

UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA – UEFS

Wilson Ramos do Carmo Junior

**ANÁLISE DE MAPEAMENTOS REALIZADOS PARA RELACIONAR
METODOLOGIAS ÁGEIS E MODELOS DE QUALIDADE**

**FEIRA DE SANTANA
2010**

Wilson Ramos do Carmo Junior

**ANÁLISE DE MAPEAMENTOS REALIZADOS PARA RELACIONAR
METODOLOGIAS ÁGEIS E MODELOS DE QUALIDADE**

Trabalho de Conclusão de Curso apresentado ao curso de Graduação de Engenharia de Computação da Universidade Estadual de Feira de Santana, como requisito parcial para a obtenção do título de Engenheiro de Computação.

Orientador: Professor Mestre José Amâncio Macedo Santos

**FEIRA DE SANTANA
2010**

RESUMO

Duas linhas de estudo que visam melhorar o processo de desenvolvimento de *software* são as metodologias ágeis (MAs) e os modelos de qualidade (MQs). Diante disso, há diversos trabalhos comparando e mapeando estas duas linhas de estudo. Este trabalho insere-se no contexto descrito, fazendo uma análise sobre os MQs (MPS.BR e CMMI) e a metodologia ágil *Extreming Programing* (XP), discutindo e identificando as relações entre o mapeamento destes dois temas. O objetivo deste trabalho, portanto, é analisar os pontos de discordância que são identificados nos trabalhos de mapeamento entre a metodologia ágil XP e os modelos de qualidade.

Palavras-Chave: CMMI. *Extreme Programming*. Metodologias Ágeis. Modelos de Qualidade. MPS.BR. XP.

ABSTRACT

Two study lines to improve the process of software development are agile methodologies (MAs) and quality models (MQs). Given this, there are several studies comparing and mapping these two study lines. This work fits into the context described making an analysis of the MQs (MPS.BR and CMMI) and agile methodologies Extreme Programming (XP), discussing and identifying the relationship between the mapping of these two themes. Therefore, the aim of this study is to analyze the points of disagreement that are identified in the work of mapping between the XP and agile quality models.

Keywords: CMMI. Extreme Programming. Agile. Quality Models. MPS.BR. XP.

SUMÁRIO

1 INTRODUÇÃO	1
2 FUNDAMENTAÇÃO TEÓRICA	4
2.1 Metodologias Ágeis.....	4
2.1.1 <i>Extreme Programming</i> (XP)	5
2.2 Modelos de Qualidade.....	7
2.2.1 <i>Capability Maturity Model Integration</i> (CMMI)	8
2.2.1.1 Nível 2 do CMMI (MPS.BR).....	10
2.2.2 Melhoria de Processo de <i>Software</i> Brasileiro (MPS.BR)	12
2.2.2.1 Nível G	13
3 METODOLOGIA	14
3.1 Mapeamento entre Metodologias Ágeis e Modelos de Qualidade de <i>Software</i>	14
3.2 Padronização dos Critérios de Avaliação.....	16
3.3 Mapeamento do CMMI e XP	17
4 RESULTADOS.....	18
4.1 Análise Comparativa entre Trabalhos	18
4.2 Pontos de Concordância e Discordância	20
5 DISCUSSÃO DOS RESULTADOS.....	31
6 CONSIDERAÇÕES FINAIS	35
7 REFERÊNCIAS	36
APÊNDICE	38

1 INTRODUÇÃO

As metodologias ágeis (MAs) surgiram através da publicação do Manifesto Ágil (AGILE MANIFEST, 2001), como forma de oferecer maior agilidade e melhorar o desempenho dos processos de desenvolvimento de projetos. Neste manifesto, foram reunidos os princípios e práticas destas metodologias de desenvolvimento. Os quatro princípios fundamentais são: indivíduos e interação mais que processos e ferramentas; *software* em funcionamento mais que documentação abrangente; colaboração com o cliente mais que negociação de contratos e responder a mudanças mais que seguir um plano (AGILE MANIFEST, 2001).

É importante notar que o surgimento das MAs não significa a rejeição dos princípios das metodologias tradicionais, como o modelo cascata (usado na década de 80), e mais recente e sofisticado o *Rational Unified Process* (RUP), mas apresenta-se como alternativa para o desenvolvimento de *software*. As metodologias ágeis não apresentam conceitos novos, porém alteram a perspectiva e os valores com que esses conceitos são encarados (HIGHSMITH; COCKBURN, 2001).

Em paralelo à disseminação das metodologias ágeis, os investimentos em qualidade de *software* vêm aumentando a cada ano (CAMPELO, 2003), o que é uma evidência forte da importância deste tema. No intuito de aumentar, medir e garantir a qualidade do *software* produzido, surgiram os Modelos de Qualidade (MQs) de *software*. Eles são modelos que estabelecem critérios para se ter qualidade na produção, no fornecimento, na aquisição e operação de um programa que visam melhoria da qualidade dos processos, produtos e serviços de *software* (SANTOS, 2010). Alguns exemplos de MQs são o *Capability Maturity Model Integration* (CMMI) e Melhoria de Processo do *Software* Brasileiro (MPS.BR). CMMI é um modelo de avaliação e melhoria da maturidade de processo de *software* respeitado mundialmente. No Brasil, tem-se o modelo de qualidade MPS.BR, criado pela SOFTEX, voltado para a realidade do mercado de micro, pequenas e médias empresas de desenvolvimento de *software*, sendo baseado no CMMI (SOFTEX, 2009).

Uma vez que tanto as MAs como os modelos de qualidade tratam de aspectos relacionados à melhoria do processo de desenvolvimento de *software*, atualmente há uma série de pesquisas visando estabelecer uma relação entre estas linhas de estudo, que podem ser considerados indícios fortes da relevância do tema (MARÇAL et al., 2010; SANTOS, 2010; SUTHERLAND; JAKOBSEN; JOHNSON, 2007). É possível perceber a partir dos trabalhos

relacionando MAs e MQs que os resultados expostos em alguns deles possuem problemas e/ou são incompletos. Não são encontrados detalhes sobre como as metodologias contemplam ou não um determinado requisito (SANTOS, 2010). No trabalho de Santana (2006) e outros, por exemplo, é dito que a MA analisada satisfaz determinado requisito, mas não faz uma discussão sobre o porquê de este requisito ser satisfeito. O mesmo ocorre no trabalho de Dias (2010) e em outros trabalhos analisados.

A partir da grande quantidade de trabalhos realizados envolvendo MAs, MQs e suas relações, é possível notar que existem alguns pontos de discordância no que se diz respeito à metodologia e conclusões dos autores acerca do mapeamento de MAs e MQs. Isso acontece, pois, não há um modelo formal para realização deste mapeamento e nem para atribuir os critérios de avaliação em cada requisito analisado. Outro fator que contribui para estes pontos de discordância é que os mapeamentos são feitos através da opinião dos autores, pois, sua realização é baseada fundamentalmente em interpretação (CAMPELO, 2003).

É importante para destacar a amplitude do tema observar que existem diversas metodologias ágeis (*Extreming Programming (XP)*, *Feature Driven Development (FDD)*, *Dynamic Systems Development Method (DSDM)*, *Adaptive Software Development (ASD)*, *Lean Development (LD)*, SCRUM, entre outras) e que, apesar destas MAs possuírem um conjunto de princípios em comum, elas apresentam diferenças no que diz respeito às práticas, artefatos, valores e papéis. A implantação dessas metodologias em um ambiente de trabalho exige conhecimentos específicos sobre cada uma das MAs, além de um tempo de prática para adaptação da metodologia à realidade da empresa. Diante da diversidade de MAs existentes e das diferentes características que elas possuem, é analisada neste trabalho apenas uma metodologia. Pelo fato do XP ser a metodologia ágil mais conhecida e utilizada (CAMPELO, 2003) e pela grande quantidade de trabalhos analisando esta MA, o foco deste trabalho será na metodologia ágil XP.

No caso das MQs, a dificuldade de obtenção de uma certificação CMMI foi um dos fatores que contribuíram para o surgimento do MPS.BR. Segundo ASR Consultoria e Assessoria em Qualidade, em 2004, os Estados Unidos possuíam 1.738 empresas certificadas, na Índia eram 294 empresas enquanto no Brasil haviam apenas 17 empresas com certificação do CMMI. O alto custo para obtenção da certificação e o longo prazo para alcançar os níveis de maturidade mais altos impossibilitavam as pequenas e médias empresas de desenvolvimento de *software* de aderir ao programa do CMMI (OLIVEIRA, 2008). Com base nesta problemática, o MPS.BR tem como objetivo a disseminação da qualidade visando a competitividade da indústria brasileira de *software*, através da melhoria e avaliação de

processos e produtos de *software*, a um custo acessível às empresas de menor porte (SOFTEX, 2009).

Dados da Associação Brasileira de Empresas de Software (ABES) informam através de pesquisas que no Brasil, até o final de 2009, existiam apenas 151 empresas que possuíam certificação MPS.BR (Figura 1) e que o maior número destas certificações estão concentradas no nível inicial, o nível G (SOFTEX, 2009). Visto que a realidade das empresas brasileiras é de serem micro, pequenas e médias e que o custo de implantação de mais de um nível do MPS.BR é relativamente alto, este trabalho direcionará suas discussões para o nível G. Outro motivo para tomada desta decisão é por que os resultados dos outros níveis são uma evolução do nível G, ou estes resultados são incorporados aos fundamentos deste nível, conforme o Guia de Implementação da SOFTEX (SOFTEX, 2009). Os níveis do MPS.BR são, basicamente, as partes que compõem este modelo de qualidade e vão do nível G ao A, sendo o nível G o inicial.

Ano	A	B	C	D	E	F	G	Totais por Ano
2005	0	0	0	0	1	3	1	5
2006	2	0	0	1	1	1	7	12
2007	1	0	0	0	1	12	41	55
Total 2005 à 2007	3	0	0	1	3	16	49	72
2008	1	0	0	0	1	9	40	51
2009	0	0	0	0	1	12	15	28
2010	0	0	0	0	0	0	0	0
Total 2008 à 2010	1	0	0	0	2	21	55	79
TOTAIS	4	0	0	1	5	37	104	151

Figura 1: Adoção do MPS.BR por empresas brasileiras (SOFTEX, 2009).

Nesse contexto, o objetivo principal deste trabalho é analisar os pontos de discordância que são notados nos trabalhos de mapeamento entre metodologias ágeis e modelos de qualidade. Têm-se como objetivos específicos analisar a gama de trabalhos acerca de mapeamento entre XP e CMMI, XP e MPS.BR. Os resultados deste trabalho evidenciam os pontos em que há discordância na literatura tanto entre o mapeamento do XP e MPS.BR quanto do XP e CMMI. Como contribuições específicas desta monografia, pode-se destacar:

- ✓ Esquematizar os vários trabalhos que realizam o mapeamento entre XP e MPS.BR, XP e CMMI.

- ✓ Comparar as diferenças e concordâncias, obtidas através do mapeamento.
- ✓ Analisar os aspectos relevantes que levaram a existir as diferenças e concordâncias.
- ✓ Auxiliar profissionais desta área de interesse na identificação dos pontos de discordância nos mapeamentos do nível G do MPS.BR com XP.

A estrutura deste trabalho é dividida em Fundamentação Teórica (Seção 2), parte do texto que contém a exposição teórica do assunto; Metodologia (Seção 3), mostrando os métodos utilizados neste trabalho; Resultados (Seção 4), parte do texto na qual se apresentam as conclusões correspondentes aos objetivos do trabalho; Discussão dos Resultados (Seção 5), mostrando os pontos relevantes sobre as diferenças e concordâncias encontradas e discutindo tais pontos; Considerações Finais (Seção 6); Referencias Bibliográficas (Seção 7) e Apêndice.

2 FUNDAMENTAÇÃO TEÓRICA

A seguir serão apresentados os conceitos fundamentais utilizados neste trabalho.

2.1 Metodologias Ágeis

A popularização dos métodos ágeis ocorreu com a criação do Manifesto para Desenvolvimento Ágil de *Software* (AGILE MANIFESTO, 2001), quando um grupo de pesquisadores reuniu-se com o objetivo de uniformizar os diversos métodos existentes para o desenvolvimento de *software* (BECK, 2004). Um ponto importante do manifesto é a definição de uma nova abordagem para o desenvolvimento de *software*, focando em agilidade, flexibilidade e na habilidade de se comunicar e de entregar novos produtos e serviços, em um tempo definido. Alguns princípios e valores comuns que são compartilhados por todos os métodos ágeis foram estabelecidos:

- ✓ Indivíduos e a interação são mais importantes do que processos e ferramentas;
- ✓ *Software* operacional é mais importante que documentação detalhada;
- ✓ A colaboração dos clientes é mais importante que negociação do contrato;

- ✓ Mudanças e adaptação são mais importantes do que seguir um planejamento.

O Manifesto Ágil ressalta o que mais tem valor para as metodologias ágeis. Processos, documentação, contratos, e planejamento são menos importantes do que saber como lidar com pessoas, do que ter o cliente colaborando para encontrar a melhor solução, do que entregar o *software* com qualidade e do que se adaptar às mudanças (DIAS *et al.*, 2010).

Para tornar o desenvolvimento de *software* mais rápido, foi necessário que os métodos ágeis modificassem as etapas do processo e a forma como os envolvidos com o desenvolvimento realizavam suas atividades.

As MAs representam uma mudança de paradigma de desenvolvimento de *software* (BECK, 2004). Os princípios das MAs são evidentes em muitas metodologias, como por exemplo, no *Extreme Programming* (XP). Esta metodologia possui técnicas diversificadas e agrupa muitas recomendações que contribuem para uma maior qualidade de *software* (BECK, 2004). Estas recomendações incluem o desenvolvimento de *software* iterativo, verificação contínua, necessidades de interação do cliente com desenvolvedores e projeto de arquitetura adaptativa.

2.1.1 *Extreme Programming* (XP)

Extreme Programming (XP) foi desenvolvido para atender as necessidades específicas de desenvolvimento de *software* conduzido por equipes pequenas diante da evolução das necessidades. Uma das mais conhecidas metodologias ágeis de desenvolvimento de *software* é o XP. Ele foi criado nos Estados Unidos, na década de 90, por Kent Beck. Esta metodologia ágil possui um sistema de práticas que a comunidade de desenvolvedores de *software* vem evoluindo para resolver os problemas de entregar programas de qualidade em um curto espaço de tempo (BECK, 1999).

Os Fundamentos do XP incluem a escrita de testes de unidade antes de programar e manter todos os testes de corrida em todos os momentos; integração e testes de todo o sistema várias vezes por dia; projetos começando com um design simples que evolui constantemente para aumentar a flexibilidade necessária e remover complexidade desnecessária; colocar um sistema mínimo em produção rapidamente e priorizar as funcionalidades (BECK, 1999).

Algumas das práticas adotadas pelo XP são (BECK, 1999):

- ✓ **Jogo do planejamento:** Deve ser elaborado de forma simples, fácil e rápido um plano para o próximo *release* (planejamento das versões), determinando o escopo baseado nas prioridades do negócio, nas possibilidades de implementação e estimativas técnicas.
- ✓ **Releases Pequenos:** Os *releases* devem ser de pouca duração.
- ✓ **Metáforas:** Guiam o desenvolvimento do projeto através de uma história que explique como o sistema funciona.
- ✓ **Testes:** São divididos em testes unitários e testes de aceitação. Os testes unitários são feitos pelo programador durante o desenvolvimento. Os testes de aceitação são especificados pelo cliente ditando o que deve funcionar para que o *software* seja aceito.
- ✓ **Cliente no Local:** O cliente é parte integrante da equipe e deverá estar presente no local de desenvolvimento sempre que necessário para tirar alguma dúvida ou priorizar alguma história.
- ✓ **Refatoração:** É uma técnica para reorganizar um código, alterando sua estrutura interna sem modificar o comportamento externo. Podem ser simples mudanças de nomes de variáveis, métodos, classes para tornar o código mais legível, remoção de código duplicado ou uma simplificação da arquitetura do sistema.
- ✓ **Programação em Pares:** O desenvolvimento de tarefas é feito em dupla a fim de promover o trabalho coletivo e colaborativo. As duplas são trocadas frequentemente para possibilitar interação, aumentar a comunicação e união do grupo. Esta prática permite reduzir a quantidade de erros na hora de programar, já que existem duas pessoas analisando o mesmo código. Enquanto um dos programadores digita, o outro revisa o código e sugere melhorias. Com o tempo, o trabalho em duplas promove o compartilhamento de conhecimentos específicos de cada participante para vários membros da equipe.
- ✓ **Test-Driven Development (TDD):** É uma prática de desenvolvimento utilizada pelo XP em que a ordem de desenvolvimento de *software* é iniciada pelo teste, para depois fazer um código funcional em que este teste seja satisfeito, e por fim, feita uma refatoração em cima do código, para então recomeçar o ciclo. Este método fornece confiança ao programador, pois ele terá certeza que seu código foi totalmente testado, o que resultará em economia de tempo futuramente.

O *Extreme Programming* também define alguns papéis, assumidos pela equipe do projeto (BECK, 1999):

- ✓ **Programador:** Responsável por projetar o *software*, codificar, programar testes e estimar suas tarefas. Todos da equipe assumem este papel no desenvolvimento do projeto;
- ✓ **Cliente:** Representante do cliente, responsável por estabelecer prioridades, escopo, escrever histórias e escrever os testes funcionais. Deve estar disponível no local do desenvolvimento sempre que necessário para esclarecer as dúvidas;
- ✓ **Técnico:** Responsável por identificar problemas e resolvê-los para que a equipe possa trabalhar da melhor forma. Não requer conhecimentos técnicos profundos.
- ✓ **Testador:** O testador é responsável por apoiar o cliente a escolher e escrever os testes funcionais, além de assegurar a execução e reportagem dos problemas identificados nestes testes.
- ✓ **Arquitetos:** analisam a estrutura do sistema para identificar melhorias de grande impacto;
- ✓ **Gerentes de Projeto (Coach):** promovem a comunicação e removem os impedimentos que dificultam o trabalho dos programadores;
- ✓ **Usuários:** contribuem com opiniões, sugestões e críticas para ajustes e novas histórias.

Espera-se que ao longo do tempo essas especialidades sejam disseminadas entre os envolvidos do projeto para evitar a concentração de conhecimento e estimular o crescimento profissional de todos os participantes (BECK, 1999).

2.2 Modelos de Qualidade

Os modelos de qualidade (MQs) são modelos que estabelecem critérios para se ter qualidade na produção, no fornecimento, na aquisição e operação de um *software* (SOFTEX, 2009). Eles visam melhoria da qualidade dos processos, produtos e serviços de *software*. O termo MQ em geral está relacionado a uma série de aspectos tais como normalização e melhoria de processo, medições, padrões e verificações. Apesar dos modelos aplicados na garantia da qualidade de *software* atuarem principalmente no processo, o principal objetivo é

garantir um produto final que satisfaça às expectativas do cliente, dentro daquilo que foi proposto inicialmente.

Os MQs não são uma forma de assegurar qualidade total na entrega do produto desenvolvido, mas sim propõem o acompanhamento da qualidade de cada etapa realizada, garantindo o controle e buscando diminuir ao máximo os riscos de falhas.

Alguns dos principais modelos de qualidade são o CMMI e MPS.BR. Estes modelos objetivam a melhoria da qualidade de produção nas empresas. Para tanto, estas empresas precisam obter suas certificações no modelo desejado, atendendo a uma série de requisitos propostos por cada um destes modelos. Só no nível inicial do MPS.BR, por exemplo, existem 17 requisitos para serem atendidos. As empresas enfrentam dificuldades em obter estas certificações por conta da grande quantidade de requisitos presentes nestes modelos e por conta do custo relativamente alto para sua aderência na empresa. A respeito do CMMI, o custo geralmente fica entre R\$200 mil a R\$1 milhão a depender da complexidade do processo. Além disso, é necessário investir tempo, para se chegar aos níveis de maturidade mais altos, levando de 4 a 8 anos para se certificar (OLIVEIRA, 2008). Essas dificuldades contrastam com a realidade das empresas brasileiras que não podem realizar um investimento tão alto na obtenção da certificação.

2.2.1 *Capability Maturity Model Integration (CMMI)*

O CMMI é um modelo desenvolvido pelo *Software Engineering Institute da Carnegie-Mellon University* que propõe um maior controle de qualidade focado em desenvolvimento de *software*. Seu objetivo é auxiliar o gerenciamento e mudança de processo, fornecer uma estrutura básica para métodos confiáveis e coerentes de avaliação de organizações de *software*, bem como auxiliar a melhoria do processo interno de *software* (SOFTWARE ENGINEERING INSTITUTE, 2009).

O foco principal do modelo CMMI é melhoria dos processos envolvidos no desenvolvimento de projeto tais como análise, planejamento e controle do projeto. CMMI traz uma abordagem de melhoria de processo que fornece às organizações os elementos essenciais de processos efetivos. O CMMI ajuda a integrar funções organizacionais separadas, definir metas e prioridades de melhoria de processos e ajuda a fornecer um ponto de referência para avaliar os processos atuais (SOFTWARE ENGINEERING INSTITUTE, 2009). *Capability*

Maturity Model Integration na sua essência não é uma metodologia, mas sim um conjunto de princípios. Este conjunto de princípios centra-se na maturação de um processo de desenvolvimento de *software*. Na Figura 2, estão representados os níveis de maturidade do CMMI.

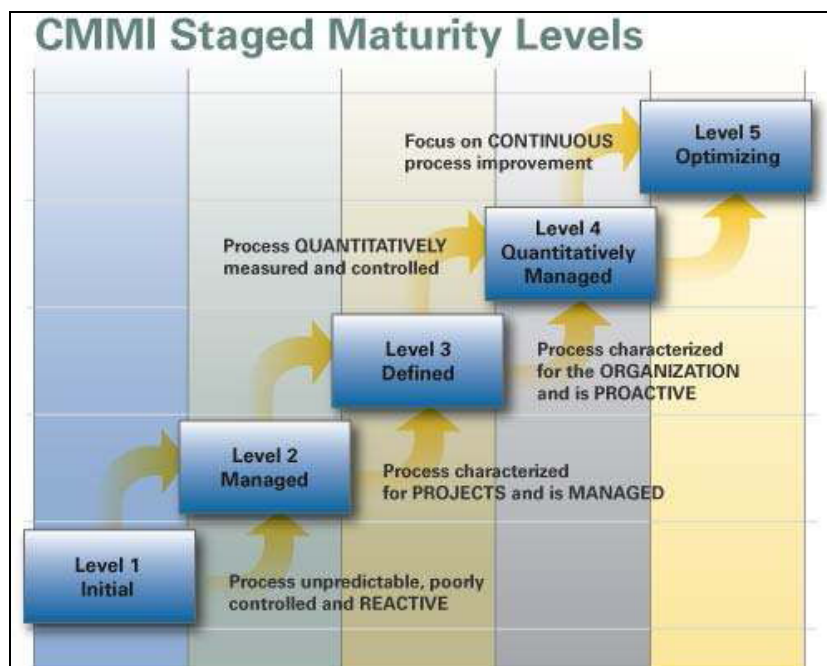


Figura 2: Níveis do CMMI (ELIAS, 2008).

O CMMI é composto de 5 níveis, conforme a Figura 2, em que o nível 1 é o mais baixo (SOFTWARE ENGINEERING INSTITUTE, 2009):

- ✓ **Nível 1: *Initial*** (Inicial) – as organizações que possuem um processo imprevisível e sem controle e estão aptas a repetir os processos passados estão no nível 1 de maturação.
- ✓ **Nível 2: *Managed*** (Gerenciado) – organizações tem um processo de gerenciamento de projeto para acompanhar o cronograma e custo, sendo capazes de gerenciar um projeto. Este nível inclui *software* de gestão de configuração e garantia de qualidade de *software*.
- ✓ **Nível 3: *Definited*** (Definido) – organizações orientadas à processo estão neste nível de maturação. Este nível engloba normas, descrição de processos e procedimentos em

vigor e inclui *software* de gestão integrada, engenharia de *software* produto, programas de treinamento e orientação de processos organizacionais.

- ✓ **Nível 4:** *Quantitatively Managed* (Qualitativamente Gerenciado) – empresas neste nível gerem o processo com métricas quantitativas através do tempo. Este nível engloba gerenciamento de processos quantitativos e de gestão da qualidade do *software*.
- ✓ **Nível 5:** *Optimizing* (Otimização) – este é o nível de maturidade mais alto definido pelo CMMI e engloba gerenciamento de mudanças de tecnologia, processo e gestão de mudança. Organizações de nível 5 controlam e avaliam o processo quantitativamente, podendo intervir em sua especificação para otimizá-lo continuamente.

O objetivo principal nas transições através desses níveis de maturidade é a realização de um processo controlado e mensurado que tem como fundamento a melhoria contínua. Cada nível de maturidade corresponde a um conjunto de práticas de *software* e de gestão específicas, denominadas *Key Process Areas* (Áreas-Chave do Processo – KPAs). As KPAs possuem metas que devem ser implantadas para que a organização possa atingir o nível de maturidade desejado. Estas metas descrevem as características da KPA com maior nível de especificidade.

Será abordado com um pouco mais de detalhes o nível 2 do CMMI, pois a argumentação realizada nos resultados deste trabalho abrange características das KPAs.

2.2.1.1 Nível 2 do CMMI (MPS.BR)

O nível 2 do CMMI envolve a gestão de projetos, controle de cronograma e custos, gestão de configuração, garantia de qualidade de *software* e gestão de subcontrato. Estas técnicas de gestão de projetos são estabelecidas para mapear custos, prazos, e funcionalidades do sistema. O nível 2 do CMMI é dividido em 5 *Key Process Areas* (KPAs): *Requirement Management*, *Software project planing*, *Software project tracking and oversight*, *Software subcontract management*, *Software quality assurance* e *Software congiguration management* (SOFTWARE ENGINEERING INSTITUTE, 2009), descritas a seguir:

- ✓ **Requeriment Management** – Significa Gerenciamento de Requisitos e tem por objetivo estabelecer um entendimento comum entre o cliente e o projeto de *software* a respeito dos seus requisitos, os quais serão abordados pelo projeto de *software*.
 - Meta-1: Os requisitos do sistema que são alocados para o *software* são controlados para estabelecer uma base para a engenharia de *software* e de gestão do uso.
 - Meta-2: Os planos de *software*, produtos e atividades são mantidos consistentes com os requisitos do sistema alocados ao *software*. O sistema entregue deve atender aos requisitos de usuário.
- ✓ **Software project planing** – Significa Planejamento de Projetos e tem por objetivo estabelecer planos razoáveis para executar a engenharia de software e para gerenciar o projeto.
 - Meta-1: Estimativas de *software* são documentadas para uso no planejamento e acompanhamento do projeto.
 - Meta-2: Atividades de projetos de *software* e os compromissos são planejados e documentados.
- ✓ **Software project tracking and oversight** – Significa Acompanhamento e Supervisão de Projeto e tem por objetivo estabelecer uma visibilidade adequada do progresso real, de modo que a gerência possa tomar ações efetivas quando o desempenho do projeto de software se desvia significativamente do planejamento de projeto.
 - Meta-1: Os resultados reais e os desempenhos são monitorados.
 - Meta-2: Alterações aos compromissos de software são decididos por toda equipe.
- ✓ **Software subcontract management** – Significa Gerenciamento de Subcontratação e tem por objetivo selecionar subcontratados qualificados de *software* e gerenciá-los efetivamente.
- ✓ **Software quality assurance** – Significa Garantia de Qualidade de Software e tem por objetivo fornecer uma gestão com visibilidade apropriada sobre o processo que está sendo utilizado pelo projeto de *software* e dos produtos que estão sendo construídos.

- ✓ **Software configuration management** – Significa Gerenciamento de Configuração e tem por objetivo estabelecer e manter a integridade dos produtos do projeto de *software* ao longo do ciclo de vida do projeto.
 - Meta-1: Atividades de gestão de configuração de *software* são planejadas.
 - Meta-2: Produtos de software selecionados são identificados, controlados e disponibilizados.
 - Meta 3: As alterações nos produtos de *software* são controladas.

2.2.2 Melhoria de Processo de *Software* Brasileiro (MPS.BR)

O MPS.BR é um modelo de referência coordenado pela Associação para a Promoção da Excelência do *Software* Brasileiro (SOFTEX), que tem por objetivo atingir maior qualidade no processo de desenvolvimento de *software*, visando a competitividade da indústria brasileira de *software* principalmente em organizações com recursos limitados para investimento em qualidade (SOFTEX, 2009). Este modelo não define novos conceitos, apenas adéqua as estratégias de implementação à realidade brasileira.

O MPS.BR pode ser dividido em guias: Guia Geral, que consiste de uma descrição geral e detalha o modelo de referência MR-MPS; Guia de Aquisição, o qual descreve o processo de aquisição do *software* e os serviços correspondentes; Guia de Avaliação, que abrange o processo e os requisitos para avaliação; e Guia de Implementação, composto por sete partes (Figura 3), uma para cada nível de maturidade, que indicam a evolução dos processos na organização do MR-MPS.

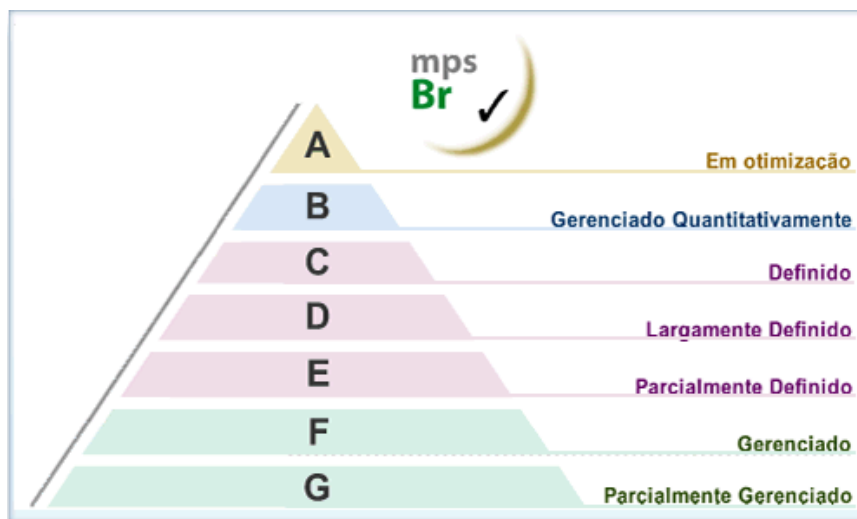


Figura 3: Níveis de Maturidade do MPS.BR (QUALIDADE BR, 2008).

Na Figura 3, estão listados os sete níveis de maturidade de *software* do MPS.BR: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Parcialmente Gerenciado). Estes níveis iniciam no nível G e progredem até o A. Ainda na Figura 3, pode-se observar a subdivisão dos processos de cada nível. Cada processo é subdividido em requisitos que juntos contemplam o processo.

Como o foco deste trabalho é no nível G, este será abordado com um pouco mais de detalhes na subseção 2.2.2.1.

2.2.2.1 Nível G

O nível de maturidade G é composto pelos processos Gerência de Requisitos e Gerência de Projetos.

O propósito do processo Gerência de Requisitos (GRE) é gerenciar os requisitos do produto e dos componentes do produto do projeto e identificar inconsistências entre os requisitos, os planos do projeto e os produtos de trabalho do projeto.

O propósito do processo Gerência de Projetos (GPR) é estabelecer e manter planos que definem as atividades, recursos e responsabilidades do projeto, bem como prover informações sobre o andamento do projeto que permitam a realização de correções quando

houver desvios significativos no desempenho do projeto. O propósito deste processo evolui a medida em que a organização cresce em maturidade (níveis). O conjunto de resultados (requisitos) esperados para o processo GPR do nível G do MPS.BR estão descritos no Apêndice A.

3 METODOLOGIA

A metodologia deste trabalho foi definida em algumas etapas. A primeira delas foi a realização do estudo bibliográfico. Foram estudados os modelos de qualidade de *software* CMMI e MPS.BR. A MA XP também foi abordada neste levantamento bibliográfico, com objetivo de fornecer os elementos necessários para a análise feita neste trabalho. A segunda etapa foi o levantamento de trabalhos de mapeamento. Uma série de trabalhos que propõem o relacionamento entre estes dois temas (MQ e MA) também foram analisados. Eles serviram de base para analisar como é feito o mapeamento das MAs e os modelos de qualidade de *software*. A terceira etapa foi a padronização dos critérios de avaliação dos trabalhos levantados na etapa dois, descrita na seção 3.2. Por fim, foi analisado o mapeamento do CMMI e XP a fim de reforçar a argumentação acerca do foco principal deste trabalho, que é a análise do GPR do MPS.BR e o XP. Esta etapa será descrita na seção 3.3.

3.1 Mapeamento entre Metodologias Ágeis e Modelos de Qualidade de *Software*

O mapeamento entre as MQs e MAs é feito basicamente através da interpretação das características do requisito em questão, e a partir dos conhecimentos teóricos/práticos da MA, é verificado se realmente existe uma satisfação do requisito analisado pela metodologia ágil em questão, utilizando como base os fundamentos destas duas linhas de estudo.

Embora este mapeamento na teoria seja realizado de uma maneira uniforme, os trabalhos de mapeamento divergem na forma como eles apresentam as conclusões de seus mapeamentos. Mesmo que mais de um autor conclua que determinado requisito seja satisfeito pela MA, alguns deles expõem esta conclusão e não a justifica. Dois exemplos serão dados para deixar claro esta linha de raciocínio. No trabalho de Santos (2010), ele faz o mapeamento do nível G do MPS.BR

e do XP. Tomando-se como exemplo o requisito GPR14 do MPS.BR (esse é o 14º requisito do processo de Gerenciamento de Projeto – GPR), o autor faz o mapeamento da seguinte forma: Primeiramente, o autor mostra a descrição do requisito GPR14:

- ✓ **GPR14 - O envolvimento das partes interessadas no projeto é gerenciado:** Este requisito relata, entre outras coisas, que os interessados no projeto podem incluir o cliente e o usuário (ou seus representantes), a direção da organização e os membros da equipe do projeto.

Feito isso, o autor estabelece o mapeamento entre o requisito em questão com a MA XP, verificando se esta atende ao determinado requisito. Abaixo está o mapeamento realizado pelo autor.

O XP possui como uma de suas práticas manter o cliente sempre presente no ambiente de desenvolvimento. A interação com o cliente deve ser intensa. Na fase inicial e ao longo do projeto, o cliente escreve as histórias de usuários e nas reuniões de planejamento de *releases* ele negocia com a equipe de desenvolvimento, priorizando as histórias a serem desenvolvidas. No XP, o cliente é convidado a participar das reuniões diárias, acompanhando o progresso e fortalecendo o envolvimento com o projeto, ou seja, em projetos XP o envolvimento dos interessados no projeto é uma de suas principais características. Desta forma o XP **atende este requisito, logo, ele é Satisfeito.** (SANTOS, 2010, p. 52, grifo do autor).

Santos (2010) faz uma análise comparativa acima e em negrito expõe sua conclusão, mostrando que o mapeamento do requisito GPR14 do MPS.BR é satisfeito pela MA XP.

Outro exemplo de como é feito este mapeamento será mostrado com base no trabalho de Santana (2006). Tomando-se o GPR14 como exemplo novamente, o autor expõe a descrição do requisito da mesma forma que Santos (2010). Porém, na hora de expor suas conclusões, ele apenas afirma que não foram identificados problemas com este requisito no mapeamento com o XP, concluindo, portanto, que esta prática é satisfeita completamente pelo XP. Este autor não apresenta argumento algum sobre o porquê deste requisito ser Satisfeito.

Além das diferenças de mapeamento referentes à interpretação, e a forma de exposição dos resultados, outra variação ocorre é quanto aos critérios de classificação adotados. De acordo com Santos (2010), seus critérios são três: Satisfeito, Parcialmente Satisfeito e Não Satisfeito (Figura 4), sendo que alguns autores utilizam cinco (ARIMOTO et al., 2009) e outros apenas dois critérios (Satisfaz ou Não Satisfaz) (SILVA et al., 2009).

Classificação	Critério
(S) Satisfeito	A prática está totalmente em conformidade
(PS) Parcialmente Satisfeito	Há evidências, embora não esteja plenamente em conformidade.
(NS) Não Satisfeito	Não há evidências

Figura 4: Critérios de classificação (SANTOS, 2010).

3.2 Padronização dos Critérios de Avaliação

Os critérios de avaliação são parâmetros utilizados pelos autores pra definir o nível de satisfação de cada requisito das MQs em relação às características da MA. Dos trabalhos estudados, foi percebido que a maioria dos autores utiliza três critérios de avaliação: Satisfeito, Parcialmente Satisfeito e Não Satisfeito, conforme a Figura 4. Como neste trabalho são comparados os trabalhos de mapeamento, foi necessário adotar um padrão para comparar os trabalhos com diferentes quantidades de critérios de avaliação. A inconsistência encontrada foi ao comparar os trabalhos de Santana (2006), Dias (2010) e Santos (2010) com o trabalho de Arimoto (2009). O autor deste artigo utiliza cinco critérios de avaliação enquanto os outros utilizam apenas três. A partir disto, e com objetivo de se fazer uma comparação mais consistente, foi realizado um mapeamento nos critérios de avaliação do Arimoto (2009) de acordo com a Tabela 1.

Tabela 1: Resultado do mapeamento dos trabalhos que utilizam 3 e 5 critérios de avaliação.

3 Critérios de Avaliação	5 Critérios de Avaliação
Não Satisfaz	Não Satisfaz
	<i>Pouco Satisfaz</i>
<i>Satisfaz Parcialmente</i>	<i>Satisfaz Parcialmente</i>
	<i>Muito Satisfaz</i>
Satisfaz (Totalmente)	Satisfaz (Totalmente)

Fonte: Próprio autor.

Conforme a Tabela 1, percebe-se que os critérios Muito Satisfaz e Pouco Satisfaz do autor Arimoto (2009), vistos na segunda coluna, equivalem ao Satisfaz Parcialmente dos outros autores. Este mapeamento foi feito de forma a não comprometer a consistência das análises que serão realizadas. Quando o autor atribuiu o critério Muito Satisfaz, foi possível perceber que em algum momento de sua análise, ele encontrou através dos fundamentos teóricos alguma evidência que impede de que esta MA seja totalmente satisfeita. Para o critério Pouco Satisfaz, a mesma conclusão foi adotada. Se uma MA Pouco Satisfaz um requisito, então existe alguma evidência na literatura que faz esta MA Satisfazer Parcialmente determinado requisito, mesmo que pouco, mas satisfaz em parte. Com base nesta conclusão e no estudo dos critérios de classificação adotados por Arimoto (2009), foi realizado o mapeamento evidenciado na Tabela 1, em que os critérios Pouco Satisfaz e Muito Satisfaz foram mapeados para Satisfaz Parcialmente.

3.3 Mapeamento do CMMI e XP

O foco principal deste trabalho é fazer uma análise sobre os pontos de discordância no Mapeamento do MPS.BR nível G com a MA XP. Ao fazer o levantamento dos trabalhos para verificar estes pontos, foram estudados também trabalhos que fazem o mapeamento do CMMI com o XP. Esta análise foi realizada com o objetivo de reforçar a comparação feita entre os trabalhos mapeamento do MPS.BR nível G com o XP. Isso é possível, pois existe uma equivalência entre os níveis do MPS.BR e CMMI (Figura 5). Portanto, ao se fazer na seção 3 as comparações entre MPS.BR e XP, será utilizado o mapeamento entre CMMI e XP para enfatizar o nível de satisfação obtido pelo mapeamento do GPR do MPS.BR com o XP.

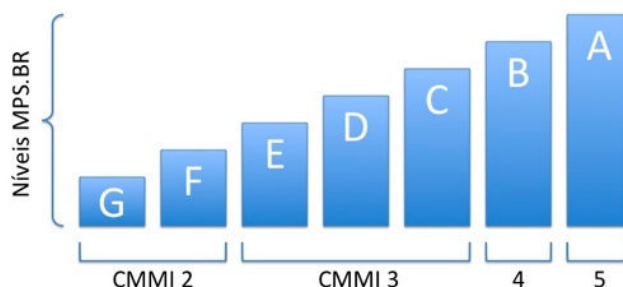


Figura 5: Mapeamento entre os níveis de maturidade do MR.MPS e CMMI (SOFTEX, 2009).

4 RESULTADOS

A seguir serão apresentadas as comparações realizadas entre os trabalhos de mapeamento e discutidos os pontos de concordância e discordância encontrados.

4.1 Análise Comparativa entre Trabalhos

Nesta seção, será feita a comparação entre os trabalhos de mapeamento tanto do XP e MPS.BR quanto do XP e CMMI. A partir das análises realizadas nos trabalhos de mapeamento entre o XP e as MQs MPS.BR e CMMI, foram encontradas inconsistências ao comparar alguns trabalhos. A Tabela 2 evidencia estas inconsistências (no Apêndice A está a descrição de todos os GPR). Para construção da Tabela 2, foram analisados cada um dos trabalhos de mapeamento, e identificada a conclusão de cada autor a respeito de todos os requisitos da Gerência de Projeto (GPR) do MPS.BR mapeados com XP. A partir desta conclusão, é possível saber quais requisitos do MPS.BR são Satisfeitos (S), Satisfeitos Parcialmente (SP) ou Não Satisfeitos (NS) pela MA XP. A obtenção dos resultados dos autores nestes mapeamentos é feita conforme mostrada na Seção 3.1 da Metodologia.

Tabela 2: Resultado do mapeamento do GPR do MPS.BR com XP de 4 trabalhos.

Requisitos	Trabalhos			
	(SANTANA et al., 2006)	(DIAS et al., 2010)	(SANTOS, 2010)	(ARIMOTO et al., 2009)
GPR 1	Satisfaz	Satisfaz	SP	Satisfaz
GPR 2	Satisfaz	Satisfaz	SP	Satisfaz
GPR 3	Satisfaz	Satisfaz	Satisfaz	Satisfaz
GPR 4	SP	Satisfaz	Satisfaz	SP
GPR 5	SP	Não Satisfaz	SP	SP
GPR 6	SP	Não Satisfaz	SP	SP
GPR 7	SP	Não Satisfaz	SP	Satisfaz
GPR 8	Não Satisfaz	Não Satisfaz	SP	Satisfaz
GPR 9	Satisfaz	Satisfaz	Satisfaz	Satisfaz
GPR 10	Satisfaz	Satisfaz	Não Satisfaz	Satisfaz
GPR 11	Satisfaz	Satisfaz	Satisfaz	Satisfaz
GPR 12	Satisfaz	Satisfaz	SP	Satisfaz
GPR 13	Satisfaz	Satisfaz	SP	Satisfaz
GPR 14	Satisfaz	Satisfaz	Satisfaz	Satisfaz
GPR 15	Satisfaz	Satisfaz	Satisfaz	Satisfaz
GPR 16	-	Satisfaz	SP	SP
GPR 17	-	Satisfaz	SP	Satisfaz

Fonte: Próprio autor.

A Tabela 2 mostra a análise comparativa de quatro trabalhos. As colunas da tabela representam os trabalhos e as linhas os requisitos GPR do MPS.BR. Observando a Tabela 2, é possível notar que existem alguns pontos de discordância nos resultados dos autores. Observa-se, por exemplo, que no trabalho de Santana (2006), o GPR8 é Não Satisfeito, enquanto no trabalho de Arimoto (2009) ele é Satisfeito. Analisando-se agora o GPR10 do trabalho de Santos (2010), percebe-se que ele é Não Satisfeito, porém nos outros trabalhos analisados este requisito é Satisfeito.

O GPR16 e o GPR17 não foram contemplados na avaliação de Santana (2006), como mostrado na Tabela 2.

Os pontos de discordância são encontrados também nos trabalhos que fazem o mapeamento de XP e CMMI. A Tabela 3 faz uma comparação de 3 trabalhos que mapeiam CMMI com XP, e é possível notar que existem discordâncias. O termo N/A (*Not Applicable*) na Tabela 3 significa que este requisito não se aplica na metodologia ágil XP. O “-“ indica que o autor não tem uma conclusão concreta sobre se a KPA é Satisfeita, Satisfeita Parcialmente ou Não Satisfeita.

Tabela 3: Resultado do mapeamento do CMMI com XP de 3 trabalhos.

Nível	Key Process Area (KPA)	Trabalhos		
		(PAULK, 2001)	(MARTINSSON, 2002)	(KOCH, 2003)
2	Requeriment Management	Satisfaz	Satisfaz	Satisfaz
2	Software project planing	Satisfaz	Satisfaz	SP
2	Software project tracking and oversight	Satisfaz	-	SP
2	Software subcontract management	Não Satisfaz	N/A	N/A
2	Software quality assurance	SP	SP	Satisfaz
2	Software congiguration management	SP	-	SP
3	Organiation process focus	SP	-	-
3	Organization process defintion	SP	Satisfaz	N/A
3	Traning program	Não Satisfaz	SP	Satisfaz
3	Integrated software management	Não Satisfaz	Não Satisfaz	Satisfaz
3	Software produtct engineering	SP	Satisfaz	Satisfaz
3	Intergroup coordination	SP	Satisfaz	SP
3	Peer reviews	SP	Satisfaz	Satisfaz
4	Quantitative process management	Não Satisfaz	-	SP
4	Software quality management	Não Satisfaz	Não Satisfaz	SP
5	Defect prevention	SP	-	SP
5	Technology change management	Não Satisfaz	Não Satisfaz	N/A
5	Process change management	Não Satisfaz	Não Satisfaz	N/A

Fonte: Próprio autor.

Da mesma forma que foi feito na Tabela 2, é possível notar diferenças de opinião nas conclusões dos autores sobre algumas KPAs na Tabela 3. Na KPA *Organization process defintion* do nível 3, por exemplo, o autor Paulk (2001) conclui que esta KPA é Satisfeita Parcialmente pelo XP, Martinsson (2002) conclui que é Satisfeita e Koch (2003) conclui que esta KPA não se aplica ao XP.

Diante dos pontos de discordância encontrados na Tabela 2, será feita uma análise de cada um destes pontos, mostrando os argumentos dos autores para chegar à conclusão que levou a tal discordância. A análise da Tabela 3 será utilizada conforme explicitada na Seção 3.2 da Metodologia, como forma de reforçar a argumentação a respeito dos pontos de discordância entre o nível G do MPS.BR e o XP.

4.2 Pontos de Concordância e Discordância

Nesta seção, serão analisadas todas as diferenças e concordâncias que foram encontradas na comparação dos 4 trabalhos, evidenciados na Tabela 2. Esta análise ocorrerá da seguinte forma: para cada requisito do GPR, será mostrado se há ou não uma discordância. Nos dois casos será discutida esta conclusão. Abaixo está a estrutura de como serão expostas as análises das diferenças e concordâncias nesta seção:

Nível: Nome do nível que está sendo analisado. Seguirá de sua descrição em casos de discordância para facilitar o entendimento do leitor.

Trabalho(s) onde se encontra a discordância: Será indicado em qual dos trabalhos da Tabela 2 existe uma discordância. Em caso de concordância, haverá aqui a ocorrência da palavra “Nenhum”.

Discordância (s): Será indicado aqui o critério de classificação do autor que apresenta a discordância e o critério de classificação dos outros autores.

Argumento(s) do(s) autor(es): Argumento utilizado pelo autor para atribuir o critério de classificação que gerou a discordância e argumentos dos demais autores defendendo suas conclusões.

Considerações: Discussão sobre a análise realizada.

Esta estrutura será utilizada para mostrar as diferenças e concordâncias nos 17 requisitos do GPR do MPS.BR, e para discutir os resultados.

Nível: GPR1 - O escopo do trabalho para o projeto está definido.

Trabalho(s) onde se encontra a discordância: Santos 2010

Discordância: Autor: Satisfaz Parcialmente / Outros autores: Satisfaz

Argumento do autor:

Santos (2010) afirma que o XP não define o escopo de todo o projeto no início porque o escopo do projeto é dividido através das *releases*, construídas no Jogo do Planejamento, e estas contêm inicialmente apenas as funcionalidades prioritárias para o cliente.

Considerações: Neste GPR um autor afirma que Satisfaz Parcialmente e argumenta sobre sua decisão. Os outros autores constatam que o GPR1 é Satisfeito pelo XP, porém, não apresentam suas argumentações. Mapeando-se o nível 2 do CMMI com o GPR1 do MPS.BR, é possível notar que o GPR1 está contido na KPA *Requirement Management*. Esta KPA estabelece o escopo através do projeto de requisitos de *software*, que deve ser claro para o cliente (SOFTWARE ENGINEERING INSTITUTE, 2009). Analisando-se esta KPA na Tabela 3, percebe-se que há uma satisfação geral por conta dos 3 autores. Observando-se as conclusões dos autores da Tabela 2 e 3 para este requisito, pode-se constatar que há uma evidência forte de que a GPR1 tende a ser satisfeita pela MA XP.

Nível: GPR2 - O escopo, os produtos de trabalho e as tarefas do projeto são estimados através de métodos apropriados.

Trabalho(s) onde se encontra a discordância: Santos 2010

Discordância: Autor: Satisfaz Parcialmente / Outros autores: Satisfaz

Argumento do autor:

Santos (2010) afirma que o XP Satisfaz Parcialmente a GPR2 devido à falta de formalismo e base histórica. Segundo o autor, o XP não define claramente o método de

decomposição de histórias em tarefas, que ocorre baseada em experiências anteriores. E estes dados históricos sobre a decomposição de requisitos não é mantida.

Considerações: Neste GPR um autor afirma que Satisfaz Parcialmente e argumenta sobre sua decisão. Os outros autores constatam que o GPR2 é Satisfeito pelo XP, porém, não apresentam suas argumentações. Mapeando-se o nível 2 do CMMI com o GPR2 do MPS.BR, percebe-se uma aproximação do GPR2 com a KPA *Software project planing*, pois uma das metas deste KPA é documentar as estimativas de *software* para serem utilizadas no planejamento e acompanhamento do projeto (SOFTWARE ENGINEERING INSTITUTE, 2009). Analisando-se esta KPA na Tabela 3, percebe que há concordância por parte de dois autores (Satisfaz) e um autor conclui que o XP Satisfaz Parcialmente esta KPA. A quantidade de satisfações obtidas pela maioria dos autores tanta da Tabela 2 quanto da Tabela 3 pode ser uma evidência de que a GPR2 seja Satisfeito.

Nível: GPR3.

Trabalho(s) onde se encontra a discordância: Nenhum

Discordância: Não há discordância.

Argumento dos autores:

Todos concordam que as fases do ciclo de vida no XP são definidas.

Considerações: Nesta GPR, todos os autores chegam à conclusão de satisfação do XP, porém apenas Santos (2010) apresenta argumentação a respeito de sua conclusão.

Nível: GPR4 - A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada.

Trabalho(s) onde se encontra a discordância: Santana 2006

Discordância: Autor: Satisfaz Parcialmente / Outros autores: Satisfaz

Argumento dos autores:

Segundo Santana (2006), não há nenhum controle dos recursos utilizados no XP. Segundo o autor, esta prática seria satisfeita se fosse realizada, durante a fase inicial de planejamento, a identificação, avaliação e planejamento de todos os recursos que seriam utilizados durante o projeto.

Segundo Santos (2010), o XP contempla este requisito devido a não obrigatoriedade de manter dados históricos de estimativas e custo das tarefas e produtos de trabalho. Segundo este autor, o XP estima o custo e esforços das tarefas na prática “Jogo do Planejamento”, onde cada tarefa é avaliada e estimada de acordo com os parâmetros de produtividade.

Considerações: Neste GPR um autor afirma que Satisfaz Parcialmente e argumenta sobre sua decisão. Os outros autores constatam que o GPR4 é Satisfeito pelo XP, porém, apenas um deles apresenta sua argumentação. Mapeando-se o nível 2 do CMMI com o GPR4 do MPS.BR, percebe-se uma aproximação do GPR4 com a KPA *Software project tracking and oversight*. Esta KPA estabelece a visibilidade de como estão as metas do projeto e permite que isto seja avaliado pela parte gerencial deste projeto (SOFTWARE ENGINEERING INSTITUTE, 2009). Observando-se a Tabela 3 é possível perceber que um autor afirma que Satisfaz, outro afirma que Satisfaz Parcialmente, e, por fim, o outro autor não chega a uma conclusão sobre se esta KPA atende o XP parcialmente ou totalmente. Esta variedade de opiniões pode indicar que ainda está em discussão na literatura se a GPR4 é atendida totalmente ou parcialmente pelo XP.

Nível: GPR5 - O orçamento e o cronograma do projeto, incluindo marcos e/ou pontos de controle, são estabelecidos e mantidos.

Trabalho(s) onde se encontra a discordância: Dias 2010

Discordância: Autor: Não Satisfaz / Literatura: Satisfaz Parcialmente

Argumento do autor:

Segundo Dias (2010), há uma falta de controle tanto sobre os custos quanto sobre o orçamento da metodologia ágil *Extreming Programing*.

Segundo Santos (2010), o XP atende parcialmente este requisito, pois o orçamento do projeto é definido de forma incremental, e é definido por *release*, em que as estórias são priorizadas pelo cliente e suas dependências analisadas pela equipe.

Considerações: Neste GPR um autor afirma que Não Satisfaz e argumenta sobre sua decisão. Os outros autores constatam que o GPR5 é Satisfeito Parcialmente pelo XP, porém, apenas um deles apresenta sua argumentação. Mapeando-se o nível 2 do CMMI com o GPR5 do MPS.BR, percebe-se uma aproximação do GPR5 com a KPA *Software project planing*. Esta KPA estabelece planos de execução do produto de software e as atividades e compromissos são planejadas e documentadas (SOFTWARE ENGINEERING INSTITUTE, 2009). Observa-se na Tabela 3 que um dos autores acredita na Satisfação Parcial desta KPA e os outros dois autores acreditam na Satisfação total desta KPA pelo XP, o que pode ser um indício de satisfação parcial do GPR5 por conta das diferentes opiniões dos autores (tanto da Tabela 2 quanto da Tabela 3).



Nível: GPR6 - Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados.

Trabalho(s) onde se encontra a discordância: Dias 2010

Discordância: Autor: Não Satisfaz / Literatura: Satisfaz Parcialmente

Argumento do autor:

Segundo Dias (2010), nenhum tratamento de riscos é abordado pelo XP.

Segundo Santos (2010), a identificação e tratamentos de riscos pelo XP, ocorre diretamente e indiretamente, através de alguns valores e práticas, tais como: entregas frequentes, *feedback* rápido, jogo do planejamento, cliente presente, TDD (*Test-driven development*), refatoração e programação em pares. Entretanto o XP não possui práticas que evidencie o registro de identificação e acompanhamento destes riscos, portanto, este autor considera que o XP Satisfaz Parcialmente.

Considerações: Neste GPR, um autor afirma que Não Satisfaz e argumenta sobre sua decisão. Os outros autores constatam que o GPR6 é Satisfeito Parcialmente pelo XP, porém, apenas um apresenta justificativa. Mapeando-se o nível 2 do CMMI com o GPR6 do MPS.BR, percebe-se uma aproximação do GPR2 com a KPA *Software congiguration management*, pois esta é a KPA que tem por objetivo estabelecer e manter a integridade dos produtos de projetos de *software* e há de certa forma uma análise dos riscos para se cumprir

tal objetivo (SOFTWARE ENGINEERING INSTITUTE, 2009). Analisando-se esta KPA na Tabela 3, percebe que dois autores indicam que ela é Satisfeita Parcialmente e o outro autor não tem uma opinião concreta sobre este requisito. Esta é uma evidência de que ainda está em discussão a satisfação da GPR6 pelo XP na literatura.

Nível: GPR7 - Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessário para executá-los.

Trabalho(s) onde se encontra a discordância: Dias 2010, Arimoto 2009

Discordância: Dias: Não Satisfaz. Arimoto: Satisfaz.

Outros autores: Satisfaz Parcialmente.

Argumento do autor:

Segundo Dias (2010), há uma falta de controle dos recursos utilizados no XP.

Segundo Santos (2010), não há no XP evidências de registros sobre alocação de recursos humanos.

Considerações: Neste GPR um autor conclui que Não Satisfaz e justifica sua conclusão, um autor conclui que Satisfaz e os outros autores concluem que o XP Satisfaz Parcialmente o GPR7, mas apenas um deles apresenta sua justificativa.

Nível: GPR8 - As tarefas, os recursos e o ambiente de trabalho necessário para executar o projeto são planejados.

Trabalho(s) onde se encontra a discordância: Santos 2010, Arimoto 2009

Discordância: Santos: Satisfaz Parcialmente. Arimoto: Satisfaz. Outros autores: Não Satisfaz.

Argumento do autor:

Segundo Santos, o planejamento formal de recurso não está no escopo do XP. O que ocorre neste sentido acontece no “Jogo do Planejamento”, onde são abordadas as tarefas a serem implementadas na *release* e os recursos necessários para implementá-las, caso seja necessário. É de responsabilidade do *Coach* organizar o ambiente de trabalho na fase de

exploração, mas não há registro deste planejamento. Desta forma, o XP atende parcialmente este requisito.

Segundo Santana, nenhum tratamento de riscos é abordado no XP. A sugestão do autor é a utilização de uma lista de riscos, apenas acrescentando as prioridades de tratamento e as probabilidades de ocorrência em cada um deles.

Considerações: Neste GPR, um autor afirma que Satisfaz e não apresenta justificativa, um autor afirma que Satisfaz Parcialmente e os outros afirmam que o XP Não Satisfaz este GPR e todos 3 destes autores apresentam suas justificativas. Mapeando-se o nível 2 do CMMI com o GPR8 do MPS.BR, percebe-se uma aproximação do GPR8 com a KPA *Software project planing*. Esta KPA estabelece planos para executar a engenharia de *software* e gerenciar o projeto de *software* (SOFTWARE ENGINEERING INSTITUTE, 2009). Observa-se na Tabela 3 que um dos autores acredita que esta KPA é Satisfeita Parcialmente pelo XP e os outros dois autores acreditam que esta KPA é Satisfeita pelo XP. Esta variedade de opiniões pode indicar que ainda está em discussão na literatura se a GPR8 é atendida totalmente ou parcialmente ou não atendida pelo XP.

Nível: GPR9.

Trabalho(s) onde se encontra a discordância: Nenhuma.

Discordância: Não há discordância.

Argumento dos autores:

Todos concordam que os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição. Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança.

Considerações: Nesta GPR, todos os autores chegam à conclusão de satisfação do XP, porém apenas Santos (2010) apresenta argumentação a respeito de sua conclusão.

Nível: GPR10 – Planos para a execução do projeto são estabelecidos e reunidos no Plano de Projeto.

Trabalho(s) onde se encontra a discordância: Santos 2010

Discordância: Autor: Não Satisfaz / Outros autores: Satisfaz

Argumento do autor:

Segundo Santos (2010), o XP não contempla este requisito, pois, não produz os documentos solicitados neste requisito para a criação do Plano de Projeto.

Considerações: A conclusão de Santos (2010) de que o XP Não Satisfaz o GPR10 influenciará diretamente nos GPR12 e GPR13, pois ambos tratam do Plano de Projeto que não é produzido pelo XP segundo este autor. Neste GPR, um autor afirma que Não Satisfaz e argumenta sobre sua decisão. Os outros autores constataam que o GPR10 é Satisfeito pelo XP, mas nenhum apresenta justificativa. Mapeando-se o nível 2 do CMMI com o GPR10 do MPS.BR, é possível notar que o GPR10 está contido na KPA *Software project planing*, a qual tem como objetivo estabelecer os planos de execução do projeto e documentá-los (SOFTWARE ENGINEERING INSTITUTE, 2009). Observa-se na Tabela 3 que um dos autores acredita que esta KPA é Satisfeita Parcialmente pelo XP e os outros dois autores acreditam que esta KPA é Satisfeita pelo XP. Apesar de um autor afirmar que o requisito GPR10 é Não Satisfeito, pela quantidade de Satisfações, é possível notar que há um indício forte de que este requisito seja realmente Satisfeito.

Nível: GPR11.

Trabalho(s) onde se encontra a discordância: Nenhuma.

Discordância: Não há discordância.

Argumento dos autores:

Todos concordam que a viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada. Se necessário, ajustes são realizados.

Considerações: Nesta GPR, todos os autores chegam à conclusão de satisfação do XP, porém apenas Santos (2010) apresenta argumentação a respeito de sua conclusão.

Nível: GPR12 - O Plano de Projeto é revisado com todos os interessados e o compromisso com ele é obtido.

Trabalho(s) onde se encontra a discordância: Santos 2010

Discordância: Autor: Satisfaz Parcialmente / Outros autores: Satisfaz

Argumento do autor:

Segundo Santos (2010), apesar do projeto ser revisado na presença de toda a equipe, o XP não produz o Plano do Projeto solicitado no GPR10. Devido à falta deste plano, o XP contempla parcialmente este requisito.

Considerações: Neste GPR, um autor afirma que Satisfaz Parcialmente e argumenta sobre sua decisão. Os outros autores constataam que o GPR12 é Satisfeito pelo XP, e nenhum destes apresenta justificativa. Mapeando-se o nível 2 do CMMI com o GPR12 do MPS.BR, percebe-se uma aproximação do GPR12 *Software project planing*. Esta KPA tem como meta acompanhar, planejar e documentar o projeto de software para que o compromisso seja obtido pelos integrantes do projeto (SOFTWARE ENGINEERING INSTITUTE, 2009). Analisando-se esta KPA na Tabela 3, percebe-se que um dos autores afirma que esta KPA é Satisfeita Parcialmente pelo XP e os outros dois autores acreditam que esta KPA é Satisfeita pelo XP. Devido à quantidade de Satisfações encontradas através das Tabelas 2 e 3, tem-se um forte indício de que o requisito GPR12 seja Satisfeito pelo XP.

Nível: GPR13 - O progresso do Projeto é monitorado com relação ao estabelecido no Plano de Projeto e os resultados são documentados.

Trabalho(s) onde se encontra a discordância: Santos 2010

Discordância: Autor: Satisfaz Parcialmente / Outros autores: Satisfaz

Argumento do autor:

No XP, o acompanhamento do projeto é realizado a todo momento através da presença do cliente no ambiente de desenvolvimento, das reuniões diárias e reuniões de planejamento das iterações. Segundo Santos (2010), o XP satisfaz este requisito parcialmente porque não evidencia o registro destes acompanhamentos e também não produz o Plano de Projeto, conforme o GPR10.

Considerações: Neste GPR, um autor afirma que Satisfaz Parcialmente e argumenta sobre sua decisão. Os outros autores constatam que o GPR13 é Satisfeito pelo XP, e nenhum destes apresenta justificativa. Mapeando-se o nível 2 do CMMI com o GPR13 do MPS.BR, percebe-se uma aproximação do GPR13 com duas KPAs: *Software configuration management*, quando se fala do monitoramento do progresso do projeto, pois esta KPA assume que o desempenho é monitorado contra os planos de *software*; *Software project planing*, quando se fala na documentação das atividades do projeto (SOFTWARE ENGINEERING INSTITUTE, 2009). Analisando-se esta KPA na Tabela 3, percebe-se que dois autores indicam que ela é Satisfeita Parcialmente e o outro autor não tem uma opinião concreta sobre este requisito. Esta é uma evidência de que ainda está em discussão a satisfação da GPR6 pelo XP. Analisando-se a KPA *Software configuration management* na Tabela 3, percebe que há dois autores indicam que ela é Satisfeita Parcialmente e o outro autor não tem uma opinião concreta sobre este requisito. Já na KPA *Software project planing*, observa-se na Tabela 3 que um dos autores acredita que esta KPA é Satisfeita Parcialmente e os outros dois autores acreditam que esta KPA é Satisfeita pelo XP. Diante de todos estes dados, nota-se uma tendência à Satisfação deste requisito já que tanto nas opiniões dos autores da Tabela 2 quanto da Tabela 3 (de ambas as KPAs) concluem que o XP Satisfaz o requisito/KPA em questão.

Nível: GPR14.

Trabalho(s) onde se encontra a discordância: Nenhum.

Discordância: Não há discordância.

Argumento dos autores:

Todos concordam que o envolvimento das partes interessadas no projeto é gerenciado.

Considerações: Nesta GPR, todos os autores chegam à conclusão de satisfação do XP, porém apenas Santos (2010) apresenta argumentação a respeito de sua conclusão.

Nível: GPR15.

Trabalho(s) onde se encontra a discordância: Nenhum.

Discordância: Não há discordância.

Argumento dos autores:

Todos concordam que as revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento.

Considerações: Nesta GPR, todos os autores chegam à conclusão de satisfação do XP, porém apenas Santos (2010) apresenta argumentação a respeito de sua conclusão.

Nível: GPR16 - Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas.

Trabalho(s) onde se encontra a discordância: Dias 2010.

Discordância: Autor: Satisfaz / Outros autores: Satisfaz Parcialmente.

Argumento dos autores:

Segundo Santos (2010), não há práticas no XP que evidenciam o registro de problemas identificados e análises realizadas sobre estes problemas. Como no Jogo do Planejamento são identificados e analisados, com a participação do cliente e a equipe, as dependências e problemas relacionados às histórias selecionadas para a próxima *release*, desta forma, o XP atende parcialmente este requisito.

Considerações: Neste GPR, um autor afirma que Satisfaz e não argumenta sobre sua decisão. Os outros dois autores constataam que o GPR16 é Satisfeito Parcialmente pelo XP, e nenhum destes apresenta justificativa. Mapeando-se o nível 2 do CMMI com o GPR16 do MPS.BR, percebe-se uma aproximação do GPR16 com *Software project tracking and oversight*. Esta KPA tem por objetivo estabelecer a visibilidade do progresso real do projeto, de modo que a parte gerencial possa tomar as devidas atitudes frente a problemas e suas questões pertinentes (SOFTWARE ENGINEERING INSTITUTE, 2009). Observando-se a Tabela 3 é possível perceber que um autor afirma que Satisfaz, outro afirma que Satisfaz Parcialmente, e, por fim, o outro autor não chega a uma conclusão sobre se esta KPA atende o

XP parcialmente ou totalmente. Esta variedade de opiniões pode indicar que ainda está em discussão na literatura se a GPR16 é atendida totalmente ou parcialmente pelo XP.

Nível: GPR17 - Ações para corrigir desvios em relação ao planejado e para prevenir a repetição dos problemas identificados são estabelecidas, implementadas e acompanhadas até a sua conclusão.

Trabalho(s) onde se encontra a discordância: Santos 2010.

Discordância: Autor: Satisfaz Parcialmente / Outros autores: Satisfaz

Argumento dos autores:

Segundo Santos (2010), embora no XP haja reuniões diárias para identificar os possíveis problemas, permitindo ao gerente aplicar medidas corretivas. O XP não possui práticas que evidenciam o registro destes problemas nem das ações corretivas, portanto, desta forma, o XP atende parcialmente este requisito.

Considerações: Neste GPR, um autor afirma que Satisfaz Parcialmente e argumenta sobre sua decisão. Os outros dois autores constatam que o GPR17 é Satisfeito pelo XP, e nenhum destes apresenta justificativa. Mapeando-se o nível 2 do CMMI com o GPR17 do MPS.BR, percebe-se uma aproximação do GPR17 com *Software project tracking and oversight*. Esta KPA tem por objetivo estabelecer a visibilidade do progresso real do projeto, de modo que a parte gerencial possa tomar as devidas atitudes frente a problemas e suas questões pertinentes (SOFTWARE ENGINEERING INSTITUTE, 2009). Observando-se a Tabela 3 é possível perceber que um autor afirma que Satisfaz, outro afirma que Satisfaz Parcialmente, e, por fim, o outro autor não chega a uma conclusão sobre se esta KPA atende o XP parcialmente ou totalmente. Esta variedade de opiniões pode indicar que ainda está em discussão na literatura se a GPR16 é atendida totalmente ou parcialmente pelo XP.

5 DISCUSSÃO DOS RESULTADOS

Nesta seção, serão discutidos os resultados evidenciados na Seção 4. A partir da análise de alguns trabalhos de mapeamento acerca do XP e MPS.BR, XP e CMMI, foi possível fazer uma comparação entre as conclusões destes autores. Estas conclusões foram expostas através das Tabelas 2 e 3, e para cada um dos 17 requisitos do GPR do MPS.BR foi

realizada uma análise, conforme mostra a Seção 4. A Tabela 4 ajuda na visualização da análise dos resultados. As ocorrências do “C” na Tabela 4 indica que houve concordância geral do mapeamento entre o XP e MPS.BR nível G, portanto, nestes casos, não foi explanada as conclusões dos autores que mapearam o XP e CMMI. Estas conclusões são utilizadas para reforçar a argumentação em ocorrências de discordâncias no mapeamento entre XP e nível G do MPS.BR. A aparição do “-” indicam que não houve um mapeamento desta KPA com as GPR por conta de interpretações dúbias na hora de se fazer tal análise. Outra consideração importante a respeito da Tabela 4 é que cada um de seus requisitos está relacionado com o Mapeamento dos KPAs do CMMI e o XP da mesma forma que foi feito nos tópicos de “Considerações” da Seção 4. Então, por exemplo, como GPR1 é equivalente à KPA *Requeriment Management*, conforme descrito na Seção 4, então foi utilizado para preencher a Tabela 4 a conclusão encontrada na Tabela 3, na KPA *Requeriment Management*, em que 3 autores tomam como Satisfeita esta KPA. Da mesma forma, no GPR2 da Tabela 4 são utilizados para preencher os dados do mapeamento dos KPAs do CMMI as conclusões da KPA *Software project planing*, pelo fato de possuir certa equivalência com o GPR2.

Tabela 4: Análise quantitativa do nível de satisfação nos trabalhos de mapeamento entre os GPRs do MPS.BR e o XP.

Requisitos	Quantidade de Satisfações, Satisfações Parciais e Não Satisfações						
	Mapeamento dos GPRs do MPS.BR e XP			Mapeamento dos KPAs do CMMI e o XP			
	Satisfaz	S. Parcialmente	Não Satisfaz	Satisfaz	S. Parcialmente	Não Satisfaz	
GPR 1	3	1	0		3	0	0
GPR 2	3	1	0		2	1	0
GPR 3	4	0	0		C	C	C
GPR 4	2	2	0		1	1	0
GPR 5	0	3	1		2	1	0
GPR 6	0	3	1		0	2	0
GPR 7	1	2	1		-	-	-
GPR 8	1	1	2		2	1	0
GPR 9	4	0	0		C	C	C
GPR 10	3	0	1		2	1	0
GPR 11	4	0	0		C	C	C
GPR 12	3	1	0		2	1	0
GPR 13	3	1	0		-	-	-
GPR 14	4	0	0		C	C	C
GPR 15	4	0	0		C	C	C
GPR 16	1	2	0		1	1	0
GPR 17	1	1	1		1	1	0
TOTAL	41	18	7		16	11	0

Fonte: Próprio autor.

A partir da Tabela 4, pode-se observar alguns pontos importantes: não houve ocorrência total de Não Satisfação, pois em cada um dos requisitos pelo menos um autor concorda com a Satisfação total ou parcial de determinado requisito; nos requisitos GPR3, GPR9, GPR11, GPR14 e GPR15 há uma satisfação total de todos os autores analisados, o que mostra uma forte evidência de aceitação da MA XP para estes processos do MPS.BR. Os aspectos básicos destas GPRs dizem respeito ao ciclo de vida, dados relevantes do projeto, avaliação das metas de projeto, gerenciamento dos interessados e revisões de projeto:

- ✓ O ciclo de vida de um projeto no XP consiste de fases e atividades que são definidas de acordo com o escopo prioritário dos requisitos. Este ciclo é claramente definido, por isso o requisito GPR3 é Satisfeito pelo XP.
- ✓ Os dados relevantes do projeto no XP são as histórias e o código, que são identificados e planejados, obedecendo às questões de privacidade e segurança exigidos pela GPR9, através dos sistemas de versões.
- ✓ Tem-se uma avaliação das metas de projeto no XP através das reuniões diárias e das reuniões realizadas no final das iterações e das *releases*.
- ✓ As partes interessadas no projeto no XP são o cliente, usuário e a equipe de desenvolvimento. O XP define claramente a participação intensa do cliente no ambiente de desenvolvimento, no qual a equipe está desenvolvendo o produto e sendo instruída pelo *Coach*, que faz o papel de técnico gerenciador desta equipe.
- ✓ A revisão de projeto no XP é feita através das iterações, que servem como pontos de controle dentro das *releases* para que programadores e clientes saibam como a execução esta em relação ao planejado.

A partir da discussão destes pontos é possível perceber o motivo da Satisfação do XP em relação ao MPS.BR, por conta dos autores, em relação a tais requisitos.

Outro ponto importante a ser notado na Tabela 4 é que nos requisitos GPR7, GPR8 e GPR17 têm-se um índice de incerteza grande por conta das diferentes opiniões dos autores tanto no mapeamento do XP com MPS.BR quanto do CMMI. O GPR7 diz respeito ao planejamento dos recursos humanos. Segundo o Guia de Implementação (SOFTEX, 2009), este planejamento de recursos humanos determina funções, responsabilidades e relações hierárquicas do projeto. Esta parte do GPR7 é atendida pelo XP, conforme a subseção 2.1.1, quando se fala dos papéis estabelecidos pelo XP. Porém, o Guia também expõe “A existência de registros das necessidades e disponibilidade de treinamento evita a alocação com base em critérios subjetivos”. Como não são encontrados no XP esse tipo de registro, não fica clara para os autores a Satisfação do requisito GPR7 pela MA XP. O GPR8 diz respeito ao planejamento das tarefas,

recursos e do ambiente de trabalho na execução do projeto. Como não há no XP este planejamento de recursos, fica à cargo do autor decidir se este requisito é Satisfeito Parcialmente ou não por esta MA. Observando-se agora o GPR17, sabe-se que este requisito diz respeito às ações para prevenção e correção de problemas identificados no projeto. Como no XP não há práticas que evidenciem o registro dos problemas identificados e as análises realizadas nestes problemas, há um índice de incerteza em relação à Satisfação deste requisito pelo XP.

Com base na Tabela 4, é fácil perceber que há uma tendência à Satisfação da MA XP, tanto pelo MPS.BR nível G quanto pelo CMMI nível 2. Nota-se que a quantidade de Satisfações em relação ao MPS.BR e o XP é igual 41, mais que o dobro das Satisfações Parciais e muito mais que o triplo das Não Satisfações. A verificação destes dados relatam a conclusão geral dos autores na maioria dos trabalhos de mapeamento envolvendo XP e MPS.BR, que é a tendência forte de satisfação desta MA com essa MQ. Embora pesquisado com menos profundidade, pode-se perceber um pouco dessa tendência verificando os dados da Tabela 4 em relação ao mapeamento do CMMI e XP.

É possível perceber com todo este estudo que alguns trabalhos de mapeamento fazem a análise acerca da MA e MQ, mas não apresentam as justificativas que levaram estes autores a esta conclusão. Isso geralmente ocorre quando o autor conclui o requisito como Satisfeito. Quando o requisito é Não Satisfeito, os autores tendem a discutir os problemas encontrados neste mapeamento, e alguns ainda sugerem uma forma de adaptação para resolução deste problema, para transformar o Não Satisfaz em Satisfaz. Quando o requisito é tomado como Satisfeito em alguns trabalhos, apesar de ter sido feita uma análise em cima deste requisito e na metodologia, esta análise não é explícita em seus trabalhos de mapeamento. Como os trabalhos de mapeamento são baseados em interpretação, a omissão desses argumentos citados prejudicam a integridade do trabalho, ainda mais que não existe um modelo formal para se realizar este mapeamento.

Apesar de toda a análise de dados consistentes apresentados neste trabalho, ele apresenta algumas limitações. A quantidade de trabalhos que mapeiam o XP e MPS.BR de forma a oferecer os dados que seriam necessários neste trabalho é uma limitação. Na tentativa de suprir tal limitação, o mapeamento do CMMI com o XP também foi abordado como forma de enfatizar as argumentações que geraram discordâncias entre os quatro autores analisados. Para atingir este objetivo, foi preciso fazer uma aproximação entre o nível G do MPS.BR e o nível 2 do CMMI. Esta aproximação é viável, pois a literatura garante 100% de compatibilidade entre estes modelos de qualidade, e garante também que todas as práticas do CMMI estão contempladas no MPS.BR (SOFTEX, 2009). Como o CMMI possui 5 níveis e o

MPS.BR 7 níveis, é possível notar que alguns dos níveis do CMMI irão abranger mais de um nível do MPS.BR (Figura 5). Consequentemente, algumas KPAs abrangem mais de requisito do MPS.BR. Esta observação fica clara ao verificar que 17 requisitos do GPR são mapeados pra 6 KPAs do nível 2 do CMMI, sendo que estas KPAs ainda abrangem o GRE e o nível F do MPS.BR. Outra limitação é quanto a questão das aproximações dos critérios de avaliação dos trabalhos de mapeamento com diferente quantidade de critérios. Alguns autores poderiam fazer o mapeamento em que Muito Satisfaz e Satisfaz são equivalentes ao critério Satisfaz e Pouco Satisfaz e Não Satisfaz sendo equivalente ao critério Não, ao invés de mapear como foi feito na Metodologia deste trabalho (Seção 3.2).

6 CONSIDERAÇÕES FINAIS

Neste trabalho, foi realizada a comparação de trabalhos de mapeamento entre a metodologia ágil XP e os requisitos GPR do MPS.BR e também do XP com o CMMI. Essas comparações foram evidenciadas em forma de tabelas para melhor visualização dos dados. Os métodos utilizados para analisar estes dados foram descritos na Seção 3 de Metodologia. Foram mostrados os pontos de discrepância quando se comparavam seus resultados, bem como as concordâncias, e por fim uma discussão acerca destes pontos e suas implicações.

Este trabalho pode servir como referência para profissionais que trabalham com desenvolvimento de *software*. Ele procurou ampliar a visibilidade dos pontos de discordância que existem na literatura quando se fala em mapear MAs e MQs. A falta de um modelo formal para fazer este mapeamento e o fato dos trabalhos serem baseado na interpretação dos autores são os principais causadores destas diferentes conclusões. Outro problema encontrado neste estudo foi a falta de argumentação sobre as conclusões de alguns autores, deixando o trabalho incompleto, pois, não são encontrados detalhes sobre como as metodologias contemplam ou não um determinado requisito nestes casos. A falta de padronização no momento de estabelecer os critérios de avaliação também contribui para esta problemática nos trabalhos de mapeamento. A diferente quantidade de critérios gera inconsistências ao se fazer comparações entre trabalhos de mapeamento ligados à MAs e MQs.

A proposta deste trabalho diante destas inconsistências foi a de comparar alguns trabalhos de mapeamento, identificar as diferenças e concordâncias obtidas e fazer uma discussão, principalmente nas discordâncias, a partir das análises de cada um dos requisitos

do GPR e estudo da metodologia ágil XP. Portanto, o objetivo principal deste trabalho foi evidenciar e analisar os pontos de discordância notados nos trabalhos de mapeamento entre a MA XP e os requisitos GPR do MPS.BR.

Como trabalhos futuros pode-se citar: ampliar a quantidade de trabalhos pesquisados que fazem mapeamento entre o MPS.BR e XP; fazer levantamento de mais trabalhos acerca do mapeamento de MAs e MQs, tratando de outras MAs, diferentes MQs ou de diferentes níveis do MPS.BR; propor soluções para que o mapeamento das MAs e MQs seja totalmente satisfeito; estabelecer um modelo formal para se fazer mapeamento entre as MAs e MQs; padronização dos critérios de classificação de requisitos em trabalhos de mapeamento.

7 REFERÊNCIAS

AGILE MANIFEST. **Manifesto for Agile Software Development**. 2001. Disponível em: <<http://www.agilemanifesto.org>>. Acesso em: 30 jul. 2010.

ARIMOTO, M. M.; MURAKAMI, E.; CAMARGO, V. V.; CAGNIN, M. I. **Adherence Analysis of Agile Methods According to the MR-MPS Reference Model**. VIII Simpósio Brasileiro de Qualidade de Software, UNIVEM, 2009.

BECK, K. **Extreme Programming Explained**. Addison Wesley, 2004.

BECK, K. **Extreme Programming Explained: Embrace Change**. 1st. ed. [S.l.]: Addison-Wesley Professional, 1999.

CAMPELO, R. E. C. **XP-CMM2: Um Guia para Utilização de Extreme Programming em um Ambiente Nível 2 do CMM**. Pós-Graduação Em Ciência Da Computação. Universidade Federal de Pernambuco, 2003.

DIAS, T. M. R.; MOITA, G. F.; SILVA, M. P.; FERREIRA, B.; SILVA, A. M. **Viabilidade do Desenvolvimento de Software Baseado no Modelo MPS.BR com a Metodologia Extreme Programming**. Nono Simpósio de Mecânica Computacional. Universidade Federal de São João Del-Rei, 2010.

ELIAS, G. S. **Métricas Associadas ao Desenvolvimento de Software Usando Métodos Ágeis**. Faculdade Cenecista Nossa Senhora dos Anjos, Gravataí, 2008.

HIGHSMITH, J.; COCKBURN, A. **Agile Software Development: The Business of Innovation**. Computer. IEEE Computer Society, Los Alamitos, CA, USA, v. 34, n. 9, 2001.

KOCH, A. S. **CMMI - Compilant XP**, 2003. Disponível em: <www.askprocess.com/resources/articles/CMM-XP.pdf>. Acesso em: 28 nov. 2010.

MARÇAL, A. S. C.; FREITAS, B. C. C.; SOARES, F. S. F., BELCHIOR, A. D. **Mapping CMMI Project Management Process Areas to SCRUM Practices**. University of Fortaleza - Master's Degree in Applied Computer Science, 2010.

MARTINSSON, J. **Maturing Extreme Programming Through the CMM**. Department of Computer Science. Lund University, 2002.

OLIVEIRA, C. S. **Comparando CMMI x MPS. BR: As Vantagens e Desvantagens dos Modelos de Qualidade no Brasil**, 2008. Disponível em: <<http://www.camilaoliveira.net/Arquivos/>>. Acesso em: 28 de nov. 2010.

PAULK, M. C. **Extreme Programming from a CMM Perspective**. IEEE Software, 2001.

QUALIDADE BR. **Qualidade BR: Teste e Qualidade de Software**, 2008. Disponível em: <<http://qualidadebr.wordpress.com/tag/mpsbr/>>. Acesso em: 13 mai. 2010.

SANTANA, C. A.; TIMÓTEO, A. L.; VASCONCELOS, A. M. L. **Mapeamento do modelo de Melhoria do Processo de Software Brasileiro (MPS.Br) para empresas que utilizam Extreme Programming (XP) como metodologia de desenvolvimento**. V Simpósio Brasileiro de Qualidade de Software – SBQS. Universidade Federal de Pernambuco (UFPE), 2006.

SANTOS, C.B.S. **Mapeamento dos Processos de Desenvolvimento Ágeis em Relação ao Modelo de Melhoria do Processo de Software do Brasil (nível G)**. Feira de Santana: UEFS, 2010.

SILVA, F. G.; HOENTSCH, S. C. P. **Uma análise das Metodologias Ágeis FDD e Scrum sob a Perspectiva do Modelo de Qualidade MPS.BR**. Universidade Federal de Sergipe, 2009.

SOFTEX. **SOFTEX**. 2009. Disponível em: <<http://www.softex.br>>. Acesso em: 13 mai. 2010.

SOFTWARE ENGINEERING INSTITUTE. **SOFTWARE ENGINEERING INSTITUTE**. 2009. Disponível em: <<http://www.sei.cmu.edu/cmmi/>> Acesso em: 13 mai. 2010.

SUTHERLAND, J.; JAKOBSEN, C. R.; JOHNSON K. **Scrum and CMMI Level 5: The Magic Potion for Code Warriors**. IEEE Computer Society, 2007.

APÊNDICE A – Requisitos de Processos do Nível G (GPR) do MPS.BR

- ✓ **GPR1:** O escopo do trabalho para o projeto é definido.
- ✓ **GPR2:** As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados.
- ✓ **GPR3:** O modelo e as fases do ciclo de vida do projeto são definidos.
- ✓ **GPR4:** O esforço e custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas.
- ✓ **GPR5:** O orçamento e o cronograma do projeto, incluindo marcos e/ou pontos de controle, são estabelecidos e mantidos.
- ✓ **GPR6:** Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados.
- ✓ **GPR7:** Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessário para executá-los.
- ✓ **GPR8:** As tarefas, os recursos e o ambiente de trabalho necessário para executar o projeto são planejados.

- ✓ **GPR9:** Os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição. Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança.
- ✓ **GPR10:** Planos para a execução do projeto são estabelecidos e reunidos no Plano de Projeto.
- ✓ **GPR11:** A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada. Se necessário, ajustes são realizados.
- ✓ **GPR12:** O Plano de Projeto é revisado com todos os interessados e o compromisso com ele é obtido.
- ✓ **GPR13:** O progresso do Projeto é monitorado com relação ao estabelecido no Plano de Projeto e os resultados são documentados.
- ✓ **GPR14:** O envolvimento das partes interessadas no projeto é gerenciado.
- ✓ **GPR15:** Revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento.

- ✓ **GPR16:** Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas.
- ✓ **GPR17:** Ações para corrigir desvios em relação ao planejada e para prevenir a repetição dos problemas identificados são estabelecidas, implementadas e acompanhadas até a sua conclusão.