



**UNIVERSIDADE FEDERAL DO PARÁ**  
**CURSO DE SISTEMAS DE INFORMAÇÃO**  
**Campus Universitário de Marabá**

**Caroline Costa da Silva**

**Melhoria da Qualidade de Processos de Software  
com Base nas Avaliações MPS-BR**

**Marabá – PA**  
**2010**

**Caroline Costa da Silva**

**Melhoria da Qualidade de Processos de Software  
com Base nas Avaliações MPS-BR**

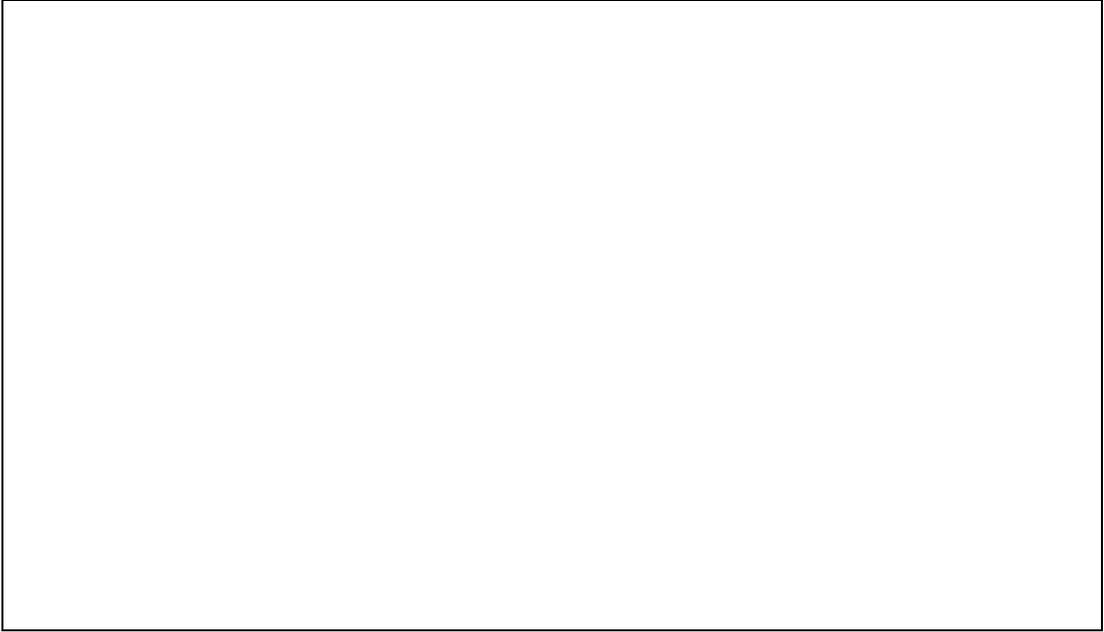
Trabalho de Conclusão de Curso,  
apresentado à Universidade Federal do  
Pará, como parte dos requisitos necessários  
para obtenção do Título de Bacharel em  
Sistemas de Informação.

**Orientador (a):** Zenaide Carvalho da Silva

**Co-orientador (a):** José Santos Barreto

**Marabá – PA  
2010**

# FICHA CATALOGRÁFICA

A large, empty rectangular box with a thin black border, intended for entering cataloging information. It occupies the upper half of the page below the title.

**Caroline Costa da Silva**

**Melhoria da Qualidade de Processos de Software  
com Base nas Avaliações MPS-BR**

Este Trabalho de Conclusão de Curso foi julgado adequado à obtenção do grau de Bacharel em Sistemas de Informação e aprovado em sua forma final pelo curso de Sistemas de Informação da Universidade Federal do Pará.

Marabá – PA, 14 de Dezembro de 2010.

---

Nome, Titulação e Assinatura de membro da Banca Avaliadora

---

Nome, Titulação e Assinatura de membro da Banca Avaliadora

---

Nome, Titulação e Assinatura de membro da Banca Avaliadora

---

Nome, Titulação e Assinatura de membro da Banca Avaliadora

# AUTORIZAÇÃO

Eu CAROLINE COSTA DA SILVA, matrícula Nº 05060002207, AUTORIZO a Universidade Federal do Pará – UFPA, a divulgar total ou parcialmente o presente Trabalho de Conclusão de Curso através de meios eletrônicos.

Marabá - PA, 17/12/2010.

---

Caroline Costa da Silva

# DEDICATÓRIA

*Aos, meus pais que sempre lutaram muito por mim  
e por meu irmão e nunca desistiram perante as  
dificuldades da vida e durante toda vida  
trabalharam para que os filhos crescessem dignos  
de ser o que quiserem. Em especial a minha avó  
Neuza, tão querida, por me ajudar muito e  
principalmente por ser minha avó.*

*A eles que formam o principal esteio da minha  
família se tornando por fim o meu tudo.*

## **AGRADECIMENTOS**

Aos meus poucos, porém, grandes e verdadeiros amigos, que me ajudaram nos momentos de dificuldade pessoal e que sempre estiveram ao meu lado nos momentos de alegria e farra.

Aos caros colegas de curso, que sempre alegraram as aulas com suas piadas altamente “coisativas”!

A minha orientadora por ter me acolhido aos “45 do segundo tempo”, ao meu co-orientador, meu amigo e incentivador.

Ao meu ex-orientador, professor “Oráculo”, que ensinou uma grande lição de vida, no sentido mais irônico possível: “Faça das suas escolhas de vida, de parceria, ou de qualquer outra coisa, sempre as melhores possíveis”.

# EPÍGRAFE

*Da luta não me retiro, me atiro do alto e que  
me atirem no peito*

*Da luta não me retiro, todo dia de manhã é  
nostalgia das besteiras que fizemos ontem*

*(Teatro Mágico)*

# RESUMO

*Qualidade é uma das palavras chaves para a competitividade das organizações contemporâneas, pois, é hoje um requisito fundamental em todas as áreas de atividade humana.*

*A indústria de software, nos últimos anos, tem procurado métodos que assegurem que seus produtos tenham qualidade e utilidade aceitável. Essa demanda por qualidade cria um cenário onde se faz necessário a utilização e o desenvolvimento de modelos de qualidade de software, que permitam que os produtos finais estejam em conformidade com especificações, padrões de desenvolvimento e com as necessidades exigidas e os objetivos propostos por quem produz software.*

*Verifica-se, então, que os esforços em melhoria da qualidade de software não devem focar só no produto, mas principalmente no processo de desenvolvimento, aprimorando-o, e no cliente, fornecendo um produto melhor e dentro das necessidades exigidas. O projeto MPS-BR provê um modelo de melhoria baseado em conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de produtos de software e serviços correlatos. É uma iniciativa de grande relevância como fator de melhoria na qualidade dos produtos oferecidos pela indústria nacional de software. O componente de avaliação (MA-MPS) deste modelo e os resultados de sua implementação, como insumo para a melhoria, são o foco deste trabalho.*

**Palavras-chave:** *qualidade de software, produtos de software, processos de software, melhoria de processos.*

# ABSTRACT

*Quality is one of the key words for the competitiveness of contemporary organizations, therefore, is today a basic requirement in all areas of human activity.*

*The software industry, in recent years, is seeking methods to ensure that their products have acceptable quality and usefulness. This demand for quality creates a scenario where it is necessary the use and development of models of software quality, enabling the final products conform to specifications, standards development and the required needs and objectives proposed by those who produce software.*

*It appears, then, that efforts to improve the quality of software should not only focus on the product, but mainly in the development process, improving it, and the client, providing a better product and within the required needs. The MPS-BR project provides an improvement model based on concepts of maturity and process capability for assessing and improving quality and productivity of software products and related services. It is an initiative of great relevance as a factor in improving the quality of products offered by the national software industry. The evaluation component (MA-MPS) in this model and the results of its implementation, as input for the improvement, the focus of this work.*

**Keywords:** *software quality, software products, processes, software process improvement.*

# LISTA DE FIGURAS

<i>Figura 1 - Camadas da Engenharia de Software</i>	9
<i>Figura 2 - O processo de software</i>	15
<i>Figura 3 - Modelo de qualidade para qualidade externa e interna</i>	22
<i>Figura 4 - Modelo de qualidade para qualidade em uso</i>	27
<i>Figura 5 - Gerenciamento de qualidade e desenvolvimento de software</i>	31
<i>Figura 6 - Atributos de qualidade de software</i>	34
<i>Figura 7 - Fatores principais de qualidade de produtos de software</i>	37
<i>Figura 8 - Qualidade baseada em processo</i>	38
<i>Figura 9 - Estrutura do modelo MPS-BR</i>	41
<i>Figura 10 - Níveis de maturidade do modelo MR-MPS</i>	42
<i>Figura 11 - Processo de avaliação MR-MPS</i>	49
<i>Figura 12 - Fluxo do subprocesso 1 - Contratar avaliação</i>	55
<i>Figura 13 - Fluxo do subprocesso 2 - Preparar a realização da avaliação</i>	58
<i>Figura 14 - Fluxo do subprocesso 4 - Realizar avaliação final</i>	65
<i>Figura 15 - Fluxo do subprocesso 4 - Documentar os resultados da Avaliação</i>	73

# LISTA DE TABELAS

<i>Tabela 1 - Atributos essenciais de um bom software</i>	12
<i>Tabela 2 - Descrição dos atributos de processo (AP)</i>	43
<i>Tabela 3 - Níveis MR-MPS, seus processos e atributos de processo</i>	44
<i>Tabela 4 - Processo de avaliação</i>	51
<i>Tabela 5 - Tempo máximo e mínimo de uma avaliação MR-MPS inicial</i>	62
<i>Tabela 6 - Escala de caracterização do grau de implementação de um resultado esperado do processo e do atributo do processo nos projetos</i>	67
<i>Tabela 7 - Regras de agregação de caracterização dos resultados esperados dos processos e dos atributos do processo nos projetos para obtenção da caracterização organizacional</i>	68
<i>Tabela 8 - Regras de caracterização do grau de implementação dos atributos do processo na organização</i>	69
<i>Tabela 9 - Caracterização de atributos do processo para satisfazer aos níveis MR-MPS</i>	70
<i>Tabela 10 - Métricas usadas na perspectiva Organizacional</i>	78
<i>Tabela 11 - Nível de satisfação dos clientes apresentado</i>	80
<i>Tabela 12 - Métricas usadas na perspectiva de projetos</i>	81
<i>Tabela 13 - Métricas usadas na perspectiva do modelo MPS</i>	83

# LISTA DE ABREVIATURAS

ABNT - Associação Brasileira de Normas Técnicas

AP - Atributo de Processo

BID - Banco Interamericano de Desenvolvimento

CMM - Capability Maturity Model (Modelo de Maturidade e Capacitação)

CMMI - Capability Maturity Model Integrated (Modelo de Integração de Maturidade e Capacitação)

FINEP - Financiadora de Estudos e Projetos

IA - Instituição Avaliadora, autorizada pela SOFTEX

IEC - International Electrotechnical Commission (Comissão Eletrotécnica Internacional)

IEEE - Institute of Electrical and Electronics Engineers (Instituto de Engenheiros Eletricistas e Eletrônicos)

II - Instituição Implementadora, autorizada pela SOFTEX

IOGE - Instituição Organizadora de Grupo de Empresas, autorizada pela SOFTEX

ISO - International Organization for Standardization (Organização Internacional de Normalização)

MA-MPS - Método de Avaliação de Melhoria do Processo de Software

MCT - Ministério da Ciência e Tecnologia

MN-MPS - Modelo de Negócios de Melhoria do Processo de Software

MR-MPS - Modelo de Referência de Melhoria do Processo de Software

MPS-BR - Melhoria do Processo de Software Brasileiro

NBR - Norma Brasileira

PME - Pequenas e Médias Empresas

RAP - Resultado do Atributo de Processo

RUP - Rational Unified Process (Processo Unificado Racional)

SCAMPI <sup>SM 3</sup> - Standard CMMI Appraisal Method for Process Improvement (Método Padrão de Avaliação do CMMI para Melhoria de Processo)

SEBRAE - Serviço Brasileiro de Apoio às Micro e Pequenas Empresas

SEI - Software Engineering Institute

SOFTEX - Associação para Promoção da Excelência do Software Brasileiro

SQA - Software Quality Assurance (Garantia da Qualidade de Software)

SPICE - Software Process Improvement and Capability dEtermination (Melhoria de Processo de Software e capacidade)

TI - Tecnologia da Informação

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>2</b>	<b>ENGENHARIA DE SOFTWARE</b>	<b>9</b>
2.1	<i>Software</i>	10
2.2	<i>Produtos de Software</i>	11
2.3	<i>Processos de Software</i>	13
2.4	<i>Melhores Práticas de Software</i>	17
2.5	<i>Modelos de Processo de Software</i>	18
<b>3</b>	<b>NBR ISO/IEC 9126-1 – UM MODELO DE QUALIDADE DE SOFTWARE</b>	<b>21</b>
3.1	<i>Qualidade externa e interna</i>	21
3.1.1	Funcionalidade	22
3.1.2	Confiabilidade	23
3.1.3	Usabilidade	24
3.1.4	Eficiência	24
3.1.5	Manutenibilidade	25
3.1.6	Portabilidade	26
3.2	<i>Qualidade em uso</i>	27
3.2.1	Eficácia	27
3.2.2	Produtividade	27
3.2.3	Segurança	28
3.2.4	Satisfação	28
<b>4</b>	<b>GARANTIA DE QUALIDADE DE SOFTWARE</b>	<b>29</b>
4.1	SQA	29
4.2	<i>Atividades de SQA</i>	31
4.3	<i>Plano de SQA</i>	33
<b>5</b>	<b>MELHORIA DE PROCESSOS DE SOFTWARE</b>	<b>37</b>

5.1	<i>MPS-BR - Modelo de Melhoria de Processo de Software Brasileiro</i>	39
<b>6</b>	<b>AVALIAÇÃO DE PROCESSOS DE SOFTWARE</b>	<b>46</b>
<b>7</b>	<b>MÉTODO DE AVALIAÇÃO MPS</b>	<b>49</b>
7.1	<i>Equipe de Avaliação</i>	52
7.2	<i>SUBPROCESSO 1 – Contratar avaliação</i>	54
7.3	<i>SUBPROCESSO 2 – Preparar realização da avaliação</i>	57
7.4	<i>SUBPROCESSO 3 – Realizar avaliação final</i>	64
7.5	<i>SUBPROCESSO 4 – Documentar os resultados da avaliação</i>	72
<b>8</b>	<b>RESULTADOS DAS EMPRESAS COM BASE NAS MELHORIAS PROPOSTAS PELO MA-MPS76</b>	
<b>9</b>	<b>CONCLUSÃO</b>	<b>86</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>87</b>
	<b>BIBLIOGRAFIA</b>	<b>90</b>

# 1 INTRODUÇÃO

Qualidade é uma das palavras chaves para a competitividade das organizações contemporâneas. Segundo Bornia e Wenke (2000), nos dias de hoje as empresas estão inseridas num ambiente competitivo, onde, a busca pela qualidade é o caminho para a sua sobrevivência e manutenção no mercado em longo prazo.

A qualidade é hoje um requisito fundamental em todas as áreas de atividade humana, pois, todos querem oferecer e receber produtos e serviços com qualidade.

Garvin (2002) destaca quatro etapas de garantia de qualidade:

- a. Inspeção;
- b. Controle estatístico;
- c. Garantia;
- d. E gerenciamento estratégico da qualidade.

As quatro etapas, citadas acima, tratam de métodos e ferramentas certificadoras da qualidade do produto e da produção, através de inspeção ou controle estatístico; da inovação ao buscar a garantia da qualidade de uma forma total, através da qualidade em todos os processos; da concepção e desenvolvimento do produto ou serviço até a entrega ao cliente, além da utilização da qualidade como fator estratégico e competitivo, criando a relação de qualidade à lucratividade.

O conceito de qualidade não é novo. Essa noção foi evoluindo ao longo do tempo mediante as especificidades que cada período apresentou na história do desenvolvimento humano (BORNIA; WERNKE, 2000 *apud* PALADINI, 1995). Conceituá-la pode ser complicado, por sua diversidade de significados populares.

O SEBRAE<sup>1</sup> (2010) define qualidade como o conjunto de características de todo produto e serviço ou relação planejada, praticada e verificada, almejando-se ir além das expectativas de satisfação das pessoas envolvidas.

Porém, por mais que se encontrem dificuldades em conceituar a qualidade, é essencial admiti-la como uma atitude que possibilita a obtenção de vantagens sobre a concorrência, além de diferencial estratégico no mercado.

Para Garvin (2002), diferentes pessoas percebem a qualidade de diferentes formas, contexto no qual é possível identificar cinco perspectivas diferentes:

- a. *A visão transcendental:* a qualidade é algo que podemos reconhecer, mas não podemos definir;
- b. *A visão do usuário:* a qualidade é adequação ao propósito pretendido;
- c. *A visão do fabricante:* a qualidade é a conformidade com a especificação do produto;

---

<sup>1</sup> Serviço Brasileiro de Apoio às Micro e Pequenas Empresas.

- d. *A visão do produto:* a qualidade está relacionada às características inerentes ao produto;
- e. *A visão do mercado:* a qualidade depende do valor que os consumidores estão dispostos a pagar pelo produto;

A atividade de software ganha maior relevância na medida em que as indústrias relacionadas à tecnologia de informação (TI) expandem sua presença na economia, assim como suas articulações com outros setores (ROSELINO, 2006). Atualmente, cada vez mais, o produto software passa a ser um componente comum em uma série de outros produtos, desde carros, fornos de micro-ondas, elevadores, telefones, até sistemas de informação organizacionais. Diante disto, a sociedade passa a ver o software como produto integrante de suas vidas, e, essa nova realidade gera uma demanda maior por qualidade (ROCHA; MALDONADO e WEBER, 2001).

Da mesma maneira que organizações em geral procuram meios para garantir a qualidade dos produtos que oferecem, a indústria de software também tem procurado métodos que assegurem que seus produtos tenham qualidade e utilidade aceitável (PFLEEGER, 2007).

A norma ISO/IEC 9126 define qualidade de software como “a totalidade de características de um produto de software que lhe confere a capacidade de satisfazer necessidades explícitas e implícitas”. A qualidade de software deve estar em conformidade com especificações e padrões de desenvolvimento, a necessidades exigidas e objetivos propostos por quem produz software, garantindo que tanto o produto software quanto um bom

processo de desenvolvimento do mesmo atinjam níveis de qualidade que mantenham as necessidades dos usuários (DEVMEDIA, 2008).

Freqüentemente nós estreitamos nosso enfoque e pensamos somente sobre os defeitos e falhas de um software, mas, a qualidade é muito mais ampla, e quando avaliamos a qualidade dos produtos do desenvolvimento devemos ampliar este quadro (PFLEEGER, 2004). Pois, falhas se apresentam sem aviso prévio e podem gerar impactos econômicos e sociais até mesmo irremediáveis, tanto para as empresas de software quanto para os usuários dos seus produtos (DEVMEDIA, 2008).

A demanda por qualidade de software cria uma situação onde se faz necessário a utilização e o desenvolvimento de modelos de qualidade software. Mesmo com este cenário, segundo Gomes (2003), ainda é elevado o número de empresas brasileiras de software que não adotam técnicas para melhoria da qualidade de seus produtos, em um mercado onde empresas que desenvolvem software de qualidade são mais competitivas.

Porém, na medida em que cresce a demanda por sistemas complexos, com grande responsabilidade no contexto das organizações, a qualidade desponta como um fator essencial no desenvolvimento de produtos de software, e isso, têm estimulado uma maior disposição para investimentos em qualidade por parte das empresas do setor em questão. Entretanto, é importante frisar que a qualidade não se obtém de forma espontânea, ela deve ser construída (DUARTE; FALBO, 2006).

Para a SOFTEX<sup>2</sup> (2009a) essas mudanças que estão ocorrendo nos ambientes de negócios têm motivado as empresas de software a modificar suas estruturas organizacionais e seus processos produtivos, saindo da visão tradicional baseada em áreas funcionais para a de redes de processos centrados nos clientes. Para estas empresas, alcançar competitividade pela qualidade implica tanto na melhoria da qualidade dos produtos de software e serviços correlatos, como dos processos de produção e distribuição de software. Qualidade é fator crítico de sucesso, e para que se tenha um setor de software competitivo, nacional e internacionalmente, é essencial que os empreendedores do setor coloquem a eficiência e a eficácia dos seus processos em foco, visando à oferta de produtos de software e serviços correlatos conforme padrões internacionais de qualidade.

Segundo Pressman (2002), a qualidade de software é uma meta importante. É uma combinação complexa de fatores, que variam de acordo com as diferentes aplicações e clientes que as solicitam. Onde, diversos fatores podem afetar a qualidade de um software, como: sua funcionalidade, sua eficiência e sua capacidade de manutenção, características mais difíceis de serem avaliadas por poderem ser mensuradas somente de forma indireta. Assim, cada caso precisa ser avaliado, porque métricas de qualidade oferecem uma indicação de quão estreitamente o software conforma-se às exigências implícitas e explícitas do cliente. Um software é medido por muitas razões, dentre elas:

- a. Indicar a qualidade do produto;

---

<sup>2</sup> Associação para Promoção da Excelência do Software Brasileiro.

- b. Avaliar a produtividade das pessoas que produzem o produto;
- c. Avaliar os benefícios derivados de novos métodos (quanto à produtividade e qualidade) e ferramentas de software;
- d. Formar uma linha básica para estimativas;
- e. Ajudar a justificar pedidos de novas ferramentas ou treinamentos.

Avaliar produtos de software e seus processos de produção é necessário porque avaliações são capazes de gerar os insumos necessários para o ajuizamento da qualidade de um software. Pois, métricas melhores e mais confiáveis levam a uma melhor avaliação e conseqüentemente representam uma abordagem consistente para a inferência da qualidade de um software (MAGELA, 2006).

É preciso, de fato, que os esforços em melhoria da qualidade de software tenham seu foco não apenas no produto, fazendo um software melhor, mas principalmente no processo de desenvolvimento deste, fazendo melhor o software, e no cliente, fazendo o software mais fácil de usar (GOMES, 2003).

O MPS-BR<sup>3</sup> é um programa mobilizador, de longo prazo, criado em dezembro de 2003 e coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), que conta com o apoio do Ministério da Ciência e Tecnologia (MCT), da Financiadora de Estudos e

---

<sup>3</sup> MPS-BR, MR-MPS, MA-MPS e MN-MPS são marcas da SOFTEX. A sigla MPS-BR está associada ao programa MPS-BR – Melhoria do Processo de Software Brasileiro e a sigla MPS está associada ao modelo MPS – Melhoria do Processo de Software.

Projetos (FINEP), do SEBRAE e do Banco Interamericano de Desenvolvimento (BID), e tem por objetivo a Melhoria de Processo do Software Brasileiro (SOFTEX, 2009a).

O modelo MPS-BR baseia-se nos conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de produtos de software e serviços correlatos. O MPS-BR possui três componentes, dentre eles o Método de Avaliação (MA-MPS), que foi definido de forma a: permitir a avaliação objetiva dos processos de software de uma organização; permitir a atribuição de um nível de maturidade do Modelo de Referência (MR-MPS) com base no resultado da avaliação; ser aplicável a qualquer domínio na indústria de software; além de ser aplicável a organizações de qualquer tamanho (SOFTEX, 2009a).

Este trabalho tem por objetivo demonstrar a importância da melhoria dos processos de software através de avaliações que produzam os insumos necessários para tanto, objetivando proporcionar melhor garantia de qualidade. E promover o MA-MPS como um método de avaliação voltado para as características e objetivos das empresas brasileiras. Porém, antes de chegarmos aos pontos principais deste trabalho, vamos conceituar os tópicos necessários ao bom entendimento da Engenharia de Software.

- a. O *Capítulo 2* conceitua a engenharia de software, o software, e sua importância nos dias de hoje, o processo software e melhores práticas de desenvolvimento.
- b. O *Capítulo 3* aborda a NBR ISO/IEC 9126-1 que define um modelo de qualidade aplicáveis a qualquer tipo de sistema.

- c. O *Capítulo 4* trata de conceitos sobre garantia de qualidade de software (SQA).
- d. O *Capítulo 5* questiona o porquê da melhoria da qualidade de software, e apresenta conceitos o MPS-BR, um modelo de melhoria de qualidade de software.
- e. O *Capítulo 6* caracteriza, de forma breve, a avaliação de processos de software como um aspecto importante no processo de melhoria de qualidade de software.
- f. O *Capítulo 7* descreve o MA-MPS.
- a. O *Capítulo 8* pretende demonstrar a variação de desempenho das organizações desenvolvedoras de software, em função da adoção do modelo MPS.

## 2 ENGENHARIA DE SOFTWARE

Existem inúmeras definições para Engenharia de Software. Para Pressman (2002), engenharia nada mais é do que a análise, projeto, construção, verificação e gestão de elementos técnicos/sociais. Definindo a engenharia de software como uma tecnologia em camadas. As quais devem dar apoio e sustentação à engenharia de software, focando sempre a qualidade.



Figura 1 - Camadas da Engenharia de Software  
Fonte – Engenharia de Software (PRESSMAN, 2002)

Para o IEEE (1994), engenharia de software é:

- a. A aplicação de uma abordagem sistemática, disciplinada e quantificável, para o desenvolvimento, operação e manutenção do software, isto é, a aplicação de engenharia ao software.
- b. O estudo de abordagens como as de (a).

Sommerville (2007) descreve a engenharia de software como uma disciplina de engenharia que se relaciona com todos os aspectos da produção

de software, desde a especificação do sistema até sua manutenção. Não está relacionada somente com os processos técnicos de desenvolvimento, pois, também está relacionada com atividades de gerenciamento de projetos de software, desenvolvimento de ferramentas, além de métodos e teorias de apoio a produção.

## 2.1 Software

Atividades em que prevalecem tecnologias de informação e comunicação (TICs) são compostas por diferentes sistemas informatizados. O software é um não-objeto, uma não-coisa, que pelos seus atributos atende necessidades humanas de qualquer espécie, que podem ser de natureza individual ou coletiva (ROSELINO, 2006).

Associar *software* a programas de computador cria uma visão muito restritiva. Pois, o software não é apenas um programa, trata-se também de dados de documentação e configuração, que juntos, são necessários ao funcionamento correto de um programa (SOMMERVILLE, 2007). Abrangem programas que executam em computadores de qualquer espécie, dados combinados de textos e números, além de diferentes representações da informação (PRESSMAN, 2002).

O software é importante porque, nos dias de hoje, afeta praticamente todos os aspectos de nossas vidas, pois, está difundido nos comércios, cultura e atividades do nosso dia a dia (PRESSMAN, 2002).

Distinto de outros produtos intensivos em informações e conhecimento, como os produzidos pela indústria gráfica, o software é um bem

funcionante, que interage e, em certos casos, comanda a operação de bens materiais, como ferramentas pessoais de trabalho (computadores) e bens intermediários (máquinas industriais) (ROSELINO, 2006).

## 2.2 Produtos de Software

Produtos de Software são definidos pela norma ISO/IEC 9126-1 como um conjunto de programas de computador, procedimentos e possível documentação e dados associados. Estes produtos incluem produtos intermediários e produtos destinados a usuários, desenvolvedores ou mantenedores.

Segundo Pressman (2002), na visão do engenheiro de software, o produto é o conjunto de programas, documentos e dados que compõem um software de computador, enquanto na visão do usuário é a informação resultante, que torna melhor o mundo deste usuário.

Para Sommerville (2007), existem dois tipos fundamentais de produtos de software:

- a. *Os genéricos*: são aqueles produzidos por uma empresa de desenvolvimento e vendidos no mercado para clientes dispostos a adquiri-lo, como os softwares para computador pessoal (PCs), pacotes gráficos, ferramentas de gerenciamento de projetos, entre outros.
- b. *Os personalizados*: são aqueles encomendados por um cliente a uma empresa de software que desenvolverá o sistema especialmente para ele. São exemplos sistemas de

apoio a processos de negócios ou de controle de dispositivos eletrônicos, etc.

A diferença entre esses produtos de software consiste no controle da especificação dos mesmos. No primeiro a empresa desenvolvedora do software controla a especificação, enquanto no segundo geralmente a especificação é desenvolvida e controlada pela organização que adquiriu o produto.

Os produtos de software também possuem atributos associados que demonstram a qualidade do software. Esses atributos estão diretamente relacionados com o comportamento do software durante sua execução, com a organização do programa-fonte, além da documentação do mesmo. O conjunto de atributos esperados de um sistema de software depende da aplicação deste. Pode-se, no entanto, generalizar um conjunto de atributos considerados essenciais a um sistema bem projetado.

Tabela 1 - Atributos essenciais de um bom software  
Fonte: Engenharia de Software (SOMMERVILLE, 2007)

<b>Características do Produto</b>	<b>Descrição</b>
<b>Facilidade de manutenção</b>	O software deve ser escrito de modo que possa evoluir para atender às necessidades de mudança dos clientes. É um atributo fundamental, pois a mudança de software é uma consequência inevitável de um ambiente de negócios em constante mutação.
<b>Confiança</b>	O nível de confiança do software tem uma série de características, incluindo confiabilidade, proteção e segurança. Um software confiável não deve causar danos físicos ou econômicos no caso de falha no sistema.
<b>Eficiência</b>	O software não deve desperdiçar os recursos do sistema, como memória e ciclos do processador. Portanto, a eficiência inclui tempo de resposta, tempo de processamento,

	utilização de memória, etc.
<b>Usabilidade</b>	O software deve ser usável, sem esforço excessivo, pelo tipo de usuário para o qual ele foi projetado. Isso significa que ele deve apresentar uma interface com o usuário e documentação adequadas.

Um dos objetivos da engenharia de software é utilizar técnicas adequadas para aperfeiçoar os produtos de software, sejam eles do desenvolvimento ou da manutenção (especificação de requisitos, documentos do projeto, código, dados de teste, documentação, manuais dos usuários), para que os produtos sejam mais fáceis de utilizar, livres de defeitos e mais efetivos em suas funções (PFLEEGER, 2004). O produto representa o alvo fundamental dos padrões de qualidade.

## 2.3 Processos de Software

Todo projeto de software passa por fases identificáveis antes de se obter um software usável. Existem diversas maneiras de progredir entre essas fases, cada uma dessas maneiras é chamada de *processo de software* (BRAUDE, 2005).

Segundo Pressman (2002), quando elaboramos um produto de software devemos percorrer uma série de passos que ajudem a criar em tempo hábil resultados de alta qualidade. O processo é um diálogo em que o conhecimento deve se transformar em software, prevendo a interação entre usuários e projetistas, entre usuários e ferramentas em desenvolvimento e entre projetistas e ferramentas em desenvolvimento. Cada processo define a abordagem que é adotada quando o software é desenvolvido, e a ferramenta funciona como o meio de comunicação entre cada novo processo.

Segundo Sommerville (2007) todo processo envolve um conjunto de ferramentas e técnicas que o viabilizem, além de possuir as seguintes características:

- a. Prescreve todas as suas principais atividades.
- b. Utiliza recursos, está sujeito a um conjunto de restrições e gera produtos intermediários e finais.
- c. Pode ser composto de sub-processos relacionados. Pode ser definido como uma hierarquia de processos, organizados de maneira que cada sub-processo tenha seu próprio modelo de processo.
- d. Cada atividade do processo possui critérios de entrada e saída, possibilitando saber quando um processo começa e termina.
- e. As atividades são organizadas em seqüência, para que a ordem de execução das atividades seja clara.
- f. Possui um conjunto de diretrizes que explicam os objetivos de cada atividade.
- g. Restrições e controles podem ser aplicados a atividades, recursos ou produtos.

A estrutura do processo orienta nossas ações, permitindo: examinar, entender, controlar e aprimorar as atividades que o compõem. Um processo é um conjunto de procedimentos organizados que permitam a construção de produtos que satisfaçam a uma série de objetivos e padrões. É importante porque imprime consistência e estrutura a um conjunto de atividades a serem

realizadas, além de permitir capturar experiências e passá-las adiante, através de processos e procedimentos documentados (PFLEEGER, 2004).

Processos de software são complexos e dependem de julgamento humano (SOMMERVILLE, 2007). Podem ser caracterizados em uma estrutura comum, que define um pequeno número de atividades aplicáveis a todos os projetos de software, independentemente de tamanho ou complexidade, onde um conjunto de tarefas constituídas de marcos de projeto, produtos do trabalho e pontos de garantia de qualidade permitem que as atividades da estrutura sejam adaptáveis as necessidades do projeto e da equipe de desenvolvimento do mesmo (PRESSMAN, 2002).

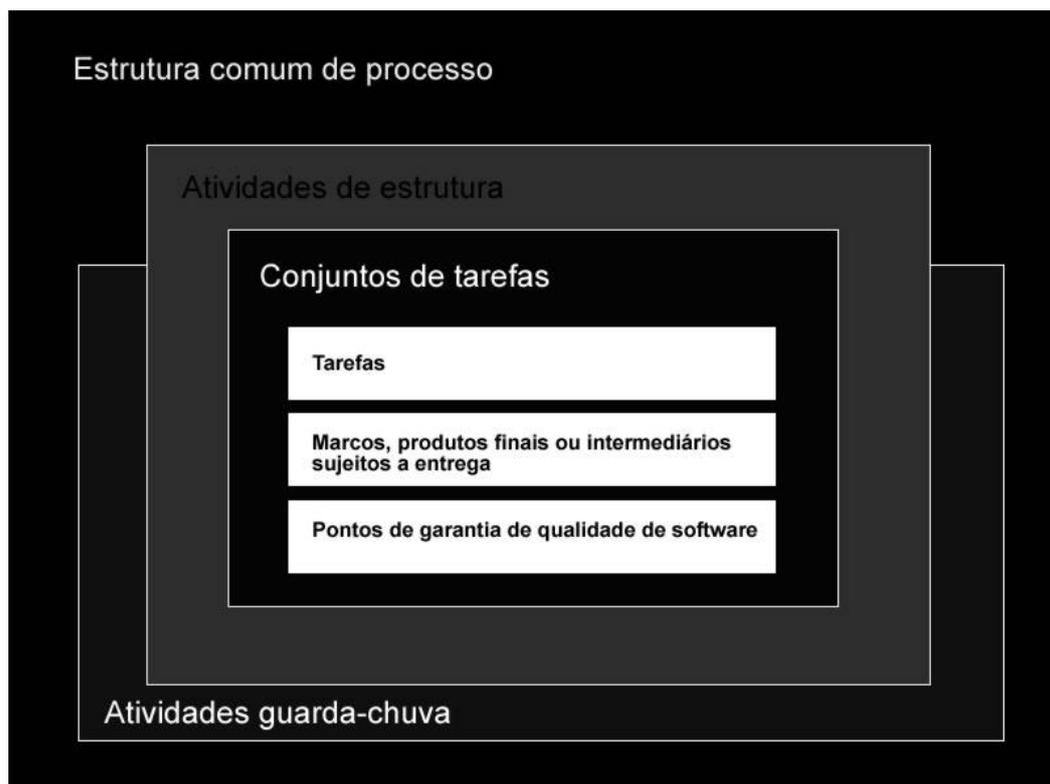


Figura 2 - O processo de software  
Fonte: Engenharia de Software, (PRESSMAN, 2002)

Não existe um processo de software ideal, por isso várias organizações desenvolveram diferentes abordagens para o desenvolvimento

de software. Porém, espaços para aprimoramentos destes processos de software nas empresas de desenvolvimento existem. Esses aprimoramentos acontecem por meio da padronização do processo, que permite que a diversidade de processos de software da organização seja reduzida. A padronização é um passo importante na introdução de novos métodos, técnicas e boas práticas de engenharia de software.

Existem diferentes processos de software, mas, algumas atividades são fundamentais a todos eles, segundo Sommerville (2007):

- a. *Especificação de software*: aqui a funcionalidade do software e suas restrições de operações devem ser definidas.
- b. *Projeto e implementação de software*: é produzido o software que atende às especificações.
- c. *Validação de software*: para garantir que o software faz o que o cliente necessita, o software deve ser validado.
- d. *Evolução do software*: as necessidades dos clientes mudam, para atendê-las o software deve evoluir.

Existem diversas atividades que afetam diretamente a qualidade final de produtos de software, as quais se não forem realizadas de forma adequada podem afetar a qualidade, de maneira que, a qualidade do processo de desenvolvimento e manutenção se torna mais importante que a qualidade do produto (PFLEEGER, 2004).

## 2.4 Melhores Práticas de Software

As melhores práticas de desenvolvimento de software são abordagens utilizadas para o desenvolvimento de software, que utilizadas em conjunto, atacam as origens dos principais problemas de desenvolvimento de software. Segundo Kruchten (2003) são elas:

- a. *Desenvolver o software iterativamente*: O desenvolvimento acontece linearmente, desde a análise de requisitos, passando pela construção, teste de código e unidade, teste de subsistema e teste do sistema.
- b. *Gerenciar requisitos*: É um processo dinâmico e contínuo que abrange três atividades: dedução, organização e documentação das funcionalidades e embaraces do sistema; avaliação de mudanças dos requisitos e a avaliação de seu impacto; localização e documentação de comercializações e decisões.
- c. *Usar arquiteturas baseadas em componentes*: Envolve a evolução contínua da arquitetura de um sistema, onde, cada iteração produz uma arquitetura executável que pode ser medida, testada e avaliada contra os requisitos do sistema. Permite que a equipe de desenvolvimento ataque, de forma contínua, os riscos mais importantes para o projeto de software.
- d. *Modelar visualmente o software*: Ajuda a equipe de desenvolvimento a visualizar, especificar, construir e documentar a estrutura e comportamento da arquitetura de

um sistema. Esta prática, utilizada em conjunto com o desenvolvimento iterativo, ajuda a expor e avaliar as mudanças de arquitetura e a comunicar essas mudanças para a equipe.

- e. *Verificar continuamente a qualidade do software*: envolve criar testes para avaliar a funcionalidade de um sistema e sua conformidade para com os requisitos exigidos.
- f. *Controlar mudanças de software*: Permite melhor alocação de recursos, com base nas prioridades e riscos do projeto, além de gerenciar o trabalho ativamente nessas mudanças. Permite monitorar continuamente as mudanças, de maneira que se possa descobrir ativamente e depois reagir aos problemas. É importante para avaliar e gerenciar o impacto das mudanças.

## 2.5 Modelos de Processo de Software

Um processo de desenvolvimento de software pode ser chamado de ciclo de vida do software, porque esboça a vida do produto do desenvolvimento desde a concepção até a implementação, entrega, utilização e manutenção. Visando, portanto, um maior controle gerencial do desenvolvimento do projeto. Cada um desses processos pode ser descrito em diversas maneiras: texto, figuras ou uma combinação destes recursos organizada como um modelo que contém as principais características do processo (PFLEEGER, 2004).

O ciclo de vida de software define um conjunto de fases, que geram produtos de trabalho necessários para o desenvolvimento de fases futuras de

um projeto. A organização em fases possibilita o planejamento do projeto, incluindo marcos importantes para controle e revisões. As fases procuram representar, de forma abstrata, a estrutura do processo, que pode ser chamado de modelo de ciclo de vida. De maneira geral, um modelo de ciclo de vida descreve a estrutura de organização de atividades do processo em fases e define como essas fases estão relacionadas (SOFTEX, 2009d).

Um modelo de ciclo vida de software é uma representação abstrata dos objetos e atividades envolvidas no processo de software, oferecendo uma forma mais abrangente e fácil de representar o gerenciamento de processo de software e, por conseguinte o progresso do projeto (SOMMERVILLE, 2007). Ele possibilita períodos planejados de avaliação e de tomada de decisões, em que compromissos significativos são realizados com relação aos recursos, abordagem técnica, reavaliação de escopo e custo do projeto (SOFTEX, 2009d). Um modelo é escolhido com base na natureza do projeto, dos métodos e ferramentas a serem utilizados, dos controles e dos produtos intermediários e finais que serão demandados (PRESSMAN, 2002).

Com este cenário, é importante conhecer diferentes modelos de ciclo de vida e em que situações eles são aplicáveis. Os principais modelos de ciclo de vida podem ser agrupados em três categorias principais: modelos seqüenciais ou cascata, modelos incrementais e modelos evolutivos (SOFTEX 2009d *apud* ISO/IEC 15271:1998). Entretanto, outros tipos de modelos de ciclo de vida têm sido cada vez mais adotados pelas organizações, como o Processo Unificado Racional (RUP - *Rational Unified Process*) e suas variações (KRUCHTEN, 2003).

Uma organização pode preestabelecer que todos os projetos utilizem o mesmo modelo de ciclo de vida, pode, ainda, predefinir que mais de um modelo de ciclo de vida será utilizado na organização, de maneira que, para cada projeto, será selecionado o modelo que melhor atender às suas características (SOFTEX 2009d *apud* ISO/IEC 15271:1998).

## 3 NBR ISO/IEC 9126-1 – UM MODELO DE QUALIDADE DE SOFTWARE

Desenvolver produtos de software de alta qualidade é importante, especificação e avaliação da qualidade de produtos de software são fatores cruciais para a garantia dessa qualidade desejada. Pode-se alcançar isto, definindo-se apropriadamente as características de qualidade, através de métricas válidas e amplamente aceitas de especificação e avaliação.

A NBR ISO/IEC 9126-1<sup>4</sup> é um modelo de qualidade de software composto por duas partes: qualidade interna e externa, e qualidade em uso. Estas partes possuem características e subcaracterísticas que são aplicáveis a todo tipo de sistema e fornecem terminologia consistente para tratar de qualidade de produto de software além de estrutura para especificação de requisitos de software e realização de comparações entre produtos de software.

### 3.1 Qualidade externa e interna

Esta primeira parte da norma especifica seis características para qualidade interna<sup>5</sup> e externa<sup>6</sup> que são subdivididas em subcategorias (resultantes de atributos internos do software) que se manifestam quando o produto final de software é utilizado.

---

<sup>4</sup> Primeira parte, de quatro, da NBR ISO/IEC 9126 que está sob o título geral de Engenharia de Software – Qualidade do Produto.

<sup>5</sup> É a totalidade das características do produto de software do ponto de vista interno. Pode ser medida e avaliada de acordo com os requisitos internos.

<sup>6</sup> É a totalidade das características do produto de software do ponto de vista externo. É a qualidade quando o software é executado, que é medido e avaliado durante sua execução em uma simulação.

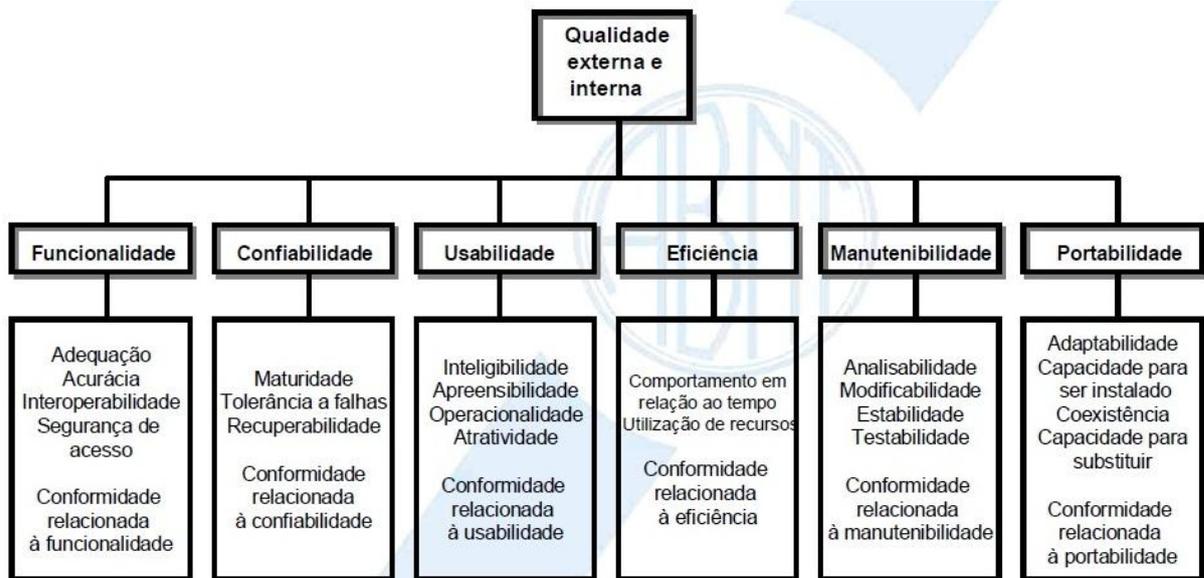


Figura 3 - Modelo de qualidade para qualidade externa e interna

Fonte: Engenharia de Software – Modelo de Qualidade, (NBR ISO/IEC 9126-1: 2003)

### 3.1.1 Funcionalidade

Trata da capacidade do produto de software fornecer funções que atendam as necessidades explícitas e implícitas, quando o software estiver sendo executado em um ambiente simulado. Está relacionada com o que o software faz para atender os requisitos que o sistema propõe. Suas subcategorias são:

- a. *Adequação*: o produto de software deve fornecer um conjunto de funções que atendam as necessidades do usuário especificadas.
- b. *Acurácia*: o produto de software deve fornecer, com grau de precisão necessário, os resultados acordados.
- c. *Interoperabilidade*: o produto de software deve ser capaz de interagir com um ou mais sistemas especificados.
- d. *Segurança de acesso*: o produto de software deve proteger as informações e dados de pessoas ou sistemas não

autorizados e deve também garantir que pessoas e sistemas autorizados não tenham seu acesso negado.

- e. *Conformidade relacionada à funcionalidade*: o produto de software deve estar de acordo com normas, regulamentações previstas em leis e preceitos relacionados à funcionalidade.

### 3.1.2 Confiabilidade

Trata da capacidade do produto de software manter o seu nível de desempenho esperado, quando usado em condições especificadas. Suas subcaracterísticas são:

- a. *Maturidade*: o produto de software deve ser capaz de evitar falhas decorrentes de defeitos no software.
- b. *Tolerância a falhas*: o produto de software deve ser capaz de manter seu nível de desempenho esperado em casos de defeitos no software e de violação de sua interface especificada.
- c. *Recuperabilidade*: o produto de software deve ser capaz de restabelecer seu nível de desempenho esperado e recuperar os dados que foram diretamente atingidos por uma falha.
- d. *Conformidade relacionada à confiabilidade*: o produto de software deve estar de acordo com normas e regulamentações relacionadas à confiabilidade.

### 3.1.3 Usabilidade

Trata da capacidade do produto de software em ser compreendido, aprendido, operado e atraente para o usuário quando utilizado nas condições especificadas. Suas subcaracterísticas são:

- a. *Inteligibilidade*: o produto de software deve ser capaz de possibilitar ao usuário a compreensão de que o software é apropriado e de que ele pode ser utilizado para as tarefas e condições que foram especificadas no projeto.
- b. *Apreensibilidade*: o produto de software deve ser capaz de possibilitar que o usuário aprenda sua aplicação.
- c. *Operacionalidade*: o produto de software deve ser capaz de permitir que o usuário o opere e controle.
- d. *Atratividade*: o produto de software deve ser atraente para o usuário.
- e. *Conformidade relacionada à usabilidade*: o produto de software deve estar de acordo com normas, e guias de estilo relacionados à usabilidade.

### 3.1.4 Eficiência

Trata da capacidade do produto de software apresentar desempenho apropriado mediante a quantidade de recursos usados sob as condições especificadas. Suas subcaracterísticas são:

- a. *Comportamento em relação ao tempo*: o produto de software deve apresentar tempos de resposta e de processamento, e

taxas de transferência, adequados, durante a execução das funções do software.

- b. *Utilização de recursos*: o produto de software deve ser capaz de utilizar tipos e quantidades de recursos apropriados durante a execução das funções do software.
- c. *Conformidade relacionada à eficácia*: o produto de software deve estar de acordo com normas e convenções relacionadas à eficiência.

### 3.1.5 Manutenibilidade

Trata da capacidade do produto de software em ser modificado, de maneira que, as modificações que ocorram, possam incluir correções, melhorias ou adaptações do software em decorrência de mudanças nos seus requisitos e especificações funcionais. Suas subcaracterísticas são:

- a. *Analisabilidade*: o produto de software deve ser capaz de possibilitar a identificação de deficiências, das causas de falhas no software e das partes que tem que ser modificadas.
- b. *Modificabilidade*: o produto de software deve ser capaz de permitir que uma modificação especificada seja implementada.
- c. *Estabilidade*: o produto de software deve evitar resultados inesperados decorrentes de modificações no software.
- d. *Testabilidade*: o produto de software deve permitir que quando modificado, o software seja validado.

- e. *Conformidade relacionada à manutenibilidade*: o produto de software deve estar em conformidade com normas e convenções relacionadas à manutenibilidade.

### 3.1.6 Portabilidade

Trata da capacidade do produto de software em ser transferido de um ambiente organizacional, de hardware ou de software, para outro. Suas subcaracterísticas são:

- a. *Adaptabilidade*: o produto de software deve ser capaz de ser adaptado em diferentes ambientes especificados, sem que haja a necessidade de aplicação de ações ou meios além dos fornecidos para essa finalidade pelo software em questão.
- b. *Capacidade de ser instalado*: o produto de software deve ser capaz de ser instalado no ambiente especificado no projeto.
- c. *Coexistência*: o produto de software deve ser capaz de coexistir com outros produtos de softwares independentes, com ambiente e recursos comuns.
- d. *Capacidade para substituir*: o produto de software deve ser capaz de ser utilizado como substituto de outro produto de software especificado, possuindo o mesmo propósito no mesmo ambiente.
- e. *Conformidade relacionada à portabilidade*: o produto de software deve estar de acordo com normas e convenções relacionadas à portabilidade.

## 3.2 Qualidade em uso

É a visão da qualidade do produto de software na ótica do usuário. A qualidade em uso é obtida com a obtenção da qualidade externa, a qual depende da obtenção da qualidade interna. Trata da capacidade do produto de software em permitir que os usuários atinjam as metas especificadas mediante quatro características, discutidas a seguir. Essas características podem ser medidas quando o produto de software é utilizado em um ambiente e contexto de uso específico.

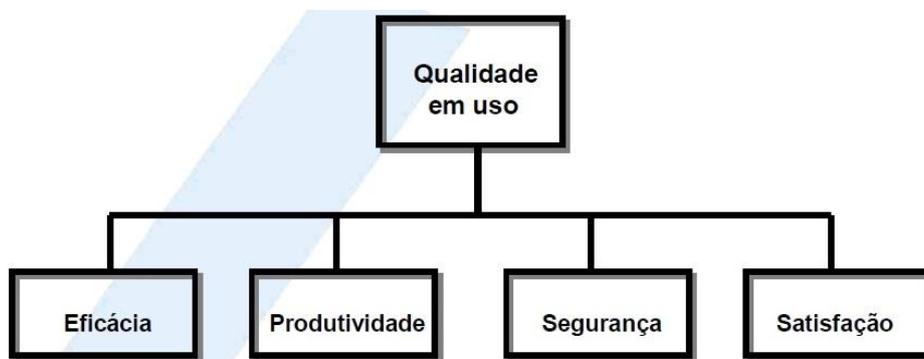


Figura 4 - Modelo de qualidade para qualidade em uso

Fonte: Engenharia de Software – Modelo de Qualidade, (NBR ISO/IEC 9126-1: 2003)

### 3.2.1 Eficácia

O produto de software deve permitir que os usuários atinjam metas especificadas com acurácia e completude, nos contextos de uso especificados no projeto.

### 3.2.2 Produtividade

O produto de software deve ter a capacidade de possibilitar que os seus usuários apliquem quantidade apropriada de recursos em relação à eficácia obtida. Recursos relevantes podem incluir tempo para completar a tarefa, esforço do usuário, materiais e custos financeiros.

### **3.2.3 Segurança**

É a capacidade do produto de software apresentar níveis aceitáveis de riscos de danos as pessoas, negócios, software, propriedades ou ao ambiente especificado. Em geral esses riscos são decorrentes de falhas na funcionalidade, confiabilidade, usabilidade ou manutenibilidade do software.

### **3.2.4 Satisfação**

É a resposta do usuário à interação com o produto de software incluindo atitudes relacionadas ao uso do produto.

## 4 GARANTIA DE QUALIDADE DE SOFTWARE

O termo qualidade de software apareceu com a necessidade de organizar e padronizar o desenvolvimento de softwares que não tinham planejamento, nem norma de qualidade aplicável (SODRÉ, 2006). Diante disto, nos últimos anos, a preocupação das empresas com a qualidade de seus produtos tornou-se uma estratégia de mercado importante, uma vez que, garantir qualidade é uma atividade essencial para qualquer tipo de negócio que ofereça produtos que serão utilizados por outras pessoas (PRESSMAN, 2002).

Ainda segundo Pressman (2002), existem dois tipos de qualidade de software:

- a. *A qualidade de projeto:* Abrange os requisitos, as especificações e o projeto de software. A qualidade do software aumenta se ele é produzido de acordo com as especificações.
- b. *A qualidade da conformidade:* Abrange principalmente a implementação do sistema. Se ela está de acordo com o projeto e o sistema satisfaz os requisitos e metas de desempenho, a qualidade da conformidade é alta.

### 4.1 SQA

A garantia de qualidade de software (*Software Quality Assurance*, SQA) consiste nas funções de auditar e relatar (PRESSMAN, 2002). É um processo que estabelece um framework de procedimentos organizacionais e padrões, que devem ser aplicados ao processo de desenvolvimento de

software ou ao produto de software, produzindo assim um software de alta qualidade. Existem dois tipos de padrões que podem ser estabelecidos como parte do processo de garantia de qualidade, segundo Sommerville (2007):

- a. *Padrões de produto*: São aplicados ao produto de software em desenvolvimento e incluem padrões de documentos, de documentação e de codificação. Está ligado à saída do processo de software.
- b. *Padrões de processo*: Definem quais processos devem ser seguidos durante o desenvolvimento de software. Incluem definições de processos de especificação, projeto e validação, e uma descrição dos documentos que devem ser produzidos ao longo destes processos assegurando que os padrões de produto sejam seguidos.

A garantia de qualidade é serviço de todos os envolvidos em um projeto, ela é obtida através da análise, projeto, codificação, testes, revisões técnicas formais, controle de produtos do desenvolvimento, modificações, além da aplicação de padrões de engenharia de software. A SQA tem por objetivo disponibilizar para a gerência de qualidade, os dados necessários para que ela tome ciência sobre a qualidade do software que está sendo desenvolvido, demonstrando se a qualidade está satisfazendo as metas do projeto (PRESSMAN, 2002).

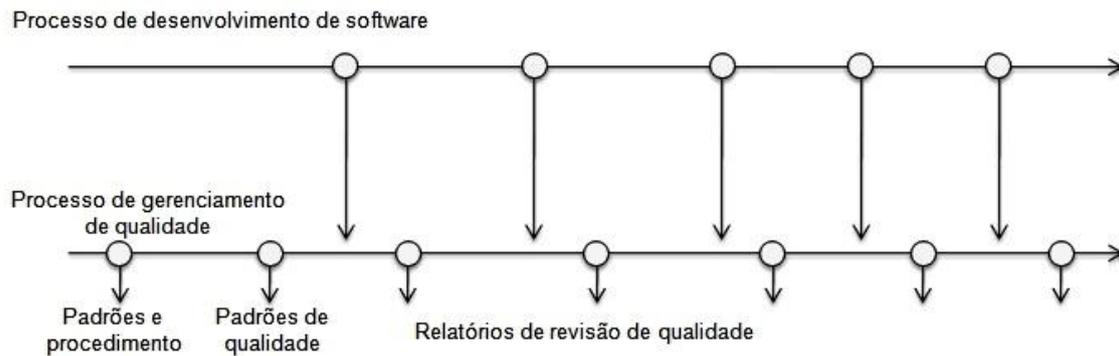


Figura 5 - Gerenciamento de qualidade e desenvolvimento de software  
Fonte: Engenharia de Software (SOMMERVILLE, 2007)

A equipe de qualidade durante o processo de gerenciamento da qualidade verifica todas as saídas do projeto, as falhas e erros encontrados devem ser reportados ao gerente de qualidade, para garantir que as saídas estejam de acordo com os padrões e objetivos definidos pela organização (SOMMERVILLE, 2007).

O gerente de qualidade, com base nos dados gerados pela equipe de gerência de qualidade, tem por obrigação monitorar o progresso de todas as atividades de qualidade, incluindo resultados, assim como cronograma para identificar desvios do plano de qualidade para que eles recebam ações corretivas o mais cedo possível (PEZZÉ; YOUNG, 2008).

## 4.2 Atividades de SQA

A garantia de qualidade de software é composta de variadas atividades que estão associadas a dois grupos diferentes, a equipe de engenharia de software e a equipe de garantia de qualidade. A equipe de engenharia busca a qualidade através da aplicação de métodos e medidas técnicas sólidas, de revisões técnicas formais e de testes de software bem planejados. A equipe de qualidade ajuda a equipe de engenharia a conseguir um produto de software de alta qualidade planejando, supervisionando,

registrando, analisando e relatando a garantia de qualidade, ela também coordena o controle e gestão de mudanças e coleta e analisa métricas de software (PRESSMAN, 2002).

O *Software Engineering Institute* (Instituto de Engenharia de Software) apresenta as atividades de garantia de qualidade de software (PRESSMAN 2002 apud PAULK 1993):

- a. *Preparo de um plano de garantia de qualidade:* O plano de garantia é desenvolvido e revisado, ele identifica as avaliações a serem realizadas, as auditorias e revisões a serem aplicadas, os padrões aplicáveis ao projeto, os procedimentos de relato e acompanhamento de erros, os documentos que devem ser produzidos pela equipe de SQA e a quantidade de realimentação fornecida à equipe de engenharia de software.
- b. *Descrição do processo de software do projeto:* A equipe de software escolhe um processo de desenvolvimento para o projeto e a equipe de garantia de qualidade revisa a descrição do processo para verificar se o processo escolhido atende os padrões organizacionais, internos de software, externamente impostos e outras partes do plano de software.
- c. *Revisão das atividades de desenvolvimento de software:* A equipe de SQA identifica, documenta e acompanha os desvios e correções do processo de desenvolvimento, garantindo que ele satisfaça os objetivos do projeto.

- d. *Auditoria dos produtos de software*: A equipe de garantia verifica os produtos do desenvolvimento, identifica, documenta e acompanha se as correções dos desvios foram feitas, e relata os resultados da verificação ao gerente do projeto.
- e. *Documentação dos desvios*: Garante que os desvios do desenvolvimento do software e dos produtos foram documentados e manipulados de acordo com um procedimento de documentação, de maneira que, os desvios possam ser identificados no plano de projeto, na descrição do processo, nos padrões aplicados ou nos produtos gerados.
- f. *Relatos SQA*: Registra qualquer processo que não tenha satisfeito os objetivos do projeto e relata a gerência. Os itens que não atenderem o padrão desejado serão acompanhados até que sejam corrigidos.

### **4.3 Plano de SQA**

O planejamento é parte integrante do processo de qualidade e é elaborado e revisado ao longo de todo o processo de desenvolvimento, ele envolve e interliga diversos assuntos como: cronograma, custos, usabilidade, dependências, entre outros. Um processo de qualidade elaborado corretamente equilibra diversas atividades durante o desenvolvimento do projeto (PEZZÉ; YOUNG, 2008).

O plano de qualidade estabelece as qualidades de software desejadas e descreve como elas devem ser avaliadas, define o que significa

*alta qualidade* de software, seleciona os padrões organizacionais e o processo de desenvolvimento apropriado para o produto a ser desenvolvido. Os planos de qualidade diferem quanto ao tamanho e o tipo de sistema que será produzido, eles admitem uma série de atributos de qualidade de software, em geral os mais importantes, que devem ser considerados durante o planejamento de qualidade (SOMMERVILLE, 2007).

Segurança	Facilidade de compreensão	Portabilidade
Proteção	Facilidade de testes	Facilidade de uso
Confiabilidade	Adaptabilidade	Facilidade de reuso
Facilidade de recuperação	Modularidade	Eficiência
Robustez	Complexidade	Facilidade de aprendizado

Figura 6 - Atributos de qualidade de software  
Fonte: Engenharia de Software, (SOMMERVILLE, 2007)

O plano deve ainda estabelecer papéis e responsabilidades as pessoas em nível estratégico e tático. O nível estratégico é representado nos documentos de qualidade e na estrutura organizacional. O nível tático é representado no plano de projeto e atribui responsabilidades para os envolvidos de acordo com a estratégia geral, este nível envolve ainda, o balanceamento do nível de esforço ao longo do processo de desenvolvimento de software e o gerenciamento das interações entre os indivíduos do projeto (PEZZÉ; YOUNG, 2008).

O plano de SQA é um roteiro para que a garantia de qualidade de software aconteça, serve ainda, como um gabarito para as atividades de qualidade definidas para cada projeto (PRESSMAN, 2002).

Possui norma definida pelo IEEE (1994) que descreve todas as seções desejáveis de um plano, dentre elas:

- a. *Seção inicial*: Descreve a finalidade e o escopo do plano de SQA, também indica as atividades do processo de software que serão cobertas pela garantia de qualidade, lista todos os documentos citados no plano e menciona todas as normas aplicáveis.
- b. *Seção gerencial*: Descreve o posicionamento da garantia de qualidade de software na estrutura da organização, as tarefas, atividades e posição de SQA durante o desenvolvimento do projeto, os papéis e responsabilidades organizacionais quanto à qualidade do produto do desenvolvimento.
- c. *Seção de documentação*: Descreve todos os produtos<sup>7</sup> de trabalho como parte do processo de desenvolvimento e define o conjunto de produtos de trabalho aceitável para a obtenção da qualidade de software.
- d. *Seção de padrões, práticas e convenções*: Lista todas as normas e práticas aplicáveis durante o processo de desenvolvimento de software, e métricas de projeto, processo e produto, que tem que ser coletadas como parte do trabalho de desenvolvimento do projeto.
- e. *Seção de revisões e auditoria*: Identifica as revisões e auditorias que devem ser feitas pelas equipes de engenharia de software e de SQA e pelo cliente.

---

<sup>7</sup> Documentos de projeto, modelos, documentos técnicos, manuais, entre outros.

- f. *Seção de teste*: Descreve o plano e procedimento de teste de software, define os requisitos para a manutenção dos registros dos testes e relata problemas e suas ações corretivas.
- g. *Demais seções*: Identificam as ferramentas e métodos que apoiarão as atividades de garantia de qualidade de software; descreve procedimentos de gestão de configuração de software para controle de modificações; define abordagens para gerência de contrato e de gerência de riscos; estabelece métodos de documentação dos registros.

As metas da qualidade são alcançadas mediante esse cuidadoso planejamento das atividades de qualidade, que estão diretamente relacionadas com os objetivos identificados para o projeto (PEZZÉ; YOUNG, 2008).

## 5 MELHORIA DE PROCESSOS DE SOFTWARE

Produzir software de qualidade é uma meta fundamental para a Engenharia de Software, que disponibiliza métodos, técnicas e ferramentas para este fim. No entanto, a qualidade não deve estar presente somente nos produtos de software produzidos, mas também nos processos utilizados para desenvolver esses produtos, haja vista, que a qualidade do processo de desenvolvimento atinge a qualidade dos produtos diretamente (ROCHA *et. al.*, 2001).

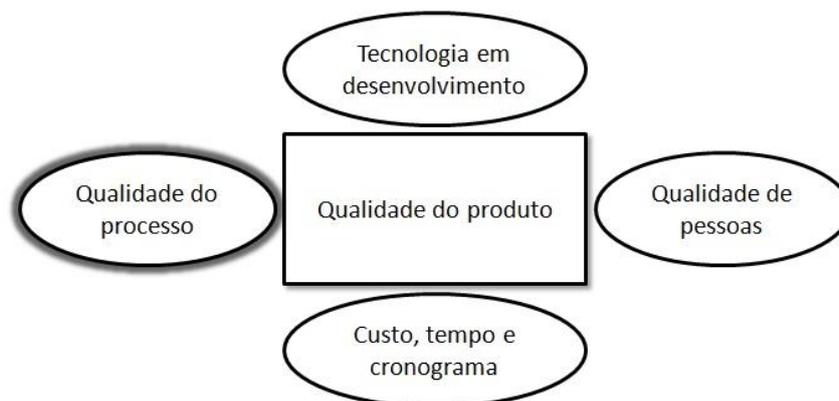


Figura 7 - Fatores principais de qualidade de produtos de software  
Fonte: Engenharia de Software (SOMMERVILLE, 2007)

A maioria das falhas que ocorrem em produtos de software tem sua origem nas falhas do processo de desenvolvimento. A ocorrência dessas falhas pode ser minimizada através da modificação dos processos ou ambiente de desenvolvimento. Essas modificações no processo de desenvolvimento, em geral, podem acarretar em melhorias no produto e reduções significativas de custo (PEZZÉ; YOUNG, 2008).

Processos de desenvolvimento de software podem possuir técnicas antiquadas ou simplesmente não aproveitar de maneira eficiente as melhores

práticas da engenharia de software. Muitas empresas ainda não se beneficiam de métodos eficazes em seus processos de desenvolvimento de software, mas, algumas têm se concentrado em aprimorar seus processos de software, como maneira de aumentar a qualidade de seus produtos (SOMMERVILLE, 2007).

A norma ISO/IEC 12207 (IEEE, 2008) define o processo de melhoria como um processo para estabelecer, avaliar, medir, controlar e melhorar um processo de software.

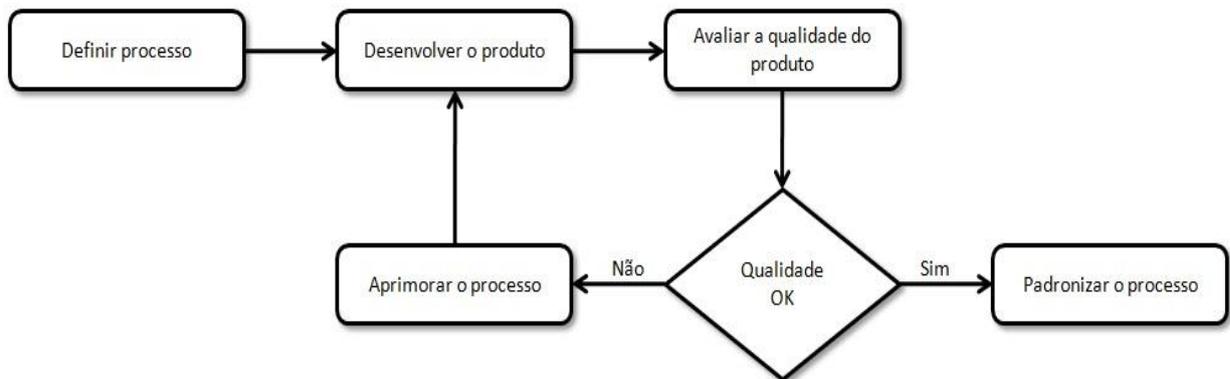


Figura 8 - Qualidade baseada em processo  
Fonte: Engenharia de software (SOMMERVILLE, 2007)

Os objetivos principais da melhoria de processos de uma organização são (SOFTEX 2009 *apud* FLORAC; CARLETON, 1999):

- a. Entender as características dos processos existentes e os fatores que afetam a capacidade do processo.
- b. Planejar e implementar ações que modifiquem o processo para atender melhor as necessidades de negócio.
- c. Avaliar os impactos e benefícios obtidos e compará-los com os custos das mudanças realizadas nos processos.

A organização que deseja aprimorar a maturidade dos seus processos de software deve adotar modelos de melhoria de processo de software, que permitam que ela melhore seu desempenho produtivo e agregue valor aos seus produtos de desenvolvimento (SANTOS; CONTE 2010 *apud* SANDHOF, 2004).

Modelos de maturidade<sup>8</sup> e capacidade<sup>9</sup> de processo de software enfocam a melhoria dos processos de desenvolvimento, e contribuem diretamente para a melhoria dos produtos produzidos e do software projetado. Muitos modelos de melhoria de processos já foram propostos, e esses modelos e seus métodos de avaliação estão se tornando padrão em muitas organizações (PFLEEGER, 2004).

Organizações ainda imaturas, que não perceberam essa necessidade visível de preocupação com os processos de desenvolvimento, têm um grande desafio pela frente quando a meta é garantir a qualidade dos produtos de software que desenvolvem.

## **5.1 MPS-BR - Modelo de Melhoria de Processo de Software Brasileiro**

O MPS-BR (Melhoria de Processo de Software Brasileiro) é um modelo para avaliação e melhoria de processos de software, que tem como base técnica as normas ISO/IEC 12207<sup>10</sup> e suas emendas 1 e 2, e a ISO/IEC

---

<sup>8</sup> A amplitude na qual um processo específico é definido, gerenciado, medido, controlado e executado.

<sup>9</sup> O intervalo de resultados esperados que possam ser alcançados com o uso de um processo.

<sup>10</sup> Esta norma diz respeito ao ciclo de vida de software.

15504<sup>11</sup>, o modelo MPS também cobre todo o conteúdo do modelo CMMI<sup>12</sup> (MORO, 2008). Descrito através de guias (geral, aquisição, avaliação e implementação), o modelo MPS baseia-se nos conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de produtos de software e serviços correlatos. Dentro desse contexto, o modelo MPS possui três componentes (SOFTEX, 2009a):

- a. *Modelo de Referência (MR-MPS)*: contém os requisitos a serem atendidos pelas organizações, bem como os processos e os níveis de maturidade
- b. *Método de Avaliação (MA-MPS)*: contém o processo de avaliação, os requisitos para os avaliadores e os requisitos para averiguação da conformidade ao MR-MPS.
- c. *Modelo de Negócio (MN-MPS)*: contém as regras para implantação do MPS-BR pelas empresas de consultoria, de software e avaliação.

---

<sup>11</sup> Esta norma diz respeito à avaliação de processos de software.

<sup>12</sup> Modelo de Maturidade e Capacidade Integrado, desenvolvido pelo Instituto de Engenharia de Software (SEI – Software Engineering Institute).

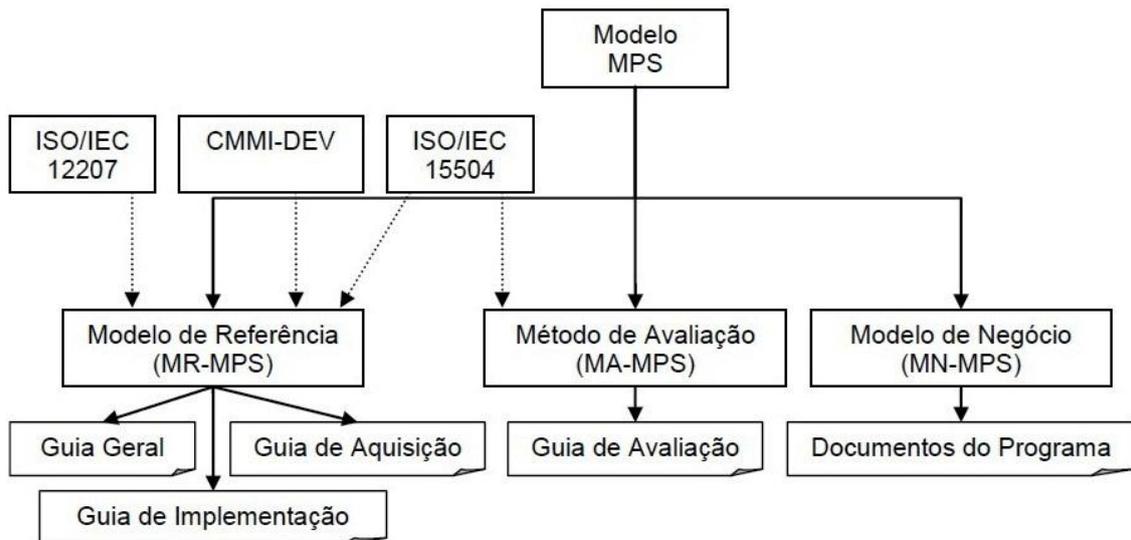


Figura 9 - Estrutura do modelo MPS-BR  
Fonte: MPS-BR - Guia Geral (SOFTEX, 2009a)

O Modelo de Referência MR-MPS define sete níveis de maturidade, que estabelecem patamares de evolução de processo, caracterizando estágios de melhoria de implementação de processos na organização (MORO, 2008). O grau de maturidade é o grau de melhoria de processo para um determinado conjunto de processos no qual todos os resultados esperados do processo e dos atributos dos processos são atendidos (SOFTEX, 2009a). Os níveis de maturidade do MR-MPS se iniciam no nível G e progridem de baixo para cima até o nível A.



Figura 10 - Níveis de maturidade do modelo MR-MPS  
Fonte: Própria

No modelo MR-MPS, os processos são descritos em termos de propósito: o propósito descreve o objetivo geral a ser atingido durante a execução do processo; e resultados: os resultados esperados do processo estabelecem os resultados a serem obtidos com a efetiva implementação do processo, onde, os resultados podem ser evidenciados por um produto de trabalho produzido ou uma mudança significativa de estado ao se executar o processo (MONTONI, 2010).

A capacidade do processo expressa o grau de refinamento e institucionalização com que o processo é executado na organização. No MR-MPS, à medida que a organização evolui nos níveis de maturidade, um maior nível de capacidade para desempenhar o processo deve ser atingido. A capacidade do processo é representada por um conjunto de atributos de

processo (AP) descritos em termos de resultados esperados (RAP<sup>13</sup>) (SOFTEX, 2009d).

Tabela 2 - Descrição dos atributos de processo (AP)  
Fonte: MPS-BR - Guia Geral (SOFTEX, 2009a)

AP	Descrição
AP 1.1	O processo é executado: Este atributo é uma medida do quanto o processo atinge o seu propósito.
AP 2.1	O processo é gerenciado: Este atributo é uma medida do quanto à execução do processo é gerenciada.
AP 2.2	Os produtos de trabalho do processo são gerenciados: Este atributo é uma medida do quanto os produtos de trabalho produzidos pelo processo são gerenciados apropriadamente.
AP 3.1	O processo é definido: Este atributo é uma medida do quanto um processo padrão é mantido para apoiar a implementação do processo definido.
AP 3.2	O processo está implementado: Este atributo é uma medida do quanto o processo padrão é efetivamente implementado como um processo definido para atingir seus resultados.
AP 4.1	O processo é medido: Este atributo é uma medida do quanto os resultados de medição são usados para assegurar que a execução do processo atinge os seus objetivos de desempenho e apóia o alcance dos objetivos de negócio definidos.
AP 4.2	O processo é controlado: Este atributo é uma medida do quanto o processo é controlado estatisticamente para produzir um processo estável, capaz e previsível dentro de limites estabelecidos.
AP 5.1	O processo é objeto de melhorias e inovações: Este atributo é uma medida do quanto as mudanças no processo são identificadas a partir da análise de defeitos, problemas, causas comuns de variação do desempenho e da investigação de enfoques inovadores para a definição e implementação do processo.
AP 5.2	O processo é otimizado continuamente: Este atributo é uma medida do quanto às mudanças na definição, gerência e desempenho do processo têm impacto efetivo para o alcance dos objetivos relevantes de melhoria do processo.

<sup>13</sup> Referente aos resultados esperados de um processo avaliado. Ex.: No nível G são esperados, dos AP 1.1 e AP 2.1, 10 RAPs, dentre eles: RAP 1 - O processo atinge seus resultados definidos; RAP 2 - Existe uma política organizacional estabelecida e mantida para o processo; RAP 3 - A execução do processo é planejada.

Sete níveis de capacidade de processos são descritos pelos nove atributos de processo descritos acima, conforme a tabela a seguir (SOFTEX, 2009a).

Tabela 3 - Níveis MR-MPS, seus processos e atributos de processo  
Fonte: MPS-BR - Guia Geral (SOFTEX, 2009a)

Nível	Processos	Atributos de Processo
A		AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1, AP 4.2 , AP 5.1 e AP 5.2
B	Gerência de Projetos - GPR (evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1 e AP 4.2
C	Gerência de Riscos – GRI	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
	Desenvolvimento para Reutilização - DRU	
	Gerência de Decisões - GDE	
D	Verificação - VER	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
	Validação - VAL	
	Projeto e Construção do Produto – PCP	
	Integração do Produto – ITP	
	Desenvolvimento de Requisitos – DRE	
E	Gerência de Projetos - GPR (evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
	Gerência de Reutilização – GRU	
	Gerência de Recursos Humanos – GRH	
	Definição do Processo Organizacional – DFP	
	Avaliação e Melhoria do Processo Organizacional – AMP	
F	Medição – MED	AP 1.1, AP 2.1 e AP 2.2
	Garantia da Qualidade – GQA	
	Gerência de Portfólio de Projetos – GPP	
	Gerência de Configuração – GCO	
	Aquisição – AQU	
G	Gerência de Requisitos – GRE	AP 1.1 e AP 2.1
	Gerência de Projetos - GPR	

O programa MPS-BR foi criado para a realidade das empresas brasileiras, com foco nas PMEs<sup>14</sup> de software e representa um esforço para agregar valor a qualidade de desenvolvimento de software e as empresas de software, potencializando a capacidade de competitividade nacional e internacional delas.

Para apoiar a adoção do modelo MPS, a SOFTEX organiza grupos, de acordo com o Modelo de Negócio Cooperado (MNC), de organizações interessadas em compartilhar os serviços e custos na implementação e avaliação do modelo MPS. Parte dos custos é subsidiado por órgãos, como o BID (Banco Interamericano de Desenvolvimento), o MCT (Ministério de Ciência e Tecnologia) e o SEBRAE (Serviço de Apoio às Micro e Pequenas Empresas) (SOFTEX, 2009a).

A implementação do MR-MPS de acordo com o MNC para as pequenas e médias empresas é uma alternativa importante para que elas obtenham benefícios com a implementação de melhorias em processos com um custo acessível (SOFTEX, 2009b).

---

<sup>14</sup> Pequenas e médias empresas.

## 6 AVALIAÇÃO DE PROCESSOS DE SOFTWARE

Para Pfleeger (2007), a melhoria de processos é a força motriz para produzir softwares melhores. Porém, algumas vezes existe a necessidade de se demonstrar os objetivos da melhoria, de forma simples, deixando-se claro o ponto em que se está e o ponto aonde se quer chegar. A avaliação de processos é um aspecto-chave nas boas práticas da engenharia de software, fazendo com que as características específicas dos processos e dos produtos do desenvolvimento fiquem mais visíveis.

De acordo com a ISO/IEC 12207, avaliação é a determinação sistemática do grau de conformidade e atendimento de uma entidade com relação aos critérios que foram estabelecidos para ela (IEEE, 2008).

As avaliações de processo de software ocorrem em diferentes contextos e são realizadas para atender a diferentes objetivos. Dependendo das características, dos modelos e métodos aplicados, podem atender a diferentes escopos e pertencer a diferentes paradigmas. Geralmente, uma avaliação de processo é realizada para atender a dois objetivos (ANDRADE, 2005):

- a. A melhoria dos processos;
- b. E a determinação da capacidade dos processos de uma organização.

Para determinar o quanto os processos padrão contribuem para a organização, avaliações disciplinadas dos processos devem ser realizadas através de um modelo de avaliação de processos que permita avaliar a

capacidade dos processos com base em um modelo de referência de processos de software. E é através da análise de resultados obtidos com essas avaliações, que podem ser identificados pontos fortes, pontos fracos e riscos envolvidos em utilizar os processos avaliados em um projeto específico ou no contexto de uma unidade organizacional (SOFTEX 2009d apud ISO/IEC 2004b).

O escopo de uma avaliação do processo de software pode cobrir: todos os processos da organização; um subconjunto selecionado dos processos; ou um projeto específico. Para a maioria das avaliações de processo baseadas nos conceitos de maturidade ou capacidade, a unidade de análise e classificação é normalmente em nível organizacional (ANDRADE 2005 *apud* KAN, 2003). A avaliação de processo baseada em maturidade de processo torna-se relevante quando uma organização tem a intenção de embarcar em uma estratégia geral de melhoria em longo prazo (ANDRADE, 2005).

A partir da análise dos resultados das avaliações, são identificadas informações de melhorias nos processos padrão da organização que poderão ser utilizadas para melhorar os processos padrão por meio da realização de mudanças nas suas capacidades existentes ou por meio da substituição de subprocessos por outros mais eficientes ou efetivos (SOFTEX 2009d *apud* FLORAC; CARLETON, 1999).

Uma vez identificadas às melhorias potenciais a serem implementadas nos processos padrão da organização, estas são analisadas e transformadas em itens de ação, que necessitam ser realistas e alinhados aos

papéis e responsabilidades definidos na organização, (SOFTEX 2009d *apud* ZAHARAN, 1998).

## 7 MÉTODO DE AVALIAÇÃO MPS

O processo e o método de avaliação MA-MPS é um componente do modelo MPS, com base técnica na norma internacional ISO/IEC 15504-2:2003<sup>15</sup>, que permite a avaliação objetiva dos processos de software de uma organização, a atribuição de um nível de maturidade do MR-MPS com base no resultado da avaliação, e que possibilita ainda, ser aplicável a qualquer domínio da indústria de software e a organizações de qualquer tamanho (SOFTEX, 2009c).

O propósito do processo de avaliação MPS é determinar o quanto os processos padrões da organização contribuem para alcançar os objetivos de negócio da organização (maturidade da organização quanto a seus processos de software) e para apoiar a organização a planejar, realizar e implantar melhorias contínuas nos processos com base no entendimento de seus pontos fortes e fracos (SOFTEX, 2009d). O processo de avaliação descreve o conjunto de atividades e tarefas a serem realizadas para atingir este propósito (SOFTEX, 2009c).



Figura 11 - Processo de avaliação MR-MPS  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

<sup>15</sup> Equivale a NBR ISO/IEC 15504-2 sob o seguinte título “Tecnologia da Informação – Avaliação de processo Parte 2: Realização de uma avaliação”.

A SOFTEX (2009c) destaca como necessidades para o sucesso de uma avaliação as seguintes prerrogativas:

- a. *Comprometimento do patrocinador*: O comprometimento do patrocinador é essencial para assegurar que os objetivos da avaliação sejam atingidos. Este comprometimento também diz respeito aos recursos necessários, tempo e pessoal disponível para executar a avaliação.
- b. *Motivação*: O responsável pela unidade organizacional deve motivar os participantes de forma aberta e construtiva, deixando claro a todos que o foco da avaliação é o processo e não o desempenho dos indivíduos que implementam o processo.
- c. *Fornecimento de feedback*: O fornecimento de feedback e o estabelecimento de uma atmosfera que encoraje a discussão aberta sobre os resultados preliminares, durante a avaliação, ajudam a assegurar que a avaliação seja significativa para a organização.
- d. *Confidencialidade*: O respeito à confidencialidade das informações obtidas durante a avaliação é essencial para que se obtenha as informações necessárias.
- e. *Percepção dos benefícios*: Os membros da organização devem perceber que a avaliação resultará em benefícios que os ajudarão direta ou indiretamente a realizar o seu trabalho.
- f. *Credibilidade*: É importante que todas as partes confiem que os avaliadores têm a experiência e competência para realizar

a avaliação, são imparciais e têm um entendimento adequado da unidade organizacional.

O processo de avaliação MA-MPS é composto de quatro subprocessos, descritos adiante, que são compostos de diversas atividades descritas através das tarefas que a compõem (SOFTEX, 2009c).

Tabela 4 - Processo de avaliação  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

<b>PROCESSO DE AVALIAÇÃO</b>	
<b>SUBPROCESSO</b>	<b>ATIVIDADE</b>
Contratar a avaliação	Pesquisar Instituições Avaliadoras (IA)
	Estabelecer contato
Preparar a realização da avaliação	Viabilizar a avaliação
	Planejar a avaliação
	Preparar a avaliação
	Conduzir a avaliação inicial
	Completar a preparação da avaliação
Realizar a avaliação final	Conduzir a avaliação final
	Avaliar a execução do processo de avaliação
Documentar os resultados da avaliação	Relatar resultados
	Registrar resultados

Como resultado do processo de avaliação espera-se que:

- a. Sejam obtidos dados e informações que caracterizem os processos de software da organização.
- b. Seja determinado o grau em que os resultados esperados são alcançados e os processos atingem o seu propósito.

- c. Seja atribuído um nível de maturidade do MR-MPS à organização.

## 7.1 Equipe de Avaliação

Para garantir que a equipe de avaliação tenha o conhecimento da organização que está sendo avaliada e que não tenha interesse direto no resultado da avaliação, a equipe deve ser composta tanto por membros internos quanto por membros externos à organização.

Os membros internos da equipe de avaliação são chamados de representantes da unidade organizacional, eles devem ser avaliadores com os mesmos deveres e direitos dos demais e contribuem com seu conhecimento da empresa para que toda a equipe entenda melhor a organização, seus processos e os artefatos apresentados. Eles devem ter independência para desempenhar o seu papel de avaliador, não devem ser superiores hierárquicos dos colaboradores que serão entrevistados e não podem ter tido uma participação significativa nos projetos que serão avaliados. Os membros internos que forem selecionados devem:

- a. Ter assistido o Curso de Introdução ao MPS-BR (C1-MPS-BR);
- b. Ter experiência em desenvolvimento de software, preferencialmente em gerência de projetos.; isenção nos julgamentos e opiniões; conhecimento da organização, seus processos e artefatos apresentados; postura de colaboração com a equipe de avaliação; compromisso com a exatidão dos

- resultados da avaliação; interpretação de termos, jargão, nomes e outros elementos peculiares da organização.
- c. Após a Avaliação Inicial devem verificar se os ajustes requeridos foram implementados corretamente e assinar o Relatório de Ajustes Realizados para a Avaliação Final.
  - d. Na Avaliação Final, devem atuar durante as entrevistas como facilitador para que as perguntas e respostas sejam compreendidas pela equipe e pelo entrevistado. Durante todo o período da Avaliação, e mesmo depois, ele deve evitar qualquer comentário, conversa ou insinuação sobre os trabalhos realizados durante a avaliação.

Impedimentos para ser o Representante da Organização na Equipe de Avaliação:

- a. Ser sócio da organização a qual pertença à organização.
- b. Ser parente em qualquer grau de um dos sócios da empresa.
- c. Ser parente em qualquer grau de um dos membros da direção da empresa.

Os membros externos são:

- a. O avaliador líder da IA ou líder em formação indicado pela SOFTEX que esteja autorizado a conduzir avaliação nos níveis indicados pela SOFTEX;
- b. O(s) avaliador(es) adjunto(s) da IA;
- c. O(s) avaliador(es) em formação indicado(s) pela SOFTEX, se for o caso.

É responsabilidade do avaliador líder verificar se os requisitos do Processo e Método de Avaliação MA-MPS para a composição da equipe estão sendo seguidos. No caso de existir uma Instituição Implementadora, esta também deverá ajudar o avaliador líder verificando esses requisitos e orientando a unidade organizacional na escolha dos projetos e do representante da organização na equipe de avaliação.

## **7.2 SUBPROCESSO 1 – Contratar avaliação**

Este subprocesso inicia o processo de avaliação, tem por objetivo estabelecer contrato para realização de avaliação, solicitada por uma organização que queira avaliar seus processos ou os processos de outras.

O contratante da avaliação, chamado patrocinador, é o representante da organização solicitante, um agente externo<sup>16</sup> ou uma outra organização que necessite conhecer a capacidade dos processos da organização avaliada.

---

<sup>16</sup> Pode ser um grupo de empresas que visem minimizar os custos de avaliação. Ex.: IOGE (Instituição Organizadora de Grupos de Empresas).

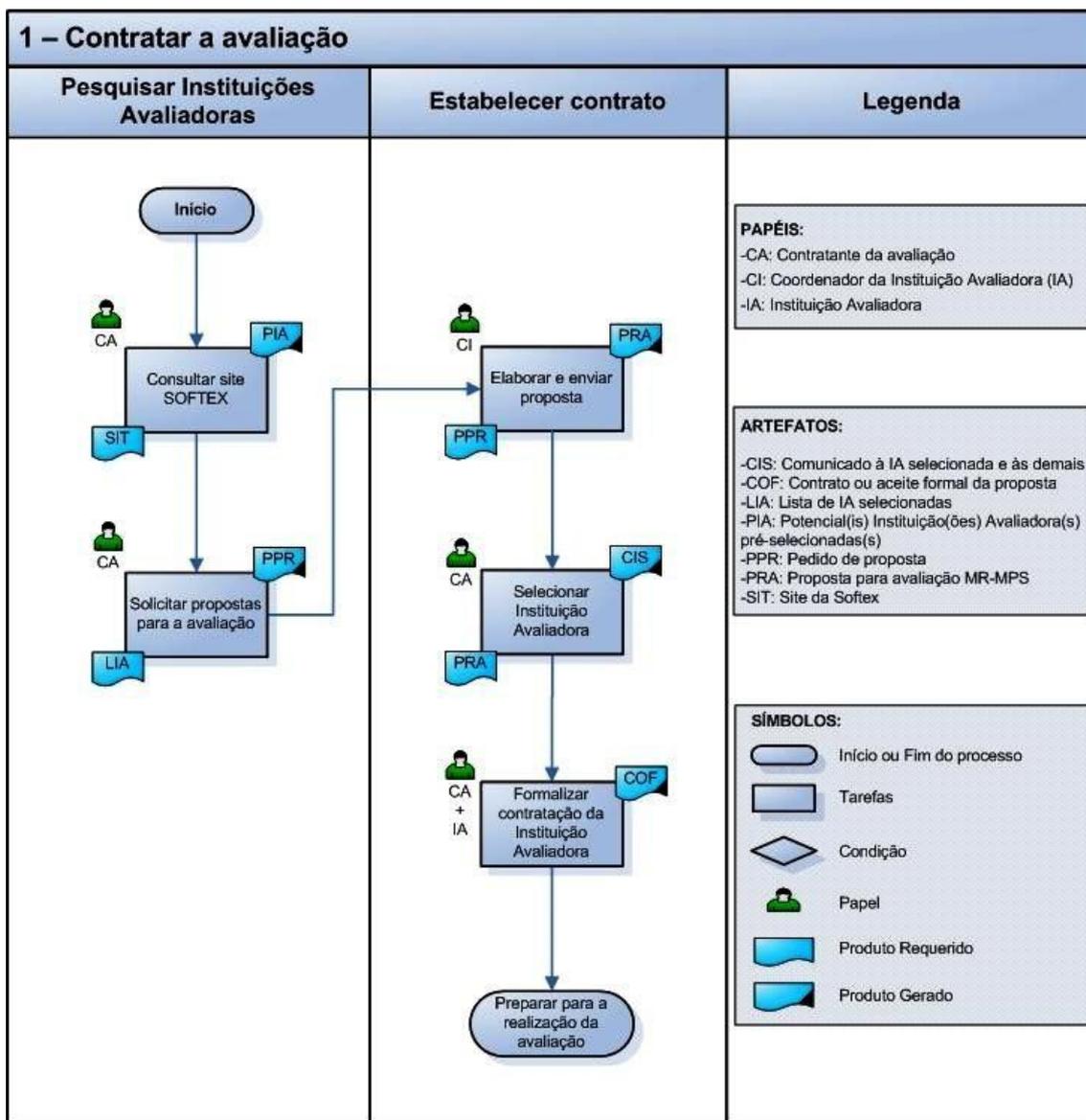


Figura 12 - Fluxo do subprocesso 1 - Contratar avaliação  
 Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

Primeiramente uma lista de Instituições Avaliadoras (IA), credenciadas junto a SOFTEX deve ser obtida e propostas para a realização de uma avaliação MR-MPS na organização devem ser realizadas.

A Instituição Avaliadora ao receber uma solicitação de avaliação, deve verificar se dispõe de avaliador líder e adjunto, dentro das competências requeridas e condições para condução da avaliação, e então elaborar proposta a ser enviada a organização solicitante. Nesta proposta devem constar: valor da taxa SOFTEX correspondente ao nível de avaliação solicitado.

Uma IA pode ser contratada quando for:

- a. A Instituição Implementadora (II) ou a Instituição Organizadora de Grupos de Empresas (IOGE) que apoiou a implementação do MR-MPS na organização solicitante.
- b. Tiver prestado consultoria em processo para a organização nos últimos 3 anos.
- c. Algum membro da equipe da IA: tiver ministrado curso para a organização a ser avaliada; trabalhar na organização a ser avaliada; tiver orientado trabalho acadêmico que tenha sido utilizado na implementação do MR-MPS; tiver realizado “*gap analysis*<sup>17</sup>” na organização a ser avaliada.
- d. Algum membro da organização a ser avaliada: tiver feito parte da equipe da IA nos últimos 3 anos.
- e. A IA tiver tido membros comuns a Instituição Implementadora que implementou o modelo ou IA que realizou a implementação nos últimos 3 anos.
- f. A organização a ser avaliada for uma IA (uma IA não pode se auto-avaliar).
- g. A IA não possuir avaliador líder credenciado para realizar avaliações no nível MR-MPS solicitado pela organização a ser avaliada.
- h. Houver qualquer outro impedimento de natureza ética.

---

<sup>17</sup> Análise da Diferença - consiste em comparar a situação atual de algo com um modelo pré-determinado. Sua proposta consiste em analisar e identificar o que já está em uso com uma metodologia e o que está faltando. O resultado da análise mostra o que é necessário para fortalecer a qualidade e a administração dos dados e da informação.

- i. Se a IA tiver uma Instituição Implementadora, com mesma razão social e CNPJ, todas as regras para a IA se aplicam a Instituição Implementadora.

### **7.3 SUBPROCESSO 2 – Preparar realização da avaliação**

O objetivo deste subprocesso é comunicar a contratação, de uma proposta, para a SOFTEX, para que esta autorize a avaliação, a planeje, prepare a documentação necessária para a realização da avaliação e faça uma avaliação inicial para verificar se a organização está pronta para uma avaliação MR-MPS no nível de maturidade solicitado.

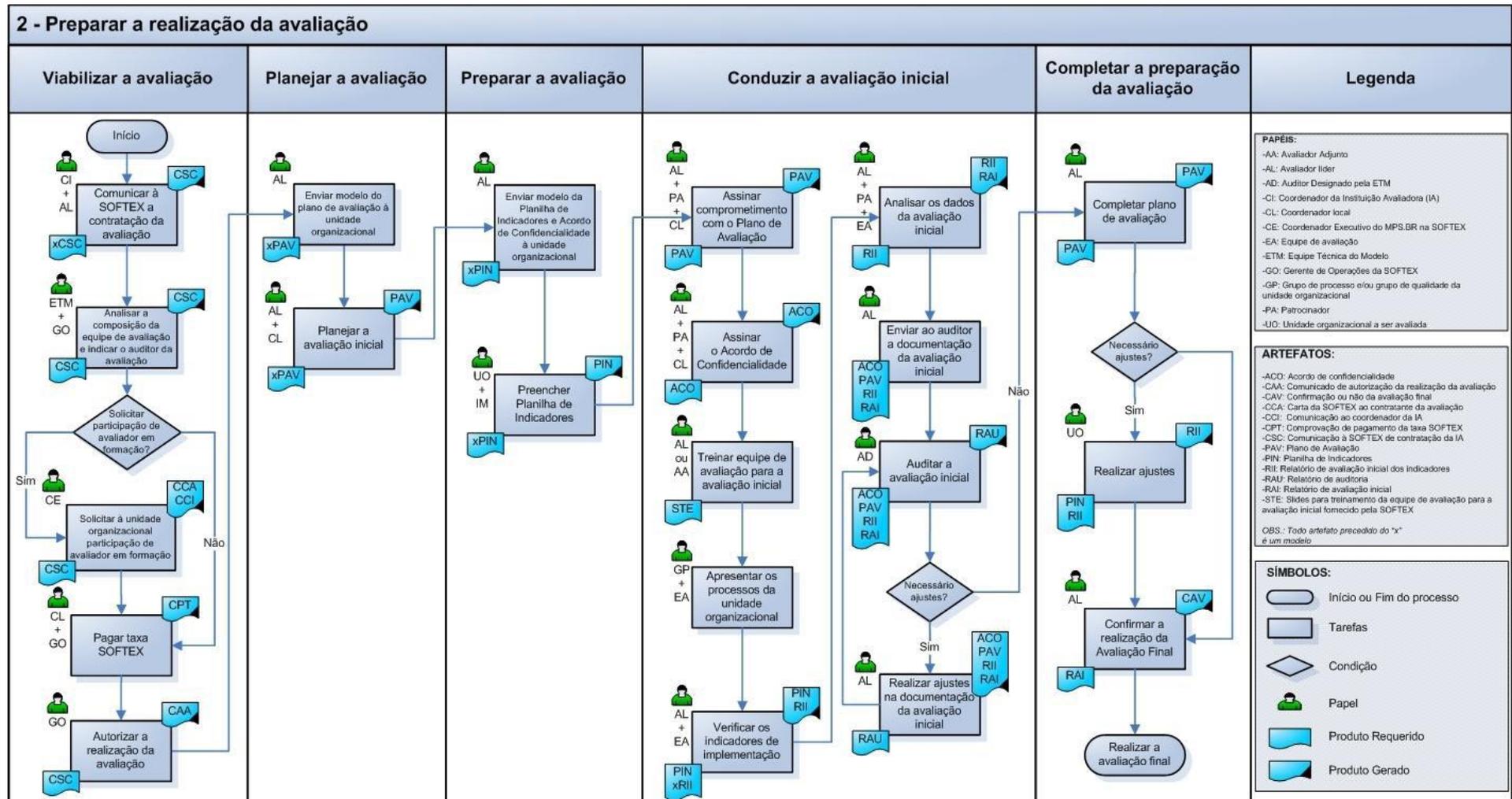


Figura 13 - Fluxo do subprocesso 2 - Preparar a realização da avaliação  
 Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

O subprocesso se inicia com a viabilização da avaliação, quando a contratação de uma IA para avaliação MR-MPS é informada à SOFTEX para obtenção de aprovação da mesma. O coordenador da IA envia o documento Comunicado à SOFTEX e aos responsáveis pelo MPS-BR, contendo a definição da equipe de avaliação e do coordenador local com 20 dias de antecedência da avaliação inicial.

A composição da equipe de avaliação é avaliada pela SOFTEX, no que diz respeito: à adequação do nível em que o avaliador líder está habilitado e o nível MR-MPS pretendido para a avaliação; ao conflito de interesse; e ao cumprimento dos requisitos dos avaliadores adjuntos. Um auditor da avaliação é indicado.

A SOFTEX pode ter o interesse de que a avaliação da organização seja observada por um avaliador líder ou adjunto, em formação e não pertencente a IA contratada pela organização avaliada, e solicitar participação junto ao contratante da avaliação e ao coordenador da IA.

A autorização da avaliação MR-MPS é efetivada mediante o pagamento da taxa SOFTEX e:

- a. Declaração de que: foi paga a taxa de avaliação à SOFTEX; os representantes da unidade organizacional participaram do curso oficial de Introdução ao MPS-BR (C1-MPS-BR); os avaliadores em formação (se for o caso), indicados pela SOFTEX para fazerem parte da equipe de avaliação, cumprem as condições estabelecidas; a unidade

organizacional e/ou IOGE concordou com participação de avaliador em formação (se for o caso).

- b. Cópia da declaração da IOGE (no caso de grupo de empresas) de que a unidade organizacional cumpriu 100% da implementação e não tem qualquer pendência que impeça receber parcelas do apoio SOFTEX.

O Plano de Avaliação (modelo SOFTEX) a ser seguido para se realizar a avaliação na unidade organizacional deve ser elaborado e enviado para a SOFTEX. Então, a avaliação inicial da organização deve ser planejada, incluindo os dados da unidade organizacional e dos projetos, a data da avaliação inicial, as tarefas da avaliação inicial e seu cronograma. A avaliação inicial presencial é obrigatória para todos os níveis de maturidade do MR-MPS.

O plano de avaliação inicial deve suprir também: a infraestrutura necessária para a realização da avaliação e a seleção das pessoas a serem entrevistadas, individualmente ou em grupo; controle de versões do plano de avaliação inicial.

A SOFTEX ao preparar a avaliação, elabora e envia para a organização, uma planilha com os indicadores que comprovam a implementação dos processos e que será utilizada na avaliação. São indicadores de implementação:

- a. *Indicadores diretos*: são o objetivo de uma tarefa, ou seja, o produto principal da realização de uma tarefa.
- b. *Indicadores Indiretos*: são artefatos que são consequência da realização de uma tarefa e que referendam a implementação

de um resultado, mas que não são o objetivo da realização da tarefa e normalmente são gerados durante a execução dos projetos.

- c. *Afirmações*: são obtidas em entrevistas e/ou apresentações e confirmam a implementação do processo, seus resultados e atributos.

Na avaliação inicial é obrigatória a presença de toda a equipe de avaliação e de um representante da Instituição Implementadora. Podem participar também, da avaliação inicial, outros colaboradores da unidade organizacional que ela considere oportuno e com a concordância do avaliador líder. Deve haver o comprometimento de todos com o plano de avaliação definido e com o acordo de confidencialidade.

A equipe de avaliação, então, é treinada pelos avaliadores, os processos de software da organização são apresentados, a planilha de indicadores de implementação é verificada por todos os envolvidos na avaliação e um Relatório de Avaliação Inicial, onde são explicitados todos os pontos a serem ajustados até a data da avaliação final, é gerado.

Tabela 5 - Tempo máximo e mínimo de uma avaliação MR-MPS inicial  
 Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

<b>NÍVEIS MR-MPS</b>	<b>DURAÇÃO DA AVALIAÇÃO INICIAL (DIAS)</b>	<b>COMPOSIÇÃO DA EQUIPE DE AVALIAÇÃO</b>
A	3-4	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 8. Total máximo de avaliadores: 9
B	3-4	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 8. Total máximo de avaliadores: 9
C	3-4	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 6. Total máximo de avaliadores: 8
D	3-4	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 6. Total máximo de avaliadores: 7
E	2-4	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 4. Total máximo de avaliadores: 6
F	2-3	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 4. Total máximo de avaliadores: 5
G	1-2	Av. líder (1), av. adjunto (1 ou mais), representante unidade organizacional (1 ou mais) – Total mínimo de avaliadores: 3. Total máximo de avaliadores: 4

Com a Planilha de Indicadores verificada e o Relatório de Avaliação Inicial, o avaliador líder e a equipe de avaliação se reúnem com o patrocinador, a equipe responsável pela implementação do MR-MPS na unidade organizacional, e o representante da Instituição Implementadora MR-MPS para analisarem o Relatório de Avaliação Inicial, os ajustes necessários e decidirem como e quando a avaliação final será realizada. O Relatório de Avaliação Inicial é preenchido e assinado, o que representa o compromisso da organização em realizar o que nele é requerido até a data da avaliação final, e enviado a SOFTEX.

Independentemente do Resultado da Avaliação Inicial, o avaliador líder deve enviar ao auditor indicado os originais do Plano de Avaliação, do Relatório de Avaliação Inicial, do Resultado de Avaliação Inicial e do Acordo de Confidencialidade. Toda essa documentação deverá ser arquivada pela Instituição Avaliadora responsável.

A Avaliação Inicial deverá ser auditada pelo grupo de auditores do MA-MPS, sob supervisão do FCC e da coordenação da SOFTEX. Um parecer é emitido pelo auditor designado e poderá ser:

- a. *Aprovado*: A documentação está completa, correta e não há dúvidas com relação ao Resultado da Avaliação Inicial.
- b. *Aprovado com modificações*: Não há dúvidas com relação ao Resultado da Avaliação Inicial, mas a documentação não está completa ou correta. Um parecer do auditor deverá ser enviado ao avaliador líder solicitando os acertos pertinentes e re-envio para verificação das correções.
- c. *Não aprovado*: Há dúvidas com relação ao Resultado da Avaliação Inicial. Uma reunião com o avaliador líder para esclarecimentos e análise da Planilha de Indicadores é convocada pelo auditor.

Se o relatório de Auditoria indicar que existem itens a serem ajustados, o avaliador líder deve realizar as correções e enviar os documentos corrigidos ao auditor para nova verificação.

Como etapa final do subprocesso preparar a realização da avaliação, o planejamento da avaliação final deve ser completado pelo

avaliador líder com os dados obtidos na avaliação inicial, os ajustes indicados no Relatório de Avaliação Inicial devem ser efetuados pela organização no período que antecede a avaliação final e um Relatório de Ajustes Realizados para a Avaliação Final é gerado. Com os ajustes feitos, a realização da avaliação final é confirmada com a aprovação do Relatório de Ajustes Realizados para a Avaliação Final, totalmente preenchido com as soluções dos itens requeridos, pelo avaliador líder.

## **7.4 SUBPROCESSO 3 – Realizar avaliação final**

Este subprocesso se inicia com a condução da avaliação final da organização de acordo com o processo e o método de avaliação MA-MPS e o Plano de Avaliação. Nesta etapa uma reunião com os colaboradores da organização é realizada para que o avaliador líder explique todos os procedimentos e tarefas que serão realizadas durante a avaliação final. O Plano de Avaliação e o Acordo de Confidencialidade são assinados, o primeiro em detrimento do comprometimento de todos com o conteúdo do plano.

A equipe de avaliação deve ser então, treinada pelo avaliador líder ou por um avaliador adjunto para a realização da avaliação final, a equipe deve ainda ser dividida em miniequipes, para as quais são atribuídos processos, tarefas e responsabilidades. As miniequipes examinam a planilha e o Relatório de Ajustes Realizados para a Avaliação Final e verificam a presença e adequação dos resultados que comprovam a implementação dos resultados esperados para cada processo e atributo do processo, nos processos que lhe cabem.

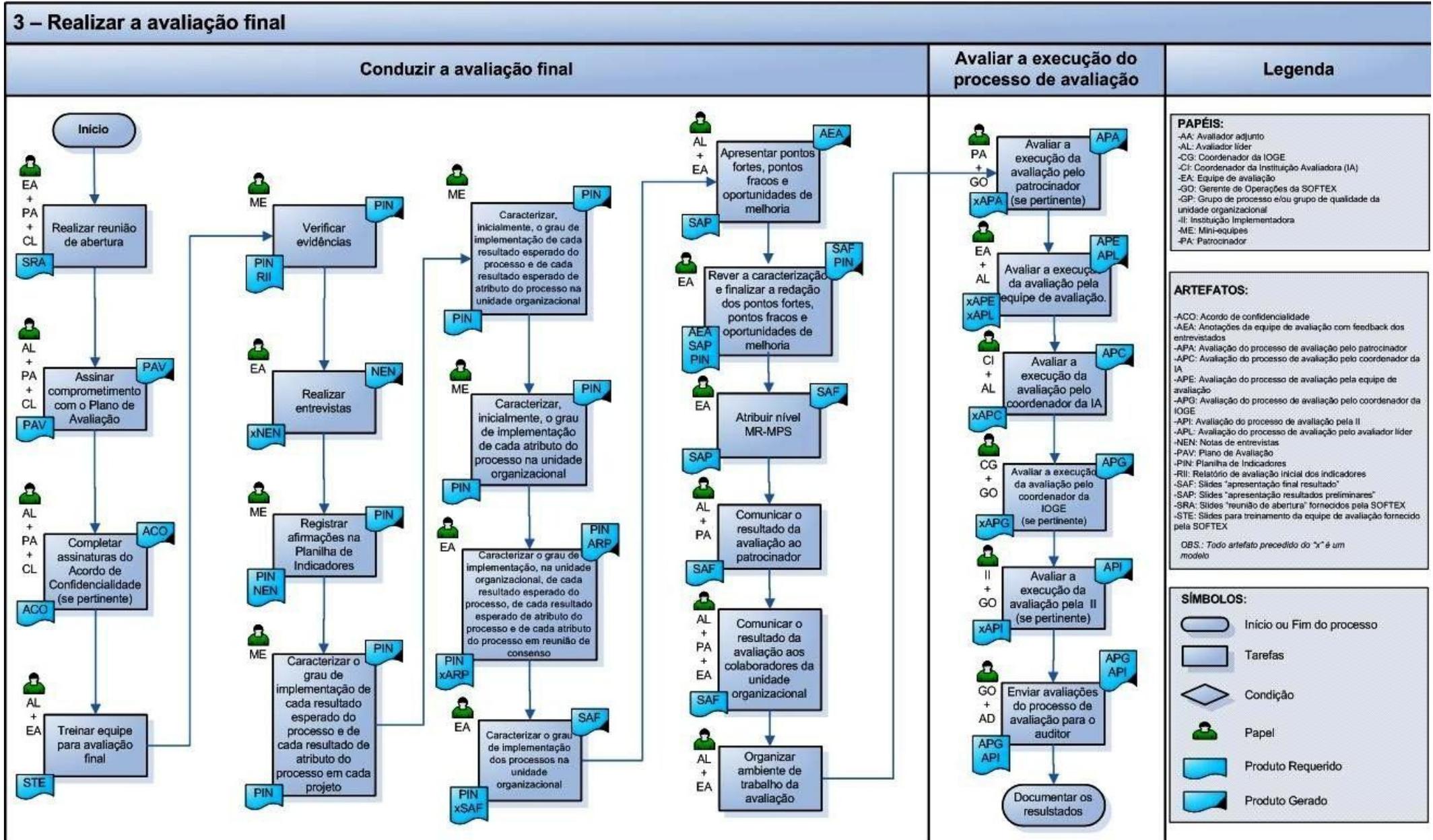


Figura 14 - Fluxo do subprocesso 4 - Realizar avaliação final  
 Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

Em seguida entrevistas são realizadas com os envolvidos nos processos (de acordo com o Plano de Avaliação) para confirmar a implementação dos processos e a geração dos indicadores. As entrevistas são um dos pontos mais importantes da avaliação, pois, ajuda a equipe de avaliação a entender e analisar como a organização implementa os seus processos, ajuda também a identificar o que pode ser melhorado e o nível de institucionalização dos processos.

Ao final de cada entrevista a equipe de avaliação deve consolidar os dados das entrevistas de maneira resumida na Planilha de Indicadores, na forma de declarações precisas, as suas observações sobre o que realmente foi afirmado, os pontos fortes, os pontos fracos e as oportunidades de melhoria. O objetivo da consolidação é, também, controlar o andamento da avaliação, permitindo verificar se existem dados suficientes para a equipe fazer um julgamento, ou se são necessárias mais entrevistas para, com mais informações, ser possível o julgamento.

Mediante os resultados da Planilha de Indicadores e das entrevistas realizadas, cada miniequipe deve caracterizar os processos sob sua responsabilidade e o grau implementação de cada resultado esperado do processo e de cada resultado esperado de atributo do processo nos projetos, com base nas atribuições contidas na tabela a seguir.

Tabela 6 - Escala de caracterização do grau de implementação de um resultado esperado do processo e do atributo do processo nos projetos  
 Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

Grau de Implementação	Caracterização
Totalmente implementado (T)	O indicador direto está presente e é julgado adequado.
	Existe pelo menos um indicador indireto e/ou afirmação confirmando a implementação.
	Não foi notado nenhum ponto fraco substancial.
Largamente implementado (L)	O indicador direto está presente e é julgado adequado.
	Existe pelo menos um indicador indireto e/ou afirmação confirmando a implementação.
	Foi notado um ou mais pontos fracos substanciais.
Parcialmente implementado (P)	O indicador direto não está presente ou é julgado inadequado.
	Artefatos/afirmações sugerem que alguns aspectos do resultado esperado estão implementados.
	Pontos fracos foram documentados.
Não implementado (N)	Qualquer situação diferente das acima.
Não avaliado (NA)	O projeto não está na fase de desenvolvimento que permite atender ao resultado ou não faz parte do escopo do projeto atender ao resultado.
Fora do escopo (F)	O resultado esperado está fora do escopo da avaliação, conforme documentado no plano da avaliação.

Cada miniequipe, com base na caracterização do grau de implementação de cada resultado esperado do processo e de cada resultado esperado de atributo do processo em cada projeto, faz uma caracterização inicial do grau de implementação de cada resultado esperado na organização e

prepara a redação inicial dos pontos fortes, pontos fracos e oportunidades de melhoria, relativos aos processos sob sua responsabilidade, para serem apresentados a toda a equipe de avaliação. A caracterização é feita de acordo com as regras do MA-MPS, conforme a tabela seguir.

Tabela 7 - Regras de agregação de caracterização dos resultados esperados dos processos e dos atributos do processo nos projetos para obtenção da caracterização organizacional  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

<b>Caracterização nos projetos avaliados</b>	<b>Caracterização agregada para a unidade organizacional</b>	<b>Observações</b>
Todos X (isto é, todos T, ou todos L, ou todos P, ou todos N)	X	Se todos os projetos têm a mesma caracterização, esta é a caracterização da unidade organizacional.
Todos os projetos terminados X (isto é, todos T, ou todos L, ou todos P ou todos N) e os incompletos NA (Não Avaliado)	X	Se pelo estágio de desenvolvimento dos projetos incompletos o resultado não puder ser evidenciado (NA), a caracterização da unidade organizacional é X.
Todos T ou L	L	Se os projetos forem caracterizados para um resultado esperado como L ou T, caracteriza-se a unidade organizacional como L para este resultado esperado.
Todos T ou L e os incompletos NA (Não Avaliado)	L	Se pelo estágio de desenvolvimento dos projetos incompletos o resultado não puder ser evidenciado, a caracterização da unidade organizacional é L.
Existem P, mas não existem N (Pode existir NA - Não Avaliado)	L ou P	A decisão é da equipe de avaliação.
Existe N	N, P ou L	A decisão é da equipe de avaliação.
Resultado Esperado F (Fora do Escopo)	F	O resultado esperado foi declarado fora do escopo da avaliação no plano da avaliação.

Com base nos resultados anteriores de caracterização, o grau de implementação de cada atributo do processo na organização deve ser também caracterizado de acordo com as regras apresentadas na próxima tabela.

Tabela 8 - Regras de caracterização do grau de implementação dos atributos do processo na organização

Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

<b>Grau de implementação</b>	<b>Caracterização</b>	<b>Porcentagem de Implementação dos resultados relacionados<sup>18</sup></b>
Totalmente implementado (T)	Existe evidência de um enfoque completo e sistemático para o atributo no processo avaliado e de sua plena implementação. Não existem pontos fracos relevantes para este atributo no processo avaliado.	>85% a 100%
Largamente implementado (L)	Existe evidência de um enfoque sistemático e de um grau significativo de implementação do atributo no processo avaliado. Existem pontos fracos para este atributo no processo avaliado.	>50% a 85%
Parcialmente implementado (P)	Existe alguma evidência de um enfoque para o atributo e de alguma implementação do atributo no processo avaliado. Alguns aspectos de implementação não são possíveis de prever.	>15% a 50%
Não implementado (N)	Existe pouca ou nenhuma evidência de implementação do atributo no processo avaliado.	0 a 15%

Em uma reunião de toda a equipe de avaliação é realizada a agregação, atribuindo-se um grau que caracterize a implementação de cada resultado esperado do processo, de cada resultado esperado de atributo do processo e de cada atributo do processo na organização. A equipe, em consenso, chega à redação dos pontos fortes, pontos fracos e oportunidades de melhoria que serão apresentados aos entrevistados e que justificam a caracterização.

---

<sup>18</sup> É ao mesmo tempo uma avaliação qualitativa e quantitativa.

Caracterizado o grau de implementação de cada resultado de processo e de cada atributo de processo na unidade organizacional, a equipe de avaliação, por meio de consenso, deve caracterizar o grau de implementação de cada processo na organização como:

- a. *Satisfeito*: Quando todos os resultados esperados para o processo foram caracterizados como T (Totalmente implementado) ou L (Largamente implementado) ou a caracterização dos atributos de processo satisfazer às regras da tabela a seguir.
- b. *Não satisfeito*: Em qualquer situação que difira das citadas anteriormente.

Tabela 9 - Caracterização de atributos do processo para satisfazer aos níveis MR-MPS  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2009c)

Nível MR-MPS	Atributos do processo	Caracterização
G	AP 1.1	T
	AP 2.1	L ou T
F	AP 1.1	T
	AP 2.1 e AP 2.2	L ou T
E	AP 1.1, AP 2.1, AP 2.2	T
	AP 3.1 e AP 3.2	L ou T
D	AP 1.1, AP 2.1, AP 2.2	T
	AP 3.1 e AP 3.2	L ou T
C	AP 1.1, AP 2.1, AP 2.2	T
	AP 3.1 e AP 3.2	L ou T
B	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2	T
	AP 4.1 e AP 4.2	L ou T

A	AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1 e AP 4.2	T
	AP 5.1 e AP 5.2	L ou T

Os pontos fortes, pontos fracos e as oportunidades de melhoria devem ser apresentados aos entrevistados, em grupos, com o objetivo de obter o feedback a respeito dos resultados preliminares elaborados pela equipe de avaliação. Se necessário, a caracterização do grau de implementação dos processos poderá ser revista. A finalização da redação dos pontos fortes, pontos fracos e oportunidades de melhoria é feita a partir do feedback obtido. A obrigatoriedade que existe é apenas da equipe de avaliação refletir e considerar o feedback.

Uma vez obtido o consenso sobre cada processo contemplado na avaliação e finalizada a redação dos pontos fortes, pontos fracos e oportunidades de melhoria, é atribuído ou não o nível MR-MPS pretendido à organização. A atribuição do nível de maturidade é feita a uma organização se cada processo pertencente ao nível pretendido e incluído no escopo da avaliação tiver sido caracterizado como *satisfeito*.

Os resultados da avaliação deverão ser comunicados ao patrocinador e aos colaboradores da organização pelo avaliador líder. E ao fim da avaliação a equipe de avaliação deve eliminar a documentação gerada que possa comprometer o Acordo de Confidencialidade. Também deve gerar uma versão da apresentação dos resultados da avaliação e da planilha de avaliação final resultante e entregar para o coordenador local.

No fim do processo, a execução da avaliação da organização deve ser avaliada como forma de fornecer feedback à SOFTEX acerca do Processo

e Método de Avaliação MA-MPS, da Instituição Avaliadora (IA) e de outros aspectos relacionados à avaliação realizada. É responsabilidade do avaliador líder fazer com que as avaliações da equipe de avaliação, do patrocinador, do coordenador da IA e a sua própria avaliação sejam realizadas. É responsabilidade da SOFTEX enviar e garantir o recebimento das avaliações da IOGE e da Instituição Implementadora e enviá-las ao auditor designado.

## **7.5 SUBPROCESSO 4 – Documentar os resultados da avaliação**

É o último subprocesso do processo de avaliação, tem por objetivo relatar os resultados obtidos na avaliação da organização e incorporá-los ao banco de dados de avaliações MR-MPS da SOFTEX.

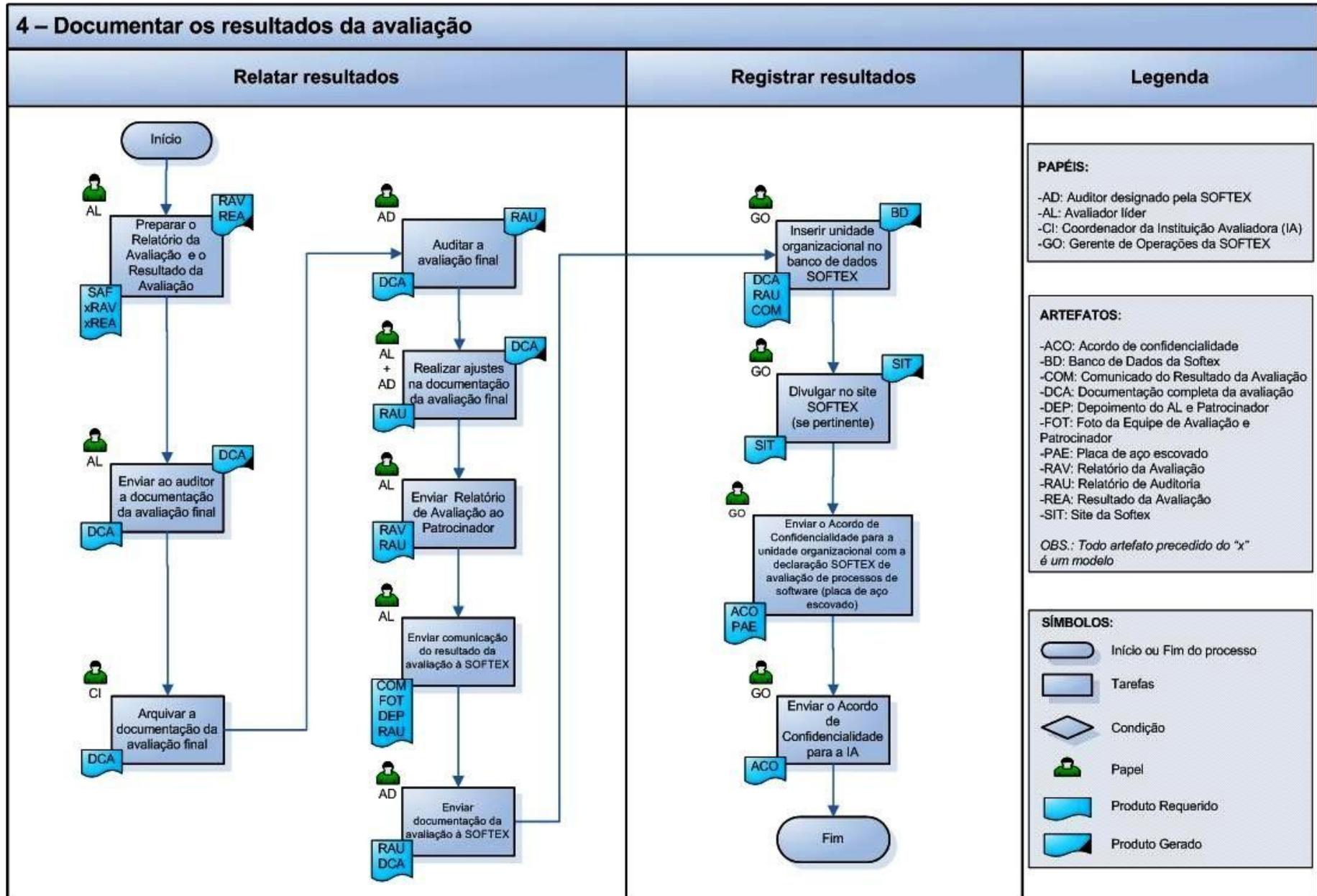


Figura 15 - Fluxo do subprocesso 4 - Documentar os resultados da Avaliação  
 Fonte: MPS-BR Guia de Avaliação (SOFTEX, 2009c)

É com base na avaliação realizada e nos resultados obtidos com ela, que o avaliador líder elabora um Relatório da Avaliação e o Resultado da Avaliação. O avaliador líder, então, reúne e organiza toda a documentação gerada pela avaliação final e a envia para o auditor designado. A Instituição Avaliadora, então, arquiva cópias dessa documentação e da versão final da planilha de indicadores de implementação.

As avaliações auditadas recebem um parecer que pode ser:

- a. *Aprovado*: A documentação está completa, correta e não há dúvidas com relação ao resultado da avaliação final. O parecer é enviado ao avaliador líder, ao coordenador da IA e à SOFTEX pelo auditor, e este autoriza o envio do Relatório de Avaliação e do Resultado da Avaliação para a organização e contratante, e para a SOFTEX, respectivamente.
- b. *Aprovado com modificações*: Não há dúvidas com relação ao resultado da avaliação final, mas a documentação não está completa ou correta. O parecer é enviado ao avaliador líder, ao coordenador da IA pelo Auditor, solicitando acertos pertinentes e envia para a verificação das correções.
- c. *Não aprovado*: Há dúvidas com relação ao resultado da avaliação final. O parecer é enviado a SOFTEX, com cópia para o Avaliador líder e para o coordenador da IA. A SOFTEX convoca o Auditor e o Avaliador líder para uma reunião de esclarecimentos e análise da Planilha de indicadores. A SOFTEX pode ainda realizar uma auditoria na organização solicitante da avaliação.

Se o Relatório de Auditoria indicar itens a serem corrigidos, o Avaliador Líder realiza as correções e envia a documentação corrigida ao Auditor para verificação. Sendo a documentação aprovada pelo Auditor, o Avaliador Líder envia o Relatório de Avaliação ao patrocinador e ao contratante da avaliação, e arquiva os documentos corrigidos na Instituição Avaliadora.

Mediante aprovação da documentação da avaliação pelo auditor MA-MPS, o avaliador líder envia o Relatório de Avaliação e o Resultado da Avaliação para o patrocinador e para a SOFTEX, respectivamente. O auditor finalizando a auditoria envia o seu parecer e toda a documentação da avaliação à SOFTEX.

Os Resultados da Avaliação na organização são, então, incorporados ao banco de dados de avaliações MR-MPS da SOFTEX e divulgados no site SOFTEX, se a organização avaliada autorizar. O Acordo de Confidencialidade é enviado para a unidade organizacional e para a IA com a declaração SOFTEX de avaliação de processos de software em placa de aço escovado.

## **8 RESULTADOS DAS EMPRESAS QUE IMPLEMENTARAM AS MELHORIAS PROPOSTAS PELO MA-MPS**

Os programas de melhoria de processo de software envolvem muitos fatores que podem ser decisivos para o sucesso da implantação ou que podem dificultá-la. Para tanto, estudos que busquem verificar a real contribuição, seja ela positiva ou negativa, para os programas de melhoria de processo se fazem necessários (SANTOS; CONTE, 2010).

O resultado de uma avaliação MPS pode ser usado tanto para orientar a organização na melhoria dos processos implementados, quanto para apoiar na definição dos próximos passos para crescimento da maturidade dos seus processos. Portanto, a equipe de avaliação deve avaliar também aspectos da implementação que, apesar de atenderem adequadamente o nível solicitado, devem ser melhorados para que a organização alcance rapidamente e de forma adequada níveis mais altos de maturidade (SOFTEX, 2008a).

Após implementar as melhorias nos processos padrão da organização, é importante que a utilização adequada dos processos e outros ativos de processo organizacional nos projetos da organização seja apoiada, bem como monitorada para garantir que as melhorias implementadas estão tendo os efeitos desejados e não têm impacto negativo nos projetos (SOFTEX 2009 apud ZAHARAN, 1998). O sucesso de uma iniciativa de melhoria baseada no modelo MPS pode ser medido pelo nível de alcance da implementação dos processos (SOFTEX, 2008a).

É muito comum ouvir que a melhoria da qualidade, principalmente através da melhoria de processos, implica no aumento da produtividade e competitividade das empresas. Mas, normalmente, faltam evidências objetivas que comprovem essa afirmação (SOFTEX, 2008b).

Foi com base nessa questão e na aceleração da adoção do modelo MPS no Brasil a partir de 2006, que a SOFTEX propôs a realização do projeto iMPS que visa apresentar os Resultados de Desempenho de organizações que adotaram o Modelo MPS, compreendendo a variação do desempenho de organizações desenvolvedoras de software, ao longo do tempo.

De acordo com o documento gerado pelo iMPS de 2008, até agosto de 2008 contava-se com 107 avaliações MA-MPS contra 79 avaliações CMMI realizadas no Brasil, com a perspectiva de que até o fim de 2010 sejam feitas mais 377 avaliações MA-MPS oficiais. A SOFTEX como responsável pela monitoração da satisfação das empresas que adotaram o modelo MPS-BR prevê os seguintes indicadores e meios de verificação em todas as empresas apoiadas:

- a. Se 50 pequenas e médias empresas de software (PMES) participantes do Programa MPS-BR aumentaram sua presença em mercados internacionais;
- b. Se houve 20 % de aumento do volume de exportação das PMES;
- c. Se houve 20% de redução dos custos de produção de software das PMES participantes do Programa, pela introdução do MPS;

- d. Pesquisa para medir a evolução dos custos de produção do software e o incremento do volume de exportações das empresas;
- e. Pesquisa de opinião nas organizações relacionada à adoção do modelo MPS; das quais se espera que, pelo menos, 80% das PMEs que adotaram o MPS-BR estejam satisfeitas, e;
- f. Pesquisa de satisfação das empresas.

Considerando os requisitos citados anteriormente e o cenário crescente de adoção do modelo MPS e das boas práticas da engenharia de software nele previstas, tem-se o interesse em compreender qualitativamente as variáveis de desempenho decorrentes da adoção destas práticas, como custo, produtividade e qualidade (SOFTEX 2008b *apud* WOHLIN *et. al.*, 2000).

Os estudos do projeto iMPS foram realizados em um total de 171 empresas que estão iniciando a implementação do modelo MPS (63), que estão em procedimento de avaliação (21), e que estão com avaliação publicada no portal da SOFTEX e com prazo de validade vigente (87). Eles verificaram que a maioria das organizações que adotaram o modelo MPS-BR ainda se encontram nos níveis iniciais. Métricas foram utilizadas nesses estudos em diferentes perspectivas como mostram as tabelas a seguir.

Tabela 10 - Métricas usadas na perspectiva Organizacional  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2008b)

MÉTRICA	INTERPRETAÇÃO
Outros modelos de referência de processo	Indica se a organização também utiliza outros modelos de referência {CMM, CMMI, ISO 9001, ...}
Número de clientes no país	Representa a quantidade de clientes da empresa no país

Número de clientes no exterior	Representa a quantidade de clientes da empresa no exterior
Número de projetos no país	Representa o número de projetos da empresa no país
Número de projetos no exterior	Representa o número de projetos da empresa no exterior
Número de funcionários total	Funcionários envolvidos em desenvolvimento de software, segundo as seguintes categorias de regime de emprego: assalariado, sócio-proprietário, cooperado, pessoa física, autônomo, outros.
Satisfação do cliente	Totalmente Satisfeitos: todos os clientes se demonstram satisfeitos com os produtos e/ou serviços da empresa. Largamente Satisfeitos: a maioria dos clientes se demonstra satisfeita com os produtos e/ou serviços da empresa. Parcialmente Satisfeitos: a minoria dos clientes se demonstra satisfeita com os produtos e/ou serviços da empresa. Não Satisfeitos: nenhum cliente se demonstra satisfeito com os produtos e/ou serviços da empresa. Satisfação Desconhecida: a empresa não conhece o grau de satisfação de seus clientes.

Com base nas métricas citadas acima acerca dos estudos realizados com as empresas que adotaram o modelo MPS-BR, observou-se que:

- a. 3,3% das empresas utilizam ou utilizavam o modelo CMM de referência.
- b. 9,8% das empresas utilizam ou utilizavam o modelo CMMI de referência.
- c. 24,4% das empresas utilizam ou utilizavam o modelo ISO de referência.
- d. 8,9% das empresas utilizam ou utilizavam outros modelos de referência.
- e. As empresas apresentavam uma variação de 16 a 40 clientes no País.

- f. 24,5% das empresas possuíam clientes no exterior.
- g. As empresas apresentavam uma variação de 7 a 14 projetos ativos no país.
- h. 12,7% das empresas possuem projetos ativos em outros países.
- i. As empresas apresentam uma variação de 29 a 75 funcionários.

Tabela 11 - Nível de satisfação dos clientes apresentado  
 Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2008b)

Agrupamento	Resultado	
Empresas Iniciando a Implementação	Totalmente Satisfeitos	18,4%
	Largamente Satisfeitos	50,0%
	Parcialmente Satisfeitos	13,2%
	Não Satisfeitos	0%
	Satisfação não conhecida	18,4%
Empresas em Processo de Avaliação	Totalmente Satisfeitos	10%
	Largamente Satisfeitos	50%
	Parcialmente Satisfeitos	0%
	Não Satisfeitos	0%
	Satisfação não conhecida	40%
Empresas Nível G	Totalmente Satisfeitos	14%
	Largamente Satisfeitos	48%
	Parcialmente Satisfeitos	8%
	Não Satisfeitos	0%
	Satisfação não conhecida	30%
Empresas Nível F	Totalmente Satisfeitos	5,3%
	Largamente Satisfeitos	68,4%
	Parcialmente Satisfeitos	0%

	Não Satisfeitos	0%
	Satisfação não conhecida	26,3%
Empresas Níveis E – A	Totalmente Satisfeitos	14,3%
	Largamente Satisfeitos	57,1%
	Parcialmente Satisfeitos	0%
	Não Satisfeitos	0%
	Satisfação não conhecida	28,6%

Um projeto é um empreendimento realizado para criar um produto ou serviço único. O projeto se caracteriza por temporalidade e resultado, serviço ou produto único e elaboração progressiva. A tabela a seguir apresenta as métricas e a interpretação que foi dada para se coletar os valores para a perspectiva de projetos.

Tabela 12 - Métricas usadas na perspectiva de projetos  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2008b)

MÉTRICA	INTERPRETAÇÃO
Custo médio de projeto	Medido em função do percentual do faturamento bruto nos últimos 12 meses.
Tamanho médio de projeto	Tamanho médio de projeto dos últimos 12 meses, medido na unidade utilizada na empresa. Exemplos: pontos de função, pontos de caso de uso, linhas de código, homem-hora.
Tempo médio dos projetos	Tempo em meses, considerando projetos realizados ou em execução nos últimos 12 meses.
Prazo médio dos projetos	Tempo estimado em meses, considerando projetos realizados ou em execução nos últimos 12 meses.
Precisão de Estimativas	<i>Dados um tempo médio estimado nos projetos dos últimos 12 meses e um tempo médio gasto nos projetos dos últimos 12 meses diferente de 0, calcule Precisão de estimativa = <math>1 -  ((\text{tempo médio gasto nos projetos dos últimos 12 meses} - \text{tempo médio estimado nos projetos dos últimos 12 meses}) / \text{tempo médio estimado nos projetos dos últimos 12 meses}) </math></i>
Produtividade	<i>Dado um tempo médio gasto nos projetos dos últimos 12 meses diferente de 0, calcule Produtividade = Tamanho médio de projeto dos últimos 12 meses / tempo médio gasto</i>

	nos projetos dos últimos 12 meses.
--	------------------------------------

Com base nas métricas citadas acima acerca dos estudos realizados com as empresas que adotaram o modelo MPS-BR na perspectiva de projetos, observou-se que:

- a. O custo médio dos projetos das empresas iniciando a implementação era de 18,76% do percentual de faturamento delas.
- b. O custo médio dos projetos das empresas em processo de avaliação era de 2,31% do percentual de faturamento delas.
- c. O custo médio dos projetos das empresas nível G era de 4% do percentual de faturamento delas.
- d. O custo médio dos projetos das empresas nível F era de 5,25% do percentual de faturamento delas.
- e. O custo médio dos projetos das empresas nível E-A era de 0,14% do percentual de faturamento delas.
- f. O tamanho médio dos projetos de todas as empresas, por ponto de função, variava de 275 a 610.
- g. O tempo médio dos projetos de todas as empresas, variava de 3 a 5 meses.
- h. O prazo médio estimado dos projetos de todas as empresas, variava de 3 a 4 meses.
- i. A precisão das estimativas dos projetos de todas as empresas variava de 0,8% a 1%.
- j. A produtividade em pontos de função por mês variava de 66,66 a 163,75.

A perspectiva do modelo MPS representa o modelo e busca obter as características efetiva e diretamente relacionadas ao modelo MPS, independente de organização e projeto.

Tabela 13 - Métricas usadas na perspectiva do modelo MPS  
Fonte: MPS-BR - Guia de Avaliação (SOFTEX, 2008b)

MÉTRICA	INTERPRETAÇÃO
Tempo de Implementação	Tempo gasto, em média, pelas organizações para implementar o MPS.
Gasto com a Implementação	Percentual do faturamento bruto obtido pelo desenvolvimento de software investido na implementação do MPS, medido através da seguinte fórmula: <i>Dado um valor de faturamento bruto da empresa nos últimos 12 meses diferente de 0, calcule</i> Percentual do faturamento bruto investido na implementação = (Valor investido na implementação MPS / Valor do faturamento bruto nos últimos 12 meses da empresa) * 100.
Gasto com a Avaliação	Percentual do faturamento bruto obtido pelo desenvolvimento de software investido na avaliação MPS, medido através da seguinte fórmula: <i>Dado um valor de faturamento bruto da empresa nos últimos 12 meses diferente de 0, calcule:</i> Percentual do faturamento bruto investido na avaliação = (Valor investido na avaliação MPS / Valor do faturamento bruto nos últimos 12 meses da empresa) * 100.
Satisfação com o Modelo	Indica a satisfação da organização com o modelo MPS {Totalmente Satisfeito, Parcialmente Satisfeito, Não Satisfeito}
Retorno de Investimento	Retorno de investimento (ROI) da implementação do MPS, medida através da seguinte fórmula. Dados o percentual do faturamento bruto investido na implementação MPS diferente de 0 e percentual do faturamento bruto investido na avaliação MPS diferente de 0, calcule: ROI = (Percentual de Variação do faturamento bruto médio obtido pelo desenvolvimento de software após a avaliação da empresa)/(Percentual do faturamento bruto médio obtido pelo desenvolvimento de software investido na implementação MPS + Percentual do faturamento bruto médio obtido pelo desenvolvimento de software investido na avaliação MPS) * 100

Com as métricas citadas acima, chegou-se aos seguintes resultados:

- a. O tempo de implementação MPS, nos níveis correspondentes, das empresas avaliadas e das de nível G e F foi de 12 meses.
- b. O tempo de implementação MPS, nos níveis correspondentes, das empresas de nível E - A foi de 5 meses.
- c. O gasto da implementação MPS com base no percentual do faturamento das empresas variou de 2% a 5% dos níveis mais altos aos mais baixos respectivamente.
- d. O gasto com a avaliação MPS com base no percentual do faturamento das empresas variou de 0,01% a 0,64% dos níveis mais altos aos mais baixos.
- e. De todas as empresa com relação a adoção do modelo MPS, em média, 70,2% estavam totalmente satisfeitas, 24,2% estavam parcialmente satisfeitas, 1,6% não estavam satisfeitas e 4% não tinham uma satisfação conhecida.
- f. Todas as empresas avaliadas possuíam uma variação de retorno de investimento no modelo MPS, com base nos percentuais já recuperados, entre 96,15% e 260,72%, dos níveis mais altos aos mais baixos respectivamente.

Com base em todas as métricas das perspectivas abordadas o estudo mostrou que à medida que as empresas adquirem mais maturidade, elas são capazes de tratar projetos maiores, com menor custo relativo, que as exigências adicionais dos níveis de maturidade mais altos aumentam a capacidade produtiva, assim, o esforço investido na realização das boas práticas da engenharia de software previstas no modelo poderia ser

recuperado ao longo do desenvolvimento dos projetos em detrimento da maior preocupação com a garantia da qualidade de software, que o nível de satisfação dos clientes é maior nos níveis mais altos de maturidade. E que o retorno do investimento no modelo de melhoria deve-se tanto à melhoria da capacidade produtiva quanto aos efeitos da credibilidade associada à obtenção de níveis mais altos, facilitando a concretização de novos negócios.

Essas informações permitiram caracterizar estas empresas avaliadas, e a importância delas está diretamente relacionada ao estabelecimento de um marco de comparação para análises futuras, até mesmo para as empresas.

## 9 CONCLUSÃO

Com base em tudo que foi discutido neste trabalho, acredita-se que os processos de software mais visíveis e controlados afetam diretamente a qualidade dos produtos de software. E que as empresas de Engenharia de Software que tenham por objetivo tornarem-se bem-sucedidas e assim permanecerem, devem ter como enfoque métodos flexíveis de crescimento para seus projetos, além de planejamento estratégico que demonstre como às tecnologias serão utilizadas e como elas produzirão melhorias para os processos, produtos e insumos da empresa.

No âmbito acadêmico, toda a discussão e caracterização dos principais pontos deste trabalho pretendem demonstrar as áreas que precisam de dedicação maior em pesquisa e principalmente de disseminação de conhecimento, tendo como meta a melhoria dos processos de software.

Nesse contexto o trabalho aqui apresentado veio contribuir para demonstrar a importância da melhoria dos processos de software através de avaliações que produzam os insumos necessários para tanto, objetivando proporcionar melhor garantia de qualidade de software. E promover o MA-MPS como um método de avaliação voltado para as características e objetivos das empresas brasileiras.

## REFERÊNCIAS BIBLIOGRÁFICAS

ANACLETO, A., WANGENHEIM C. G. V., SALVIANO C. F., 2004, "Avaliação de Processos para Início de Programas de Melhoria em Micro e Pequenas Empresas de Software", Universidade Federal de Santa Catarina, VI SIMPROS, São Paulo, Brasil. In: Anais do VI Simpósio Internacional de Melhoria de Processos de Software.

ANDRADE, J. M. S., 2005, "Avaliação de Processos de Software em Ambientes de Desenvolvimento de Software Orientados à Organização", Universidade Federal do Rio de Janeiro, Dissertação de Mestrado. Disponível em: [http://ramses.cos.ufrj.br/taqa/index.php?option=com\\_phocadownload&view=category&download=154:dissertacao\\_andrade&id=3:dissertacao-de-mestrado&Itemid=130](http://ramses.cos.ufrj.br/taqa/index.php?option=com_phocadownload&view=category&download=154:dissertacao_andrade&id=3:dissertacao-de-mestrado&Itemid=130). Acesso em: 30 de novembro de 2010.

ANDRADE, J. M. S., ALBUQUERQUE, A. B., CAMPOS, F. B., ROCHA, A. R., 2004, "Conseqüências e Características de um Processo de Desenvolvimento de Software de Qualidade e Aspectos que o Influenciam: Uma Avaliação de Especialistas", SBQS, Brasília, Brasil. In: Anais do III Simpósio Brasileiro de Qualidade de Software.

BARCELLOS, M. P., 2009, "Uma Estratégia Para Medição de Software e Avaliação de Bases de Medidas Para Controle Estatístico de Processos de Software em Organizações de Alta Maturidade". Tese de Doutorado, Rio de Janeiro, Brasil. Disponível em: <http://www.inf.ufes.br/~monalessa/TeseDoutoradoMonalessa.pdf>. Acesso em: 02 de novembro de 2010.

BORNIA, A. C.; WENKE, R., 2000, "Considerações acerca dos Conceitos e Visões sobre os Custos da Qualidade", FAE Centro Universitário, Revista FAE - Business, Curitiba, Brasil. Disponível em: [http://www.fae.edu/publicacoes/pdf/revista\\_da\\_fae/fae\\_v3\\_n2/consideracoes\\_acerca\\_dos.pdf](http://www.fae.edu/publicacoes/pdf/revista_da_fae/fae_v3_n2/consideracoes_acerca_dos.pdf). Acesso em: 18 de agosto de 2010.

BRAUDE, E., 2005, "Projeto de Software – Da programação à arquitetura: uma abordagem baseada em Java", Editora Bookman.

CAMPOS, F. B.; CONTE, T. U.; KATSURAYAMA, A. E.; ROCHA, A. R. C., 2007, "Gerência Quantitativa para o Processo de Desenvolvimento de Requisitos", Porto de Galinhas, Brasil. In: Anais do VI Simpósio Brasileiro de Qualidade de Software.

COSTA, A. F. B., EPPRECHT, E. K., CARPINETTI, L. C. R., 2005, "Controle Estatístico de Qualidade", Editora Atlas, São Paulo, Brasil.

DUARTE, K. C., FALBO, R. A., 2000, "Uma Ontologia de Qualidade de Software", XIV Simpósio Brasileiro de Engenharia de Software, João Pessoa, Brasil. In: Anais do VII Workshop de Qualidade de Software.

DEVMEDIA, 2008, "Qualidade de Software", Brasil. Disponível em: <http://www.devmedia.com.br/articles/viewcomp.asp?comp=5916>. Acesso em: 18 de outubro de 2010.

GARVIN, D. A., 2002, "Gerenciando a qualidade: a visão estratégica e competitiva", Editora Qualitymark.

GOMES, N. S., 2003, "Qualidade de Software - Uma Necessidade", Ministério da Fazenda. Brasília, DF. Disponível em: [http://www.fazenda.gov.br/ucp/pnafe/cst/arquivos/Qualidade\\_de\\_Soft.pdf](http://www.fazenda.gov.br/ucp/pnafe/cst/arquivos/Qualidade_de_Soft.pdf). Acesso em: 18 de agosto de 2010.

GONÇALVES, L. P.; RIBEIRO, T. V.; NASCIMENTO, L. M. A.; REIS, C. A. L.; SANTOS, G., 2010, "Planejamento de Gerência Quantitativa em uma Organização com Maturidade Inicial de Qualidade – Relato de Experiência", In: Anais do IX Simpósio Brasileiro de Qualidade de Software, Belém, Brasil.

IBM, 2007, "SOA terminology overview: Development processes, models, and assets", International Business Machines, IBM, Artigo. Disponível em: <http://www.ibm.com/developerworks/webservices/library/ws-soa-term2>. Acesso em: 10 de novembro de 2010.

IEEE, 1994, "Software Engineering Standards", IEEE Computer Society. Disponível em: <http://ieeexplore.ieee.org>. Acesso em: 10 de novembro de 2010.

IEEE ISO/IEC 12207, 2008, "Systems and software engineering - Software life cycle processes", IEEE Computer Society. Disponível em: <http://ieeexplore.ieee.org>. Acesso em: 10 de novembro de 2010.

KALINOWSKI, M., SANTOS, G., REINEHR, S., MONTONI, M., ROCHA, A. R., WEBER, K. C., TRAVASSOS, G. H., 2010, "MPS-BR: Promovendo a Adoção de Boas Práticas de Engenharia de Software pela Indústria Brasileira", CIBSE, Brasil. Disponível em: [http://www.softex.br/portal/softexweb/uploadDocuments/CIBSE2010\\_MPSBR\\_CameraReady.pdf](http://www.softex.br/portal/softexweb/uploadDocuments/CIBSE2010_MPSBR_CameraReady.pdf). Acesso em: 20 de novembro de 2010.

KOSCIANSKI, A.; SOARES, M. S., 2007, "Qualidade de Software: Aprenda as tecnologias e técnicas mais modernas para o desenvolvimento de software", 2ª Ed., Editora Novatec.

KRUCHTEN, P., 2003, "Introdução ao RUP - Rational Unified Process", 2ª Ed., Editora Ciência Moderna.

NBR ISO/IEC 9126-1, 2003, "Engenharia de Software – Qualidade de produto", Associação Brasileira de Normas Técnicas, Rio de Janeiro, Brasil.

MAGELA, R., 2006, "Engenharia de Software Aplicada – Fundamentos", Editora Alta Books.

MCT, 2001, "Qualidade e Produtividade no Setor de Software Brasileiro", Ministério da Ciência e Tecnologia, Secretaria de Política de Informática. Brasília, DF. Disponível em: <http://www.mct.gov.br/index.php/content/view/2867>. Acesso em: 18 de agosto de 2010.

MONTONI, M. A., KALINOWSKI, M., LUPO, P., ABRANTES, J. F., FERREIRA, A., ROCHA, A. R. C., 2007, "Uma Metodologia para Desenvolvimento de Modelos de Desempenho de Processos para Gerência Quantitativa de Projetos de Software", Porto de Galinhas, Brasil. In: Anais do VI Simpósio Brasileiro de Qualidade de Software.

MORO, R. D., 2008, "Avaliação e Melhoria de Processos de Software: Conceituação e Definição de um Processo para Apoiar a sua Automatização", Universidade Federal do Espírito Santo, Dissertação de Mestrado, Vitória, Brasil. Disponível em: <http://labes.inf.ufes.br/falbo/files/DissertacaoDal%20MoroRodrigo.pdf>. Acesso em: 20 de novembro de 2010.

PALADINI, E. P., 2000, "Gestão da qualidade: Teoria e Prática", Editora Atlas, São Paulo.

PEZZÉ, M., YOUNG, M., 2008, "Teste e Análise de Software: processos, princípios e técnicas", 1ª Ed., Editora Bookman.

PFLEEGER, S. L., 2004, "Engenharia de Software: Teoria e Prática", 2ª Ed., Editora Prentice Hall.

PRESSMAN, R. S., 2002, "Engenharia de Software", 5ª Ed., Editora MC Graw Hill.

ROCHA, A. R. C., MALDONADO, J. C., WEBER, K. C., 2001, "Qualidade de Software: Teoria e Prática". Editora Prentice Hall.

ROSELINO, J. E., 2006, "A Indústria de Software: O "Modelo Brasileiro" em Perspectiva Comparada", Universidade Estadual de Campinas, Instituto de Economia, Tese de Doutorado, Campinas, Brasil. Disponível em: [http://www.softex.br/portal/softexweb/uploadDocuments/\\_observatorio/Roselino%20-%20Tese.pdf](http://www.softex.br/portal/softexweb/uploadDocuments/_observatorio/Roselino%20-%20Tese.pdf). Acesso em: 18 de agosto de 2010.

SANTANA, A. F. L., 2007, "Problemas em Iniciativas de Melhoria de Processos de Software sob a Ótica de uma Teoria de Intervenção", Universidade Federal de Pernambuco, Dissertação de Mestrado, Recife, Brasil. Disponível em: [http://www.di.ufpe.br/~hermano/download/dissertacoes/AF\\_Santana-Dissertacao\\_Vfinal.pdf](http://www.di.ufpe.br/~hermano/download/dissertacoes/AF_Santana-Dissertacao_Vfinal.pdf). Acesso em 15 de novembro de 2010.

SANTOS, D. V., CONTE, T. U., 2010, "Fatores Críticos de Sucesso em Iniciativas MPS – Uma Visão Qualitativa", In: Anais do IX Simpósio Brasileiro de Qualidade de Software, Belém, Brasil.

SEBRAE, 2010, "Qualidade Total", Serviço Brasileiro de Apoio às Micro e Pequenas Empresas, Brasil. Disponível em: <http://www.sebrae.com.br/exibeBia?id=681&searchterm=qualidade>. Acesso em: 18 de agosto de 2010.

SODRÉ, C. C. P., 2006, "Norma ISO/IEC 9126: Avaliação de Qualidade de Produtos de Software", Universidade Estadual de Londrina, Departamento de Computação, Trabalho de Conclusão de Curso, Londrina, Brasil. Disponível em: <http://www2.dc.uel.br/nourau/document/?view=625>. Acesso em: 15 de novembro de 2010.

SOFTEX, 2008a, "MPS-BR – Lições Aprendidas", Disponível em: [http://www.softex.br/mpsbr/\\_livros/licoes/mpsbr\\_pt.pdf](http://www.softex.br/mpsbr/_livros/licoes/mpsbr_pt.pdf). Acesso em: 18 de agosto de 2010.

SOFTEX, 2008b, "Imps – Resultado de Desempenho das Organizações que Adotaram o Modelo MPS", Disponível em: [http://www.softex.br/mpsbr/\\_livros/imps/imps.pdf](http://www.softex.br/mpsbr/_livros/imps/imps.pdf). Acesso em: 18 de agosto de 2010.

SOFTEX, 2009a, "MPS-BR: Guia Geral v.2009", Disponível em: [http://www.softex.br/mpsbr/\\_guias/guias/MPS-BR\\_Guia\\_Geral\\_2009.pdf](http://www.softex.br/mpsbr/_guias/guias/MPS-BR_Guia_Geral_2009.pdf). Acesso em: 18 de agosto.

SOFTEX, 2009b, "MPS-BR: Guia de Aquisição v.2009", Disponível em: [http://www.softex.br/mpsbr/\\_guias/guias/MPS-BR\\_Guia\\_de\\_Aquisicao\\_2009.pdf](http://www.softex.br/mpsbr/_guias/guias/MPS-BR_Guia_de_Aquisicao_2009.pdf). Acesso em: 18 de agosto.

SOFTEX, 2009c, "MPS-BR: Guia de Avaliação v.2009", Disponível em: [http://www.softex.br/mpsbr/\\_guias/guias/MPSBR\\_Guia\\_de\\_Avaliacao\\_2009.pdf](http://www.softex.br/mpsbr/_guias/guias/MPSBR_Guia_de_Avaliacao_2009.pdf). Acesso em: 18 de agosto.

SOFTEX, 2009d, "MPS-BR: Guia de Implementação – Parte 1 v.2009", Disponível em: [http://www.softex.br/mpsbr/\\_guias/guias/MPS-BR\\_Guia\\_de\\_Implementacao\\_Parte\\_1\\_2009.pdf](http://www.softex.br/mpsbr/_guias/guias/MPS-BR_Guia_de_Implementacao_Parte_1_2009.pdf). Acesso em: 18 de agosto.

SOMMERVILLE, I., 2007, "Engenharia de Software", 8ª Ed., Editora Pearson.

WEBER, K. C., ROCHA, A. R., ALVES, A., AYALA, A. M., GONÇALVES, A., PARET, B., SALVIANO, C., MACHADO, C. F., SCALET, D., PETIT, D., ARAÚJO, E., BARROSO, M. G., OLIVEIRA, K., OLIVEIRA, L. C. A., AMARAL, M. P., CAMPELO, R. E. C., MACIEL, T., 2004, "Modelo de Referência para Melhoria de Processo de Software: uma abordagem brasileira", XXX Conferência Latino Americana de Informática, Arequipa, Peru. Disponível em: [http://pos.facom.ufu.br/~willian/uniube/eng\\_software/arteng2.pdf](http://pos.facom.ufu.br/~willian/uniube/eng_software/arteng2.pdf). Acesso em: 20 de novembro de 2010.

## BIBLIOGRAFIA

NBR ISO/IEC 9126-1, 2003, "Engenharia de Software – Qualidade de produto", Associação Brasileira de Normas Técnicas, Rio de Janeiro, Brasil.

SOFTEX, 2008, "Imps – Resultado de Desempenho das Organizações que Adotaram o Modelo MPS", Disponível em: [http://www.softex.br/mpsbr/\\_livros/imps/imps.pdf](http://www.softex.br/mpsbr/_livros/imps/imps.pdf).

SOFTEX, 2009, "MPS-BR Guia de Avaliação", Disponível em: [http://www.softex.br/mpsbr/\\_guias/guias/MPSBR\\_Guia\\_de\\_Avaliacao\\_2009.pdf](http://www.softex.br/mpsbr/_guias/guias/MPSBR_Guia_de_Avaliacao_2009.pdf).