

# Adoption of CMMI and agile methodologies in Brazilian companies

## La adopción de CMMI y metodologías ágiles en empresas brasileñas

## Adoção do CMMI e das metodologias ágeis em empresas brasileiras

Guilherme S. Lacerda<sup>1</sup> Me., Anderson Baum Barbosa<sup>2</sup> Bel., Vinicius G. Ribeiro<sup>3</sup> Ph. D.

1. Ciência da Computação; Faculdade de Informática - Centro Universitário Ritter dos Reis Porto Alegre – RS – Brasil

2. Sistemas de Informação; Faculdade de Informática - Centro Universitário Ritter dos Reis Porto Alegre – RS – Brasil

3. Ciência da Computação Faculdade de Informática - Centro Universitário Ritter dos Reis Porto Alegre – RS – Brasil

*guilherme\_lacerda@uniritter.edu.br, anderson.baum@gmail.com, vinicius@uniritter.edu.br*

Recibido para revisión 3 de marzo de 2011, aceptado 18 de octubre de 2011, versión final 4 de noviembre de 2011

**Resumo** — Inúmeros são os trabalhos, pesquisas e teses existentes na área de qualidade, especialmente os relacionados a desenvolvimento

de software. Alguns fatores como aumento da exigência dos clientes, necessidade de construir mais produtos com menos recursos e, principalmente, responder rapidamente às mudanças demandadas pelo negócio faz com que seja necessário, cada vez mais, identificar formas de melhorar os processos, a fim de atender essas exigências do mercado. Dessa forma, o presente trabalho exploratório buscou identificar uso do modelo CMMI e das metodologias ágeis em empresas brasileiras - uma vez que integrados, possibilitam melhorias nos processos das equipes de desenvolvimento de software e, por conseguinte, aumentam a qualidade de seus produtos.

**Palavras chave:** Qualidade de software; CMMI; Métodos ágeis.

**Abstract** — There are countless studies, researches and theses in the area of quality, especially those related to software development. Some factors such as increased demand from customers, need to build more products with fewer resources and, especially, respond rapidly to changes demanded by the business makes it necessary, increasingly, to identify ways to improve processes in order to meet these market requirements. Thus, this exploratory study sought to identify the use of CMMI and agile methods in Brazilian companies - once integrated, provide improvements in the processes of software development teams and therefore increase the quality of their products.

**Keywords** — software quality; CMMI, Agile methods.

**Resúmen** — Hay innumerables estudios, investigaciones y tesis en el área de calidad, especialmente las relacionadas con el desarrollo

de software. Algunos factores como el aumento de la demanda de los clientes, la necesidad de crear más productos con menos recursos y, sobre todo, responder con rapidez a los cambios exigidos por la empresa hace que sea necesario, cada vez más, para identificar formas de mejorar los procesos con el fin de cumplir con estas necesidades del mercado. Por lo tanto, este estudio exploratorio trató de identificar el uso de CMMI y los métodos ágiles en las empresas brasileñas - una vez integrados, proporcionan mejoras en los procesos de los equipos de desarrollo de software y por lo tanto aumentar la calidad de sus productos.

**Palabras clave** — Calidad del software, CMMI, Métodos Ágiles.

### I. INTRODUÇÃO

A qualidade de um projeto de software é muitas vezes determinada pela verificação se o produto desenvolvido por este está funcionando sem defeitos, se atende às necessidades propostas inicialmente no projeto, ou então, se o processo utilizado para produzi-lo foi seguido conforme o estipulado.

Existe uma variedade de modelos, normas reconhecidas internacionalmente e metodologias que visam um objetivo comum de prover práticas, que aplicadas ao longo do ciclo de vida do projeto, buscam aumentar a qualidade do produto final. Porém, ainda que muitos avanços tenham sido feitos na área de qualidade de software, existem lacunas a serem preenchidas, como mostra o *Chaos Report* [23]: 44% dos projetos analisados

foram concluídos ultrapassando o custo, prazo ou com redução de escopo, enquanto 24% falharam, ou seja, foram cancelados ou nunca foram usados.

Considerados os fatos descritos pelo *Standish Group* [23], questiona-se: como fazer ao longo das fases de desenvolvimento, que a produção atenda os padrões mínimos de qualidade necessários? Além disso, como garantir que o processo utilizado é o melhor? Que a equipe de desenvolvimento está sendo eficiente e eficaz, e está produzindo algo com qualidade?

Considerando que a qualidade de um produto é fortemente influenciada pela qualidade do processo utilizado para produzi-lo [12], o presente trabalho descreve as práticas e processos identificados no mais reconhecido modelo para melhoria de processos – CMMI –, e em metodologias ágeis. Para tanto, é analisado o que possuem em comum e como as empresas brasileiras de desenvolvimento de software os estão utilizando de forma integrada, a fim de contribuir no acréscimo dos níveis de qualidade do produto final por elas desenvolvido.

O trabalho está estruturado da seguinte forma: a seção dois apresenta o referencial teórico no que tange a qualidade, gestão de processos e métodos ágeis.

## II. REVISÃO LITERÁRIA

Estabelecer uma ligação entre qualidade e processo, parte do entendimento sobre a definição de ambos os conceitos. A complexidade em determinar o que é qualidade é extremamente alta, pois é fruto de uma combinação de elementos subjetivos e quantitativos. Com relação a processo, em sua essência, é a forma de executar alguma tarefa envolvendo pessoas, ferramentas e métodos ou procedimentos para cumprir um determinado objetivo. As empresas têm buscado constantemente melhorar os níveis de qualidade de seus produtos, apoiados em modelos e metodologias reconhecidas no mercado. Abaixo serão desdobrados alguns conceitos importantes sobre como os processos possuem um papel fundamental sobre a qualidade.

### A. Melhoria de Processos e a Qualidade

Em geral, qualidade significa excelência de um produto ou serviço em suportar e satisfazer as necessidades explícitas e implícitas, propostas no início da construção do produto [9]. Essa construção ocorre através da execução de inúmeros processos integrados, ou seja, por uma sequência de passos realizados para um determinado propósito, utilizando métodos, aplicados seguindo técnicas, com o apoio de ferramentas e sob a responsabilidade de pessoas.

Em uma análise subjetiva, o entendimento sobre um produto entregue com qualidade está no fato de atender as funcionalidades solicitadas, enquanto que outro entendimento sobre qualidade poderá estar demonstrada através de números baixos de defeitos durante a produção, levando a definição

de Crosby [5] sobre o que é qualidade: “Qualidade é a conformidade aos requisitos”.

Sommerville [20], reforçando o caráter subjetivo da qualidade, define que:

“Diferentes *stakeholders* tem em mente diferentes requisitos e podem expressá-los de maneira distintas. Os engenheiros de requisitos precisam descobrir todas as possíveis fontes de requisitos e encontrar pontos comuns e os conflitos”.

Além do aspecto individual de interpretação do que é qualidade, existe a questão quantitativa, ou seja, a que pode ser expressa em números. Os dados numéricos são muito importantes para o gerenciamento de projetos, pois reduzem a possibilidade de erros de avaliação, por sofrerem menos efeitos da subjetividade, que podem levar a decisões errôneas com consequências negativas ao projeto [11].

Por trás desses aspectos acerca da busca pela qualidade, estão os processos e a preocupação em constantemente melhorá-los. Um dos métodos mais utilizados, quando se fala em melhoria de processos, é o ciclo PDCA (*Plan, Do, Check, Act*) [6].

Cada um desses passos representa uma etapa na melhoria do processo. O *Plan* está relacionado ao planejamento de uma melhoria. Já o *Do* diz respeito a colocar esse planejamento em prática. O *Check* refere-se à constantemente monitorar e avaliar os resultados obtidos conforme a execução do que foi planejado. Por fim, o *Act* fecha o ciclo provendo a possibilidade de realizar mudanças na forma de planejar e agir ou mesmo de institucionalizar e manter a forma de execução de um determinado processo, baseado nos resultados obtidos.

Kolmel e Eisenbiegler [12] demonstraram através de um estudo, em pelo menos 100 empresas européias, que a aplicação de melhoria de processos de desenvolvimento contribuiu significativamente no aumento da qualidade entregue por estas empresas, como ilustra a figura 1:

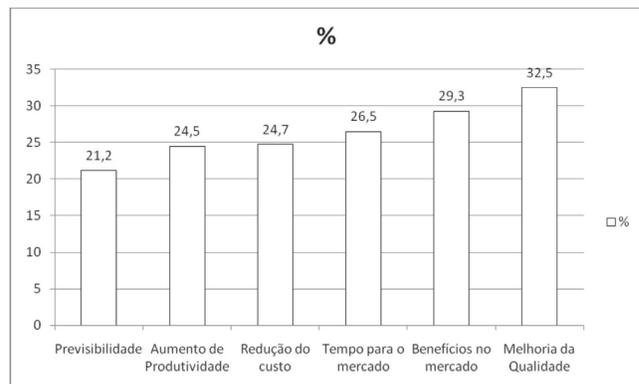


Figura 1 - Resultado do Estudo de Kolmel e Eisenbiegler (fonte: KOLMEL e EISENBIEGLER, 2000)

De fato uma organização que queira obter uma maior qualidade em seus produtos, necessitará planejar, corrigir e

melhorar, ao longo do ciclo de vida do projeto de construção de um produto, a melhoria contínua de seus processos. Devido a isso é que atualmente há uma busca constante sobre modelos e metodologias de mercado, principalmente sobre o CMMI e as Metodologias Ágeis, que sirvam de referência para melhoria nos processos do desenvolvimento de software.

### B. Modelo de Maturidade de Processos CMMI

O objetivo do CMMI (*Capability Maturity Model Integration*) é servir de guia para melhoria de processos nas organizações, no que diz respeito ao desenvolvimento, aquisição e manutenção de produtos ou serviços. Ao utilizar o CMMI, uma organização busca ser mais eficiente, respeitando seus próprios prazos e construindo software com menos erros [4].

Em vista de entender o modelo de maturidade de processos CMMI, é necessário compreender algumas definições de elementos contidos nele:

- área de processo: conjunto de práticas executadas coletivamente, satisfazendo um conjunto de objetivos que repercutem em uma melhora nessa área;
- objetivos específicos: identificam características únicas que descrevem o que deve ser implementado para satisfazer uma área de processo;
- práticas específicas: atividades importantes para atingir um determinado objetivo, o que estão associadas com o nível de maturidade;
- objetivos genéricos: existe um para cada nível de maturidade, descrevendo o que a organização deve fazer para atingir um determinado nível;
- práticas genéricas: são categorizadas por nível de maturidade e asseguram que os processos associados com as áreas de processo serão efetivos e repetíveis.

O CMMI possui duas representações: a contínua e a por estágios. Estas representações permitem à organização utilizar diferentes caminhos para a melhoria de seus processos de acordo com seu interesse ou necessidade (SEI, 2006).

- representação contínua: possibilita à organização utilizar a ordem de melhoria que melhor atende os objetivos de negócio da empresa. Essa representação é caracterizada por Níveis de Capacidade (*Capability Levels*);
- representação por estágios: disponibiliza uma sequência pré-determinada para melhoria baseada em estágios que não deve ser desconsiderada, pois cada estágio serve de base para o próximo. Essa representação é caracterizada por Níveis de Maturidade (*Maturity Levels*).

O modelo CMMI versão 1.2 (CMMI-DEV) contém 22 áreas de processo. Em sua representação por estágios as áreas são divididas da seguinte forma (Figura 2):

- nível 1 – Inicial (*Ad-hoc*): nesse nível os processos são caóticos. Normalmente não existe um ambiente estável de

desenvolvimento de software, padrões não existem ou se existem, não são seguidos;

- nível 2 – Gerenciado: os projetos possuem requisitos gerenciados e processos planejados, medidos e controlados. As práticas permitem que a organização cumpra os planos no desenvolvimento dos projetos. São definidas práticas de gerenciamento de requisitos, planejamento, acompanhamento e controle de projetos, gerenciamento de acordo com fornecedores, gerência da configuração, garantia da qualidade de processos e produtos e medição e análise de informações de processos e produtos;
- nível 3 – Definido: os processos são bem caracterizados e entendidos. A padronização de processos possibilita maior consistência nos produtos gerados pela organização. Algumas práticas são descritas, como o desenvolvimento de requisitos, solução técnica, integração de produto, verificação e validação de produto, definição e foco de processo organizacional, treinamento organizacional, análise de decisão e resolução de problemas, bem como gerenciamento integrado de projetos e de riscos;
- nível 4 – Quantitativamente Gerenciado: os processos são selecionados para contribuir com o desempenho geral dos demais processos. São controlados usando métodos estatísticos e outras técnicas quantitativas, bem como analisando o desempenho dos processos organizacionais;
- nível 5 – Otimizado: os processos são continuamente melhorados com base em um entendimento quantitativo das causas comuns de alterações de desempenho. Para isso são feitas análises causais e de resolução de problemas nos processos, bem como constante inovação organizacional e de implantação de processos.

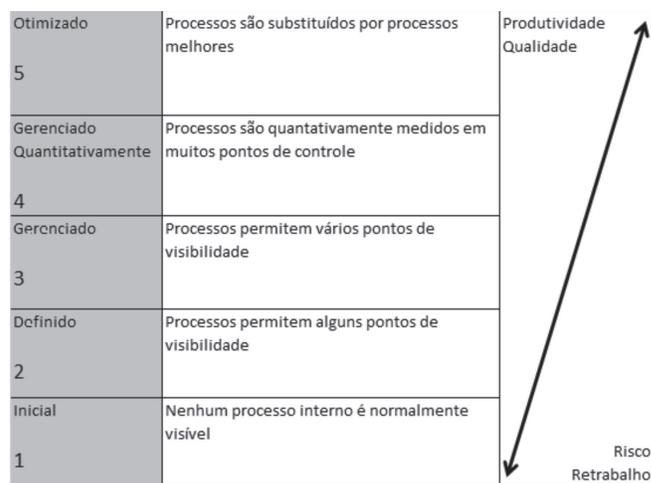


Figura 2 - Níveis de Maturidade dos Processos (fonte: KOCIANSKI e SOARES, 2007)

Em um contexto em que se necessite o uso otimizado de documentação e mais dinamicidade para atender às mudanças no produto, empresas que queiram aplicar esse modelo de maturidade de processos, acabam contratando consultorias especializadas [11], consumindo tempo e dinheiro. Dessa forma, as empresas começaram a encontrar nas metodologias ágeis uma forma de atingir os objetivos das áreas de processos do CMMI, uma vez que elas prezam pela redução no desperdício de tempo com atividades que não trazem valor ao cliente.

### C. Metodologias Ágeis

Existem inúmeras metodologias ágeis de desenvolvimento de software, cada uma destas expostas pelo *The Agile Alliance*. Em sua essência, essas metodologias tentam minimizar o risco de forma a fazer com que funcionalidades mais importantes sejam desenvolvidas primeiramente e em um curto período de tempo, chamado de iteração. Cada iteração é como um projeto de software em miniatura de seu próprio, e inclui todas as tarefas necessárias para implantar uma nova funcionalidade: planejamento, análise de requisitos, *design*, codificação, teste e documentação [13].

As metodologias ágeis estão alicerçadas em alguns princípios, conforme o *Manifesto Ágil* [2]:

- garantir a satisfação do cliente entregando rapidamente e continuamente softwares em funcionamento;
- softwares funcionais são a principal medida de progresso do projeto, sendo mais importantes que descrever uma documentação extensa, não significando que documentos não sejam necessários;
- até mesmo mudanças tardias de escopo no projeto são bem-vindas, mas para isso precisam novamente ser planejadas e incluídas em novas iterações;
- desenho do software deve prezar pela excelência técnica;
- indivíduos e interações mais do que processos e ferramentas, não significando que deva ser deixado de lado qualquer processo ou ferramenta, pelo menos um mínimo deve existir;
- colaboração com clientes mais do que negociação de contratos.

As duas metodologias ágeis mais utilizadas atualmente são o XP (*Extreme Programming*), focado nos aspectos de do desenvolvimento de produto, e *Scrum*, focado no gerenciamento e acompanhamento do projeto de desenvolvimento.

#### 1) XP (*Extreme Programming*)

*Extreme Programming (XP)* é uma metodologia ágil para equipes pequenas e médias, que irão desenvolver software com requisitos vagos e em constante mudança. Para isso, se adota a estratégia de constante acompanhamento e realização de vários pequenos ajustes durante o desenvolvimento de software [3].

Possui como valores e princípios básicos a comunicação,

trabalho de alta qualidade com simplicidade, feedback rápido e constante, coragem para abraçar as mudanças, e respeito entre os envolvidos no projeto de software.

Dentre as variáveis de controle em projetos (custo, tempo, qualidade e escopo), XP possui um foco explícito em escopo. Para isso, recomenda-se a priorização de funcionalidades que representem maior valor possível para o negócio. Desta forma, caso seja necessária a diminuição de escopo, as funcionalidades menos valiosas serão adiadas ou canceladas.

Para aplicar os valores e princípios durante o desenvolvimento de software, XP propõe uma série de práticas [25]:

- jogo de planejamento (*Planning Game*): sessão para priorizar as funcionalidades a serem desenvolvidas na iteração;
- pequenas versões (*Small Releases*): a liberação de pequenas versões funcionais do projeto auxilia no processo de aceitação por parte do cliente;
- metáfora (*Metaphor*): é a tradução das palavras do cliente para o significado que ele espera dentro do projeto, a fim de facilitar a comunicação;
- projeto simples (*Simple Design*): fazer o código exato para que a funcionalidade seja implementada;
- time coeso (*Whole Team*): a equipe de desenvolvimento é formada pelo cliente e pela equipe de desenvolvimento;
- testes de aceitação (*Customer Tests*): são testes construídos com base em critérios estabelecidos pelo cliente, para aceitar um determinado requisito do sistema;
- ritmo sustentável (*Sustainable Pace*): trabalhar com qualidade, buscando ter ritmo de trabalho saudável (40 horas por semana), sem horas extras;
- reuniões em pé (*Stand-up Meeting*): as reuniões são rápidas, realizadas em pé, sendo que são abordadas apenas tarefas realizadas e a realizar pela equipe;
- posse coletiva (*Collective Ownership*): o código fonte não tem dono e ninguém precisa solicitar permissão para poder modificar o mesmo;
- programação em pares (*Pair Programming*): é a programação em dupla utilizando um único computador;
- padrões de codificação (*Coding Standards*): a equipe de desenvolvimento precisa estabelecer regras para programar e todos devem seguir estas regras;
- desenvolvimento orientado a testes (*Test Driven Development*): primeiro cria-se os testes unitários e depois se cria o código para que eles funcionem;
- refatoração (*Refactoring*): refabricar melhora a clareza do código, divide-o em módulos mais coesos e de maior reaproveitamento;
- integração contínua (*Continuous Integration*): sempre

que produzir uma nova funcionalidade, nunca esperar para integrar à versão atual do sistema.

## 2) SCRUM

A função primária do *Scrum* é ser utilizado para o gerenciamento de projetos de desenvolvimento de software, porém, teoricamente ele pode ser aplicado em qualquer contexto no qual um grupo de pessoas necessite trabalhar juntas para atingir um objetivo comum [16].

### O *Scrum* possui as seguintes características:

- cada *sprint* é uma iteração que segue um ciclo (PDCA) e entrega incremento de software pronto;
- um *backlog* é conjunto de requisitos, priorizado pelo *Product Owner* (responsável pelo ROI – *Return over Investment* ou Retorno sobre o Investimento - e por conhecer as necessidades do cliente);
- há entrega de um conjunto fixo de itens do *backlog* em série de interações curtas ou *sprints*;
- breve reunião diária, ou *daily scrum*, em que cada participante fala sobre o progresso conseguido, o trabalho a ser realizado e o que o impede de seguir avançando;
- breve sessão de planejamento, na qual os itens do *backlog* para uma *sprint* (iteração) são definidos;
- retrospectiva, na qual todos os membros da equipe refletem sobre a *sprint* passada;

O *Scrum* é facilitado por um *Scrum Master*, que tem como função primária remover qualquer impedimento à habilidade de uma equipe de entregar o objetivo do *sprint*. , bem como o de utilizar corretamente as práticas de *Scrum*, motivando-os e mantendo o foco na meta da iteração.

*Scrum* permite a criação de equipes auto-organizadas, encorajando a comunicação verbal entre todos os membros da equipe e entre todas as disciplinas que estão envolvidas no projeto [17].

De fato, as metodologias ágeis definem, através de práticas relativamente simples, como equipes de desenvolvimento devem planejar, monitorar, controlar e desenvolver um determinado produto. Estabelecer a sua ligação com o CMMI é importante, para que as empresas possam se valer dos níveis de maturidade de processo desse modelo combinados com a agilidade das metodologias ágeis em responder as mudanças. A próxima seção abordará essa integração.

## 3) Integrando CMMI e Metodologias Ágeis

O ciclo de melhoria de processos PDCA serviu de base para muitas normas, modelos e métodos serem desenvolvidos, dentre os quais o modelo CMMI e as metodologias ágeis. Enquanto as metodologias ágeis estão focadas em como os projetos devem desenvolver os produtos [7], o CMMI preocupa-se em determinar o que os projetos fazem ou devem fazer para conquistar níveis elevados de maturidade e melhoria de processos, garantindo gradativamente, níveis elevados de qualidade do produto em desenvolvimento.

Uma série de fatores faz com que um modelo como o CMMI não seja adotado por organizações: tamanho da organização, custo para implementação das práticas descritas no modelo e senso comum a respeito de não haver benefício aos projetos [21]. Em contra partida, cada vez mais as empresas adotem as metodologias ágeis como padrão de condução de projetos de desenvolvimento, por serem dinâmicas e responderem rapidamente às mudanças nos requisitos, além de serem relativamente mais baratas de serem introduzidas nas organizações.

Um estudo provou que as áreas de processos do CMMI podem ser implementadas por práticas ágeis, bastando para isso que a organização tenha como base de sua política os valores e princípios dessas metodologias, bem como disponibilizando repositórios de processos, documentos, técnicas e práticas de ambos os modelos [1]. O resultado desse estudo também demonstrou, através de um mapeamento baseado na análise de percentual de cobertura de uma área de processo do CMMI por práticas de metodologias ágeis, que é possível e recomendado utilizarem ambos para melhorar o processo de desenvolvimento. A tabela 1 apresenta esse mapeamento.

**Tabela 1.** cobertura das áreas de processos do cmmi pelas metodologias ágeis (fonte: alegria e bastarrica, 2006)

Área	Métodos com maior contribuição	Percentual de satisfação (nível de capacidade ou NC + percentual)
Gerenciamento de Requisitos	XP	NC-2 em 75%, NC-3 em 50%
Medição e Análise	XP, ASD, SCRUM	NC-2 em 75%, NC-3 em 50%
Planejamento de Projeto	XP, SCRUM	NC-3 em 100%
Monitoramento e Controle	XP, SCRUM	NC-3 em 100%
Gerenciamento de Subcontratado	Não se aplica	-
Garantia de Qualidade de Produto e Processo	FDD, Crystal, EVO	NC-2 em 75%, NC-3 em 50%
Gerenciamento da Configuração	Nenhum	NC-1 em 10%
<b>Percentual de satisfação para nível de maturidade 2</b>		<b>72%</b>
<b>Percentual de satisfação para nível de maturidade 3</b>		<b>60%</b>

Uma visão macro desse mapeamento demonstra, através da divisão dos níveis de maturidade, onde as práticas ágeis estão

associadas ao CMMI, conforme a Tabela 2:

**Tabela 2.** aderência ao cmmi utilizando métodos ágeis (fonte: <http://www.blogcmmi.com.br>)

Maturity Level	Process Area	SCRUM	XP
Level 2: Managed	Requirements Management (REQM)	LA	LA
	Project Planning (PP)	LA	LA
	Project Monitoring and Control (PMC)	PA	PA
	Supplier Agreement Management (SAM)	NA	NA
	Measurement and Analysis (MA)	LA	LA
	Process and Product Quality Assurance (PPQA)	NA	PA
	Configuration Management (CM)	NA	PA
Level 3: Defined	Requirements Development (RD)	LA	LA
	Technical Solution (TS)	NA	NA
	Product Integration (PI)	NA	LA
	Verification (VER)	NA	LA
	Validation (VAL)	LA	LA
	Organizational Process Focus (OPF)	PA	PA
	Organizational Process Definition (OPD)	PA	PA
	Organizational Training (OT)	NA	NA
	Integrated Project Management (IPM)	NA	NA
	Risk Management (RSKM)	NA	NA
Decision Analysis and Resolution (DAR)	NA	NA	
Level 4: Quantitatively managed	Organizational Process Performance (OPP)	NA	NA
	Quantitative Project Management (QPM)	NA	NA
Level 5: Optimizing	Organizational Innovation and Deployment (OID)	NA	NA
	Causal Analysis and Resolution (CAR)	NA	NA

LA = Largely Addressed / PA = Partially Addressed / NA = Not Addressed

Analisando algumas áreas de processo do CMMI, dos níveis 2 e 3 de maturidade, chega-se ao mapeamento das práticas

utilizadas nas metodologias (*Scrum* e *XP*) ágeis que cumprem com os objetivos dessas áreas [15]:

**Quadro 1** - Objetivos CMMI e as Práticas Ágeis (fonte: PIKKARAINEN e MÄNTYNIEMI, 2006)

Objetivo do CMMI	Práticas Ágeis
Gerenciamento de Requisitos	Stories, Product Backlog; Planning Games; Information Radiator; On-line Customer; Self-organizing Teams
Estabelecimento de Estimativas	Planning Games; Tasks and Effort Estimations for one to two weeks iterations on information radiator
Desenvolvimento do Plano de Projeto	Planning Games; Tasks on information radiator; Product Backlog
Aprovação do Projeto	Planning Games; On-line Customer; Self-organizing Teams; Reflection Workshops

Tendo em vista esses estudos comparativos que relatam a possibilidade de utilizar os processos do CMMI com as práticas das metodologias ágeis, buscou-se identificar, nas empresas de desenvolvimento de software, como estão tratando dessa integração.

### III. USO DO CMMI E DAS METODOLOGIAS ÁGEIS NAS EMPRESAS BRASILEIRAS

Para verificar se o modelo CMMI e as metodologias ágeis podem ser adotados de forma integrada, foi realizado levantamento com algumas empresas brasileiras de desenvolvimento de software. A forma de identificação das empresas elegíveis a participar da pesquisa se deu através de uma consulta à base de dados do *Software Engineering Institute (SEI)*[19], mantenedora do modelo CMMI.

Ao todo, foram contatadas 52 empresas – escolhidas por conveniência -, entre os dias 6 de outubro e 15 de novembro de 2010, convidando-as a responderem um questionário (variáveis apresentadas no ANEXO I) disponibilizado no *Google Docs*, sendo que 15 dessas empresas disponibilizaram suas informações, onde apenas uma não possuía certificação CMMI.

Em geral, tratam-se de empresas que desenvolvem produtos para os mercados financeiros (bancos, corretoras), indústria (de todos os tipos), governo (federal, estadual e municipal), telecomunicações, saúde e energia. Algumas empresas também citaram o fato de serem *software houses*, desenvolvendo produtos que atendem diversos mercados.

Com relação à localização dessas empresas, cinco são de São Paulo, quatro do Rio Grande do Sul, duas da Bahia, uma de Santa Catarina, uma do Paraná, uma do Ceará e uma de Pernambuco. No que diz respeito a tempo de atuação no mercado, nove dessas empresas possuem de 6 a 15 anos, cinco empresas possuem de 16 a 25 anos e uma empresa possui mais de 26 anos de atuação.

Analisando apenas as organizações com certificação, 13 utilizaram a abordagem por estágio, enquanto que uma utiliza a abordagem contínua. A empresa que não está certificada também informou utilizar a abordagem contínua. Outro ponto interessante é o nível de certificação, nove empresas possuem o nível 2, três empresas possuem o nível 3 e duas empresas possuem o nível 5. Em termos da representatividade do **nível de certificação**, essa amostra de empresas é bastante condizente com a base de informações da SEI, onde a maioria das empresas brasileiras utiliza a abordagem por estágio e está nos níveis 2 e 3 de certificação.

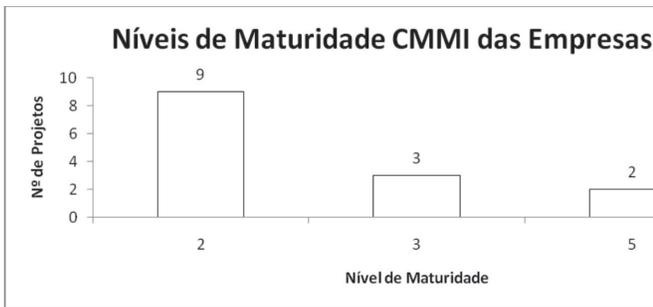


Figura 3 - Níveis de maturidade CMMI das empresas que responderam a pesquisa

No que diz respeito à utilização de metodologias ágeis, duas empresas realizaram o seu processo de certificação CMMI utilizando metodologias ágeis, o que demonstra a aderência das práticas ágeis às áreas de processo do CMMI. A Figura 4 mostra o percentual de projetos, nas empresas certificadas, que utilizam metodologias ágeis. Ao todo, são nove organizações que informaram utilizar essas metodologias:

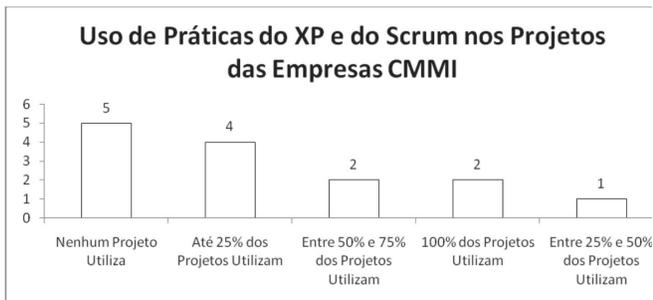


Figura 4 - Percentual de projeto utilizando Metodologias Ágeis em empresas CMMI

Dentre as práticas ágeis, observou-se que as que estão ligadas ao *Scrum* são largamente utilizadas, conforme mostrado na figura 3, onde se verifica certa relação entre os níveis de certificação CMMI e as práticas empregadas:

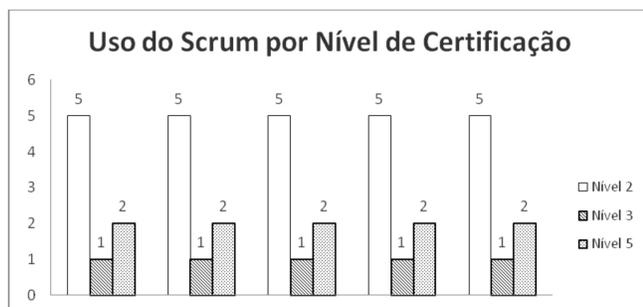


Fig. 5 – Utilização das práticas do Scrum por nível de certificação CMMI

Já no que diz respeito ao XP, observou-se que há grande enfoque nas práticas voltadas ao desenvolvimento, como *Unit Tests*, *Continuous Integration* e *Coding Standards* (Figura 6). Essa observação é importante, haja vista que o foco do *Scrum* está na gestão do projeto, enquanto que as práticas do XP focam mais nas atividades de desenvolvimento do produto.

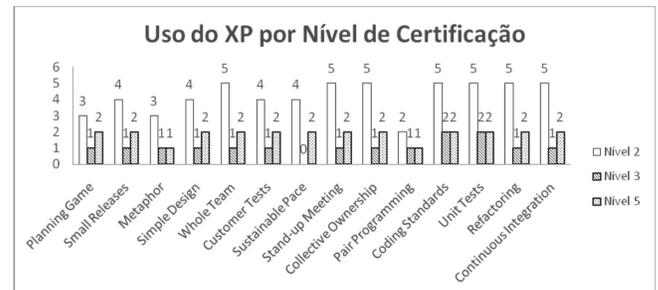


Fig.6 – Utilização de práticas do XP por nível de certificação CMMI

Assim, pode-se verificar, nos estudos mapeando as práticas ágeis para áreas de processo CMMI, a combinação de práticas do XP e do *Scrum* – o que possibilita atender objetivos dos níveis de maturidade do CMMI.

#### IV. CONSIDERAÇÕES FINAIS

A garantia dos níveis de qualidade em projetos de desenvolvimento de software, não é uma tarefa fácil. Cientes dessa dificuldade, as empresas adotaram modelos como o CMMI e metodologias ágeis para utilizar práticas e processos reconhecidos pelo mercado, de forma a garantir a qualidade de seus produtos. Enquanto o CMMI foca em determinar o que deve ser feito para desenvolver um produto, as metodologias ágeis descrevem práticas de como desenvolver e gerenciar o desenvolvimento de um produto.

As práticas das metodologias ágeis são, em sua maioria, aderentes às áreas de processos do CMMI, nos níveis 2 e 3 de maturidade. O resultado da pesquisa realizada com as empresas de desenvolvimento brasileiras demonstrou que os estudos existentes que mapeiam as práticas dos métodos ágeis com o CMMI, estão em uma direção correta. De fato, é possível adotar práticas ágeis em empresas certificadas CMMI e, além disso, é possível utilizar essas práticas para gerar evidências em processos de certificação desse modelo, como foi observado em duas das empresas consultadas.

Existe uma tendência, caracterizada por seis empresas que utilizam pouca ou nenhuma prática ágil, em utilizar um número maior ou todas as práticas em um futuro próximo. Isso abre um campo de pesquisa e atuação de consultorias que possam se beneficiar dessas metodologias para facilitar a aderência aos processos do CMMI. Outro fator importante dos métodos ágeis é que possuem em seus princípios a extrema disciplina e necessidade de comprometimento de todo o time de desenvolvimento, elementos muito importantes para conquistar níveis elevados de maturidade de processo e, por conseguinte, de qualidade.

Embora apenas 29% das empresas consultadas tenham respondido a pesquisa consolidada nesse artigo, dificultando a obtenção de um conjunto maior de informações para análise,

fica como possibilidade futura a ser explorada a realização desse tipo de investigação em um número maior de organizações, de forma a comparar não só o uso de um modelo de melhoria de processos como o CMMI, mas também outros modelos reconhecidos no mercado brasileiro, como o MPS.BR. Outra possibilidade seria expandir essa pesquisa de forma global, ao invés de estar restrito apenas ao mercado brasileiro, como é o caso ocorrido nessa investigação.

Em suma, em nenhuma área de processos do CMMI é exigido que um projeto empregue uma determinada metodologia, seja ela cascata ou ágil. Além disso, não há nenhuma determinação que os artefatos gerados pelos processos devam ser extensos, esse é um problema muitas vezes gerado pelas empresas, que acabam colocando mais informações do que as necessárias. Dessa forma, adotar as práticas do XP e do *Scrum*, de forma gradativa em um ritmo constante de melhoria de processos, permite que a empresa, ao necessitar certificar-se no modelo CMMI, o faça de forma menos traumática e com custos não tão elevados.

Em termos de metodologia, o presente estudo exploratório não detém a representatividade ideal, visto a baixa taxa de respondentes, além da distribuição geográfica das organizações dos respondentes. Contudo, foi possível obter uma visão inicial da realidade desejada, suficiente para um primeiro levantamento exploratório.

#### REFERÊNCIAS

- [1] Alegria, J. A., e Bastarrica, M. C. , 2006. Implementing CMMI using a Combination of Agile Methods. En: CLEI Electronic Journal, Vol. 9, N. 1, paper 7.
- [2] Beck, K. , 2001. Acessado em 22 de abril de 2010, na página Manifesto for Agile Software Development: <http://agilemanifesto.org/>
- [3] Beck, K., e Andres, C., 2004. Extreme Programming Explained: Embrace Change. Boston: Addison-Wesley Professional. 2<sup>nd</sup> Edition. 224 p.
- [4] Chrissis, M. B., Konrad, M., e Shrum, S., 2006. CMMI: Guidelines for Process Integration and Product Improvement, 2<sup>nd</sup> edition. Boston: Addison-Wesley Professional. 663 p.
- [5] Crosby, P., 1992. Quality is Free: the Art of Making Quality Certain. São Paulo, SP: Mentor Books. 309 p.
- [6] Deming, W. E., 1990. A revolução da administração. Rio de Janeiro: Saraiva. 116 p.
- [7] Glazer, H., Dalton, J., Anderson, D., Konrad, M., e Shrum, S. (2008, Novembro). CMMI or Agile: Why Not Embrace Both! p. 28.
- [8] International Organization for Standardization (ISO) & International Electrotechnical Commission, 1998. ISO/IEC 12207: Software process lifecycle.
- [9] International Organization for Standardization (ISO) & International Electrotechnical Commission, 1994. ISO/IEC 8402: Quality Management and Quality Assurance - Vocabulary.
- [10] Jeffries, R., Anderson, A., & Hendrickson, C., 2000. Extreme Programming Installed. Boston: Addison-Wesley Professional. 288 p.
- [11] Kocianski, A., e Soares, M., 2007. Qualidade de Software. São Paulo, SP: Novatec. 186 p.
- [12] Kolmel, B., e Eisenbiegler, J., 2000. Impact Measurement of SPI-Projects. En: Views on Software Development . Reykjavik Conference, Iceland: University of Iceland. (31 de Agosto de 2000)
- [13] Kroll, P., e MacIsaac, B., 2006. Agility and Discipline Made Easy: Practices from OpenUP and RUP. Boston: Pearson Education. 448 p.
- [14] Machado, C. F., 2006. Definindo Processos do Ciclo de Vida de Software Usando a Norma NBR ISO/IEC 12207 e Suas Ementas 1 e 2. Lavras: UFLA/FAEPE.
- [15] Pikkarainen, M., e Mäntyniemi, A., 2006. An Approach for Using CMMI in Agile Software Development Assessments: Experiences from Three Case Studies. In SPICE 2006 conference. Luxemburg(2006, May).
- [16] Schwaber, K., 2004. Agile Project Management with SCRUM. Redmond, WA: Microsoft Press. 192 p.
- [17] Schwaber, K., 2002. Agile Software Development with SCRUM. Upper Saddle River, NJ: Prentice Hall. 158 p.
- [18] Scrum Alliance. Acessado em 22 de Março de 2010, em: <http://www.scrumalliance.org>
- [19] SEI – Software Engineering Institute, 2006. Acessado em 22 de Março de 2010, em <http://www.sei.cmu.edu/reports/O6tr008.pdf>
- [20] Sommerville, I., 2003. Engenharia de Software. São Paulo, SP: Prentice-Hall. 8<sup>a</sup> edição. 568 p.
- [21] Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P., & Murphy, R., 2007. An exploratory study of why organizations do not adopt CMMI. The Journal of Systems and Software .
- [22] Takeuchi, H., & Nonaka, I. , 1986. The New New Product Development Game. Harvard Business Review. (Janeiro-Fevereiro 1986).
- [23] The Standish Group, 2009. Chaos Report. Boston, Massachusetts.
- [24] Vriens, C., 2003. Certifying for CMM Level 2 and ISO9001 with XP@Scrum. En: Agile Development Conference (ADC '03).
- [25] Wells, D., 1999. Acessado em 22 de Março de 2010 - A gentle introduction: <http://www.extremeprogramming.org/>.

**Guilherme S. Lacerda** nasceu em Bagé (RS), Brasil, em 1979. É graduado em Informática pela Urcamp (Bagé, 2000), e é Mestre em Sistemas de Informação, área de Engenharia de Software pelo PPG da UFRGS (Porto Alegre, 2005). Dedicava-se atualmente em atividades de consultoria e treinamento em Engenharia de Software, Gerência de Projetos e Metodologias Ágeis. É Professor Universitário em nível de Graduação (Facensa, UniRitter) e Pós-Graduação (UniRitter, Unisinos). Ele atuou como diretor de tecnologia de uma empresa do ramo de software livre e open source durante 9 anos. Desenvolveu diversos projetos com tecnologias C/C++, PHP e Java, usando Metodologias Ágeis. É pioneiro no emprego de Metodologias Ágeis no Brasil, onde atua desde 2001, com especial ênfase em Lean, SCRUM e eXtreme Programming. Certified SCRUM Master (CSM) e Certified SCRUM Professional (CSP) pela SCRUM Alliance. Palestrante em dezenas de eventos nacionais e internacionais sobre o tema. Participou da revisão técnica do livro “eXtreme Programming Explained”, do Kent Beck, lançado em 2004 pela Bookman. Fundador do XP-RS/GUMA, onde atua na vice-coordenação. Me. Lacerda é Membro do IASA (International Association Software Architects).

**Anderson B. Barbosa** nasceu em Porto Alegre (RS), Brasil, em 1985. É graduado em Sistemas de Informação pelo Centro Universitário Ritter dos Reis (2010). Atualmente, cursa Pós-graduação em Tecnologias Aplicadas a Sistemas de Informação com Métodos Ágeis, no Centro Universitário Ritter dos Reis. Ele atua na Hewlett Packard do Brasil, no centro de desenvolvimento e pesquisa em Porto Alegre, atuando como Quality Engineer na área de processos e garantia da qualidade.

**Vinicius G. Ribeiro** nasceu em Porto Alegre (RS), Brasil, em 1962. É Graduado em Ciências Náuticas, pelo Ministério da Marinha (Rio de Janeiro, 1984), e em Ciência da Computação, pela UFRGS (Porto Alegre, 1994). Mestre em Administração – área de Concentração de Produção e Sistemas, pela UFRGS (Porto Alegre, 1996) e Doutor em Ciência da Computação pela UFRGS (Porto Alegre, 2005). É professor Adjunto no Centro Universitário Ritter dos Reis, em disciplinas de Graduação, Especialização e Mestrado, e professor Titular na Faculdade Cenecista Nossa Senhora dos Anjos, em disciplinas de Graduação. É Coordenador do Mestrado em Design do Centro Universitário Ritter dos Reis.

Ele foi consultor de empresas de desenvolvimento de software – tanto privadas, como estatais. É autor do livro *Pesquisa em Computação: uma Abordagem Metodológica para Trabalhos de Conclusão de Curso e de Iniciação Científica* (Porto Alegre, RS: UniRitter, 2010). Suas áreas de interesse incluem Sua área de interesse incluem as metodologias de desenvolvimento de software, segurança computacional, metodologias de investigação em Computação e modelagem de fenômenos da realidade.

Dr. Ribeiro é membro da ACM - Association for Computing Machinery - e da SBC – Sociedade Brasileira de Computação.

## ANEXO I

### Questões componentes do questionário – questões fechadas

Dados identificatórios e CMMI	Tecnologia XP	Tecnologia SCRUM
Nome da Empresa	XP Planning game	SCRUM Product Backlog
Estado	XP Smal releases	SCRUM Sprint Backlog
Tempo atuação no mercado	XP metaphor	SCRUM Planning Game
e-mail	XP Simples Design	SCRUM Daily Scrum Meeting
Principais TI empregadas	XP Whole Team	SCRUM Sprint Review
Foco dos produtos	XP Customer Tests	
Certificada CMMI	XP Sustainable Pace	
Abordagem para verificação CMMI	XP Stand-up Meeting	
Se “Por Estágio”, qual nível	XP Collective Ownership	
Empregados métodos ágeis na certificação	XP Pair Programming	
Se não certificada, pretende?	XP Coding Standards	
	XP Unit Tests	
	XP Refactoring	
	XP Continuous Integration	



Cisco Networking Academy es un programa ampliamente conocido de e-doing que enseña a los estudiantes las habilidades tecnológicas de Internet en una economía global. El programa proporciona contenido basado en la Web, pruebas en línea, seguimiento del desempeño de los estudiantes, laboratorios con equipos reales y con simuladores, soporte y entrenamiento por parte de los instructores, así como preparación para las certificaciones estándares de la industria.



#### Oferta de cursos

- ✓ Mantenimiento de PC: IT Essentials
- ✓ Redes básicas: Cisco Certified Network Associate
- ✓ Redes avanzadas: Cisco Certified Network Professional
- ✓ Seguridad en routers: CCNA Security
- ✓ Voz sobre IP
- ✓ Asterisk básico

#### Programación 2011

Ciclo 48: Inicia 17 de enero. Finaliza 12 de marzo  
 Ciclo 49: Inicia 22 de marzo. Finaliza 23 de mayo  
 Ciclo 50: Inicia 30 de mayo. Finaliza 29 de julio  
 Ciclo 51: Inicia 8 de agosto. Finaliza 3 de octubre  
 Ciclo 52: Inicia 10 de octubre. Finaliza 12 de diciembre

Consulte los horarios de cada nivel a través de nuestros canales informativos al pie de página



#### Además...

- ✓ Alquiler de laboratorios virtuales para auto-estudio o cursos empresariales
- ✓ Presentación de exámenes de certificación para múltiples áreas bajo el Centro Pearson VUE
- ✓ Cursos exclusivos para su empresa
- ✓ Pregunte por nuestros descuentos

#### CATC - Academia Regional - Academia Local

Universidad Nacional de Colombia sede Medellín  
 Calle 65 78-28 Bloque M1 Oficina 101. Facultad de Minas

Teléfono: +57 4 4255268 Fax: +57 4 2341002 E-mail: [catc@unal.edu.co](mailto:catc@unal.edu.co) Web: <http://cnap.unalmed.edu.co>

Facebook: [fb.me/catcunal](https://www.facebook.com/catcunal) Twitter: [@catcunal](https://twitter.com/catcunal) Buzz: [google.com/profiles/catcunal](https://www.google.com/profiles/catcunal)  
 Medellín, Colombia

