contém todas as necessidades do negócio e os requisitos (funcionais e não funcionais) que servirão de base para o desenvolvimento do software, espera-se descrever toda a seqüência de eventos que poderão ocorrer com o uso deste sistema, gerando, de certa forma um cenário para o desenvolvimento do software. A Figura 7 apresenta os produtos de trabalho resultantes da prática específica RD SP 3.1-1 do modelo CMMI.

Produtos de Trabalho Resultantes	NA	PA	A
1) Conceitos operacionais	-	-	X
2) Instalação de produto, operação, manutenção e conceitos de suporte	-	-	X
3) Disposição dos conceitos utilizados	-	X	-
4) Construção de Cenários	-	X	-
5) Casos de Uso	X	-	-
6) Novos requisitos	-	-	X

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 7 Produtos de trabalho da RD SP3.1-1

Com o *Product Backlog*, os itens 1 e 2 da Figura 7, estão Atendidos pelo Scrum, e o *Scrum Master* coloca a disposição dos *stakeholders* todos os conceitos utilizados durante o projeto, e, assim, o item 3 da Figura 7 é considerado parcialmente atendido pelo Scrum. Como este método não detalha a construção de cenários, mas descreve a seqüência de eventos deste sistema, o item 4 é parcialmente atendido pelo Scrum. Pelo fato do Scrum não definir técnicas para eliciar os requisitos, como casos de uso, o item 5 da Figura 7 não é atendido pelo Scrum. Por outro lado, novos requisitos, podem ser inseridos após a *Sprint Review Meeting*, pois o *Scrum Master* reporta os resultados do trabalho com os participantes para avaliação, e caso surja um novo requisito é neste momento que ele é inserido, assim, o item 6 da Figura 7 é atendido no Scrum. Como não existe, neste caso, um consenso sobre que categoria deve ser enquadrada esta prática devido à ocorrência de várias categorias, optou-se por selecionar o típico produto de trabalho que tenha o objetivo mais próximo ao da prática específica em questão. Desta forma, como o típico produto de trabalho, representado pelo item 1 da Figura 7, foi definido como atendido, esta prática também será considerada *Atendida* pelo Scrum.

### SP 3.2-1 Estabelecer uma definição das funcionalidades requeridas

A criação de uma lista das funcionalidades, denominada *Product Backlog*, é o ponto inicial no Scrum. Além destas funcionalidades, nesta lista, são inseridas as características, padrões, tecnologias e estratégias do produto que será desenvolvido. Schwaber (2002, p.33) considera que a definição do *Product Backlog* e o *Sprint* são considerados as fases mais importantes do Scrum. A seguir, a Figura 8 apresenta os produtos de trabalho resultantes da prática específica RD SP 3.2-1 do modelo CMMI.



Produtos de Trabalho Resultantes	NA	PA	A
1) Arquitetura funcional	-	-	X
2) Diagrama de Atividades e casos de uso	X	-	-
3) Análise Orientada a Objeto com identificação de serviços	X	-	-

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 8 Produtos de trabalho da RD SP3.2-1

A arquitetura funcional, apontada no item 1 da Figura 8, é indicada na criação do *Product Backlog*. Como o Scrum não determina a técnica que será adotada para especificar a funcionalidade requerida pelo cliente, fica a critério do desenvolvedor a utilização de casos de uso e diagramas de atividades, bem como a identificação de serviços na análise orientada a objeto. Como não existe, neste caso, um consenso sobre que categoria em que deve ser enquadrada esta prática devido à ocorrência de várias categorias, optou-se por selecionar o típico produto de trabalho que tenha o objetivo mais próximo ao da prática específica em questão. Desta forma, como dois típicos produtos de trabalhos, representados pelo itens 2 e 3 da Figura 8, foram definidos como não atendidos, esta prática também será considerada *Não Atendida* no Scrum.

### SP 3.3-1 Analisar os requisitos

No Scrum, o processo de análise dos requisitos procura identificar se todas as funcionalidades do sistema, descritas no *Product Backlog List*, resultado da análise do *Product Backlog*, serão resolvidas com os requisitos descritos e se os mesmos se sobrepõem ou estão em conflito, não definindo explicitamente um documento de requisitos acordados entre os *stakeholders*. A seguir, a Figura 9 apresenta os produtos de trabalho resultantes da prática específica RD SP 3.3-1 do modelo CMMI, com a determinação da escala de classificação do atendimento pelo Scrum.

Produtos de Trabalho Resultantes	NA	PA	A
1) Relatórios dos defeitos dos requisitos	X	-	-
2) Propor mudanças nos requisitos para resolver defeitos	-	-	X
3) Apresentar os requisitos chaves	-	-	X
4) Elaborar medidas técnicas de performance	X	-	-

Fonte: primária Legenda: NA-Não Atendido PA-Parcialmente Atendido A-Atendido

Figura 9 Produtos de trabalho da RD SP3.3-1

Percebe-se que a análise dos requisitos está presente no projeto Scrum, porém, como não apresenta relatórios de defeitos, como sugere o item 1 da Figura 9, este produto de trabalho não é atendido pelo Scrum. Propor mudanças nos requisitos para resolução de defeitos, item 2 da Figura 9, é uma das principais características do Scrum, assim como dos outros métodos ágeis. Apresentar os requisitos chaves, item 3 da Figura 9, é atendido no Scrum porque neste método existe a preocupação no desenvolvimento dos requisitos de acordo com as necessidades dos *stakeholders*. Para que isto ocorra, é necessário elencar os requisitos chaves ou fundamentais para a execução do projeto. Os requisitos analisados desde a elaboração das funcionalidades do software até o seu desenvolvimento não detalham prováveis medidas técnicas de performance que poderão ser utilizadas no projeto. Como não existe, neste caso, um consenso sobre que categoria em que deve ser enquadrada esta prática devido à ocorrência de várias categorias, optou-se por selecionar o típico produto de trabalho que tenha o objetivo mais próximo ao da prática específica em questão. Desta forma, como o típico produto de trabalho, representado pelo item 2 da Figura 9, foi definido como atendido, esta prática também será considerada *Atendida* pelo Scrum.

### SP 3.4-3 Analisar os requisitos para avaliação

O Scrum não define modelos, simuladores ou protótipos para analisar o risco das necessidades dos *stakeholders* e suas restrições. O único produto de trabalho resultante da prática específica RD SP 3.4-3 do modelo CMMI é *Avaliar os riscos relatados nos requisitos*. E como o Scrum não avalia os riscos, pois entende que os prováveis riscos são eliminados rapidamente através das reuniões diárias e na entrega rápida do sistema, este produto de trabalho não é atendido no Scrum. Desta forma, a prática específica RD SP 3.4-3 do modelo CMMI é considerada *Não Atendida* no Scrum.

### SP 3.5-1 Validar os requisitos

A validação dos requisitos ocorre na fase conhecida como *Post Sprint Demonstration and Meeting*, onde os requisitos são verificados se estão descritos de forma apropriada, procurando, com isso, eliminar problemas como requisitos incompletos, ambíguos ou inconsistentes. Ressalta-se, que esta prática específica é somente aplicada na Representação Contínuo do modelo CMMI. Os resultados da validação dos requisitos, único produto de trabalho desta prática, são expostos para que a equipe Scrum inicie o processo de desenvolvimento. Desta forma, o produto de trabalho da prática específica RD SP 3.5-1 do modelo CMMI é atendido no Scrum, e esta prática é considerada *Atendida*.

### SP 3.5-1 Validar os requisitos com métodos válidos

A reunião conhecida como *Post Sprint Demonstration and Meeting* é um método que o Scrum utiliza para validar os requisitos. Esta prática específica diferencia-se da prática anterior, somente por exigir a definição de um método válido para validação dos requisitos. Então, o único produto de trabalho, *Registro dos métodos de análise e seus resultados*, é atendido pelo Scrum, logo, esta prática é considerada *Atendida*.

Finalizando a análise da área de processo Desenvolvimento de Requisitos, a Figura 10 ilustra a classificação do atendimento do Scrum em relação às práticas específicas da área de processo RD do modelo CMMI.

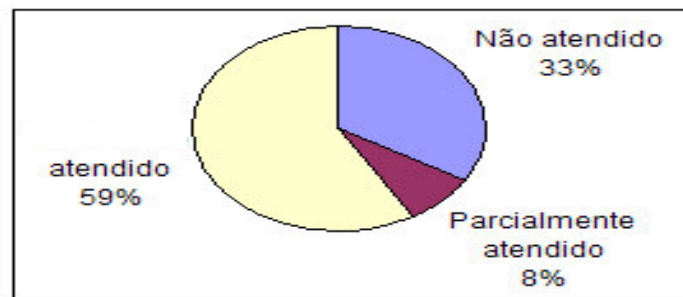


Figura 10 Escala de classificação do atendimento do Scrum em relação às práticas específicas da área de processo RD do CMMI.

Com base na análise realizada, percebe-se que a maioria das práticas específicas, cerca de 59%, encontra-se na posição de *Atendido*, e apenas 8% na posição de *Parcialmente Atendido* e 33% na posição de *Não Atendido*. Verifica-se, de certa forma, que apesar do Scrum encontrar-se na classificação de *Atendido*, este método *não está completamente* em conformidade com as exigências de todas as práticas específicas da área de processo RD do modelo CMMI.

## 4.3. Resultados finais da análise realizada nas áreas de processo REQM e RD

Analisando-se as duas áreas de processo em conjunto, percebe-se que de um total de 17 SP avaliadas, 6 encontram-se na posição de *Não Atendido* e apenas 2 como *Parcialmente Atendido*, ou seja, quase metade destas práticas necessitam de um estudo mais detalhado para adequação ao

modelo CMMI. Vale ressaltar que dentre as práticas na situação de *Não Atendido* ou *Parcialmente Atendido*, a maioria encontra-se como *Não Atendido* demonstrando que uma atenção especial deverá ser atribuída para estas práticas.

## 5. Considerações finais

Mediante a análise realizada, pode-se concluir que o método ágil Scrum necessita a inclusão de algumas atividades ou diretrizes em seu processo para atender as exigências das práticas específicas que foram enquadradas nas categorias de *Não Atendida* e *Parcialmente Atendida*. Nesse sentido, algumas soluções foram propostas para serem agregadas (como uma extensão) ao método ágil Scrum visando atender estas práticas. Estas soluções são apresentadas na Figura 11.

Descrição da SP	Solução
SP 1.3-1 Gerenciar as mudanças dos Requisitos	Desenvolver um documento que deverá basicamente registrar novos requisitos ou alterações: as solicitações de novos requisitos ou alterações serão recebidas formalmente. Também deverá elaborar um relatório de impacto visando a documentação da inclusão ou alteração, mantendo um histórico de alteração de cada requisito, inclusive de seus atributos.
SP 1.4-2 Manter a rastreabilidade bidirecional dos requisitos	Desenvolver uma matriz de rastreabilidade
SP 1.5-1 Identificar inconsistências entre o Projeto de Trabalho e os Requisitos	Desenvolver um documento que será utilizado durante o <i>Sprint</i> do Scrum, que indique as inconsistências entre o projeto de trabalho e os requisitos alocados.
SP 1.1-2 Eliciar as necessidades	Definir uma técnica de elicitação de requisitos
SP 2.2-1 Alocar os requisitos dos componentes dos produtos	Desenvolver um documento para registrar os componentes do produto e os requisitos alocados a eles. Cada projeto de trabalho terá um documento que armazenará, também, os requisitos temporários, as derivações dos requisitos e seus relacionamentos.
SP 2.3-1 Identificar as requisitos de interface	Elaborar um documento de requisitos de software, descrevendo as restrições quanto às características do produto, a interface com outras aplicações e a descrição sob o domínio .
SP 3.2-1 Estabelecer uma definição das funcionalidades requeridas	Agregar a técnica de caso de uso presente na UML
SP 3.4-3 Analisar os requisitos para avaliação	Desenvolvimento de um documento que auxilie no processo de análise do risco das necessidades dos <i>stakeholders</i> e suas restrições.

Fonte: primária

Figura 11 Soluções propostas

Conforme mostra a Figura 11, para cada inconsistência identificada na análise realizada, foi proposta uma solução. Estas soluções utilizam como base os seguintes fundamentos: i) estar alinhada à “cultura” Scrum – as soluções deverão estar de acordo com a filosofia Scrum, de forma a não descaracterizá-la e a realizar o menor impacto possível no método; e ii) utilizar práticas da engenharia de software que possam solucionar os problemas encontrados, como, por exemplo, a documentação de requisitos.

Caso uma organização que utiliza o método ágil Scrum no seu processo de desenvolvimento de software desejar uma adequação deste método com as áreas de processo REQM e RD do modelo CMMI, deverá desenvolver as soluções propostas para as práticas específicas que foram enquadradas nas categorias de *Não Atendida* e *Parcialmente Atendida*. Um exemplo para estas soluções é apresentado em Zanatta (2005, p.111).

## 6. Bibliografia

ABRAHAMSSON, Pekka, SALO Outi. *Agile Software Development Methods – Review and Analysis*. Espoo 2002, VTT Publications 478 107.

AHERN, Dennis. CLOUSE, Aaron. TURNER, Richard. *CMMI Distilled: a practical introduction to integrated process improvement*. Boston: Addison Wesley, 2004.

BOEHM, Barry. DeMARCO, Tom. *The Agile Methods Fray*. IEEE Computer Science, June, p. 90-91. 2002.

CHRISISS, Mary B.; KONRAD M.; SHRUM S. *CMMI: Guidelines for Process Integration and Product Improvement*. SEI, Addison Wesley, 2003.

FIORINI, Soeli. STAA, Arndt, BAPTISTA R. *Engenharia de software com CMM*. Rio de Janeiro: Brasport, 1998.

FOWLER, Martin. *The New Methodology*. Disponível em: <  
<http://www.martinfowler.com/articles/newMethodology.html>>. Acesso em : set 2003.

PAETSCH, F. EBERLEIN, A. MAURER, F. *Requirements Engineering and Agile Software Development*, WETICE 2003, IEEE Computer Science. 2003. p.308.

PAULK, Mark. C. *Extreme Programming from a CMM Perspective*, IEEE Software, vol. 18, no. 6, 2001. p. 19-26.

SCHWABER, Ken; BEEDLE, Mike. *Agile Software Development with SCRUM*. Prentice Hall, 2002.

TURNER, Richard. JAIN, Apurva. *Agile Meets CMMI: Culture clash or common cause*. XP/Agile Universe. 2002, p.153-165.

ZANATTA Lazaretti, Alexandre. *xScrum: uma proposta de extensão de um Método Ágil para Gerência e Desenvolvimento de Requisitos visando adequação ao CMMI* Florianópolis, 2004. 180 páginas. Dissertação (Mestrado em Ciência da Computação) - Curso de Pós-Graduação em Ciência da Computação, Universidade Federal de Santa Catarina.