

**UNIOESTE – Universidade Estadual do Oeste do Paraná**

**CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS**

**Colegiado de Ciência da Computação**

***Curso de Bacharelado em Ciência da Computação***

**Proposta de um processo de gerenciamento de  
requisitos para uma empresa desenvolvedora de  
software com foco na satisfação do Nível G do MPS.BR**

*Marcelo Rodrigues Gomes*

**CASCAVEL**

**2010**

**MARCELO RODRIGUES GOMES**

**PROPOSTA DE UM PROCESSO DE GERENCIAMENTO DE  
REQUISITOS PARA UMA EMPRESA DESENVOLVEDORA DE  
SOFTWARE COM FOCO NA SATISFAÇÃO DO NÍVEL G DO MPS.BR**

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação, do Centro de Ciências Exatas e Tecnológicas da Universidade Estadual do Oeste do Paraná - Campus de Cascavel.

Orientador: Prof. Dr. Victor Francisco Araya Santander

CASCADEL

2010

**MARCELO RODRIGUES GOMES**

**PROPOSTA DE UM PROCESSO DE GERENCIAMENTO DE  
REQUISITOS PARA UMA EMPRESA DESENVOLVEDORA DE  
SOFTWARE COM FOCO NA SATISFAÇÃO DO NÍVEL G DO MPS.BR**

Monografia apresentada como requisito parcial para obtenção do Título de *Bacharel em Ciência da Computação*, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos professores:

---

Prof. Dr. Victor Francisco Araya

Santander (Orientador)

Colegiado de Ciência da Computação,

UNIOESTE

---

Prof. Msc. Carlos José Maria Olguín

Colegiado de Ciência da Computação,

UNIOESTE

---

Prof. Dr. Clodis Boscarioli

Colegiado de Ciência da Computação,

UNIOESTE

Cascavel, 04 de Novembro de 2010.

"A qualidade nunca se obtém por acaso; ela é sempre o resultado do esforço inteligente".  
(John Ruskin)

## **AGRADECIMENTOS**

Agradeço a todos que tornaram possível a realização desse trabalho, em especial meus pais, Sonia e Valdir, que me apoiaram não só durante o TCC, mas em toda a minha vida. E a minha namorada, Jaqueline, que foi importantíssima, sempre carinhosa e compreensiva comigo.

# Lista de Figuras

Figura 1	Níveis de Maturidade do MPS.BR comparados ao CMMI
Figura 2	Processo de gerenciamento de mudanças
Figura 3	Campo de Rastreabilidade
Figura 4	Matriz de Rastreabilidade “ <i>Requisito X Casos de Teste</i> ”
Figura 5	Matriz de Rastreabilidade “ <i>Requisito X Requisito</i> ”
Figura 6	Rastreabilidade Horizontal
Figura 7	Plano de Revisão
Figura 8	Controle de Mudanças
Figura 9	Fases para a Implantação

# Lista de Tabelas

Tabela 1	Níveis do MPS.BR e seus Atributos de Processo
Tabela 2	Atividades de um Processo de Gerência de Requisitos
Tabela 3	Fatores de Mudanças de Requisitos
Tabela 4	Avaliação das técnicas para a satisfação da GRE1

# Lista de Abreviaturas e Siglas

BID	Banco Interamericano de Desenvolvimento
CMMI	<i>Capability Maturity Model Integration</i>
DRE	Documento de Requisitos
FINEP	Financiadora de Estudos e Projetos
GPR	Gerência de Projeto
GRE	Gerência de Requisitos
IEC	<i>International Electrotechnical Commission</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
ISO	<i>International Organization for Standardization</i>
MCT	Ministério da Ciência e Tecnologia
MPS.BR	Melhoria de Processo do Software Brasileiro
NBR	Norma Brasileira
MPE	Micro Pequenas Empresas
SEBRAE	Serviço Brasileiro de Apoio às Micro e Pequenas Empresas
SEI	<i>Software Engineering Institute</i>
SOFTEX	Associação para Promoção da Excelência do Software Brasileiro

# Sumário

<b>Lista de Figuras .....</b>	<b>vi</b>
<b>Lista de Tabelas.....</b>	<b>vii</b>
<b>Lista de Abreviaturas e Siglas .....</b>	<b>viii</b>
<b>Sumário.....</b>	<b>ix</b>
<b>Resumo.....</b>	<b>xii</b>
<b>1. Introdução .....</b>	<b>1</b>
1.1 Contexto.....	1
1.2 Motivações.....	3
1.3 Proposta .....	4
1.4 Contribuições .....	4
1.5 Estrutura do Trabalho .....	5
<b>2. Qualidade de Software .....</b>	<b>6</b>
2.1 Contexto.....	6
2.2 Padrões e Modelos de Maturidade.....	7
2.3 Melhoria de Processo do Software Brasileiro (MPS.BR) .....	9
2.4 Nível de Maturidade G .....	12
2.4.1 Gerência de Projeto (GPR) .....	12
2.4.2 Gerência de Requisitos (GRE).....	14
2.5 Considerações Finais .....	14
<b>3. Gerência de Requisitos (GRE).....</b>	<b>16</b>
3.1 Introdução .....	16
3.2 GRE1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos .....	18
3.2.1 Técnicas de Levantamento de Requisitos .....	18
3.2.1.1 Entrevistas .....	19

3.2.1.2	Questionários .....	19
3.2.1.3	Joint Application Development (JAD).....	20
3.2.1.4	Brainstorming.....	21
3.2.1.5	Etnografia.....	22
3.2.1.6	Prototipagem .....	23
3.2.1.7	Pesquisa de mercado (SWOT) .....	24
3.2.2	Validar Requisitos utilizando critérios objetivos.....	24
3.3	GRE2 - Obtenção do comprometimento da equipe técnica com os requisitos aprovados .....	25
3.3.1	Reuniões.....	25
3.3.2	E-mail.....	26
3.3.3	Contratos .....	26
3.4	GRE3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida .....	27
3.1.6	Tipos de Rastreamento.....	28
3.1.6.1	Rastreamento Para Frente e Para Trás .....	28
3.1.6.2	Rastreamento Pré-Especificação e Pós-Especificação de Requisitos ...	29
3.1.6.3	Extra-Rastreamento e Intra-Rastreamento de Requisitos .....	29
3.5	GRE4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos.....	29
3.5.1	Técnicas existentes.....	31
3.5.1.1	Leitura Baseada em perspectiva (PBR).....	31
3.5.1.2	Revisões Formais .....	32
3.6	GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto .....	33
3.7	Considerações Finais .....	35
<b>4.</b>	<b>Proposta de um processo de GRE para satisfação do MPS.BR .....</b>	<b>36</b>
4.1	Introdução .....	36
4.2	GRE1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos .....	38
4.2.1	Entender o domínio do cliente .....	38
4.2.2	Registrar os requisitos elicitados e seus fornecedores .....	38
4.2.3	Validar os requisitos registrados utilizando critérios objetivos .....	39

4.2.4	Fornecer informações suficientes para a organização aceitar ou não a produção dos projetos .....	41
4.2.5	Avaliação das técnicas .....	41
4.3	GRE2 - Obtenção do comprometimento da equipe técnica com os requisitos aprovados .....	42
4.4	GRE3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida .....	42
4.4.1	Ferramentas de Rastreabilidade .....	43
4.4.2	Rastreabilidade Vertical .....	44
4.4.3	Rastreabilidade Horizontal .....	46
4.5	GRE4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos .....	47
4.6	GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto .....	49
4.7	Processo para Implantação da Engenharia de Requisitos visando a melhoria da Qualidade de Software .....	52
4.8	Considerações Finais .....	55
<b>5.</b>	<b>Estudo de Caso .....</b>	<b>57</b>
5.1	A empresa .....	57
5.2	O projeto .....	58
5.2.1	Entender o domínio do cliente .....	59
5.2.2	Registrar os requisitos elicitados e seus fornecedores .....	60
5.2.3	Validar os requisitos registrados utilizando critérios objetivos .....	60
5.2.4	Fornecer informações suficientes para a organização aceitar ou não a produção dos projetos .....	60
5.3	GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto .....	63
5.4	Processo de Implantação da Engenharia de Requisitos visando a melhoria da Qualidade de Software .....	63
5.5	Considerações Finais .....	65
<b>6.</b>	<b>Considerações Finais e Trabalhos Futuros .....</b>	<b>68</b>
<b>7.</b>	<b>Referencias. ....</b>	<b>70</b>

# Resumo

A necessidade de criar softwares com qualidade é algo que reflete diretamente em melhoria de produto e de serviço, fazendo com que as empresas desenvolvedoras possam ter um destaque no mercado, buscando a competitividade pela qualidade. Sabe-se que a maioria dos problemas no desenvolvimento de software estão associados aos requisitos, não sendo possível desenvolver um sistema de qualidade, cumprir prazos e custos, e atender as expectativas do cliente, sem ter um processo definido para determinar e gerenciar os requisitos. Neste contexto, este trabalho propõe uma abordagem para definir o processo de gerenciamento de requisitos em uma empresa desenvolvedora de software de pequeno porte, almejando auxiliá-la na melhoria da qualidade dos serviços e produtos prestados. Também descreve as principais etapas do processo de gerenciamento de requisitos visando atender os resultados esperados do modelo de qualidade MPS.BR nível G, bem como os passos necessários para concretizar o estabelecimento do processo na empresa. Essa proposta foi aplicada em um estudo de caso e a análise dos resultados alcançados e as dificuldades encontradas são elencadas no final do trabalho unidas a um conjunto de lições aprendidas. Com estas, foi possível verificar que abordar a melhoria de processo como um projeto, utilizar técnicas corretas e respeitar os requisitos são as diretrizes mais importantes que se pode tirar deste trabalho.

**Palavras-chave:** Qualidade de Software, Engenharia de Requisitos, Gerência de Requisitos.

# Capítulo 1

## Introdução

Este capítulo tem como objetivo a apresentação do trabalho, indicando as motivações para o seu desenvolvimento, bem como a sua estruturação. Inicialmente, na Seção 1.1, a proposta é contextualizada no escopo da Engenharia de Software, destacando a importância do processo de Engenharia de Requisitos na obtenção de produtos de software de qualidade. Na Seção 1.2, são descritas as principais motivações que levaram a execução deste estudo. Na Seção 1.3, a proposta em si é explanada, bem como seus principais objetivos. Já na Seção 1.4, são relatadas as contribuições esperadas com o desenvolvimento do presente trabalho. Por fim, na Seção 1.5 a estrutura do trabalho é apresentada.

### 1.1 Contexto

Queira em casa, no trabalho ou no governo o software é tão fundamental que se tornou uma das tecnologias mais importantes do mundo e tal importância faz com que o número de empresas desenvolvedoras de software aumente e produzam mais sistemas computacionais para várias áreas de negócio (PRESSMAN, 2006). Contudo, o fato de lançar um produto antecipadamente no mercado ou aumentar a quantidade e a velocidade de produção não são suficientes, pois o software precisa apresentar qualidade. Não tem sentido produzir um sistema computacional que não funcione, não atenda as necessidades do cliente ou que não seja confiável (YOURDON, 1995).

A necessidade de criar softwares com qualidade é algo que reflete diretamente em melhoria de produto e de serviço, fazendo assim, com que as empresas possam ter um destaque no mercado, buscando a competitividade pela qualidade (SOFTEX, 2009a).

Tal atributo é importante, tanto para usuários, quanto para empresas que desenvolvem softwares. Assim, para auxiliar o desenvolvimento de softwares de qualidade, mantendo as prioridades de usuários (preocupados com preço e funcionalidades do produto) e de

desenvolvedores (preocupados com custo de desenvolvimento e otimização de recursos, visando lucro) foram criados modelos de qualidade, que apresentam metas que as empresas devem atingir baseadas em boas práticas da engenharia de software e da engenharia de requisitos.

Neste contexto destaca-se o modelo CMMI (*Capability Maturity Model Integration*) proposto pela SEI (*Software Engineering Institute*), um modelo criado na década 80 e que vem sendo aprimorado desde então. O CMMI apresenta procedimentos que, quando implementados pela empresa, proporcionam uma melhoria contínua para o processo de desenvolvimento e produtos de software gerados. Porém, seu custo e prazo de implantação são altos, o que foge da realidade de algumas empresas tornando-o inviável (SEI, 2010).

Por outro lado, em nível nacional destaca-se o MPS.BR (Melhoria de Processo do Software Brasileiro) um modelo de qualidade que foi criado para atender a realidade das empresas brasileiras. Com foco em atender micro e pequenas empresas (MPEs), o MPS.BR apresenta padrões de qualidade aceitos internacionalmente e com a mesma competência de outros modelos existentes (SOFTEX, 2009a).

Este modelo é composto por sete níveis de maturidade: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Parcialmente Gerenciado) (SOFTEX, 2009a). A evolução do nível de maturidade ocorre desde o nível inicial G até o nível A. Cada nível apresenta atributos que devem ser cumpridos para que a empresa possa ser qualificada.

Para o nível inicial, nível G, existem dois atributos de processos: Gerência de Projetos e Gerência de Requisitos. Aos requisitos estão associados à maioria dos problemas no desenvolvimento de software. Não é possível desenvolver um sistema de qualidade, cumprir prazos e custos, e atender as expectativas do cliente sem ter um processo definido para determinar e gerenciar os requisitos (BLASCHEK, 2002).

Assim, no contexto de Qualidade de Software é importante estudar meios eficazes de definir, avaliar e gerenciar requisitos, bem como possibilitar que boas práticas sejam incorporadas efetivamente no processo de produção de software.

Neste trabalho, observando a realidade nacional de MPEs que buscam melhorar seus processos via MPS.BR, detecta-se uma grande dificuldade de melhorar a gerência de requisitos no processo de desenvolvimento. Assim, estudar e propor técnicas, ferramentas e estratégias de gerência de requisitos com foco na realidade destas empresas passa a ser um desafio atual.

## **1.2 Motivações**

Considerando a importância da Gerência de Requisitos na satisfação de exigências definidas no MPS.BR já no seu nível inicial de maturidade, bem como, o reconhecimento mundial, tanto no ambiente acadêmico, quanto industrial, de que esta área é de vital importância, é necessário esclarecer melhor esta atividade.

Sabe-se que grande parte dos problemas com a satisfação do cliente, em relação a produtos de software, estão relacionados a erros, inconsistências e ambiguidade de documentos de requisitos. Muito desses erros poderiam ser evitados se a empresa tivesse um processo definido de como avaliar e gerenciar os requisitos do projeto (KOTONYA & SOMMERVILLE, 1998). Tais características podem ser alcançadas pela empresa por meio da aplicação de modelos de maturidade como o MPS.BR.

Porém, para muitos profissionais isso parece ser algo difícil, pelo fato de não existirem propostas mais concretas que guiem as empresas nesse processo, já que o modelo MPS.BR contém apenas resultados exigidos para o processo de desenvolvimento e seus atributos, não determinando métodos, técnicas ou ferramentas que devem ser utilizadas para alcançar estes resultados.

Esta característica do modelo garante uma sinergia com a academia, atraindo-a para exercer a função de gerar conhecimento a respeito de métodos, técnicas e ferramentas que permitam alcançar os resultados esperados. Estas propostas devem contemplar a realidade das empresas desenvolvedoras e o estado da arte na área de gerência de requisitos (SAYÃO et al., 2003). Também é necessário fazer uma análise crítica de custo-benefício das técnicas utilizadas neste processo, e observar atentamente os benefícios reais do uso das mesmas no processo de evolução de sistemas computacionais (HAZAN & LEITE, 2003).

Portanto, estudar a gerência de requisitos em um contexto mais específico para o MPS.BR, bem como propor e viabilizar a utilização de uma abordagem que auxilie empresas de médio e pequeno porte em âmbito nacional é o principal foco deste trabalho.

De forma mais específica, consideradas as principais características comuns de empresas nessa categoria e com base nessas características, uma empresa representativa das mesmas será escolhida para validar a proposta.

## **1.3 Proposta**

No contexto de Qualidade de Software a gestão de requisitos se torna um procedimento indispensável no desenvolvimento de software. É importante para qualquer empresa ter um processo definido a ser usado como base que permita fazer a gerência dos requisitos.

Desta forma, o objetivo deste trabalho é realizar um estudo da área de Engenharia de requisitos para propor técnicas, processos, artefatos e ferramentas, no âmbito de MPEs, que possam ser utilizadas como base para a Gerência de Requisitos. Também traz um estudo de caso em uma empresa produtora de software de pequeno porte para validar a proposta apresentada no trabalho e verifica-se a possibilidade que esta tem de atender os resultados esperados pelo MPS.BR para o processo Gerência de Requisitos (GRE) do nível de maturidade G.

## 1.4 Contribuições

As principais contribuições pelo projeto são:

- Seleção de técnicas, processos, e ferramentas que possam ser utilizadas com sucesso para a satisfação do processo de gerência de requisitos exigido pelo MPS.BR nível G por MPEs;
- Definição de artefatos, com respectivas estruturas, visando viabilizar a gerência de requisitos no contexto de MPEs;
- Elaborar uma proposta de gerência de requisitos com validação prática em um cenário real;
- Obter um *feedback* técnico de uma organização que desenvolve softwares para terceiros em relação à proposta apresentada;
- Identificação das principais dificuldades em elaborar e aplicar essas práticas.

## 1.5 Estrutura do Trabalho

Este trabalho se divide em seis capítulos incluindo o presente, estruturados da seguinte maneira:

**Capítulo 2 – Qualidade de Software:** neste capítulo são abordados os principais conceitos da Qualidade de Software. Apresenta-se uma introdução ao tema, descreve-se Modelos de Maturidade, bem como o MPS.BR e seus níveis. Ao final, os problemas que abordam esses modelos e sua aplicação são descritos.

**Capítulo 3 – Gerência de Requisitos:** neste capítulo é apresentada uma visão geral a respeito da importância da gerência de requisitos. São apresentados também os cinco resultados esperados pelo MPS.BR para este atributo de processo, e quais princípios utilizados da Engenharia de Requisitos permitem satisfazê-los.

**Capítulo 4 – Proposta GRE para satisfação do MPS.BR:** neste capítulo a proposta de como realizar a gerência de requisitos é apresentada, bem como o estudo comparativo das técnicas existentes.

**Capítulo 5 – Estudo de Caso:** neste capítulo é apresentado o estudo de caso em um projeto real de uma empresa de pequeno porte, objetivando a validação da nova proposta. Para tanto, os resultados obtidos são descritos e é verificado se a nova proposta deve ser readequada com base nas experiências obtidas a partir da realização dos estudos de casos.

**Capítulo 6 – Considerações Finais:** o último capítulo relata as conclusões obtidas no desenvolvimento deste trabalho, assim como suas principais contribuições para a Engenharia de Software, particularmente ao Processo de Gerência de Requisitos em um contexto de satisfação do nível G do MPS.BR.

# Capítulo 2

## Qualidade de Software

Esse capítulo aborda os principais conceitos com relação à Qualidade de Software, ressaltando a importância da qualidade no desenvolvimento de sistemas. O capítulo é composto por uma introdução ao tema (Seção 2.1), seguida pela Seção de Padrões e Modelos de Maturidade (Seção 2.2). Após as definições iniciais, descreve-se o Modelo de Maturidade MPS.BR (Seção 2.3), com foco no Nível G de Maturidade (2.4). Ao final, são apresentados alguns dos problemas que abordam esses modelos e sua aplicação, seguido pelas considerações finais (Seção 2.5).

### 2.1 Contexto

O termo qualidade é definido ambigualmente e diferentes significados podem ser atribuídos a ele, por diferentes situações ou opiniões. Para maior clareza, considera-se para este trabalho a definição de qualidade descrita no dicionário MICHAELIS, (2009), que define: “*Qualidade: Grau de perfeição, de precisão, de conformidade a um certo padrão*”.

Trazendo essa definição para a computação, conclui-se que um software de qualidade seria aquele que apresentasse um alto grau de precisão e de conformidade em relação aos requisitos (necessidades do usuário). Existem outras definições na literatura que descrevem de forma similar, como:

**NBR ISO 8402, (1994):** “*A totalidade das características de uma entidade que lhe confere a capacidade de satisfazer às necessidades explícitas e implícitas*”.

**PRESSMAN, (2006):** “*Concordância com os requisitos funcionais e de desempenho, com padrões de desenvolvimento explicitamente documentados e com as características implícitas em todo software desenvolvido profissionalmente*”.

**ISO/IEC 9126-1 (2003):** “*Qualidade é a totalidade de características e critérios de um produto ou serviço que exercem suas habilidades para satisfazer as necessidades declaradas ou envolvidas*”.

Logo, nota-se que a qualidade do software está diretamente ligada à forma de como são tratados os requisitos a serem atendidos. Se os requisitos não refletem as reais necessidades dos usuários, ou são incompletos, inconsistentes, se as mudanças em requisitos não são controladas, ou há dificuldade para conseguir um entendimento comum entre usuários e desenvolvedores, provavelmente o software terá sua qualidade ameaçada. A qualidade do produto de software não se atinge de forma espontânea, esta depende fortemente da qualidade do processo de desenvolvimento (CARVALHO et al., 2001).

Por isso, é necessário entender que ao falar de software e de sua produção, estamos falando de produtos e de processos. Destaca LEITE (2001) que de nada adianta centrarmos nossa atenção só no produto ou só no processo. É necessário que os dois caminhem juntos. Portanto, para lidar com qualidade é necessário termos claro que ambos, processo e produto, devem ter qualidade.

Assim, um bom processo de software não garante que os produtos de software produzidos sejam de boa qualidade, mas é um indicativo de que a organização é capaz de produzir bons produtos de software (KOSCIANSKI & SOARES, 2007).

Portanto, é meta e dever da Engenharia de Software produzir e disponibilizar métodos, técnicas e ferramentas para que o produto e o processo de produção de software adquiram a qualidade necessária. Para suprir essa necessidade, órgãos como IEEE (*Institute of Electrical and Electronics Engineers*), SEI (*Software Engineering Institute*), ISO/IEC (*International Organization for Standardization / International Electrotechnical Commission*), entre outros, produzem padrões, métodos, técnicas e modelos que auxiliam o processo.

## **2.2 Padrões e Modelos de Maturidade**

Atualmente existem vários métodos, padrões e boas práticas criadas pela comunidade de Engenharia de Software, e acredita-se que o uso destas possa ajudar as empresas a melhorar seu desempenho quanto a custo, prazo, qualidade, entre outros. Porém, NOGUEIRA (2006) constatou que normalmente as organizações só implementam tais práticas quando são exigidas em avaliações de processos. De forma resumida, as técnicas existiam, mas nada obrigava ou estimulava as empresas a usarem.

Mesmo estando claro que tais práticas têm o objetivo de melhorar tarefas ou características de produto e processo, muitas empresas se recusam a aplicá-las, algumas vezes, empresas criam obstáculos e tomam que essas técnicas não são apropriadas para seu tipo de negócio. Argumentos comuns como: “*é muito caro*”, “*minha empresa não precisa disso*”, “*não acredito no benefício*”, fazem com que surja uma imagem ruim da Engenharia de Software e que suas práticas não sejam bem vistas pelas empresas, que alegam que tais são para o âmbito acadêmico e não para o prático (CARVALHO et al., 2001).

Com esse contexto foi criada uma iniciativa de explicitar os benefícios reais dessas técnicas e desenvolver métodos que avaliassem e melhorassem a capacidade de desenvolvimento de software nas organizações. Esta iniciativa teve início com o SEI, (2010), que criou, no fim da década de 80, o CMM (*Capability Maturity Model*), para avaliar a qualidade dos softwares desenvolvidos pelas empresas.

Desde a década de 90, surgiram diversos CMMs, com foco em desenvolvimento de sistemas, engenharia de software, desenvolvimento de produtos e processos, entre outros. Apesar de esses modelos serem úteis para as organizações, a grande diversificação tornou-se um problema. Cada um desses modelos tinha que ser avaliado separadamente, gerando um custo elevado. Para resolver este problema de diversidade de CMMs foi criado o CMMI (*Capability Maturity Model Integration*), que é nada mais que a unificação dos diversos CMMs (SEI, 2010).

Porém, o CMMI ainda envolve um grande custo para a realização de suas avaliações e obtenção de certificado. Dificuldades que contrastam com a realidade das empresas brasileiras que não podem realizar um investimento tão alto na obtenção da certificação.

Estes aspectos podem se tornar obstáculos para organizações que querem melhorar seus processos, principalmente para micro e pequenas empresas que trabalham com menor disponibilidade financeira. No Brasil, onde aproximadamente 70% da indústria de software é constituída por MPEs, onde, poucas organizações têm condições de adotar modelos de maturidade o que prejudica a qualidade do software nacional, dificultando as empresas nacionais terem destaque no mercado internacional (MCT, 2010).

Tendo em vista esse cenário, em 2003, a SOFTEX com ajuda do governo, universidades e empresas desenvolveu o programa MPS.BR (Melhoria de Processo do Software Brasileiro), com o objetivo de reunir as melhores práticas do CMMI, ISO/IEC 12207 e ISO/IEC 15504 e ainda manter um caminho economicamente viável e um intervalo de tempo razoável (SOFTEX, 2009a).

Cabe lembrar que o foco deste trabalho está no modelo MPS.BR, motivo pelo qual o mesmo é descrito na próxima seção.

## **2.3 Melhoria de Processo do Software Brasileiro (MPS.BR)**

O MPS.BR é um programa coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), que conta com apoio do Ministério da Ciência e Tecnologia (MCT), da Financiadora de Estudos e Projetos (FINEP), do Serviço Brasileiro de Apoio às Micro e Pequenas Empresas (SEBRAE) e do Banco Interamericano de Desenvolvimento (BID). Seu objetivo é reunir as melhores práticas do CMMI, ISO/IEC 12207 e ISO/IEC 15504 para implantar nas empresas de pequeno e médio porte brasileiras, e ainda, manter um caminho economicamente viável e um intervalo de tempo de aplicação razoável.

Como o CMMI o MPS.BR é composto por níveis de maturidade. Embora ambos tenham as mesmas metas o MPS.BR apresenta um número maior de níveis, o que facilita a aplicação de técnicas de engenharia de software, ajudando assim, as MPEs conseguirem suas certificações (SOFTEX, 2009a).

Os níveis do MPS.BR são: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Parcialmente Gerenciado).

- O nível inicial é o G e vai progredindo até o nível A. Cada nível apresenta atributos que devem ser cumpridos para que a empresa possa ser qualificada conforme apresentado na tabela 1, essas normas são as mesmas que o CMMI é baseado é por isso pode se dizer que os dois modelos tem equivalência como mostra a figura 1.

O modelo MPS possui três componentes principais: o modelo de referência MPS (MR-MPS); o método de avaliação MPS (MA-MPS); e o modelo de negócios MPS (MN-MPS). A discussão do método de avaliação e do modelo de negócios está fora do escopo deste trabalho já que este aborda os resultados esperados e os atributos de processo descritos no modelo de Referência. Todos estes modelos estão disponíveis no site da SOFTEX (SOFTEX, 2009a).

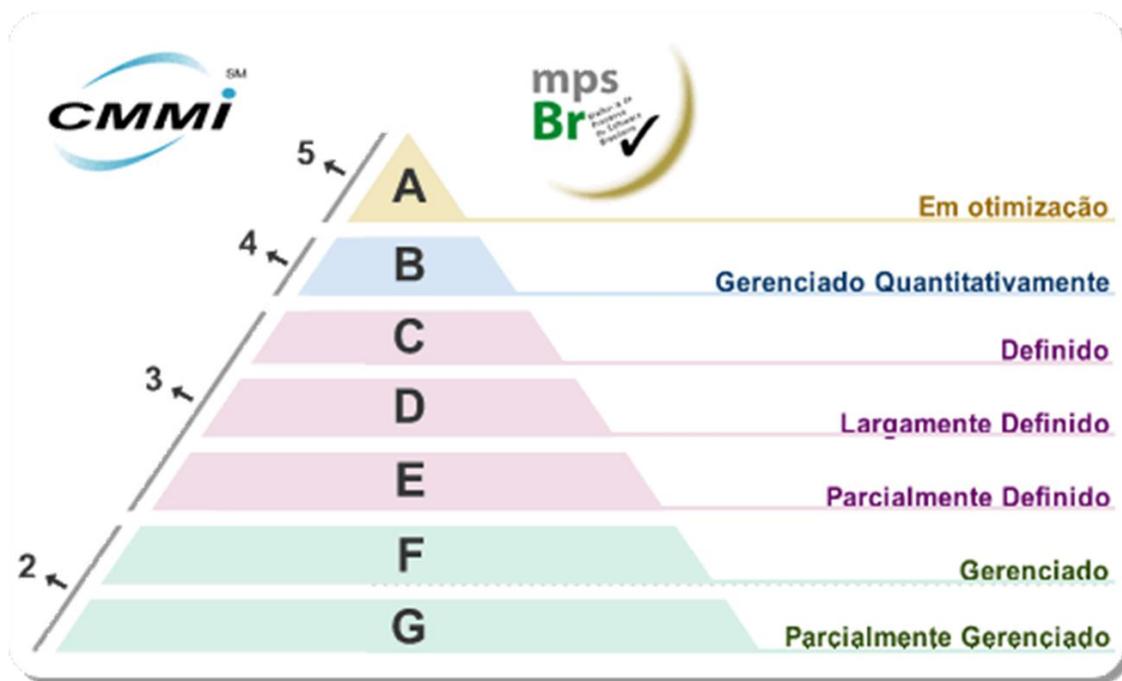


Figura 1. Níveis de Maturidade do MPS.BR Comparados ao CMMI  
(Fonte: [http://www.pentagrama.com.br/images/new\\_piramide.gif](http://www.pentagrama.com.br/images/new_piramide.gif))

**Nível de Maturidade G:** É o primeiro nível de maturidade (Parcialmente Gerenciado). Ao implementar esse nível a organização deve focar os esforços na criação de mecanismos adequados para o planejamento de projetos, monitoramento e controle, e gerenciar os requisitos ao longo do desenvolvimento.

**Nível de Maturidade F:** O segundo nível deve garantir que todo o processo é gerenciado, que a organização consiga assegurar a qualidade dos seus produtos e processos, que obtenha indicadores quantitativos do desempenho e seja capaz de gerenciar as configurações dos produtos.

**Nível de Maturidade E:** O nível E (Parcialmente Definido) faz com que a empresa comece a se desprender do conhecimento ou desempenho individual. Este nível possui processos que ajudam o processo organizacional da empresa e administra recursos humanos fazendo com que o processo de produção não dependa de pessoas chave.

**Nível de Maturidade D:** O nível D (Largamente Definido) foca na melhoria de processos de engenharia de software. Ele exige ações mais técnicas para determinadas tarefas como estabelecer métodos para validação e verificação de projeto (incluindo revisões e testes), entre outros como definição de arquitetura e estratégias.

**Nível de Maturidade C:** O Nível C (Definido) deve garantir que o processo está todo definido, até mesmo sobre quais decisões tomar em relação a desvios e riscos de projeto.

Além de definir um programa de reutilização de artefatos e ativos de projeto com o fim de reaproveitar esforços.

**Nível de Maturidade B e A:** São considerados os níveis de alta maturidade, trazendo uma melhoria contínua de processos. Seus processos focam em uma crescente capacidade competitiva através da criação de inovações. Dificilmente uma empresa alcança apenas o penúltimo nível, tanto no MPS.BR, quanto no CMMI.

Tabela 1. Níveis do MPS.BR e seus Atributos de Processo (SOFTEX, 2009a)

<b>Nível</b>	<b>Atributo de Processo</b>
A	<ul style="list-style-type: none"> <li>• <i>(sem processos adicionais)</i></li> </ul>
B	<ul style="list-style-type: none"> <li>• <b>Gerência de Projetos</b> <i>(evolução)</i></li> </ul>
C	<ul style="list-style-type: none"> <li>• <b>Gerência de Riscos;</b></li> <li>• <b>Desenvolvimento para Reutilização;</b></li> <li>• <b>Gerência de Decisões;</b></li> </ul>
D	<ul style="list-style-type: none"> <li>• <b>Desenvolvimento de Requisitos;</b></li> <li>• <b>Integração do Produto;</b></li> <li>• <b>Projeto e Construção do Produto;</b></li> <li>• <b>Validação;</b></li> <li>• <b>Verificação;</b></li> </ul>
E	<ul style="list-style-type: none"> <li>• <b>Avaliação e Melhoria do Processo Organizacional;</b></li> <li>• <b>Gerência de Projetos</b> <i>(evolução)</i>;</li> <li>• <b>Gerência de Recursos Humanos;</b></li> <li>• <b>Gerência de Reutilização;</b></li> <li>• <b>Definição do Processo Organizacional;</b></li> </ul>
F	<ul style="list-style-type: none"> <li>• <b>Aquisição;</b></li> <li>• <b>Garantia de Qualidade;</b></li> <li>• <b>Gerência de Configuração;</b></li> <li>• <b>Gerência de Portfólio de Projetos;</b></li> <li>• <b>Medição;</b></li> </ul>
G	<ul style="list-style-type: none"> <li>• <b>Gerência de Projetos;</b></li> <li>• <b>Gerência de Requisitos;</b></li> </ul>

## 2.4 Nível de Maturidade G

O nível G corresponde ao primeiro nível de maturidade (Parcialmente Gerenciado), é composto pelos processos mais críticos de gerência, e por esse motivo causa um grande impacto na cultura e na organização da empresa, visto muitas vezes como o nível mais difícil de ser implementado devido à forte resistência que as organizações apresentam para adotar novas técnicas e mudanças de cultura. Ao implementar esse nível a organização deve focar os esforços na criação de mecanismos adequados para o planejamento de projetos, monitoramento e controle, e gerenciar os requisitos ao longo do desenvolvimento.

Dois pontos são desafiadores na implantação do nível G: mudança de cultura organizacional e definição do que é “projeto” para a organização. No nível G a empresa pode usar os seus próprios padrões e caso a organização possui processos já definidos talvez esses necessitem de adaptação. Essas adaptações podem incluir alteração em processos, atividades, ferramentas, técnicas, medidas, dentre outras. Diversas organizações de software trabalham com melhoria de produtos e precisam adequar as suas formas de trabalhar para se tornarem organizações orientadas a projetos (SOFTEX, 2009b).

Para auxiliar essas mudanças o nível G traz as atividades de processo: Gerência de Projeto (GPR) e Gerência de requisitos (GRE), descritos a seguir.

### 2.4.1 Gerência de Projeto (GPR)

O propósito do processo Gerência de Projetos é estabelecer e manter planos que definem as atividades, os recursos e responsabilidades do projeto, bem como fornecer informações sobre o andamento do projeto que permitam a realização de correções quando houver desvios significativos. Haverá novas regras de GPR nos próximos níveis de maturidade, fazendo assim, com que o propósito deste processo evolua a medida que a organização cresce em maturidade.

Para o nível G existem 17 (dezessete) resultados esperados em relação à GPR:

- **GPR 1:** O escopo do trabalho para o projeto é definido;
- **GPR 2:** As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados;
- **GPR 3:** O modelo e as fases do ciclo de vida do projeto são definidos;

- **GPR 4:** (Até o nível F) O esforço e o custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas;
- **GPR 5:** O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos;
- **GPR 6:** Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados;
- **GPR 7:** Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo;
- **GPR 8:** Os recursos e o ambiente de trabalho necessários para executar o projeto são planejados;
- **GPR 9:** 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;
- **GPR 10:** Um plano geral para a execução do projeto é estabelecido com a integração de planos específicos;
- **GPR 11:** A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada. Se necessário, ajustes são realizados;
- **GPR 12:** O Plano do Projeto é revisado com todos os interessados e o compromisso com ele é obtido;
- **GPR 13:** O projeto é gerenciado utilizando-se o Plano do Projeto e outros planos que afetam o projeto e os resultados são documentados;
- **GPR 14:** O envolvimento das partes interessadas no projeto é gerenciado;
- **GPR 15:** Revisões são realizadas em marcos do projeto conforme estabelecido no planejamento;
- **GPR 16:** Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas;
- **GPR 17:** 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.

## 2.4.2 Gerência de Requisitos (GRE)

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

Para o nível G existem cinco (5) resultados esperados em relação à GRE:

- **GRE 1:** Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos;
- **GRE 2:** Obtenção do comprometimento da equipe técnica com os requisitos aprovados;
- **GRE 3:** A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida;
- **GRE 4:** Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos;
- **GRE 5:** Mudanças nos requisitos são gerenciadas ao longo do projeto.

Para todos estes resultados esperados, na proposta, foi realizada uma revisão debatendo os meios que possibilitam alcançar esses resultados, destacando as vantagens e desvantagens destes sobre as principais características comuns dentre as MPEs.

## 2.5 Considerações Finais

A aplicação de modelos de maturidade é importante, por ser um mecanismo para que as organizações alcancem os benefícios da implementação de boas práticas da Engenharia de Software. O MPS.BR vai além permitindo às empresas nacionais estabelecer um caminho economicamente viável para estes benefícios possibilitando-as estabelecer vantagens competitivas tanto no mercado local quanto global.

Resumidamente, o objetivo desses modelos é fazer com que as empresas consigam um processo de desenvolvimento **transparente** e **gerenciável**. Transparente no sentido que o processo se torne claro para todos na empresa, independente de qual fase que ele se encontre, evitando assim que existam processos “caixa preta”, ou seja, que sua execução seja desconhecida. Se o projeto é totalmente transparente, então ele é compreendido por todos e a empresa não fica dependente do conhecimento ou desempenho individual. E gerenciável para que possa ser controlado e se manter dentro do planejado, independente dos desvios que aconteçam durante o desenvolvimento. Isso possibilita que os projetos permaneçam dentro do

prazo, custo e escopo desejáveis. Tais características são vitais para a qualidade do processo e do produto.

Por fim, ressalto que o modelo MPS.BR contém apenas resultados exigidos para o processo de desenvolvimento e seus atributos, não determinando métodos, técnicas ou ferramentas que devem ser utilizadas para alcançar estes resultados. Esta característica do modelo garante uma sinergia com a academia, atraindo esta a exercer a função de gerar conhecimento a respeito de métodos, técnicas e ferramentas que permitam alcançar os resultados esperados. Esta é uma das principais motivações da presente proposta.

## Capítulo 3

# Gerência de Requisitos (GRE)

O presente capítulo fornece uma visão geral a respeito da Gerência de requisitos (3.1). Logo a seguir, apresenta os resultados esperados pelo MPS.BR para esse atributo do processo de desenvolvimento (Seções 3.2 a 3.6) e quais técnicas propostas na literatura da área permitem alcançar estes resultados. Por fim, a Seção 3.7 é destinada às considerações finais.

### 3.1 Introdução

Antes de abordar o Processo de GRE, é importante conceituar requisito, termo frequentemente citado e debatido em contratos, sem que as partes possuam uma compreensão única de seu significado.

A literatura possui várias definições para requisitos de software. SOMMERVILLE (2003) define requisitos como *“um conjunto de atividades que o software deve desempenhar, com suas limitações e restrições, além de características não ligadas diretamente às funções desempenhadas pelo software”*.

Segundo DORFMANN & THAYER (1990), requisito de software representa a capacidade requerida pelo usuário que deve ser encontrada ou possuída por um determinado produto ou componente de produto para resolver um problema ou alcançar um objetivo ou para satisfazer a um contrato, a um padrão, a uma especificação ou a outros documentos formalmente impostos.

Já MACAULAY (1996) possui uma definição bastante simples, na qual diz que requisito é simplesmente algo que o cliente necessita.

KOTONYA & SOMMERVILLE (1998) associam aos requisitos os principais problemas do desenvolvimento de software onde as principais dificuldades relatadas são: a dificuldade

em entender os requisitos; requisitos incompletos e/ou inconsistentes; e não refletirem as reais necessidades dos usuários, o que provoca retrabalho, atrasos, custos ultrapassados e a insatisfação dos clientes. Para BLASCHEK (2002) muitos desses erros poderiam ser evitados se as organizações dispusessem de um processo de engenharia de requisitos definido, controlado, medido e apropriado. No entanto, percebe-se que para muitos profissionais de informática esses conceitos não são claros, o que dificulta a ação dos gerentes no sentido de aprimorar seus processos de desenvolvimento.

Neste contexto, a SOFTEX (2009b) concorda com o cenário e traz no MPS.BR uma série de metas que guiam a empresa na formalização de seu processo de produção e auxilia assim, na solução dos problemas citados por KOTONYA & SOMMERVILLE (1998) e outros do mesmo domínio.

Considerando a importância de controlar os requisitos, o MPS.BR traz no seu primeiro nível de maturidade metas, ou resultados esperados, sobre a GRE, atividade na qual está baseada na Engenharia de Requisitos, que tem como principais atividades: elicitação, análise e negociação, especificação, validação e gerenciamento de requisitos (KOTONYA & SOMMERVILLE, 1998).

A Engenharia de Requisitos é um processo que envolve todas essas atividades para criar e manter o documento de requisitos do sistema. Os cinco resultados esperados para a GRE no nível G do MPS.BR são fundamentados nela.

O primeiro resultado esperado (GRE1) tem como objetivo elicitar os requisitos, analisá-los e negociá-los com o objetivo de esclarecer o escopo do projeto e entender melhor os requisitos, além de gerar um documento para que possa ser avaliado utilizando os critérios definidos pela empresa. Este resultado está presente em quatro das cinco atividades da ER, ficando de fora apenas do gerenciamento durante o processo.

Já o segundo (GRE2) tem como objetivo obter o comprometimento da equipe técnica com os requisitos aprovados obtidos pelo GRE1. A obtenção deste resultado inclui as atividades de análise e validação dos requisitos.

O terceiro resultado (GRE3) faz parte da modelagem dos requisitos, com o objetivo de mapear a relação entre os mesmos e entre os produtos de trabalho. Esta modelagem é de grande importância e útil para o gerenciamento durante o processo.

Os resultados GRE4 e GRE5, são responsáveis pelo monitoramento dos requisitos e o controle de mudanças, respectivamente. Resultados que são relacionados ao gerenciamento de requisitos.

Uma discussão mais detalhada sobre cada um dos resultados esperados do nível G para a GRE está descrita nas seções seguintes, nas quais também aponta-se algumas técnicas existentes que permitem alcançar os resultados e quais os benefícios que o cumprimento destes resultados traz a organização. Vale ressaltar, que estas atividades na prática não seguem uma ordem explícita de realização, elas fazem parte de um processo iterativo e incremental e que a empresa adotará, onde a ordem das técnicas adotadas pode ser alterada.

## **3.2 GRE1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos**

O objetivo deste resultado é estabelecer o uso de mecanismos para obter, com o fornecedor, o conhecimento dos requisitos, avaliá-los e validá-los. As atividades de concepção dos requisitos asseguram que a compreensão compartilhada e compatível dos requisitos foi atingida. Como comprovação do entendimento dos requisitos, deve-se ter um documento de requisitos, que pode ter diferentes formas de acordo com as necessidades da organização.

Após a identificação dos requisitos do produto e dos componentes do produto do projeto é necessário que estes sejam validados utilizando critérios objetivos pelas partes interessadas.

A seguir apresenta-se um conjunto de técnicas propostas pela comunidade de engenharia de requisitos que podem ser utilizadas neste resultado.

### **3.2.1 Técnicas de Levantamento de Requisitos**

Em seu trabalho, KENDALL (2010) afirma que uma boa comunicação com os fornecedores de requisitos é fundamental para assegurar um bom entendimento das necessidades do cliente e dos requisitos do projeto e, conseqüentemente, aumentar as chances de sucesso do projeto.

Existem diversas técnicas que auxiliam o levantamento de requisitos e que permitem a organização conseguir o entendimento necessário cobrado pelo MPS.BR. Algumas destas estão descritas a seguir.

#### **3.2.1.1 Entrevistas**

Segundo KENDALL (2010), “*Entrevista é uma conversa entre duas ou mais pessoas com uma finalidade específica*”. A entrevista é uma das técnicas mais tradicionais no processo de levantamento de requisitos, através dela que o responsável pela obtenção dos requisitos

conhece as opiniões, as expectativas e os problemas a serem tratados e é através dela que deve ser traçada a estratégia para o levantamento.

KENDALL (2010) expõe que as entrevistas devem seguir as seguintes etapas: Planejamento, Condução e Registro, podendo ser gravada ou não.

Antes de realizar a entrevista, o entrevistador deverá estabelecer os objetivos, especificando assim o que perguntar a formulação das perguntas e o tipo (objetivas, subjetivas, aprofundamento). Outro ponto importante é averiguar se o entrevistado é a pessoa mais indicada a responder as perguntas.

A formulação das perguntas deve ser feita conforme o objetivo esperado de quem entrevista. Para criar um ambiente mais espontâneo, obter riqueza de detalhes use perguntas subjetivas, contudo, o uso desta técnica pode fornecer detalhes irrelevantes e levar a perda de tempo e do controle da entrevista. Já para limitar as respostas possíveis ou ganhar tempo utilize questões objetivas, e questões de aprofundamento para a necessidade de detalhamento a respeito de algum tópico. Conforme GOGUEN & LINDE (1993), a utilização das questões anteriormente comentadas deve ser estruturada visando uma forma de melhor conduzir a entrevista.

Após a entrevista é necessário validar se o que foi documentado pelo analista está de acordo com a necessidade do usuário, que o usuário não mudou de opinião e que o usuário entende a notação ou representação gráfica de suas informações. Identificar o validador e o fornecedor dessas informações é uma das boas práticas cobradas pelo MPS.BR.

A entrevista permite que este resultado seja amplamente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos e depois serem aceitos junto aos fornecedores de requisitos.

### **3.2.1.2 Questionários**

Em alguns casos, como a indisponibilidade física ou com a dispersão das pessoas envolvidas, o questionário é a uma boa forma de levantamento, além de, segundo KENDALL (2010), ser uma técnica de levantamento que permite obter informações de várias pessoas afetadas pelo sistema como, obter um *feedback* a respeito de problemas ou identificar possíveis melhorias.

Em seu trabalho PARASURAMAN (1991) mostra que a elaboração de um questionário é um processo mais complexo do que possa aparentar. Um questionário mal formulado pode levar a considerações erradas, o que acaba sendo prejudicial ao projeto, daí a importância de

fazer um planejamento adequado visando o conteúdo, formato e ordem das questões. Um questionário deve possuir questões claras e sem ambiguidade, ter fluxo definido, prever antecipadamente dúvidas que possam surgir e evitar ser muito extenso, pois pode ocasionar desinteresse das pessoas que irão respondê-lo.

A forma de entrega ou apresentação do questionário também é abordada por PARASURAMAN (1991). Uma entrega via correio ou e-mail, onde sua principal vantagem, é o envio a um grande número de pessoas, tem provada a baixa taxa de respostas, obrigando assim ao remetente enviar uma quantidade de e-mails/cartas maior do que a quantidade desejada para o mínimo de respostas. Já a entrega em grupo, é um método que reúne um grupo de pessoas e as perguntas são feitas simultaneamente, mas cada um responde o questionário individualmente, a vantagem é que se obtêm as respostas rapidamente.

Ao analisar as respostas dos participantes é feito uma consolidação das informações fornecidas no questionário, documentando as principais descobertas e enviando uma cópia com estas informações para o participante como forma de consideração pelo tempo dedicado a pesquisa. Cabe ao responsável pela pesquisa, escolher a melhor técnica conforme a situação encontrada.

O questionário permite que este resultado seja parcialmente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos, porém, dificulta que estes sejam aceitos junto aos fornecedores de requisitos. Para que os fornecedores possam validar os requisitos levantados, o documento de requisitos criado deve ser enviado aos fornecedores, para que estes possam avaliar e enviar uma resposta se concorda ou não com os requisitos. Este processo pode ser demorado para alguns casos, o que pode ser prejudicial para a empresa.

### **3.2.1.3 *Joint Application Development (JAD)***

A técnica denominada por *Joint Application Development (JAD)*, desenvolvida pela IBM, é uma reunião realizada com as partes interessadas, sua finalidade é elicitare dúvidas que são levantadas em áreas distintas como Banco de Dados, Desenvolvimento e Usuários do Sistema.

Segundo PADUA (2001), esta técnica apresenta as seguintes vantagens: obtêm um maior comprometimento dos usuários com os requisitos; reduz o tempo necessário ao levantamento da especificação dos requisitos; retira do projeto requisitos de valor questionável; esclarece

dúvidas de interpretação dos requisitos entre usuários e desenvolvedores; identifica e discute o mais cedo possível, problemas políticos que possam interferir no projeto.

PRESSMAN (1996) atribui à JAD uma abordagem popular da técnica *Facilitated Application Specification Techniques* (FAST), na qual pede aos participantes que elaborem uma lista dos objetos do sistema, dos que são produzidos pelo sistema e dos objetos que são usados para que o sistema execute suas funções, além de uma lista de operações, que manipulam ou interagem com os objetos, ou seja, levantar anteriormente, o máximo de informações ou expectativas a respeito do que será discutido na reunião.

Conforme PADUA (2001), a técnica JAD deve ser aplicada com cuidado, pois existem pontos que podem comprometer sua eficácia como: o não envolvimento de pessoas que desempenham papéis-chaves no processo de uso do produto; A participação de pessoas não comprometidas com o produto e o número excessivo de participantes, além disso, o líder do JAD deve possuir experiência e envolvimento com os participantes.

JADs de sucesso mostram um índice de mudança de requisitos baixíssimo e a redução de tempo e esforço gastos no levantamento de requisitos, o que a torna uma técnica a ser considerada. Porém, a maioria das técnicas JAD funciona melhor em projetos pequenos ou médios. Para um sistema grande e complexo podem ser usadas múltiplas sessões JAD para acelerar a definição dos requisitos do sistema.

A técnica JAD permite que este resultado seja amplamente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos e depois serem aceitos junto aos fornecedores de requisitos.

#### **3.2.1.4 Brainstorming**

Segundo LEFFINGWELL (2003), *brainstorming* (tempestade de ideias) é uma técnica para geração de ideias. Ela consiste em uma ou várias reuniões que permitem que as pessoas sugiram e explorem ideias. É muito utilizada para a criação de novos produtos ou efetuar melhoramentos nos existentes e solucionar de problemas.

De autoria de Alex Faickney Osborn, prega a técnica, que se reúnam de uma até dez pessoas de preferência de setores e competências diferentes, para que o grupo possua diversas visões. O mais importante é que nenhuma ideia deve ser julgada como errada ou absurda, contribuindo assim para que sejam geradas muitas ideias e principalmente não constranger os integrantes. Os participantes também devem ser encorajados a combinar ou enriquecer as

ideias de outros e, para isso, é necessário que todas as ideias permaneçam visíveis a todos os participantes.

Nesta técnica é designada uma pessoa para registrar todas as ideias em uma lousa ou em papel. À medida que cada folha de papel é preenchida, ela é colocada de forma que todos os participantes possam vê-la.

Analisar as ideias é a fase final do *brainstorming*. Nessa fase é realizada uma revisão das ideias, uma de cada vez. As consideradas valiosas pelo grupo são mantidas e classificadas em ordem de prioridade.

O *brainstorming* é uma técnica mais utilizada para criação de novos produtos, situação na qual, pode ainda não existir um cliente ou fornecedor específico. Por isso ela permite que este resultado seja parcialmente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos, porém, dificulta que estes sejam aceitos junto aos fornecedores de requisitos. Para que os fornecedores possam validar os requisitos levantados, a empresa deve escolher um método que possibilite a avaliação do cliente. A prototipagem é uma técnica recomendada para essa atividade.

### 3.2.1.5 Etnografia

A etnografia é uma técnica de observação que permite entender a política organizacional bem como a cultura de trabalho da organização com objetivo de familiarizar-se com o sistema e sua história (GOGUEN & LINDE, 1993).

Nesta técnica, o analista se insere no ambiente de trabalho em que o sistema será utilizado. O trabalho diário é observado e são anotadas as tarefas reais em que o sistema será utilizado. O principal objetivo da etnografia é de descobrir requisitos de sistema implícitos, que refletem os processos reais, em vez de os processos formais, onde as pessoas estão envolvidas.

Para KENDALL (2010), alguns itens importantes que devem ser executados antes, durante e depois do estudo de observação:

- **Antes:** é indispensável identificar as áreas dos usuário a serem observadas; obter a aprovação das gerências apropriadas para executar as observações; obter os nomes e funções das pessoas chave que estão envolvidas no estudo de observação; e explicar a finalidade do estudo;
- **Durante:** é necessário familiarizar-se com o local de trabalho que está sendo observado; observar os agrupamentos organizacionais; as facilidades manuais e

automatizadas; coletar amostras de documentos e procedimentos escritos que são utilizados em cada processo; e acumular informações estatísticas a respeito das tarefas, como: frequência que ocorrem, estimativas de volumes, tempo de duração para cada pessoa que está sendo observada. Além de observar as operações normais de negócios acima é importante observar as exceções;

- **Depois:** é preciso documentar as descobertas resultantes das observações feitas. Para consolidar o resultado é preciso rever os resultados com as pessoas observadas e/ou com seus superiores.

A análise de observação tem algumas desvantagens como, consumir bastante tempo e o analista ser induzido a erros em suas observações. Mas em geral a técnica de observação é muito útil e frequentemente usada para complementar descobertas obtidas por outras técnicas.

A etnografia permite que este resultado seja amplamente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos e depois serem aceitos junto aos fornecedores de requisitos.

### **3.2.1.6 Prototipagem**

A prototipação é uma técnica que permite a obtenção rápida de sugestões, inovações, revisões ou mudanças e permite captar o *feedback* do usuário a respeito do sistema em desenvolvimento.

Protótipos tem por objetivo explorar aspectos críticos dos requisitos de um produto, implementando de forma rápida um pequeno subconjunto de funcionalidades deste produto. O protótipo é indicado para estudar as alternativas de interface do usuário; problemas de comunicação com outros produtos; e a viabilidade de atendimento dos requisitos de desempenho (KENDALL, 2010).

Em sua construção, podem ser utilizadas ferramentas de programação que possuam biblioteca de componentes, processadores de texto, ferramentas de manipulação de imagens ou até mesmo um lápis e papel.

Alguns dos benefícios da prototipagem são as reduções dos riscos na construção do sistema, pois o usuário chave já verificou o que o analista captou nos requisitos do produto.

A prototipagem permite que este resultado seja amplamente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos e depois serem aceitos junto aos fornecedores de requisitos.

### **3.2.1.7 Pesquisa de mercado (SWOT)**

A Pesquisa de mercado, análise de competidores, ou SWOT (*Strengths, Weaknesses, Opportunities, Threats*), foi criada por dois professores de Havard, Kenneth Andrews e Roland Christensen. Consiste em analisar os pontos fortes e fracos além de ameaças e oportunidades, com fins de se projetar um produto que maximize os pontos fortes levantados e minimizar os pontos fracos encontrados nos produtos semelhantes existentes, resumindo (FINE, 2009).

Existem alguns pontos que devem ser observados como: verificar se as informações são recentes e/ou isentas; as fontes devem ser adequadas e desprovidas de viés; todos os participantes devem conhecer os conceitos envolvidos;

O *SWOT* é uma técnica mais utilizada para criação de novos produtos, situação na qual, pode ainda não existir um cliente ou fornecedor específico. Por isso ela permite que este resultado seja parcialmente atendido, possibilitando o entendimento dos requisitos, documentá-los para uma avaliação utilizando os critérios previamente definidos, porém, dificulta que estes sejam aceitos junto aos fornecedores de requisitos. Para que os fornecedores possam validar os requisitos levantados, a empresa deve escolher um método que possibilite a avaliação do cliente. A prototipagem é uma técnica recomendada para essa atividade.

### **3.2.2 Validar Requisitos utilizando critérios objetivos**

Após listados os requisitos é preciso garantir que os requisitos listados atendam às necessidades e expectativas do cliente e dos usuários. Para isso, os requisitos identificados devem ser validados seguindo critérios objetivos, previamente estabelecidos. Alguns exemplos de critérios são: possuir identificação única, estar claro, não ser ambíguo, ser relevante, ser completo, estar consistente com os demais requisitos, ser implementável, testável e rastreável (IEEE std 830, 1998).

A organização pode utilizar os critérios sugeridos pela IEEE std 830 (1998), ou utilizar novos critérios de acordo com as necessidades da organização.

## **3.3 GRE2 - Obtenção do comprometimento da equipe técnica com os requisitos aprovados**

Após a aprovação dos requisitos, um comprometimento formal da equipe técnica com os requisitos aprovados deve ser obtido e registrado. Por isso, o gerente do projeto deve

apresentar para a equipe quais foram os requisitos definidos e deixar claro o que deverá ser produzido.

Essa prática é importante para manter a integridade do trabalho da empresa. Fábricas de softwares não são formadas por grandes maquinários, sua produção é fruto do recurso humano presente na organização. Obter o compromisso dessas pessoas significa obter o compromisso da empresa.

A mesma empresa pode utilizar várias formas de registro dependendo da ocasião. Este registro pode ser, por exemplo, um e-mail, ata de reunião ou algum outro mecanismo que permita o registro. Mudanças nos requisitos aprovados podem afetar nos compromissos já estabelecidos pela equipe. Nesse caso, um novo comprometimento deve ser obtido em relação aos requisitos modificados.

A formalidade desse registro pode mudar de acordo com a política organizacional da empresa, ou de acordo com a metodologia de gerência de projeto e de desenvolvimento de software. Empresas que exijam um registro mais formal ou utilizem uma forma de desenvolvimento mais restrita precisam de uma maior formalidade do que empresas que não se preocupam tanto com o registro em si ou que utilizam metodologias mais livres, metodologias ágeis por exemplo. Independentemente da formalidade ou não o registro da equipe técnica ainda deve existir.

Os meios mais conhecidos e utilizados de registro estão descritos a seguir.

### **3.3.1 Reuniões**

Uma prática bastante usada nas empresas é a reunião das pessoas participantes do projeto. O gerente de projeto pode requerer uma reunião onde será apresentado os requisitos acordados, após a exposição das informações, discuti-las e esclarecer possíveis dúvidas. Por fim coletar a assinatura dos presentes em ata de reunião e usá-la como evidência do compromisso dos membros participantes. A vantagem da reunião é o fato de todos estarem juntos e permitir um melhor esclarecimento e um registro de compromisso de uma forma mais rápida, porém, ela é de um custo maior pra empresa, que precisa reservar um horário vago para todos além de ter a garantia de que todos tem a disponibilidade de estar presente.

### **3.3.2 E-mail**

Uma forma mais barata de comunicado e que permite o registro do compromisso é o e-mail. Através dele o gerente pode esclarecer o plano de projeto, que pode estar em anexo ou

de forma resumida no corpo do e-mail, e salvar a resposta como forma de registro do compromisso. Para isso o gerente precisa enviar um e-mail para todos os participantes do projeto, nele deve conter todas as informações do projeto de forma clara e um aviso lembrando que o e-mail deve ser respondido como forma de entendimento. O e-mail tem um custo praticamente nulo e permite o comunicado para pessoas com indisponibilidade física e quando há uma dispersão das pessoas envolvidas, porém, a garantia de que a pessoa recebeu o e-mail, irá responder dentro do prazo e que está realmente entendeu o projeto são ameaçadas.

### **3.3.3 Contratos**

Uma maneira de registrar o acordo com uma maior formalidade é através de contratos. Um contrato é um vínculo jurídico entre dois ou mais sujeitos de direito correspondido pela vontade, da responsabilidade do ato firmado, resguardado pela segurança jurídica em seu equilíbrio social, ou seja, é um negócio jurídico bilateral ou plurilateral. É o acordo de vontades, capaz de criar, modificar ou extinguir direitos.

As cláusulas contratuais criam lei entre as partes, porém são subordinados ao Direito Positivo (princípios e regras que regem a vida social). As cláusulas contratuais não podem estar em desconformidade com o Direito Positivo, sob pena de serem nulas. No Brasil, cláusulas consideradas abusivas ou fraudulentas podem ser invalidadas pelo juiz, sem que o contrato inteiro seja invalidado.

Geralmente, contratos são mais usados quando terceiros participam do desenvolvimento, onde sua “quebra” pode gerar em multa ou outro tipo de ônus.

## **3.4 GRE3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida**

A intenção da rastreabilidade é manter o controle dos requisitos, desde a fase de obtenção até a sua liberação, ver como estes se relacionam com os demais requisitos (rastreabilidade horizontal) e com os artefatos que são gerados no decorrer do projeto (rastreabilidade vertical).

A rastreabilidade deve ser feita de forma bidirecional, ou seja, é possível verificar todo o processo tanto “para trás” quanto “para frente”. De forma mais clara, a rastreabilidade bidirecional na vertical, permite acompanhar o requisito desde o escopo do projeto, passando

pelo documento de requisitos e outros, até chegar ao código onde este foi implementado. Já a rastreabilidade bidirecional na horizontal torna possível analisar todos os requisitos que influenciam naquele que estamos avaliando e ver quais este influencia.

Em seu trabalho TORANZO (2002) mostra que uma mudança de requisito implicaria em modificações nos demais requisitos que estão relacionados àquele modificado. Assim, com os requisitos devidamente mapeados qualquer mudança que ocorra pode ter seu impacto calculado devido à rastreabilidade.

Ainda em seu trabalho, TORANZO (2002) propôs uma classificação das informações a serem rastreadas em quatro níveis: ambiental, organizacional, gerencial e de desenvolvimento. O nível de informação ambiental representa as informações do contexto político, econômico e padrão (normas) que são externas à organização e que podem afetar alguns dos seus sistemas. O nível de informação organizacional representa as informações (recurso, processo, objetivo, regra, etc.) que podem afetar o desenvolvimento dos sistemas da própria organização. O nível de informação gerencial representa algumas das informações (tarefa, objetivo, restrição, etc.) empregadas pela gerência de projetos. Finalmente, o nível de informação de desenvolvimento inclui as informações que representam os diferentes elementos/artefatos produzidos no desenvolvimento de software, por exemplo, requisito, documento, diagrama e programa.

A classificação fornece dois grandes benefícios. Primeiro, a identificação, elicitação e validação das informações rastreadas podem ser organizadas e entendidas em função dos quatro níveis. Segundo, a classificação é independente do pré e pós-rastreamento e pode ser aplicada sobre as duas. Embora a estratégia proposta por TORANZO (2002) seja ampla e apresente benefícios claros, esta proposta não a toma como aplicável para uma empresa que busca o nível G do MPS.BR, por ser muito complexa e de difícil aplicabilidade. Geralmente as empresas que buscam um sistema de rastreabilidade para atender GRE3 optam por estratégias mais simples para uma melhor adaptação evitando assim um impacto cultural na empresa. Porém, toma a proposta de TORANZO (2002) é apta para empresas que já apresentem familiaridade com um sistema de rastreamento ou buscam altos níveis de maturidade.

O estudo de quais requisitos são afetados com a mudança é indispensável para saber se a mudança é viável ou não para a continuação do projeto. Por esse motivo a rastreabilidade deve proporcionar uma análise de qual o impacto que a mudança causaria.

É importante ressaltar que este resultado estabelece a criação de um sistema de rastreamento e que não necessariamente envolve a criação de uma matriz de rastreabilidade

específica para atendimento ao resultado esperado. Contudo, deve existir um mecanismo que possibilite a realização da rastreabilidade bidirecional entre os requisitos e os demais produtos de trabalho (SOFTEX, 2009b).

O grau de descrição dos elementos do rastreamento indicará com que capacidade pode-se estabelecer e manter o rastreamento. Pois, quanto maior o número de informações na descrição dos elementos de rastreamento mais fácil fica para estabelecer esse processo, levando-se em consideração que o rastreamento deve ser feito de forma detalhada para que não haja falhas no processo.

O rastreamento pode ser feito de várias formas sendo recomendado o uso de uma ferramenta para o auxílio desse controle. Existem várias ferramentas voltadas para a rastreabilidade, tanto livres quanto proprietárias, ou ainda, a empresa pode produzir a sua própria ferramenta para uma melhor adaptação, caso necessário. Embora uma ferramenta seja extremamente importante para auxiliar a rastreabilidade, esta deve apenas servir como auxílio para uma estratégia predefinida e nunca ao contrário. Na próxima Seção apresentamos os tipos de rastreamento existentes.

### **3.4.1 Tipos de Rastreamento**

Existem diversas maneiras de realizar um rastreamento de requisitos, levando-se em consideração:

- A direção do rastreamento, um requisito pode ser rastreado “para frente” ou “para trás”;
- A evolução do requisito, um requisito pode ser rastreado para aspectos que ocorrem “antes” ou “depois” da sua inclusão na especificação de requisitos;
- Os tipos dos objetos envolvidos, tem-se um rastreamento “interno” ou “externo”.

#### **3.4.1.1 Rastreamento Para Frente e Para Trás**

De acordo com WIERINGA (1995) as definições de rastreamento para frente e para trás são as seguintes:

- *“Rastreamento para frente é a habilidade de rastrear um requisito para componentes de design ou implementação.”*
- *“Rastreamento para trás é a habilidade de rastrear um requisito para sua fonte, isto é, para uma pessoa, instituição, lei, argumento, etc.”*

O rastreamento deve ser realizado para frente quando, por exemplo, um requisito é alterado e queremos investigar o impacto da mudança nos produtos de trabalho.

Podem-se rastrear quais classes ou testes sofrem com a alteração do requisito, quais linhas de código precisam ser alteradas ou quais documentos precisam ser revisados e ajustados.

O rastreamento deve ser realizado para trás quando, por exemplo, ocorre uma mudança e queremos compreendê-la, investigando a informação utilizada para elicitado o requisito alterado.

Podem-se rastrear quais as pessoas interessadas na alteração do requisito ou a partir de quais documentos o requisito foi extraído ou ainda, com quais departamentos da organização o requisito está relacionado.

#### **3.4.1.2 Rastreamento Pré-Especificação e Pós-Especificação de Requisitos**

Uma especificação de requisitos é o resultado de um processo de elicitação. Segundo GOTEL & FINKELSTEIN (1994) o rastreamento de um requisito pode ser realizado de duas formas: localizando informação sobre o processo de elicitação, antes da conclusão da especificação de requisitos, ou localizando informação relacionada ao seu uso, depois do requisito ter sido elicitado e incluído na especificação de requisitos.

O rastreamento pré-especificação de requisitos é útil, por exemplo, quando ocorre uma mudança num requisito e queremos localizar as fontes dos requisitos ou as pessoas responsáveis a fim de validar a mudança, descobrir como o requisito foi produzido.

O rastreamento pós-especificação de requisitos é útil, por exemplo, para localizar o módulo de design no qual o requisito foi alocado ou os procedimentos de teste criados para verificar o requisito, descobrir como ele foi utilizado.

#### **3.4.1.3 Extra-Rastreamento e Intra-Rastreamento de Requisitos**

Durante a elaboração de uma especificação, os requisitos são comumente apurados, derivados e alguns são descartados. A habilidade de rastrear os relacionamentos entre os requisitos é chamada de intra-rastreamento de requisitos, ou rastreabilidade horizontal, a qual é importante para a análise e tratamento de mudanças e evolução dos requisitos. Sendo possível, observar através dessa rastreabilidade quais requisitos são afetados por alguma mudança de requisito.

Já os relacionamentos entre os requisitos e outros artefatos são capturados pelo extra-rastreamento de requisitos, ou rastreabilidade vertical, que é importante para a análise e

tratamento de mudanças e evolução dos produtos de trabalho. Sendo possível, observar através dessa rastreabilidade quais artefatos sofrem por alguma mudança de requisito.

### **3.5 GRE4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos**

O objetivo desse resultado é a definição de um método que descreva como a organização fará revisões em planos e produtos de trabalho do projeto, visando identificar e corrigir inconsistências em relação aos requisitos. Tais inconsistências devem ser registradas, avaliadas e acompanhadas até que sejam resolvidas.

Assim essa tarefa tem como objetivo principal controlar a evolução dos requisitos, seja por constatação de novas necessidades, seja por constatação de deficiências nos requisitos registrados até o momento (THAYER & DORFMAN, 1997). A Tabela 2 apresenta o conjunto de atividades de um processo de monitoramento de requisitos. Como mostra o trabalho de HAZAN & LEITE (2003), estas atividades visam apoiar a identificação, controle e rastreamento dos requisitos, bem como o tratamento das mudanças nos requisitos.

Segundo KOTONYA & SOMMERVILLE (1998), as principais preocupações de monitorar de requisitos são: Gerenciar mudanças nos requisitos acordados; Gerenciar os relacionamentos entre os requisitos; Gerenciar as dependências entre o documento de requisitos e outros documentos produzidos ao longo do processo.

Tabela 2. Atividades de um Processo de Gerência de Requisitos (HAZAN & LEITE, 2003)

<b>ATIVIDADES</b>	<b>DESCRIÇÃO</b>
Receber as solicitações de alteração de requisitos.	O grupo de engenharia de requisitos recebe as solicitações de alteração de requisitos, ou por formulário padronizado, ou por meio de um sistema de solicitação de demandas.
Registrar novos requisitos	Novos requisitos também devem ser recebidos formalmente, seja por formulário padronizado, ou por meio de controle sistemático.
Analisar impacto da mudança de requisitos	Uma análise criteriosa deve ser conduzida para avaliar o impacto do requisito a ser incluído, alterado ou excluído sobre cada um dos seus requisitos relacionados, os quais podem ser identificados pelos métodos citados no GRE3. Caso o impacto seja significativo, os requisitos (analisado e relacionado) devem ser revistos.

Elaborar relatório de impacto	Deve ser mantido um histórico de alterações para cada requisito, permitindo uma visão cronológica das principais mudanças nos requisitos.
Notificar os envolvidos	Os envolvidos são um conjunto de pessoas para as quais pode haver um impacto devido à alteração de requisitos (alteração, inclusão ou exclusão de requisitos) e devem ser notificados.
Coletar métricas	As métricas devem ser utilizadas e coletadas periodicamente para o acompanhamento das atividades de Gerência de Requisitos.

Exemplos de revisões com esses objetivos são: revisões de monitoração e controle do projeto; e inspeções baseadas em critérios explícitos para identificar inconsistências entre os planos, atividades e produtos de trabalho. Estas atividades de revisão podem ser feitas ao mesmo tempo pelo gerente de projeto ou pelo responsável pelas revisões, a prática de unir o monitoramento do projeto e o monitoramento dos requisitos pode trazer uma agilidade à organização.

Algumas técnicas e práticas sugeridas pelo PMBOK (*Project Management Body of Knowledge*) ou algumas já utilizadas para a Gerência de Projetos pela empresa podem dar suporte para essa atividade. Como Gerenciamento de integração, Gerenciamento do escopo, Gerenciamento de tempo do projeto, Gerenciamento de custos do projeto, Gerenciamento de recursos humanos, Gerenciamento das comunicações do projeto e Gerenciamento de riscos.

### **3.5.1 Técnicas existentes**

A seguir apresenta-se um conjunto de técnicas propostas pela comunidade de engenharia de requisitos que podem ser utilizadas neste resultado. O uso dessas técnicas pode ajudar a empresa guiando o processo de revisões, porém, a empresa está livre para criar o seu próprio método de acordo com as necessidades da organização.

#### **3.5.1.1 Leitura Baseada em perspectiva (PBR)**

A técnica de leitura baseada em perspectivas fornece um conjunto efetivo de procedimentos que os revisores podem seguir para detecção dos defeitos na inspeção de documentos de especificação de requisitos.

Para a aplicação da técnica é preciso identificar diferentes usuários do documento de requisitos. A seleção desses papéis varia de acordo com a necessidade do projeto ou da

organização, sendo geralmente utilizados o usuário, o projetista e o testador (FREITAS, 2003).

Cada usuário do documento pode considerar determinadas características da informação em um grau diferente de importância, dependendo do uso que faz dela dentro do ciclo de desenvolvimento do produto.

Uma vez determinadas tais perspectivas, os revisores que atuam em cada uma avaliam o documento sob seus diferentes pontos de vista, isto é, assumem a perspectiva de um usuário específico do documento de especificação de requisitos (projetista, testador e usuário final). Dessa forma, cada inspetor recebe um cenário, além de uma descrição procedural das atividades para direcionar seu trabalho de inspeção, segundo cada perspectiva.

Utiliza-se uma taxonomia para classificação dos defeitos encontrados. Cada defeito pode ser atribuído a mais de uma categoria e novas classes podem ser criadas conforme necessidades específicas. As perspectivas cobrem, coletivamente, todos os aspectos relevantes do documento. O objetivo é evitar duplicação de trabalho, fazendo com que cada revisor busque por defeitos que influenciarão no uso do documento pelo seu respectivo papel. A inspeção através destas perspectivas garante uma completa cobertura do documento, eliminando sobreposição de trabalho e ausência de inspeção em pontos do documento (FREITAS, 2003).

### **3.5.1.2 Revisões Formais**

Uma revisão formal de requisitos é uma sessão de trabalho em grupo frequentada pelas partes interessadas do projeto, durante a qual os requisitos são verificados em termos de precisão, relevância, clareza, probabilidade de execução e conformidade com os padrões da organização. As revisões de requisitos são especialmente importantes nos estágios iniciais de um projeto, mas podem ser conduzidas a qualquer momento do ciclo de vida do projeto, de acordo com a necessidade.

A audiência para uma revisão dos requisitos consiste das partes interessadas, que podem incluir os patrocinadores do projeto e os usuários dos recebíveis do projeto. Participantes apropriados incluem as partes interessadas (ou seus representantes) que estavam envolvidas na especificação dos requisitos e os acionistas (ou seus representantes) que irão trabalhar na construção da solução, baseando-se nos requisitos.

De forma ideal, os participantes de uma revisão de requisitos deveriam ler e entender a documentação dos requisitos previamente à reunião de revisão a fim de que o tempo seja

empregado para focar questões que necessitam de uma resposta e mudanças que podem ser feitas nos requisitos. Isto requer o recebimento, por parte dos participantes, de um documento de requisitos.

Como a revisão dos requisitos é uma sessão formal que deve ser frequentada por partes interessadas influentes, os analistas de negócios precisam preparar-se para revisar cuidadosamente cada detalhe. Detalhes mínimos, como avisar com grande antecedência os convidados, guiar os participantes por todo o processo de revisão (se necessário) e assegurar-se de que os participantes compreendem seus papéis e os objetivos da sessão são itens importantes para o sucesso da revisão.

A sessão de revisão é conduzida juntamente com as diretrizes de uma apresentação formal. É importante criar anotações da reunião ou da sessão para documentar as discussões e decisões acordadas. Geralmente, poderá haver algumas mudanças na documentação dos requisitos ou aos próprios requisitos.

### **3.6 GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto**

Mudanças nos requisitos são inevitáveis no processo de desenvolvimento, por isso, requisitos podem ser modificados, adicionados ou retirados do projeto a qualquer momento. O controle dessas mudanças é o objetivo deste resultado, que pede que a empresa estabeleça e mantenha um plano com atividades, recursos e responsabilidades definidas para o controle das mudanças de requisitos.

Segundo HAZAN & LEITE (2003), estas mudanças ocorrem enquanto os requisitos estão sendo elicitados, analisados e após o sistema ter entrado em produção, sendo resultantes da combinação de fatores descritos na Tabela 3.

Tabela 3. Fatores de Mudanças de Requisitos (HAZAN & LEITE, 2003)

<b>Fator de Mudança</b>	<b>Descrição</b>
Erros em requisitos, conflitos e inconsistências.	Conforme os requisitos são analisados e implementados, erros e inconsistências surgem e devem ser corrigidas.
Evolução do conhecimento do cliente	Conforme os requisitos são desenvolvidos, clientes e usuários finais desenvolvem uma melhor compreensão do que desejam.
Problemas técnicos, de custo ou cronograma.	Problemas podem ser encontrados na implementação dos requisitos. Pode ser muito custoso implementar certos

	requisitos.
Mudanças nas prioridades do cliente	As prioridades do cliente podem mudar durante o desenvolvimento do sistema como resultado de mudanças no ambiente de negócios.
Mudanças de ambiente	O ambiente no qual o sistema será instalado pode mudar, assim os requisitos devem ser modificados para manter compatibilidade.
Mudanças organizacionais	A Organização que pretende usar o sistema pode mudar sua estrutura e processos, resultando em novos requisitos de sistema.

Segundo SOMMERVILLE (2003), para realizar um gerenciamento de mudanças eficiente é preciso conhecer os estágios principais do processo, como mostra a Figura 2:

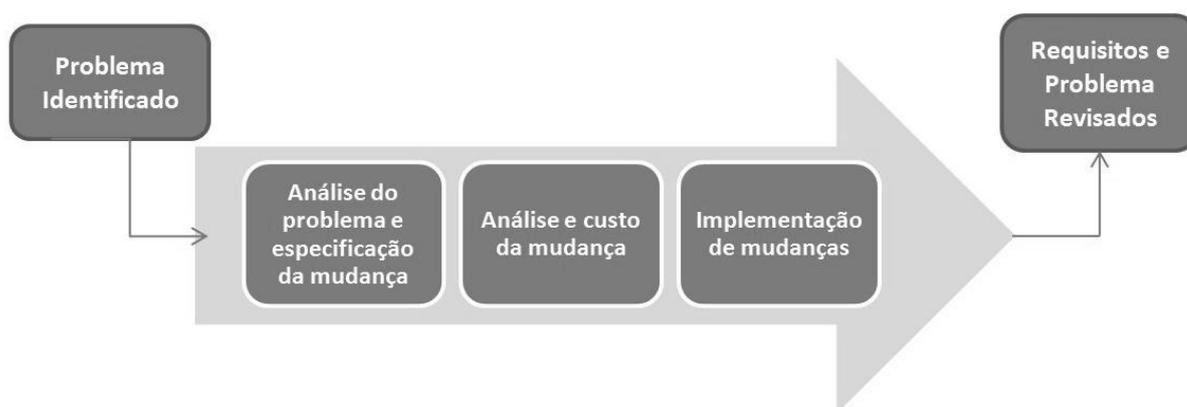


Figura 2. Processo de gerenciamento de mudanças. (SOMMERVILLE, 2003, p. 122)

**1. Análise do problema e especificação da mudança:** começa quando é identificado um problema no requisito. O problema é analisado e é verificada a sua validade.

**2. Análise e custo da mudança:** o impacto da mudança é avaliado e seu custo é estimado; a prática da rastreabilidade de requisitos (GRE3) é fundamental para o auxílio dessa análise. Com essa análise toma-se possível tomar a decisão de continuar ou não com a alteração do requisito.

**3. Implementação de mudanças:** procede-se com a alteração no documento de requisitos, projeto e implementação (se necessários), de forma que as alterações possam ser incorporadas sem muito esforço.

Ter definido o processo de gerenciamento de mudanças é importante, assim, para toda e qualquer mudança proposta nos requisitos o método definido deve ser utilizado para controlá-las. A vantagem desta definição é que como há um processo que monitora as mudanças e exige que essas sejam documentadas, as mudanças nos requisitos são realizadas de forma controlada.

Algumas técnicas e práticas sugeridas pelo PMBOK (*Project Management Body of Knowledge*) ou algumas já utilizadas para a Gerência de Projetos pela empresa podem dar suporte para essa atividade. Como Gerenciamento de integração, Gerenciamento do escopo, Gerenciamento de tempo do projeto, Gerenciamento de custos do projeto, Gerenciamento de recursos humanos, Gerenciamento das comunicações do projeto e Gerenciamento de riscos.

### **3.7 Considerações Finais**

Este capítulo apresentou uma visão geral a respeito da Gerência de requisitos. Foram apresentados os cinco (5) resultados esperados para GRE do nível G do MPS.BR e diversas técnicas que possibilitam a organização alcançar estes resultados, citando suas principais fases e características.

Uma organização que deseja empregar boas práticas de engenharia de software deve considerar a possibilidade de incluir as atividades indicadas nesse capítulo dentro das suas atividades de desenvolvimento. Para isso, é necessário que estas sejam vistas como atividades tão importantes como as outras atividades do desenvolvimento, mesmo que o custo do investimento nelas seja alto e talvez não tenha um retorno tão imediato. O recomendável é sempre analisar os requisitos durante o processo de desenvolvimento, e não depois, ou apenas quando existir um problema ou proposta de mudança.

No próximo capítulo, tendo como base os estudos realizados nesse capítulo, será apresentada a proposta de GRE que pretende atender as exigências do nível G do MPS.BR. O estudo agora é voltado para de comparação dessas técnicas no âmbito das MPEs, a fim de capturar as principais características e qualidades que são interessantes e desejáveis no contexto da Engenharia de Requisitos, e que servirão de base para a elaboração da nova proposta de GRE. Por fim, aplica-se a nova abordagem gerada a um estudo de caso.

## **Capítulo 4**

# **Proposta de um processo de GRE para satisfação do MPS.BR**

Este capítulo apresenta a proposta do trabalho iniciando com um esclarecimento sobre o domínio das MPE (Seção 4.1). Logo a seguir, apresenta-se algumas técnicas que podem ser utilizadas para atender os cinco resultados esperados pela GRE do MPS.BR nível G (Seções 4.2 a 4.6) e quais estratégias de processo para Implantação da Engenharia de Requisitos podem ser usadas para o sucesso da implantação das técnicas (Seção 4.7). Por fim, a Seção 4.8 é destinada às considerações finais.

### **4.1 Introdução**

É importante ressaltar que a proposta exposta a seguir é baseada nas técnicas citadas no Capítulo 3, é destinada, mas não restrita, às micro e pequenas empresas.

De acordo com órgãos competentes, como o SEBRAE (Serviço Brasileiro de Apoio às Micro e Pequenas Empresas) e o IBGE (Instituto Brasileiro de Geografia e Estatística), determina-se micro e pequena empresa por algumas características em comum, tais como: capital da empresa, faturamento e número de empregados. Para este estudo será adotado o critério de número de empregados, conjugado com o faturamento líquido da empresa, pois se entende que em projetos de software o número de pessoas envolvidas é relevante no que se refere a riscos, e seu faturamento poderá ser considerado no que diz respeito aos investimentos que cada tipo de empresa poderá fazer para melhorar seus processos.

Segundo informações do SEBRAE (2010), a adoção de critérios para a definição de tamanho de empresa constitui importante fator de apoio às MPEs, isto permite que as empresas classificadas dentro dos limites estabelecidos possam usufruir os benefícios e

incentivos previstos nas legislações que dispõem sobre o tratamento diferenciado ao segmento, e que buscam alcançar objetivos prioritários de políticas públicas, como o aumento das exportações, a geração de emprego e renda, a diminuição da informalidade dos pequenos negócios, entre outras.

No Estatuto de 1999, o critério adotado para conceituar MPE é a receita bruta anual, cujos valores foram atualizados pelo Decreto nº 5.028/2004, de 31 de março de 2004, que corrigiu os limites originalmente estabelecidos (R\$ 244.000,00 e R\$ 1.200.000,00, respectivamente). Os limites atuais são os seguintes:

- **Microempresa:** receita bruta anual igual ou inferior a R\$ 433.755,14 (quatrocentos e trinta e três mil, setecentos e cinquenta e cinco reais e quatorze centavos);

- **Empresa de Pequeno Porte:** receita bruta anual superior a R\$ 433.755,14 e igual ou inferior a R\$ 2.133.222,00 (dois milhões, cento e trinta e três mil, duzentos e vinte e dois Reais).

Além do critério adotado no Estatuto, o SEBRAE utiliza ainda o conceito de pessoas ocupadas nas empresas, principalmente nos estudos e levantamentos sobre a presença da MPE na economia brasileira, conforme os seguintes números:

- **Microempresa:**

1. Na indústria e construção: até 19 pessoas ocupadas;
2. No comércio e serviços, até 09 pessoas ocupadas;

- **Pequena empresa:**

1. Na indústria e construção: de 20 a 99 pessoas ocupadas;
2. No comércio e serviços, de 10 a 49 pessoas ocupadas.

Para que estas empresas possam utilizar a presente proposta e atender os cinco resultados esperados para GRE do nível G é necessário que estas utilizem um bom processo para a implantação das técnicas. Para isso, as próximas seções (4.2 a 4.6) descrevem inicialmente como atender aos resultados esperados (GRE1 a GRE5) e posteriormente na Seção 4.7 propõe-se um processo de implantação para as novas técnicas. Desta forma, as MPEs podem fazer uso destas propostas considerando a necessidade de atingir o nível G do MPS.BR. Finalmente na Seção 4,8 são apresentadas as considerações finais do Capítulo.

## **4.2 GRE1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos**

Como dito anteriormente (Seção 3.2) o objetivo deste resultado é estabelecer o uso de mecanismos para obter, com o fornecedor, o conhecimento dos requisitos, avaliá-los e validá-los. Para isso a técnica escolhida pela organização deve permitir:

- Entender o domínio do cliente;
- Registrar os requisitos elicitados e seus fornecedores;
- Avaliar os requisitos registrados utilizando critérios objetivos;
- Fornecer informações suficientes para a organização aceitar ou não a produção dos projetos.

#### **4.2.1 Entender o domínio do cliente**

Uma boa comunicação com os fornecedores de requisitos é fundamental para assegurar um bom entendimento das necessidades do cliente e dos requisitos do projeto e, conseqüentemente, aumentar as chances de sucesso do projeto. Uma técnica que não permite o entendimento completo do domínio que será trabalhado pode arruinar o projeto.

Uma boa extração de requisitos ajuda na compreensão das decisões tomadas durante o desenvolvimento evitando surpresas quando o sistema for construído e entregue. Por isso, a técnica escolhida pela organização deve permitir um contato com o cliente que possibilite entender realmente a sua necessidade.

Técnicas que não permitem um contato direto com o cliente ou que geram requisitos a partir de outras fontes, como pesquisas, dificultam ou evitam a comunicação com os fornecedores e pode prejudicar seriamente a qualidade dos requisitos elicitados. Também é importante salientar que neste contexto a utilização de uma técnica de modelagem organizacional como i\* (YU, 1993) (SANTANDER & CASTRO, 2002) ou BPMI (2010) pode melhorar o entendimento do domínio do cliente através da representação de informações estratégicas da organização na qual o sistema computacional fará parte.

#### **4.2.2 Registrar os requisitos elicitados e seus fornecedores**

Após entender o domínio do cliente, a técnica escolhida deve permitir registrar os requisitos elicitados e seus fornecedores. Como comprovação do entendimento dos requisitos,

deve-se ter um Documento de Requisitos, que pode ter diferentes formas de acordo com as necessidades da organização.

O Documento de Requisitos é a base para o desenvolvimento do software; os requisitos aqui registrados delimitam a abrangência do software, estabelecem funcionalidades requisitadas pelo conjunto de clientes e usuários, impõem restrições de qualidade, fornecem subsídios para o processo de verificação e validação do software construído. Nas metodologias clássicas, o processo de requisitos é fundamental para o sucesso de um projeto de software; um bom documento de requisitos possibilita estimativas de custos razoavelmente precisas, o cronograma de execução não deverá sofrer variações significativas, usuários podem participar ativamente do processo de validação do software. E, principalmente, pode-se colocar o software em operação com a convicção que os principais requisitos foram atendidos.

Uma boa prática é registrar os requisitos de uma forma que facilite sua avaliação pelos critérios definidos pela empresa, ou seja, cria-lo tentando atender previamente os critérios que serão avaliados. No nosso entendimento os requisitos devem ser registrados textualmente contendo como mínimo os seguintes itens na sua descrição: **ID. Única**, para referenciar os requisitos; **Descrição Textual**, para explicar e contextualizar os requisitos; **Nome do Fornecedor**, para poder rastrear de onde surgiram as informações e saber exatamente com quem falar caso necessite de mais informações; e **Nível de Importância** para dar prioridade aos requisitos mais importante, podendo assim, liberar versões executáveis do sistema com antecedência.

### 4.2.3 Validar os requisitos registrados utilizando critérios objetivos

Todos os requisitos que foram definidos no DRE (Documento de Requisitos) devem ser validados para que estes estejam compreensíveis para todos. Assim, usando as características citadas pela IEEE std 830 (IEEE, 1998) é proposto um *checklist* onde se verificam as seguintes características para todos os requisitos:

- **ID. Única:** Apresentar uma forma única de referência. Por exemplo: RF001, RNF003.
- **Claro:** Um requisito é claro se, e somente se, sua descrição não deixar dúvidas, mesmo para aqueles que o lerem pela primeira vez.

- **Correto:** Um requisito é correto se, e somente se, descreve uma ação que o software deve cumprir.
- **Não Ambíguo:** Um requisito é não ambíguo se, e somente se, a sua descrição não apresentar mais de uma interpretação.
- **Relevante:** Um requisito é relevante se, e somente se, ele está relacionado a uma funcionalidade, desempenho, restrições de design, atributos ou interfaces externas.
- **Coerente:** Coerente refere-se à coerência interna. Se um requisito não concorda com algum documento ou outros requisitos, então não é correto.
- **Nível de Importância:** Normalmente, todos os requisitos que dizem respeito a um produto de software não são igualmente importantes. Alguns requisitos podem ser essenciais, especialmente para aplicativos críticos no negócio, enquanto outros podem ser desejáveis.
- **Completo:** Um requisito é completo se apresenta todas as informações necessárias em sua descrição e cumpre todos os itens citados acima. Se possível, é desejável que este especifique as respostas válidas e inválidas para valores de entrada e saída.
- **Implementável:** O requisito deve apresentar características que permitam sua implementação.
- **Testável:** O requisito deve apresentar características que permitam seus testes.
- **Rastreável:** O requisito deve apresentar características que permitam sua rastreabilidade horizontal e vertical.

Se algum requisito falhar em algum item listado este deve ser revisto, talvez uma melhor descrição ou divisão destes em mais requisitos possa facilitar o entendimento do escopo e a importância deste requisito. Novos critérios podem ser agregados aos propostos de acordo com as características e necessidades da organização.

#### 4.2.4 Fornecer informações suficientes para a organização aceitar ou não a produção dos projetos

A validação dos requisitos após avaliação indica que estes estão compreensíveis para todos, assim, conclui-se que o domínio do cliente foi entendido e os requisitos estão registrados da forma correta. Tais informações servem para assegurar o desenvolvimento do projeto evitando surpresas indesejáveis. Portanto, as técnicas devem fornecer no mínimo os seguintes itens para possibilitar o entendimento: Em qual contexto o projeto se insere; qual o escopo do projeto; permitir calcular os recursos necessários para o projeto; as prioridades do projeto; e o prazo para a entrega.

Na Seção seguinte é feito uma avaliação das técnicas já descritas na Seção 3.2.

#### 4.2.5 Avaliação das técnicas

Estando claros os objetivos que a técnica a ser utilizada deve satisfazer visando alcançar a GRE1, a tabela 4 mostra um resumo da avaliação das técnicas já descritas na Seção 3.2 e que podem ser usadas neste contexto. Observando a tabela 4 podemos visualizar que para cada técnica e objetivos analisados o grau de satisfação foi definido em, NA (Não atendido), PA (Parcialmente atendido) ou TA (Totalmente atendido) de acordo com sua competência para cada item avaliado.

Tabela 4. Avaliação das técnicas para a satisfação da GRE1

Requisitos	Entrevista	Questionários	JAD	<i>Brainstorming</i>	Etnografia	Prototipagem	SWOT
Entendidos	TA	PA	TA	PA	TA	TA	PA
Registrados	TA	TA	TA	TA	TA	PA	TA
Avaliados	TA	TA	TA	TA	TA	TA	TA
Aceitos	TA	TA	TA	TA	TA	TA	TA

Para o entendimento do domínio do cliente apenas três técnicas foram classificadas como PA (Parcialmente atendido), o Questionário, por dificultar a comunicação do cliente e limitar as respostas do cliente; o *Brainstorming* e o SWOT por serem técnicas mais voltadas para produtos novos e não exigirem a presença ou a consulta do cliente para o entendimento dos requisitos.

A maioria das técnicas estudadas permitem o registro dos requisitos de uma forma correta, para que este possa ser avaliado após concluído. Apenas a prototipagem peca nesse registro por utilizar geralmente telas ou subprodutos como forma de registro o que pode dificultar na

avaliação de alguns critérios. Resumidamente, a prototipagem tem como objetivo gerar telas ou subprogramas para um melhor entendimento do domínio e não para documentação, onde muitas vezes os protótipos gerados são descartados.

Embora a prototipagem apresente dificuldades em relação à documentação, a avaliação dos requisitos não fica comprometida, porque permite ao cliente visualizar os requisitos e opinar se concorda ou não com eles. Assim, todas as técnicas comportam uma avaliação por critérios objetivos.

Após a validação dessa avaliação a empresa possui informações suficientes sobre a complexidade do projeto com todas as técnicas, possibilitando assim, aceitar ou recusar a implementação do projeto.

Considerando os fatores anteriormente citados esta proposta declara como aptas para certificação de nível G do MPS.BR as técnicas que apresentam para todos os itens avaliados TA (Totalmente atendido). Assim, são estas: Entrevista, JAD e Etnografia. Porém, as demais técnicas citadas podem ajudar a organização e serem complementares.

### **4.3 GRE2 - Obtenção do comprometimento da equipe técnica com os requisitos aprovados**

Após a aprovação dos requisitos, um comprometimento formal da equipe técnica com os requisitos aprovados deve ser obtido e registrado. Por isso, o gerente do projeto deve apresentar para a equipe quais foram os requisitos definidos e deixar claro o que deverá ser produzido.

Levando em consideração o número de funcionários de uma MPE, esta proposta sugere que a apresentação do projeto seja feita em uma reunião, por ser um método adequado como mostra o trabalho de SERRANO et al. (2008). Assim, após a exposição das informações, os participantes podem discuti-las e esclarecerem possíveis dúvidas. Por fim, coletada a assinatura dos presentes em ata de reunião, esta será usada como evidência do compromisso dos membros participantes. A vantagem da reunião é o fato de todos estarem juntos e permitir um melhor esclarecimento e um registro de compromisso de uma forma mais rápida.

Caso algum membro do projeto não possa estar presente em reunião o e-mail deverá ser usado como forma de comunicado e sua resposta como forma de registro. Ele também pode ser utilizado para projetos menos complexos, porém, neste caso a empresa deve deixar explícito qual a complexidade mínima em critérios objetivos.

Se a empresa necessita de uma formalidade maior um contrato entre as partes pode garantir uma maior segurança para o andamento do projeto. Lembrando que mudanças nos requisitos aprovados podem afetar nos compromissos já estabelecidos pela equipe. Nesse caso, um novo comprometimento deve ser obtido em relação aos requisitos modificados.

#### **4.4 GRE3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida**

Como dito anteriormente (Seção 3.4) o objetivo deste resultado é a definição de um método que descreva como a organização fará a rastreabilidade bidirecional entre os requisitos e os produtos de trabalho, visando manter o controle dos requisitos e calcular o impacto deles durante o projeto.

O rastreamento pode ser feito de várias formas, geralmente, devido ao grande número de informações e o tamanho das matrizes de rastreabilidade, é usado uma ferramenta computacional para o auxílio desse controle. Essa ferramenta pode ser proprietária, livre ou criada pela empresa.

Uma ferramenta proprietária tem um custo, porém traz consigo uma maior robustez e suporte quando necessário; ferramentas livres geralmente não possuem custo de adesão ou manutenção, mas podem ser limitadas em relação a funcionalidades e correções de erros; ou então a empresa pode desenvolver sua própria ferramenta, adicionando a ela todas as funcionalidades e características necessárias. Essa proposta sugere que a organização adote uma ferramenta livre ou opte por criar sua própria ferramenta, por evitar gastos com softwares proprietários e principalmente, por poder gerar uma ferramenta exclusiva para as características e necessidades da organização. O uso de ferramentas livres, também pode evitar o custo com softwares, e caso *Open Source*, permitir os benefícios da customização.

Embora uma ferramenta seja extremamente importante para auxiliar a rastreabilidade, esta deve apenas servir como auxílio para uma estratégia predefinida e nunca ao contrário. A empresa deve ter clara qual a estratégia que será usada para a rastreabilidade bidirecional e quais artefatos serão rastreados.

Assim, essa proposta mostra a seguir as funcionalidades básicas mínimas necessárias para uma ferramenta deste tipo, uma estratégia de rastreabilidade bidirecional, bem como quais os potenciais artefatos que deverão ser rastreados.

#### 4.4.1 Ferramentas de Rastreabilidade

Uma ferramenta automatizada para rastreamento de requisitos deve armazenar e manter os requisitos acordados, durante todo o ciclo de vida do software, gerenciando as mudanças ocorridas nos requisitos, permitindo rastrear os relacionamentos entre requisitos e entre requisitos e documentos produzidos no processo de desenvolvimento do sistema.

Uma opção para armazenar as informações necessárias para a rastreabilidade bidirecional vertical e horizontal é a criação de um “Campo de Rastreabilidade” (Figura 3). O “Campo de Rastreabilidade” é mais comum em organizações que possuem um sistema que controla o processo de desenvolvimento. Assim, quando os serviços são passados para a equipe desenvolvedora o “Campo de Rastreabilidade” está junto ao serviço que deve ser executado. Além de manter a rastreabilidade, esse campo ajuda o desenvolvedor a localizar melhor onde devem ser feitas alterações e mostra quais partes do sistema podem sofrer influência com as alterações que ele irá realizar.

Este campo contém as informações necessárias para a rastreabilidade bidirecional vertical e horizontal. Onde no subcampo maior superior é registrado os níveis de implementação que vão da aplicação até o código e nos três subcampos menores inferiores estão os requisitos que são implementados pelo serviço e quais se relacionam a este.

Além desses dados armazenados é necessário adotar uma ferramenta de busca responsável por localizar as informações de cada campo e a matriz de rastreabilidade para mapear as relações “*Requisito X Requisito*” e “*Requisito X Produto de Trabalho*”.

<b>Aplicação / Package / Units, Formulários, Classes:</b> <nome da aplicação que sofreu alterações> / <nome dos pacotes onde houve alterações> / <nome das unidades, formulários ou classes onde houve alterações>		
<b>Requisito do Projeto:</b> <ID do requisito implementado>	<b>Impactado por:</b> <ID dos requisitos que impactam este>	<b>Impacta os:</b> <ID dos requisitos que são impactados por este>

Figura 3: Campo de Rastreabilidade

#### 4.4.2 Rastreabilidade Vertical

A estratégia para a rastreabilidade bidirecional vertical é como manter o controle dos requisitos, desde a fase de obtenção até a sua liberação, e ver como estes se relacionam com os artefatos que são gerados no decorrer do projeto de forma bidirecional.

Para isso a empresa deve definir quais artefatos rastrear e de qual forma eles se relacionam com os requisitos e entre si. Por exemplo, para uma empresa que utilize o campo de rastreabilidade citado anteriormente, teríamos:

- 1) Dividir o **escopo** de cada projeto em **requisitos**, essa divisão já é realizada no “Documento de Requisitos”.
- 2) Logo, esses **requisitos** são passados para o sistema em forma de “**serviços**”, ou seja, cada requisito do projeto se torna *n* “**serviços**” no sistema. Para o controle, é necessário que cada “**serviço**” aberto no sistema tenha explicito no seu campo de rastreabilidade qual **requisitos** ele representa.
- 3) Assim, cada “**serviço**” será passado para seu desenvolvedor para ser implementado no **sistema**. De forma mais macro com a finalidade de facilitar a localização da implementação, no campo de rastreabilidade de cada requisito é necessário listar quais foram os pacotes que sofreram alterações.
- 4) Seguindo a organização do sistema os *packages* (pacotes) contêm *units*, formulários, ou classes, dependendo da linguagem e do estilo de programação utilizada.
- 5) Por fim, cada **requisito** está implementado em **Units, classes ou Formulários** do sistema. Para maior precisão as *Units* e formulários devem ter seu código todo **comentado**. Estes comentários descrevem o objetivo de cada função e permite o acesso exato a determinado grupo de linhas de código.

Resumindo, para esse caso, a rastreabilidade dentro do código se dá para:

- Escopo → Requisitos → Serviços → Pacotes → *Units*, Classes ou formulários → código fonte.

Para rastrear os produtos de trabalho a empresa deve definir quais os produtos que devem ser rastreados, dentre eles estão Casos de Teste, Casos de Usos, entre outros. Para indicar este relacionamento existente entre os requisitos e os produtos de trabalho pode ser utilizada uma matriz de rastreabilidade (Figura 4).

Matriz de Rastreabilidade Requisito x Casos de Teste										
	CT 01	CT 02	CT 03	CT 04	CT 05	CT 06	CT 07	CT 08	CT 09	CT 10
RF001	X	X	X							
RF002				X						
RF003					X			X	X	
RF004						X		X		
RF005							X	X		X

Figura 4. Matriz de Rastreabilidade “Requisito X Casos de Teste”

#### 4.4.3 Rastreabilidade Horizontal

A estratégia para a rastreabilidade bidirecional horizontal é como manter o controle dos requisitos, desde a fase de obtenção até a sua liberação, e ver como estes se relacionam com os demais requisitos do projeto de forma bidirecional.

Para isso a empresa deve definir qual a forma de conhecer o relacionamento entre os requisitos. Para indicar este relacionamento existente entre os requisitos do software pode ser utilizada uma matriz de rastreabilidade (Figura 5).

Matriz de rastreabilidade Requisitos x Requisitos										
	RF 01	RF 02	RF 03	RF 04	RF 05	RF 06	RF 07	RF 08	RF 09	RF 10
RF 01			X	X	X	X				
RF 02			X	X	X	X	X			
RF 03				X	X	X	X			
RF 04					X	X				
RF 05						X				
RF 06										
RF 07										
RF 08		X								
RF 09										
RF 10										

Figura 5. Matriz de Rastreabilidade “Requisito X Requisito”

Todos os requisitos são listados na primeira coluna e na primeira coluna e a intersecção linha/coluna representa a existência ou não de relacionamento entre eles. Diversos autores comentam a importância e a necessidade desse tipo de relacionamento para o processo de

desenvolvimento de software. Essa matriz permite a previsão do impacto de uma mudança ou inserção e exclusão de um requisito no sistema. TORANZO (2002) enfatiza a dificuldade de se obter e manter esse tipo de matriz e, além disso, propõe que a indicação da dependência entre os requisitos determine, além da existência ou não de relação entre eles, uma forma de registrar, subjetivamente, se essa relação é forte ou fraca.

Depois de listados todos os requisitos do projeto, o responsável deve descrever as relações entre os requisitos, essa relação pode ser expressa numa matriz de rastreabilidade. Esta relação permite analisar o impacto que ocorrerá nos demais requisitos, tanto para aqueles que o requisito faz referência, quanto para os referenciados como mostra a Figura 6.

Caso os requisitos sejam passados para o sistema em forma de “serviços”, é importante continuar com essas informações para que a rastreabilidade não se perca com a evolução dos projetos. O campo de rastreabilidade permite que as informações de relação entre os requisitos continuem durante o desenvolvimento, para isso, é obrigatório colocar a ID do requisito que aquele “serviço” atende, e a lista de todos os requisitos que influenciam e são influenciados.



Figura 6. Rastreabilidade Horizontal

## **4.5 GRE4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos**

Como dito anteriormente (Seção 3.5) o objetivo deste resultado é a definição de um método que descreva como a organização fará revisões em planos e produtos de trabalho do projeto, visando identificar e corrigir inconsistências em relação aos requisitos.

Esse mecanismo deve controlar a evolução dos requisitos, seja por constatação de novas necessidades, seja por constatação de deficiências nos requisitos registrados até o momento. Tais revisões são uma das praticas responsáveis por encontrar necessidades de mudanças

Com o intuito de revisar planos e produtos durante o projeto foi criado um fluxograma que apresenta os passos necessários para a administração das mudanças (Figura 7). Tal processo é baseado na proposta de HAZAN & LEITE (2003) e algumas práticas da Revisão Formal, apresentada na Seção 3.5.1.2.

Para que o processo proposto possa ser usado visando à certificação algumas informações adicionais devem ser definidas, como forma de comunicação e papéis responsáveis, sempre levando em consideração as características e as necessidades da organização.

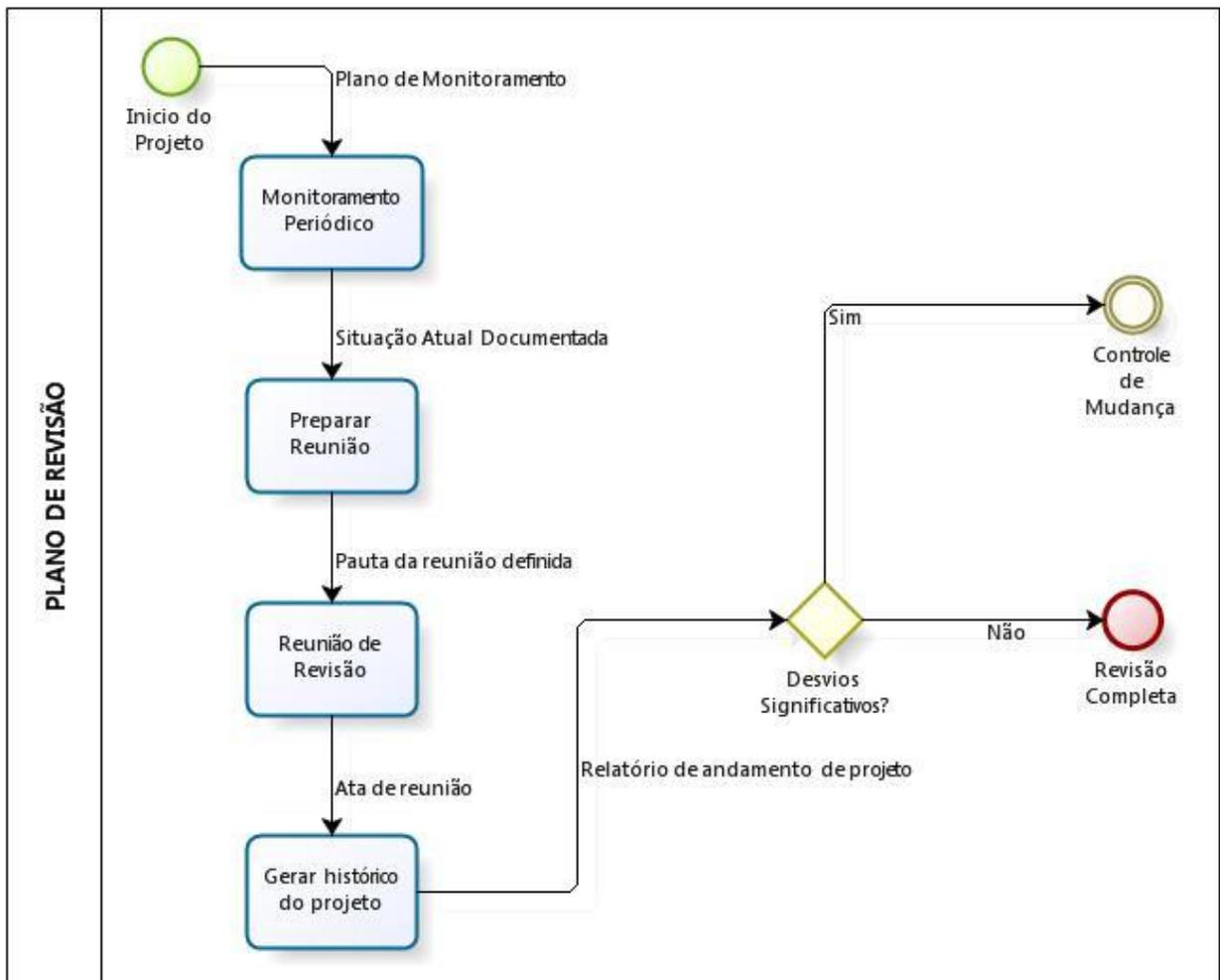


Figura 7. Plano de Revisão

**Monitoramento Periódico:** Essa atividade deve ser repetida periodicamente até o encerramento do projeto, nela o responsável irá monitorar os requisitos a fim de identificar desvio, alteração, adição ou exclusão. Caso identifique algo, ele os registrará no histórico do projeto, o qual será a pauta das reuniões de revisão.

**Preparar Reunião:** De acordo com as informações registradas no histórico do projeto, o responsável irá definir a pauta mínima para a reunião de revisão e envia-la para os participantes. Além de definir a pauta da reunião de revisão este responsável, se possível, deve levar a reunião opções de ações corretivas para os desvios identificados, para decidir sob eles com o resto da equipe.

**Reunião de Revisão:** A finalidade desta reunião é identificar se desvios ocorreram, porque ocorreram, como corrigir o desvio e como evitar que ele volte a acontecer. Além de focar em questões que necessitam de uma resposta e nas mudanças que podem ser feitas nos requisitos.

Caso se perceba alguma irregularidade significativa para a organização deve-se criar uma “Solicitação de Mudança” para a correção.

**Gerar histórico do projeto:** Após a realização da reunião de revisão, o responsável deve criar um relatório do andamento do projeto, além de ser um registro histórico, este relatório serve para ter noção do grau de conformidade com o documento de requisitos, o plano de projeto e com o andamento do mesmo. Esse relatório deve estar sempre acessível.

## **4.6 GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto**

Como dito anteriormente (Seção 3.6) o objetivo deste resultado é estabelecer o uso de mecanismos para gerenciar as mudanças nos requisitos ao longo do projeto, já que, mudanças nos requisitos são inevitáveis durante o processo de desenvolvimento.

Esse mecanismo deve auxiliar a empresa em como criar uma solicitação de mudança e nas decisões a serem tomadas. Estas decisões devem ser tomadas baseadas no impacto que causarão, por exemplo, em outros requisitos, cronograma ou custos. Todas as necessidades de mudança solicitadas devem ser registradas e um histórico com essas solicitações e suas decisões tomadas deve ficar disponível.

É importante ressaltar que o monitoramento e controle dos requisitos (GRE4) é uma das práticas responsáveis por encontrar necessidades de mudanças e que o controle de rastreabilidade (GRE3) é um importante mecanismo para facilitar a análise de impacto.

Com o intuito de controlar as mudanças nos requisitos durante o projeto foi criado um fluxograma que apresenta os passos necessários para a administração das mudanças (Figura 8). Tal processo é baseado na proposta de SOMMERVILLE (2003) e nas características das MPEs.

Para que o processo proposto possa ser usado visando à certificação algumas informações adicionais devem ser definidas, como forma de comunicação e papéis responsáveis, sempre levando em consideração as características e as necessidades da organização.

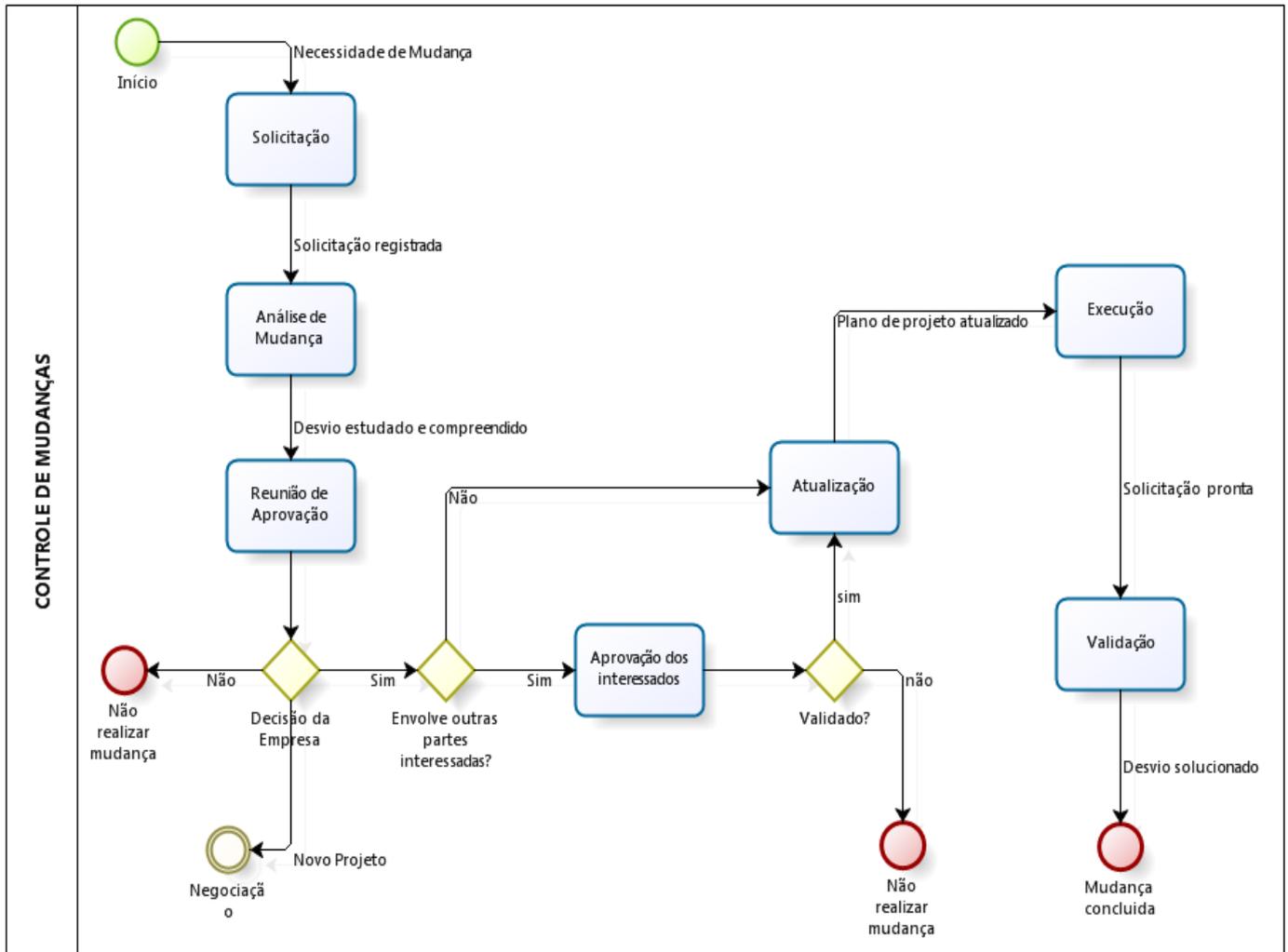


Figura 8. Controle de Mudanças

**Solicitação:** Qualquer mudança que modifique algum requisito ou que possa alterar o custo, prazo, tamanho e/ou esforço do projeto é considerado um desvio e este deve ser tratado até sua conclusão. Na identificação de um desvio uma “Solicitação de Mudança” deve ser realizada. Cada solicitação precisa ser registrada no histórico da empresa, meio por onde esta será comunicada aos interessados.

**Análise de Mudança:** Após o conhecimento do desvio um estudo para conhecer o impacto e a viabilidade deverá ser realizado. Este estudo não deve levar em conta apenas os requisitos afetados diretamente, e sim, todos os afetados direta ou indiretamente, para isso o responsável deve verificar o impacto para identificar todos os requisitos afetados pela mudança. Assim, na “Análise de Mudança”, o responsável verifica as consequências de modificar o projeto/requisito ou não, criando um “Estudo de Viabilidade” que leva em consideração

custos, esforços, prazos, impacto e se é válido estrategicamente para a empresa, com o objetivo de saber se a mudança é viável ou não.

**Reunião de Aprovação:** É importante, sempre que possível, que a empresa tenha sua opinião diante da solicitação de mudança antes do cliente, isso pode dar uma vantagem na hora da negociação. Para isso, uma reunião que tenha como pauta a análise de mudança deve ser feita pelos responsáveis de dentro da empresa, cujo, devem decidir se a mudança será aceita ou não, ou ainda, se esta será tratada como um novo projeto. Após a decisão a resposta final é registrada em ata e as partes interessadas, como cliente, fornecedor de requisitos entre outros, serão comunicadas quando necessário.

**Aprovação dos interessados:** Para que a mudança possa ter início, além da validação da empresa, é necessário existir a validação das outras partes interessadas no projeto. As mudanças relevantes são expostas para as partes envolvidas e estas, se concordarem, confirmam junto à empresa o compromisso com o requisito/projeto. Este compromisso pode ser um e-mail, ata de reunião ou algum outro mecanismo que permita o registro de acordo com a formalidade exigida pela organização. Caso não haja acordo a mudança não deve ser realizada.

**Atualização:** A aprovação da “Mudança Solicitada” precisa ser passada para a equipe desenvolvedora, as alterações de escopo, custo, prazo e esforço, causadas pelo desvio deverão ser documentadas, ou seja, cronograma, plano do projeto, documento de requisitos, matriz de rastreabilidade e todos os ativos de processo impactados devem ser ajustados para continuarem atualizados.

**Execução:** Inicia-se a fase de programação e testes de acordo com a metodologia de desenvolvimento da empresa.

**Validação:** A validação encerra o processo de “Solicitação de Mudanças”. Nela o desvio solicitado deve estar totalmente implementado e testado. O responsável deve avaliar a versão final apresentada e verificar se esta satisfaz totalmente os requisitos solicitados pelo desvio.

## **4.7 Processo para Implantação da Engenharia de Requisitos visando a melhoria da Qualidade de Software**

Para que as empresas possam utilizar a presente proposta com o intuito de uma certificação MPS.BR, além de adotar técnicas iguais ou semelhantes às apresentadas no trabalho, um processo de implantação correto deve ser utilizado.

Existem vários modelos de implantação, como o AINSI de BRIAND et al. (1999) que além de ser um bom modelo para a implantação auxilia na melhoria contínua do processo de produção de Software, porém, sua abordagem mais complexa, tempo dedicado a projetos piloto e a necessidade de mais pessoas para papéis definidos, são fatores que fogem da realidade das MPEs.

Um processo de implantação para técnicas de Engenharia de Requisitos mais adaptado à realidade das empresas estudadas é o modelo proposto pela SERPRO (2010). Este modelo apresenta passos explícitos e lógicos, a princípio sugere uma conscientização e preparação da empresa para o início dos trabalhos, e a forma de definir as melhorias dos processos atuais da empresa minimiza o impacto cultural da implantação, além de exigir um grupo pequeno de pessoas para cada tarefa, fato a qual se adapta bem ao número de pessoas que trabalham nas MPEs. Tais fatores fazem esse processo possuir uma boa aplicabilidade, tendo-se em consideração o tipo de empresa que este trabalho foca.

Existem vários outros modelos mais genéricos, se, porventura a organização possuir processos já definidos e os projetos necessitarem adaptar os processos existentes, esse fato deverá ser declarado durante o planejamento do projeto. Essas adaptações podem incluir alteração em processos, atividades, ferramentas, técnicas, procedimentos, padrões, medidas, dentre outras.

Em seu processo, a SERPRO (2010) indica nove fases (Figura 9) que devem ser seguidas até que se consiga o resultado esperado. Ao longo de toda a implantação os processos devem ser avaliados e continuamente ajustados de acordo com o *feedback* obtido.

Convém ressaltar que não existe um processo ideal que possa ser utilizado em qualquer organização. Será sempre necessário realizar adaptações, em função das características do sistema, dos usuários, da equipe, da organização, do tipo de sistema e da tecnologia a ser utilizada, dentre outros fatores.

A seguir, são descritas as nove fases indicadas pela SERPRO (2010) para implantação dos processos:

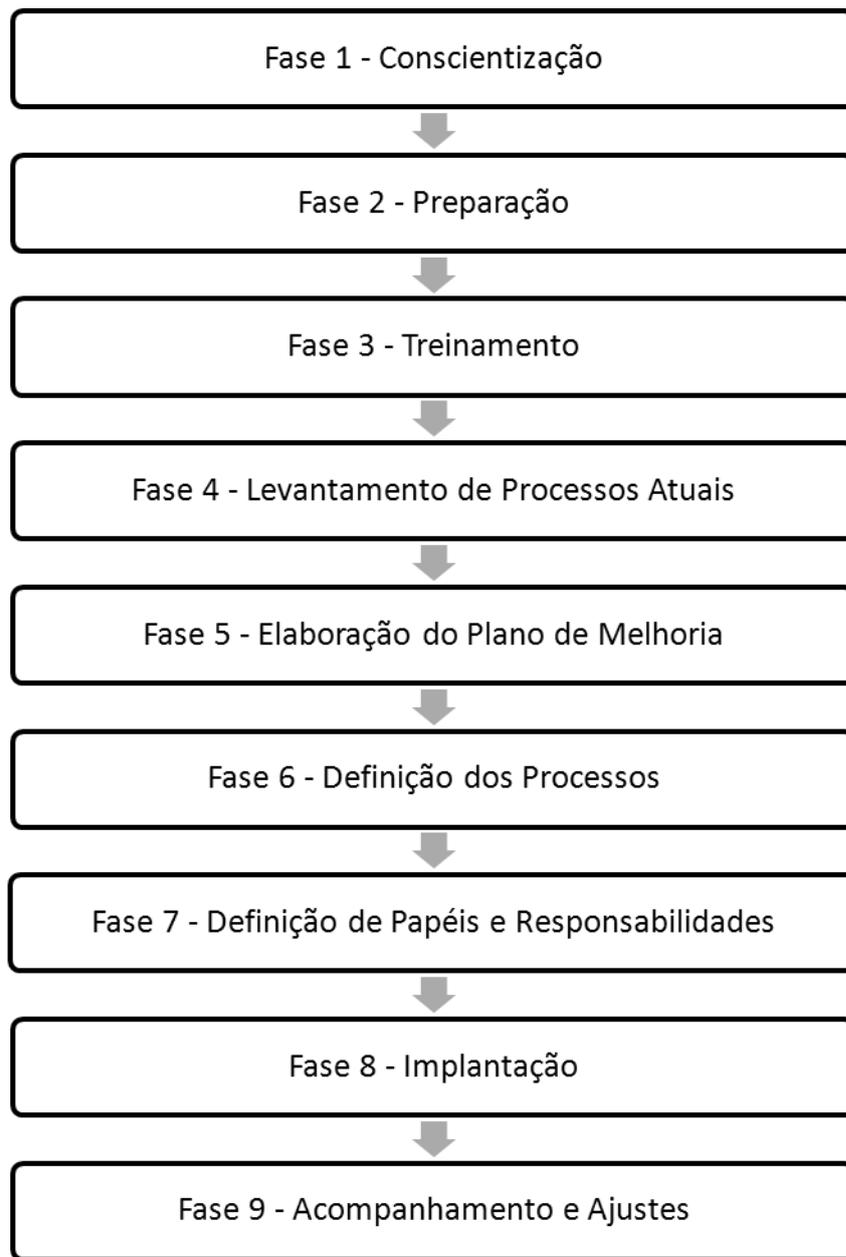


Figura 9. Fases para a Implantação (SERPRO, 2010)

• **Fase 1 - Conscientização:** nesta fase, deve-se identificar a situação atual, estabelecendo metas, e tomar ações para reduzir a distância entre o status atual e o almejado. O estado atual pode ser obtido por meio de observações e de questionários e entrevistas aos desenvolvedores em relação ao conhecimento e utilização dos Processos de Engenharia de Requisitos. Uma vez obtidas as informações, estas devem ser consolidadas e apresentadas aos gerentes e desenvolvedores, de forma a conscientizar e mobilizar os gerentes e desenvolvedores para a promoção das práticas e mudanças requeridas pelo programa. É fundamental o apoio gerencial para as ações e iniciativas de aprimoramento de processos de Engenharia de

Requisitos. Manter esse comprometimento e repassá-lo para os analistas é parte fundamental do processo de melhoria.

- **Fase 2 - Preparação:** durante a preparação devem ser escolhidos os grupos de qualidade. O grupo de qualidade é definido para planejar e acompanhar todas as tarefas do projeto de melhoria da capacidade, visando garantir que o processo de implantação do modelo como um todo flua de maneira adequada. Dependendo do tamanho da organização podem ser definidos subgrupos. Estes grupos recebem os primeiros treinamentos, de acordo com a necessidade e disponibilidade. É importante que nesta fase esteja circulando na empresa informações a respeito do trabalho sendo desenvolvido, o que pode ser feito por meio de jornais internos, intranet, e-mail ou circulação de revistas e livros que tratem sobre o tema.

- **Fase 3 - Treinamento:** O grupo de qualidade também deve definir um Plano de Treinamento, priorizando a participação das pessoas que compõe os grupos de qualidade. O treinamento deve preferencialmente ser preparado e aplicado por integrantes do grupo, pois deve ser dada uma atenção especial para a adaptação dos termos do modelo de melhoria para a cultura da empresa. Todos os termos devem ser definidos e explicados fazendo, quando possível, um mapeamento para os termos equivalentes utilizados na empresa. Durante o treinamento, é importante que um componente do grupo de qualidade anote as sugestões e considerações que porventura apareçam, pois podem ser de grande utilidade tanto para os próximos treinamentos como para utilização ao longo do processo de melhoria.

- **Fase 4 - Levantamento de Processos atuais:** O conhecimento de como está o seu processo de desenvolvimento atual é importante para a melhoria. É preciso que o grupo de qualidade tome conhecimento de todos os processos de Engenharia de Requisitos existentes na empresa e de novas técnicas que podem trazer melhorias ou preencher lacunas existentes. Este conhecimento deverá ser provido por reuniões, entrevistas e coletas de documentos com a alta gerência. Deve-se avaliar as políticas, padrões e processos atuais de Engenharia de Requisitos existentes na organização, identificando os pontos fortes e fracos.

- **Fase 5 - Elaboração do Plano de Melhoria:** o Plano de Melhoria do Processo de Engenharia de Requisitos é a base para o futuro esforço de melhoria e deve estar de acordo com os recursos da empresa. O grupo de qualidade deve preparar o Plano de Melhoria contendo os processos e documentos existentes de modo que fiquem claros os pontos em que há necessidade de melhoria e o impacto benéfico que pode causar ao processo. Pode recomendar ações para solucionar as carências identificadas e até estimar o tempo e recursos necessários para cada atividade de melhoria.

• **Fase 6 - Definição dos Processos:** a definição dos Processos a serem implantados deve ser elaborada tendo como base com as boas práticas existentes na empresa, o Plano de Melhoria definido, o tipo, as restrições e a cultura da organização. São importantes pesquisas em bibliografia existente, instituições de pesquisa e, se possível, a utilização de benchmarking.

• **Fase 7 - Definição de Papéis e Responsabilidades:** nesta fase devem ser especificados os papéis e as responsabilidades das pessoas que executarão as atividades de desenvolvimento e gerenciamento de requisitos. Deve-se definir um mapeamento para as funções existentes na Empresa.

• **Fase 8 - Implantação:** Para iniciar a implantação é importante que o grupo de qualidade defina um Plano para Projeto-piloto, onde seja especificado o projeto escolhido, as pessoas envolvidas, os treinamentos necessários, além os prazos esperados. A implantação do projeto-piloto deve ser acompanhada pelo grupo de qualidade e as dificuldades encontradas devem ser documentadas para posterior análise e avaliação dos Processos definidos.

• **Fase 9 – Acompanhamento e Ajustes:** esta fase deve acompanhar a aplicação das práticas de Gerência de Requisitos implantadas, avaliando-as e promovendo os refinamentos necessários. Nesta fase, deverão ser executadas ações para atender os resultados esperados do modelo de maturidade utilizado.

## 4.8 Considerações Finais

Este capítulo apresentou uma proposta de como atender os resultados esperados da GRE no nível G do MPS.BR e uma estratégia para a implantação de uma gerência de requisitos, ambas baseadas em três pilares básicos: qualidade, requisitos e processos. Para que a qualidade seja alcançada é primordial que os requisitos tenham sido bem definidos e controlados e, para isto, devem haver processos estabelecidos e implantados.

Por isso, o gerenciamento de requisitos é reconhecido como um importante pré-requisito para desenvolver software de alta qualidade além de permitir à organização obter a habilidade de descrever e seguir a vida de um requisito, em ambas as direções. Por essas análises, o responsável pela gerência do projeto é capaz de negociar com o cliente alterações nos planos do projeto para atender às solicitações de mudanças de requisitos e, ao mesmo tempo, minimizar os riscos do projeto, como por exemplo, desvios de cronograma e de custos.

No Capítulo 5 será apresentada a empresa escolhida para aplicar a proposta gerada. Descreve-se também o projeto no qual a proposta foi aplicada, e os resultados desses estudos de caso.

# Capítulo 5

## Estudo de Caso

Neste capítulo apresenta-se a empresa escolhida (Seção 5.1) e o projeto (Seção 5.2) analisado para o estudo de caso. As seções 5.3 a 5.7 descrevem quais foram as atividades da empresa em relação aos resultados esperados abordados nesse trabalho. Na Seção 5.8 descreve-se o processo de implantação das mudanças na empresa e, por fim, nas considerações finais (Seção 5.9) analisam-se os resultados alcançados até o momento e as dificuldades encontradas no projeto.

### 5.1 A empresa

A empresa escolhida para o estudo de caso foi a TECINCO, localizada na cidade de Cascavel, PR, é uma empresa de pequeno porte e foi escolhida como representante das MPE do Oeste do Paraná que buscam a certificação MPS.BR. Cabe ressaltar que a categorização deste tipo de empresa foi apresentada no capítulo 4.

A TECINCO, Tecnologia em Informática Corporativa, foi criada em 1994. Seu objetivo principal era o desenvolvimento de software e assessoria voltados à área de veículos e transportes. A intimidade com o negócio de veículos e transportes transformou a TECINCO, em especialista na gestão de concessionárias e agenciadoras de cargas. Hoje, acompanhando a tendência do mercado, ampliou sua área de atuação, desenvolvendo softwares voltados para a área comercial, industrial e serviços. Procura sempre maximizar o investimento dos clientes, visando oferecer ferramentas cada vez mais modernas e operacionais TECINCO (2010).

A TECINCO iniciou um processo de certificação MPS.BR Nível G em Dezembro de 2009, desde de então, vem adaptando seu processo de desenvolvimento para a conquista do certificado. A empresa espera que com o novo processo atingido pelo modelo de qualidade,

fatores como satisfação do cliente; custo médio do projeto; prazos; faturamento; produtividade e qualidade possam ser melhorados.

## **5.2 O projeto**

É importante salientar que o presente estudo de caso foi realizado como parte do trabalho de estágio do autor desta monografia. Desta forma, também houve participação do mesmo na definição e execução do projeto. Assim, pôde observar que algumas das sugestões indicadas pela proposta apresentada neste trabalho foram aceitas e outras não, o que indica que a visão da empresa difere da acadêmica por dar prioridades a critérios diferentes.

O projeto utilizado no estudo de caso aborda a necessidade de uma empresa, cliente da TECINCO, que precisava de um novo módulo no sistema controlador de frota. Era necessário desenvolver um novo módulo que permitisse ao sistema limitar o crédito dos clientes e na cobrança nos embarques feitos.

Tendo em vista que a empresa busca a certificação MPS.BR Nível G, as seções seguintes descrevem quais foram as atividades da empresa em relação aos resultados esperados abordados nesse trabalho e compara essas atividades com a proposta apresentada.

## **5.3 GRE1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos**

Como dito anteriormente (seções 3.2 e 4.2) o objetivo deste resultado é estabelecer o uso de mecanismos para obter, com o fornecedor de requisitos, o conhecimento dos requisitos, avaliá-los e validá-los. Para isso a organização deve:

- Entender o domínio do cliente;
- Registrar os requisitos elicitados e seus fornecedores;
- Avaliar os requisitos registrados utilizando critérios objetivos;
- Obter informações suficientes para aceitar ou não a produção dos projetos.

Tendo claros os objetivos pedidos nesse resultado, para o projeto estudado a empresa utilizou as seguintes atividades para resolvê-los:

### **5.3.1 Entender o domínio do cliente**

A empresa não definiu nenhum método fixo para a extração dos requisitos. Esta atividade fica sob responsabilidade do analista responsável pelo projeto, ou seja, o analista escolhe a maneira como irá levantar informações sobre o domínio do cliente para poder elicitá-los os requisitos a serem desenvolvidos.

Neste contexto cabe ressaltar o que a empresa alega defendendo esta sistemática, conforme segue: *“O fato de deixar que o analista escolha o modo de levantar os requisitos pode trazer vantagens por ser um modo flexível, permitindo ao profissional adotar métodos diferentes que sejam mais apropriados para diferentes clientes”*. Questionada sobre esta afirmação, a empresa defende que a mesma confia na qualidade de seus profissionais e isso não é um problema pra ela. Contudo, observando esta situação e com base na proposta apresentada neste trabalho, considera-se que a abordagem adotada pela empresa deixa a mesma dependente da qualidade do analista, algo que uma empresa madura não deve apresentar. A qualidade do produto também deve depender da qualidade do processo e não apenas da qualidade dos profissionais.

Outro perigo de adotar tal estratégia é que o analista pode adotar um método não eficiente para alguns casos e faltar informações para a compreensão total do projeto. Isto traria a necessidade de entrar mais vezes em contato com o fornecedor de requisitos e pode aumentar o número de mudanças no projeto. Se a fase inicial do processo de desenvolvimento de software não for efetuada de maneira consistente, parte ou todo o trabalho pode ser desperdiçado.

Por isso, sugere-se neste estudo de caso que a organização adote uma das técnicas propostas na Seção 4.2, seguindo critérios objetivos independentes de desejos pessoais e que a escolha torne-se um padrão na empresa.

### **5.3.2 Registrar os requisitos elicitados e seus fornecedores**

Como a forma de levantar os requisitos é livre, a empresa definiu um modelo de DRE (Documento de Requisitos) para assegurar que algumas informações sejam obtidas, contendo informações como prioridade dos requisitos e seus fornecedores. A descrição dos requisitos é feita em linguagem natural pelo analista de sistemas, para facilitar o entendimento das partes

envolvidas no projeto. Após a construção, o DRE é revisado pelo gerente do projeto e pelo fornecedor de requisitos. Qualquer alteração no documento é registrada em um histórico que diz a data, o motivo da alteração e o autor.

A forma de registro utilizada na empresa é a mesma proposta neste trabalho conforme apresentado na Seção 4.3 e é considerada válida por permitir que os requisitos sejam documentados de uma forma que facilite sua avaliação pelos critérios definidos pela empresa.

### **5.3.3 Validar os requisitos registrados utilizando critérios objetivos**

Todos os requisitos que foram definidos no DRE (Documento de Requisitos) devem ser validados utilizando critérios objetivos definidos pela empresa. Para essa avaliação a empresa utiliza os critérios propostos pela IEEE std 830 (IEEE, 1998), utilizando um *checklist* baseado no que foi citado nesse trabalho (Seção 4.2.3).

A forma de avaliação utilizada na empresa é a mesma proposta nesse trabalho e é considerada válida por seguir os padrões indicados pela IEEE para se obter um bom DRE.

### **5.3.4 Fornecer informações suficientes para a organização aceitar ou não a produção dos projetos**

A validação dos requisitos após avaliação indica que estes estão compreensíveis para todos. Assim, o gerente de projeto solicita uma reunião junto ao comitê da empresa para avaliar se há informações suficientes para iniciar o projeto, a viabilidade do projeto e verificar os recursos necessários. Assim, considera-se este procedimento indicado para atender a GRE1. Embora a empresa tenha um leve desvio em relação ao modo de levantar os requisitos, isso não afetaria a certificação, pois o MPS.BR não cobra no nível G como deve ser feita a negociação com o cliente.

## **5.4 GRE2 - Obtenção do comprometimento da equipe técnica com os requisitos aprovados**

Após a aprovação dos requisitos, um comprometimento formal da equipe técnica com os requisitos aprovados deve ser obtido e registrado. Por isso, o gerente do projeto solicita uma reunião com a equipe técnica para apresentar quais foram os requisitos definidos e deixar claro o que deverá ser produzido.

A apresentação é feita em forma de forma oral com ajuda de slides e o comprometimento da equipe é registrado em ata de reunião. A forma de apresentação e obtenção de compromisso utilizado na empresa é a mesma proposta nesse trabalho e é considerada válida porque permite o registro do compromisso da equipe quando consciente das exigências do projeto.

## **5.5 GRE3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida**

Como dito anteriormente (seções 3.4 e 4.4) o objetivo deste resultado é a definição de um método que descreva como a organização fará a rastreabilidade bidirecional entre os requisitos e os produtos de trabalho, visando manter o controle dos requisitos e calcular o impacto deles durante o projeto.

A empresa identificou como importante rastrear as relações entre: “*Requisito x Requisito*”; “*Requisitos x Caso de Teste*”; “*Requisitos x Código*”; e “*Requisitos x chamado*”. Onde “chamado” é o nome dado pela empresa para cada tarefa descrita em seu ERP (*Enterprise Resource Planning*), ou seja, todo o serviço da empresa é representado em forma de “chamado” no ERP, lá é possível obter informações sobre as tarefas que serão executadas e daquelas que já foram executadas.

As relações entre Requisito x Requisito; Requisito x Caso de Teste; e Requisito x “chamado” são feitas em planilha eletrônica. A relação de dependência é feita de maneira simples onde um “X” na célula marca a relação entre os elementos da linha e da coluna.

Foi proposto para a empresa uma mudança na forma de registrar as dependências com base na estratégia proposta neste trabalho (Seção 4.4) e originalmente apresentada em TORANZO (2002). Essa nova forma de indicação da dependência entre os requisitos e artefatos, mostraria além da existência ou não de relação entre eles, uma forma de registrar, subjetivamente, o tipo de relação ou se essa é forte ou fraca. A empresa negou o proposto declarando que estava em fase de adaptação ao sistema de rastreabilidade, por ser novo na empresa, e que futuramente ela poderia aceitar a proposta caso houvesse a necessidade de conhecer melhor a relação.

Para as relações “*Requisitos x Código*” a empresa adotou o “Campo de Rastreabilidade”, proposto nesse trabalho (Seção 4.4), adicionando-o em cada chamado do seu ERP. Periodicamente o gerente de projeto revisa e monitora esses artefatos para mantê-los válidos.

A forma de rastreabilidade utilizada na empresa é a mesma proposta nesse trabalho e é considerada válida.

## **5.6 GRE4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos**

Como dito anteriormente (Seções 3.5 e 4.5) o objetivo deste resultado é a definição de um método que descreva como a organização fará revisões em planos e produtos de trabalho do projeto, visando identificar e corrigir inconsistências em relação aos requisitos.

Para essa atividade a empresa adotou o fluxo proposto por esse trabalho (Seção 4.5, Figura 7), onde o monitoramento periódico é feito pelo gerente do projeto e este também é encarregado pela preparação da reunião de revisão. A reunião é feita junto ao comitê da empresa em marcos pré-determinados ao final de cada fase do ciclo de vida, possibilitando identificar e corrigir inconsistências em relação aos requisitos e ao projeto antes de iniciar uma nova fase. Algumas vezes, devido à distância, os clientes, ou seu representante, não participam desta reunião.

A forma de revisar os planos e produtos de trabalho utilizada na empresa é a mesma proposta nesse trabalho e é considerada válida porque integra a revisão ao resultado do monitoramento, permite um estudo e uma preparação antes da reunião, além de tornar explícito para todos os interessados e possibilitar o registro da situação que servirá como fonte histórica.

## **5.7 GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto**

Como dito anteriormente (Seções 3.6 e 4.6) o objetivo deste resultado é estabelecer o uso de mecanismos para gerenciar as mudanças nos requisitos ao longo do projeto, já que, mudanças nos requisitos são inevitáveis durante o processo de desenvolvimento.

Para essa atividade a empresa adotou o fluxo proposto por esse trabalho (Seção 4.6, Figura 8), onde a solicitação de mudança, que pode vir do cliente, da empresa, ou outra forma, é analisada pelo gerente do projeto, que estuda o impacto da mudança utilizando os artefatos de rastreabilidade. Com o desvio e seu impacto estudados e compreendidos, o gerente do projeto faz uma reunião com o comitê da empresa para decidir se a mudança será aceita ou não, ou ainda, se esta será tratada como um novo projeto. Após a decisão a resposta final é registrada em ata e as partes interessadas, como cliente, fornecedor de requisitos entre outros, serão comunicadas. Certas vezes, devido à distância, alguma parte interessada, ou sua representante, não participa desta reunião, mas são avisados por e-mail, telefonemas ou visitas.

A forma de gerenciar as mudanças nos requisitos utilizada na empresa é a mesma proposta nesse trabalho e é considerada válida porque aborda todo o tipo de solicitação, permite uma análise prévia esclarecendo a causa e conhecendo o impacto, permite a organização se adiantar perante o cliente, além de tornar explícito para todos os interessados e possibilitar o registro da situação que servirá como fonte histórica.

## **5.8 Processo de Implantação da Engenharia de Requisitos visando a melhoria da Qualidade de Software**

A TECINCO não seguiu nenhum processo para implantar as novas técnicas de engenharia de requisitos e outras mudanças no seu processo de desenvolvimento. Todas as mudanças foram implantadas seguindo a orientação da consultora que acompanhou a empresa.

No entanto, é possível comparar algumas características apresentadas durante o processo de adaptação da empresa com as fases do processo de implantação proposto neste trabalho (Seção 4.7), no intuito de verificar se a empresa teria vantagens se adotasse este processo de implantação.

- **Fase 1 - Conscientização:** Não houve conscientização, as pessoas da empresa mal sabiam responder o que era o MPS.BR. A coleta de informações do estado atual não aconteceu, assim a empresa não sabia qual era exatamente a diferença do modo como ela desenvolvia com o esperado pelo nível, ou seja, a empresa não sabia o que tinha que mudar.

- **Fase 2 - Preparação:** Um grupo para trabalhar com o projeto de melhorias foi escolhido pela empresa e este grupo foi treinado. Porém como a empresa não sabia ao certo o que mudar aceitou o treinamento proposto pela consultora que acompanhava a empresa, infelizmente o curso não teve o resultado desejado, ajudando pouco o grupo de qualidade. É importante que nesta fase inicie na empresa a circulação das informações a respeito do trabalho sendo

desenvolvido. Entretanto, como o grupo de qualidade não estava orientado para isso, as informações ficaram restritas ao mesmo.

- **Fase 3 - Treinamento:** O grupo de qualidade não definiu um Plano de Treinamento para a empresa, fato que foi sentido na implantação, quando os funcionários sentiam dificuldades com o novo modelo de desenvolvimento. Por isso, a implantação teve que parar por um momento até que todos fossem treinados e conhecessem as mudanças e suas novas responsabilidades.

- **Fase 4 - Levantamento de Processos atuais:** O conhecimento de como o processo atual acontecia era feito de maneira individual e fragmentada o que atrasou um pouco para conhecer realmente o que precisava ser mudado ou complementado. À medida que o processo atual era descoberto, as técnicas adequadas para cada caso iam sendo estudadas.

- **Fase 5 - Elaboração do Plano de Melhoria:** Não houve um Planejamento de Melhoria do Processo, as mudanças eram guiadas pelos resultados esperados no guia de implementação SOFTEX (2009b). Um membro do grupo escolhia um resultado para atender e trabalhava na conclusão deste, esse modo se repetiu para todos os resultados do Nível G, sem se preocupar em estimar o tempo e recursos necessários para cada atividade de melhoria.

- **Fase 6 - Definição dos Processos e Fase 7 - Definição de Papéis e Responsabilidades:** Estas fases aconteceram de maneira simultânea na empresa, a definição dos processos, recursos e responsabilidades a serem implantados foram elaboradas tendo como base as boas práticas estudadas, as restrições e a cultura da organização. Várias correções feitas pela consultora que acompanhava a empresa foram de grande ajuda.

- **Fase 8 - Implantação:** Para iniciar a implantação foi escolhido um Projeto-piloto e definidos os recursos, responsabilidades e prazos conforme definido no novo plano. No Projeto-piloto foram encontradas várias dificuldades em andar com o projeto, estas foram documentadas e usadas como base para os ajustes.

- **Fase 9 – Acompanhamento e Ajustes:** Com as dificuldades encontradas durante o acompanhamento do Projeto-piloto foi possível realizar ajustes para que um próximo pudesse ser melhorado. As principais dificuldades foram falta de treinamento, resistência de adaptação ao novo método, dificuldade de relacionar os artefatos gerados.

Embora a empresa tenha chegado a um processo melhorado e que tende a atender as exigências do MPS.BR, a perda dos benefícios pela ausência de um processo de implantação foi notada. Muito processo tornou-se obscuro para quem estava fora do grupo de qualidade e principalmente, muitos erros e dificuldades descobertos apenas na fase de implantação

poderiam ter sido evitados, além do ganho com tempo e esforço que a empresa poderia ter conseguido.

## 5.9 Considerações Finais

Atualmente, há dois projetos sendo desenvolvidos na TECINCO que seguem o novo modelo de desenvolvimento, modificado para a adaptação do MPS.BR. Ainda não foi feita a avaliação de aderência ao MPS.BR por parte da empresa responsável pela consultoria. O primeiro projeto (estudo de caso abordado) começou a ser desenvolvido no mês de agosto encontrando-se atualmente no início da fase de implantação. O segundo está iniciando a análise da proposta de projeto. Em ambos os casos, ainda não foi possível fazer uma avaliação precisa a respeito dos benefícios da implementação do processo, pois ainda estão sendo feitos ajustes no processo. A análise dos resultados alcançados até o momento e as dificuldades encontradas permitem elencar um conjunto de lições aprendidas que são descritas a seguir.

Em relação à equipe de projeto e responsáveis pelo processo de qualidade e engenharia de software:

- É fundamental descrever claramente, no início dos trabalhos, as competências de cada equipe e seus membros. Esta iniciativa potencializa a participação individual e torna as ações coletivas ágeis e produtivas reduzindo o retrabalho e discussões recursivas;
- É necessário tratar melhor as diferenças de competência dos membros da equipe. Uma forma de contornar estas diferenças é respeitar a capacidade de cada membro e reduzir as deficiências com um processo contínuo de capacitação;
- É necessário treinamento mais profundo em processo de software. As características da implantação do modelo impõem que a equipe tenha conhecimentos de engenharia de software;
- Dependendo da cultura e contexto organizacional pode ser útil ter um “grupo de qualidade” como unidade formal, o que evita que outras unidades tenham iniciativas parecidas.

Em relação ao processo de trabalho:

- A estratégia de descrever o processo em linhas gerais e depois detalhar mostra-se efetiva, mas gera discussões. Entender o caráter iterativo e incremental do projeto de melhoria não é tão simples quanto parece;

- O trabalho inicial de discutir a organização dos componentes do processo (fluxos, atividades, tarefas, passos, etc.) facilita os trabalhos seguintes. Vale a ideia de dividir para conquistar, pois cada componente bem definido se torna um elemento de composição valioso para os demais componentes.

Em relação às partes interessadas:

- Gerenciar as expectativas dos diversos interessados pelo projeto deve ser uma prioridade. Mostrar resultados parciais do projeto tem que ser feito sempre e com cuidado;
- Equilibrar as demandas por processos e os objetivos do projeto de melhoria exige grande esforço. O que se percebe é um desejo ubíquo por um processo pronto de imediato;
- Deve-se observar que a melhor iniciativa de processo não é o bastante. A melhor forma de se avaliar os resultados é execução do processo em um projeto piloto.

Os próximos passos do projeto de Melhoria do Processo de Software da TECINCO, são a execução de um número maior de projetos piloto no sentido de amadurecer o processo e a constante melhoria do processo tentando atingir altos níveis de maturidade do MPS.BR.

## Capítulo 6

# Considerações Finais e Trabalhos Futuros

A dificuldade de entender o cliente, quanto às suas necessidades é constatada em qualquer empresa desenvolvedora de software, independente do segmento de mercado de atuação.

A etapa de obtenção dos requisitos de clientes é uma tarefa que exige pesquisas e adaptações à realidade da organização. A eficácia na utilização de técnicas e ferramentas adotadas por uma empresa pode não resultar de forma eficiente em outras organizações, porque a realidade das empresas difere devido a aspectos de cultura, localização, atuação no mercado, metodologias utilizadas, tipos de clientes, entre outros.

Constata-se, através da revisão bibliográfica deste, que a fórmula da solução ideal não existe e, as técnicas disponíveis devem ser adequadas à organização, onde serão dosadas na medida certa até que seja encontrada uma gama de processos que seja, de fato, efetivo e funcional. Os resultados serão sentidos e experimentados em toda a organização que se beneficiará ao encantar seus clientes com resoluções rápidas e eficientes.

O presente trabalho contribui na área de gerenciamento de requisitos:

- Ao ressaltar a importância do levantamento de requisitos executado de forma adequada e correta;
- Ao alertar para o fato de que se a fase inicial do processo de desenvolvimento de software não for efetuada de maneira consistente, todo um trabalho pode ser desperdiçado;
- Ao retratar que quando o requisito é entendido e definido corretamente, a etapa de gerenciamento de requisitos se torna eficiente e poderosa para manter a organização destes;

- Quando ressalta a importância da identificação e análise dos impactos gerados pela rastreabilidade dos requisitos e a importância dessa atividade para o projeto;
- Ao mostrar a importância de um processo correto para a implantação das mudanças.

Outro ponto que merece destaque foi a percepção de ser altamente recomendado que a organização tenha um “Programa de Qualidade” antes de buscar a certificação. Este programa pode ser baseado no nível de maturidade que se pretende atingir e funcionaria como uma prévia à certificação, onde as mudanças necessárias para alcançar os resultados esperados teriam início, evitando assim incômodos com o tempo destinado à certificação e possivelmente reduziria o impacto cultural na empresa e os custos com consultorias.

De uma maneira geral, abordar a melhoria de processo como um projeto e respeitar os requisitos são as diretrizes mais importantes que se pode tirar das lições aprendidas deste trabalho. Isso envolve principalmente, gerenciar as expectativas dos diversos envolvidos no projeto e manipular a restrição de qualquer projeto, principalmente o que diz respeito ao escopo e tempo disponível.

É importante ressaltar que a proposta apresentada é destinada, mas não limitada, às micro e pequenas empresas, categoria na qual a empresa estudada é representativa. Para trabalhos futuros, existe a possibilidade de aplicar a proposta a outros estudos de caso; estender a proposta para outros níveis de maturidade ou investigar mais profundamente ferramentas computacionais de apoio à proposta.

## Referências

BLASCHEK, J. R. Gerência de Requisitos: O principal problema dos projetos de Software. *Developer's Magazine*, Rio de Janeiro, Agosto, 2002.

BPMI. *Business Process Management Initiative*. Disponível em: < <http://www.bpmi.org>>, Acessado em: 03 de Novembro de 2010.

BRIAND, L. et al. AINSI: An Inductive Method for Software Process Improvement: Concrete Steps and Guidelines. *IEEE Computer Society Press*, Los Alamitos, Califórnia, 1999.

CARVALHO, A. E. S. et al. Uma Estratégia para Implantação de uma Gerencia de Requisitos Visando a Melhoría Dos Processos de Software. In: *Anais do WER01 - Workshop em Engenharia de Requisitos*, 2001, Buenos Aires, 2001, p. 32-54.

DORFMANN, M. e THAYER, R. Standards, Guidelines, and Examples of System and Software Requirements Engineering. *IEEE Computer Society Press*, Los Alamitos, Califórnia, 1990.

FINE, L. G. *The SWOT Analysis: Using your Strength to overcome Weaknesses, Using Opportunities to overcome Threats*, 1 ed. Booksurge llc, 2009.

FREITAS, M. E. et al. Inspeção de Documentos de Requisitos Baseado em Técnica de Leitura PBR: Experiência Prática no CPqD. In: *Anais do Simpósio Brasileiro de Qualidade de Software*, Brasília, 2003.

GOGUEN, J. A. e LINDE, C. Techniques for Requirements Elicitation. In: *Proceedings, Requirements Engineering 1993*, IEEE CS Press, 1993, p. 152 - 164.

GOTEL, O. e FINKELSTEIN, A. An Analysis of the Requirements Traceability Problem. In: *Proceedings of the First International Conference Requirements Engineering (ICRE'94)*; Colorado Springs, IEEE CS Press, 1994, p. 94 - 101.

HAZAN, C. e LEITE, J. C. S. P. Indicadores para a Gerência de Requisitos. In: *Anais do WER03 - Workshop em Engenharia de Requisitos*, 2003, Piracicaba, 2003, pp 285-301.

IEEE Std 830-1998. *IEEE recommended practice for software requirements specifications*, Institute of Electrical and Electronics Engineers, 1998.

ISO/IEC 9126-1, *Engenharia de software - Qualidade de produto*, 2003.

- KENDALL, *Systems Analysis and Design*, 8 ed., Pearson, 2010.
- KOSCIANSKI, A. e SOARES, M. S. “Qualidade de Software, Aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software”. 2 ed. Novatec Editora, 2007.
- KOTONYA, SOMMERVILLE. *Requirements Engineering: Processes and Technique*, 1 ed, Wiley. 1998.
- LEFFINGWELL, D. *Managing Software Requirements - A Use Case Approach*, 2 ed., Hardcover, 2003.
- LEITE, J. C. S. P. *Gerenciando a Qualidade de Software com Base em Requisitos*. In: *Qualidade de Software: Teoria e Prática*. 1ª. Edição. Prentice Hall. p 238-246, 2001.
- MACAULAY, L., *Requirements Engineering*, 2 ed. Springer, 1996.
- MCT – Ministério da Ciência e Tecnologia, Disponível em: <[www.mct.gov.br](http://www.mct.gov.br)>, Acessado em: 16 de Junho 2010.
- MICHAELIS, *Moderno Dicionário da Língua Portuguesa*, 3 ed, Melhoramentos, 2009.
- NBR ISO 8402, *Gestão da qualidade e garantia da qualidade – Terminologia*, 1994.
- NOGUEIRA, M. O. *Qualidade no Setor de Software Brasileiro*, Tese de Doutorado, COPPE/UFRJ, abril 2006.
- PÁDUA, W. *Engenharia de Software fundamentos, métodos e padrões*. 1 ed. LCT Livros técnicos e científicos, 2001.
- PARASURAMAN, A. Z. *Marketing Research*. 1 ed. Addison Wesley, 1991.
- PRESSMAN, R. S. *Engenharia de Software*. 6 ed, McGraw Hill, 2006.
- SANTANDER, V. F. A., CASTRO, J. F., Deriving Use Cases from Organizational Modeling. In: *IEEE JOINT INTERNATIONAL REQUIREMENTS ENGINEERING CONFERENCE – RE02*, 2002. University of Essen, Germany, 2002, p. 32–39
- SAYÃO, M., STAA, A. V. e LEITE, J. C. S. P., *Qualidade em Requisitos*, Monografia em Ciência da Computação, DI/PUC-Rio, Rio de Janeiro, 2003.
- SEBRAE, Disponível em: <<http://www.sebrae.com.br/>>. Acessado em: 16 de Setembro de 2010.
- SEI (2010) “Página Oficial CMMI”, Disponível em: <[www.sei.cmu.edu/cmmi/index.cfm](http://www.sei.cmu.edu/cmmi/index.cfm)>, Acessado em: 14 de Junho 2010.

SERPRO, Disponível em: <<http://www.serpro.gov.br/>>. Acessado em: 18 de Setembro de 2010.

SERRANO, M. et al. Uma Proposta para Avaliação de Equipes de Requisitos. In: *Anais do WER08 - Workshop em Engenharia de Requisitos*, 2008, Barcelona, 67-75, 2008.

SOFTEX (2009a), “MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral” Setembro 2009. Disponível em: <[http://www.softex.br/mpsBr/\\_guias/default.asp](http://www.softex.br/mpsBr/_guias/default.asp)>, Acessado em: 25 de Fevereiro de 2010.

SOFTEX (2009b), “Guia de Implementação, Guia de Avaliação, Guia de Aquisição” Setembro 2009. Disponível em: <[http://www.softex.br/mpsBr/\\_guias/default.asp](http://www.softex.br/mpsBr/_guias/default.asp)>, Acessado em: 25 de Fevereiro de 2010.

SOMMERVILLE, I. *Engenharia de Software*. 6 ed, Editora Prentice Hall, 2003.

TECINCO, Disponível em: <<http://www.tecinco.com.br/>>, Acessado em: 26 de Setembro de 2010.

THAYER & DORFMAN. *Software Requirements Engineering*. 2 ed, IEEE CS, 1997.

TORANZO, M. A. *Uma Proposta para Melhoria de Requisitos de Software*. Tese de Doutorado, UFPE, setembro 2002.

WIERINGA R. J. *An introduction to requirements traceability*. Faculty of Mathematics and Computer Science, University of Vrije, Amsterdam, Setembro 1995. Relatório: IR-389.

YOURDON, *Declínio e queda dos analistas e programadore*, 3 ed, Makron Books, 1995.

YU, E., Modelling organizations for informations systems requirements engineering. In: *IEEE INTERNATIONAL SYMPOSIUM ON REQUERIMENTS ENGINEERING*, 1993. San Diego: IEEE Computer Society, 1993. p. 34-41.