

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ – UTFPR
CURSO DE TECNOLOGIA EM DESENVOLVIMENTO DE SISTEMAS DE
INFORMAÇÃO

JOSE CARLOS GESING

**MELHORIA DO PROCESSO DE SOFTWARE NO BRASIL:
UTILIZAÇÃO DO MODELO MPS.BR EM EMPRESAS DE PEQUENO PORTE**

TRABALHO DE DIPLOMAÇÃO

MEDIANEIRA

2011

JOSE CARLOS GESING
JOSÉ CARLOS GESING

**MELHORIA DO PROCESSO DE SOFTWARE NO BRASIL:
UTILIZAÇÃO DO MODELO MPS.BR EM EMPRESAS DE PEQUENO PORTE**

Trabalho de Diplomação apresentado à disciplina de Trabalho de Diplomação, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas – CSTADS – da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Cláudio Leones Bazzi

MEDIANEIRA

2011



TERMO DE APROVAÇÃO

MELHORIA DO PROCESSO DE SOFTWARE NO BRASIL: UTILIZAÇÃO DO MODELO MPS.BR EM EMPRESAS DE PEQUENO PORTE

Por

JOSÉ CARLOS GESING

Este Trabalho de Diplomação (TD) foi apresentado às 16:40 h do dia 17 de junho de 2011 como requisito parcial para a obtenção do título de Tecnólogo no Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Universidade Tecnológica Federal do Paraná, *Campus* Medianeira. O candidato foi argüido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Cláudio Leones Bazzi M.Sc.
UTFPR – *Campus* Medianeira
(Orientador)

Prof. Romualdo Rubens de Freitas
UTFPR – *Campus* Medianeira
(Convidado)

Prof. Alan Gavioli
UTFPR – *Campus* Medianeira
(Convidado)

Prof. Juliano Rodrigo Lamb
UTFPR – *Campus* Medianeira
(Responsável pelas atividades de TCC)

A folha de aprovação assinada encontra-se na Coordenação do Curso.

RESUMO

GESING, José Carlos. Melhoria do Processo de Software no Brasil, utilização do modelo MPS.BR em empresas de pequeno porte. Trabalho de conclusão de Curso, Universidade Tecnológica Federal do Paraná. Medianeira, 2011.

Este trabalho apresenta um estudo sobre as normas ISO/IEC 12207 e ISO/IEC 15504, sobre o modelo para a melhoria do processo de software MPS.BR com ênfase no nível G, aplicado em uma empresa de pequeno porte, assim como um breve comparativo ao modelo CMMI (*Capability Maturity Model Integration for Development*).

Palavras-chave: MPS, CMM, CMMI, Processos software.

ABSTRACT

GESING, José Carlos. Brazilian software process improvement, use of the MPS.BR model in small business companies. Course completion assignment, Universidade Tecnológica Federal do Paraná. Medianeira, 2011.

This paper presents a study about the ISO/IEC 12207 and the ISO/IEC 15504, about the software process improvement MPS.BR with emphasis on the G level of maturity, applied to a small business company, as well as a brief comparison to the CMMI (Capability Maturity Model Integration for Development) model.

Keywords: MPS, CMM, CMMI, Process software.

LISTA DE SIGLAS

AP: Atributo de Processo

AQU: Aquisição

CMMI-DEV: Capability Maturity Model Integration Development

CMMI: Capability Maturity Model Integration

CMM: Capability Maturity Model

DEP: Desempenho do Processo Organizacional

DFP: Definição do Processo Organizacional

DRE: Desenvolvimento de Requisitos

EAP: Estrutura Analítica do Projeto

ETM: Equipe Técnica do Modelo

TI: Tecnologia da Informação

GQA: Garantia de Qualidade

GCO: Gerência de Configuração

GPR: Gerência de Projetos

GQP: Gerência Quantitativa do projeto

GRE: Gerência de Requisitos

ISO/IEC: International Organization for Standardization/International Eleetrotechnical Comission

MA-MPS: Modelo de Método de Avaliação para Melhoria de Processo de Software

ODS: Organização desenvolvedora de software

RAP: Resultado do atributo do processo

SEI: Software Engineering Institute

SOFTEX: Associação para Promoção da Excelência do Software Brasileiro

TER: Terminamento

VAL: Validação

LISTA DE FIGURAS

Figura 1: Melhoria de processos	13
Figura 2: ISO/IEC 12207 - Estrutura.....	14
Figura 3: Modelo SPICE - Níveis de capacidade	18
Figura 4: Atributos e níveis de Processo	19
Figura 5: Características das fases do Modelo CMMI.	25
Figura 6: Modelo CMMI-Dev – Níveis e resultados.....	26
Figura 7: Estrutura do Modelo de Referência MR-MPS	33
Figura 8: Comparativo entre as fase dos modelos MPS.BR e CMMI-Dev.....	36
Figura 9: Modelo e fundamentos de Guias MPS.....	37
Figura 10: Avaliações MPS.BR ao longo dos anos (Fevereiro de 2011)	55
Figura 11: Variação de desempenho das organizações com MPS que Revalidaram/Mudaram de Nível.	56
Figura 12: Director – Tela de cadastro de requisição	66
Figura 13: Director – Tela de consulta de requisição	67
Figura 14: Director – Tela de Cadastro de dados de análise.....	68
Figura 15: Director – Tela de cadastro de Atendimento/Suporte	69
Figura 16: Figura Ciclo de vida Iterativo e Incremental.....	71
Figura 17: Ciclo de vida Iterativo e Incremental 2	71
Figura 18: Diagrama do processo de rastreabilidade	79
Figura 19: Processos – Planejamento.....	82
Figura 20: Processos – Construção	84
Figura 21: Processos - Implantação	85
Figura 22: Processos – Finalização	86
Figura 23: Processos - Mudanças.....	90
Figura 24: Processos - Monitoramento	91

SUMÁRIO

1	INTRODUÇÃO.....	9
1.1	OBJETIVO GERAL.....	10
1.2	OBJETIVOS ESPECÍFICOS	10
1.3	JUSTIFICATIVA	10
2	QUALIDADE DE SOFTWARE	12
2.1	ISO / IEC 12207	13
2.2	ISO / IEC 15504	17
3	CMMI (CAPABILITY MATURITY MODEL).....	24
3.1	ESTÁGIOS	24
3.2	PONTOS A SEREM TRABALHADOS PARA A OBTENÇÃO DO CMMI.....	29
3.3	ETAPAS DAS AVALIAÇÕES PARA O CMMI.....	29
4	MPS.BR (MELHORIA DO PROCESSO DE SOFTWARE).....	31
4.1	MODELO MPS	32
4.2	MODELO DE REFERÊNCIA MR-MPS.....	33
4.3	NÍVEL DE MATURIDADE: G – PARCIALMENTE GERENCIADO	37
4.4	ESTATÍSTICAS DO MODELO MPS.BR.....	55
5	IMPLEMENTAÇÃO DO MPS.BR NÍVEL G EM UMA EMPRESA DE PEQUENO PORTE.....	57
5.1	PORQUE O MODELO MPS.BR?	57
5.2	PLANO DE PROJETO (CERTIFICAÇÃO MPS.BR NÍVEL G).....	57
5.3	CRONOGRAMA E EAP	60
5.4	DEFINIÇÃO DE CARGOS E PERFÍS.....	61
5.5	DESENVOLVIMENTO DE GUIAS	63
5.6	CONTROLE DE REQUISIÇÕES E SUPORTE: DIRECTOR	64
5.7	CICLO DE VIDA DOS PROJETOS.....	70
5.8	CONTROLE DE VERSÕES E ATUALIZAÇÕES	73
5.9	REQUISITOS E RASTREABILIDADE	75
5.10	DEFINIÇÃO DOS PRINCIPAIS PROCESSOS (FLUXO DE TRABALHO)	79
6	CONSIDERAÇÕES FINAIS	92
6.1	CONCLUSÃO	92
6.2	TRABALHOS FUTUROS/CONTINUAÇÃO DO TRABALHO	92
7	REFERÊNCIAS BIBLIOGRÁFICAS.....	93

1 INTRODUÇÃO

Obter qualidade nos processos e produtos de engenharia de software não é uma tarefa fácil. São vários os fatores que dificultam atingir os objetivos de qualidade, e quando se trata de empresas de pequeno porte o ponto crucial sempre acaba sendo o lado financeiro, o que acaba por dificultar a melhoria dos processos e conseqüentemente de seus produtos. Geralmente, o que se vê em empresas que estão ingressando no mercado de TI, é que elas procuram focar no produto e não nos processos internos, personalizando seus produtos diretamente para seus primeiros clientes, o que pode vir a causar muitos problemas com uma futura expansão.

Ao falar-se de software e de sua engenharia se está falando ao mesmo tempo de produtos e de processos. De nada adianta centrarmos a atenção só no produto ou só no processo. É necessário que as duas visões caminhem juntas. Portanto, para lidar com qualidade é necessário sabermos que o processo de produção deve ter qualidade e que o produto deve ter qualidade. Entender que o processo de produção de software é composto por diferentes subsistemas. (FREEMAN, 1987)

É importante ressaltar que a escolha de quais métodos serão utilizados no processo de produção é uma escolha gerencial ligada a qualidade, porque a qualidade resultante será função dos métodos de produção escolhidos e seguidos. As tecnologias de produção de software, nas quais estão incluídas as tecnologias de gerência, tem por objetivo automatizar ao máximo a produção de software, mas ainda são fundamentalmente dependentes da qualidade das equipes. (WEBER, 2008)

Na procura de padronizações de processos e da melhoria da qualidade do *software* nacional, o MPS.BR (Melhoria de Processos do *Software* Brasileiro) foi criado para o perfil de empresas com diferentes tamanhos e características, públicas e privadas, mas dando atenção especial às micro, pequenas e médias empresas.

O modelo MPS.BR baseia-se nas normas ISO/IEC 12207 e ISO/IEC 15504 e no modelo CMMI-Dev sendo compatível com os padrões de qualidade aceitos internacionalmente sendo que em seu desenvolvimento foi envolvida toda a competência existente nos padrões e modelos de melhoria de processo já disponíveis. Dessa forma, ele tem

como base os requisitos de processos definidos nos modelos de melhoria de processo e atende a necessidade de implantar os princípios de engenharia de *software* de forma adequada ao contexto das empresas, concordando com as principais abordagens internacionais para definição, avaliação e melhoria de processos de *software*.

1.1 OBJETIVO GERAL

Um estudo sobre modelo para melhoria do processo de software MPS.BR e um breve comparativo com o modelo CMMI-Dev (*Capability Maturity Model Integration for Development*), assim como o acompanhamento da implantação dos processos MPS.BR nível G em uma empresa de pequeno porte, tendo como objetivo identificar e relatar a mudança de processos internos.

Este trabalho não tem como objetivo detalhar sobre prazos e custos da implantação e da avaliação do modelo MPS.BR.

1.2 OBJETIVOS ESPECÍFICOS

- Estudo das normas ISO/IEC 12207 e ISO/IEC 15504;
- Estudo do modelo CMMI-Dev;
- Estudo do modelo MPS.BR e seus níveis;
- Estudo e apresentação do modelo MPS.BR nível G;
- Apresentar resultados obtidos com a implementação do MPS.BR nível G em uma empresa de pequeno porte;

1.3 JUSTIFICATIVA

As mudanças que estão ocorrendo no mercado de tecnologia têm motivado as empresas a reavaliar suas estruturas organizacionais e processos de produção, saindo cada vez mais dos modelos tradicionais baseados em tarefas e da definição de processos centrados no cliente. Tendo como ponto de vista a competitividade, as empresas dependem cada vez mais

de sua capacidade de organização, produtividade e principalmente da qualidade de seu produto.

Para o mercado atual torna-se indispensável uma revisão contínua de serviços e processos, qualidade é fator crítico de sucesso e para que se tenha um setor de *software* competitivo é essencial que os empreendedores do setor coloquem a eficiência e a eficácia dos seus processos em foco nas empresas, visando à oferta de produtos de *software* e serviços, conforme padrões internacionais de qualidade.

Após algum tempo trabalhando com desenvolvimento de software em uma empresa que começa a almejar números mais altos, ficou visível a necessidade de mudanças e padronizações nos processos internos. A princípio, com poucos clientes e um produto ainda em fase de amadurecimento, os esforços eram voltados para o próprio produto e em atender os clientes diretamente da maneira mais rápida possível, sem utilizar processos e metodologias de controle, o que, com o passar do tempo veio a gerar retrabalho impedindo que a empresa usasse sua estrutura para produzir mais e atender novos clientes, ou seja, a empresa chegou rapidamente em um ponto em que não poderia expandir sem rever seus processos internos.

Dessa maneira surge o interesse pelo modelo MPS.BR, não apenas para garantir a qualidade e valorizar seus produtos com a certificação, mas para organizar-se de maneira em que não tenha retrabalhos e esteja preparada para futuras expansões, visando a diminuição de custos e para que possa aproveitar melhor o tempo e o potencial de seus funcionários com inovações.

2 QUALIDADE DE SOFTWARE

Um dos problemas com que se defronta a engenharia de software é a dificuldade de se medir a qualidade de software. A qualidade de um dispositivo mecânico é freqüentemente medida em termos de tempo médio entre suas falhas, que é uma medida da capacidade de o dispositivo suportar desgaste. O software não se desgasta, portanto tal método de medição de qualidade não pode ser aproveitado. (GOMES, 2000)

Os desenvolvedores de software não podem esquecer das necessidades de seus clientes, tendo em vista que eles podem apresentar diferentes necessidades em relação ao mesmo tipo de produto. Os usuários estão interessados no uso do software, em seu desempenho e nos benefícios que o seu uso possa trazer para a organização, não querem conhecer aspectos internos do software ou como o software foi desenvolvido.

Mesmo sendo possível afirmar que o desenvolvimento foi feito com qualidade, o usuário procura resposta para questões como: As funções requeridas estão disponíveis e são executadas eficientemente? O software é seguro, evita que pessoas ou sistemas não autorizados tenham acesso aos dados para leitura ou modificação? Permite que pessoas ou sistemas autorizados para acessar os dados não tenham acesso negado a eles? É fácil de integrar com outros sistemas existentes? Aceita trabalhar com arquivos de outros sistemas ou enviar dados para outros sistemas? É capaz de restabelecer seu nível de desempenho e recuperar dados afetados em casos de falha? É fácil de usar ou requer muito treinamento? O suporte técnico é confiável e atende com a rapidez necessária?

Atingir um alto nível de qualidade de produto ou serviço é o objetivo da maioria das organizações. Atualmente não é mais aceitável entregar produtos com baixa qualidade e reparar os problemas e as deficiências depois que os produtos foram entregues ao cliente. (SOMMERVILLE, 2003).

Segundo Machado (2001), para muitos engenheiros de software, a qualidade do processo de software é tão importante quanto à qualidade do produto. Assim na década de 1990 houve uma grande preocupação com a modelagem e melhorias no processo de software.

Para ajudar nessa questão a *International Organization for Standardization* – ISO e a *International Electrotechnical Commission* – IEC, que são organismos normalizadores com importância internacionalmente reconhecida no setor de software, se uniram para editar normas internacionais conjuntas.

Abordagens importantes como as normas ISO / IEC 12207 e a ISO / IEC 15504, assim como modelo CMM (*Capability Maturity Model*) sugerem que melhorando o processo de software, podemos melhorar a qualidade dos produtos. A Figura 1 demonstra o ciclo de avaliação e melhoria de processos.

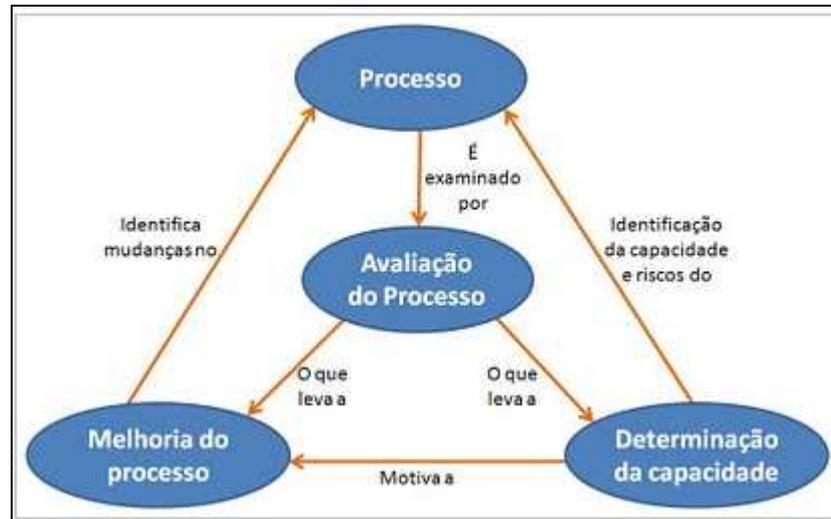


Figura 1: Melhoria de processos

Fonte: <http://balancedscorecard.blogspot.com/2008/06/abordagem-por-processos-da-norma-spice.html>

2.1 ISO / IEC 12207

A norma ISO/IEC 12207 foi criada em 1995 e tem como objetivo principal estabelecer uma estrutura comum para os processos de ciclo de vida e de desenvolvimento de softwares visando ajudar as organizações a compreenderem todos os componentes presentes na aquisição e fornecimento de software e, assim, conseguirem firmar contratos e executarem projetos de forma mais eficaz. Estabelece uma arquitetura de alto nível do ciclo de vida de software que é construída a partir de um conjunto de processos e seus inter-relacionamentos. Os processos são descritos tanto em nível de propósito/saídas como em termos de atividades. (MACHADO, 2006).

A ISO/IEC 12207 não possui nenhuma ligação com métodos, ferramentas, treinamentos, métricas ou tecnologias empregadas. Esta determinação é útil para permitir que a norma seja utilizada mundialmente e possa acompanhar a evolução da engenharia de

software nas diversas culturas organizacionais. Ela pode ser utilizada com qualquer modelo de ciclo de vida, método ou técnica de engenharia de software e linguagem de programação. Sua flexibilidade é uma característica importante, as atividades e tarefas do processo de ciclo de vida do software especificam "o que fazer" e não "como fazer".

2.1.1 ESTRUTURA

Os processos são agrupados, por uma questão de organização, de acordo com a sua natureza, ou seja, o seu objetivo principal no ciclo de vida de software. Esse agrupamento resultou em quatro diferentes classes de processos identificados na Figura 2 e listados a seguir.

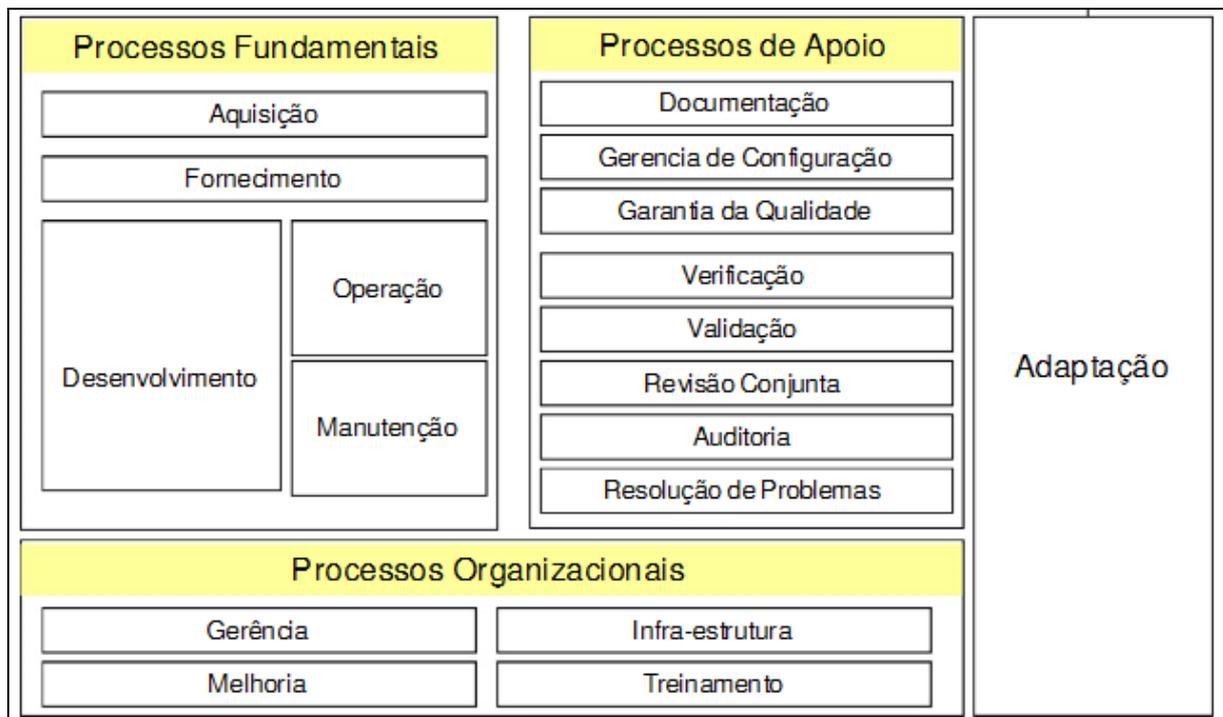


Figura 2: ISO/IEC 12207 - Estrutura

Fonte: (FRANCO, A.L) <http://pt.scribd.com/doc/12714065/ISOIEC-12207-Visao-Geral>

2.1.1.1 PROCESSOS FUNDAMENTAIS

Os processos fundamentais são necessários para que um software seja executado. Eles iniciam o ciclo de vida e comandam outros processos. São eles:

- Aquisição: possui o propósito de obter o produto e/ou serviço que satisfaça suas necessidades;
- Fornecimento: possui o propósito de prover um produto e/ou serviço;
- Desenvolvimento: possui o propósito de transformar um conjunto de requisitos em um produto ou sistema de software;
- Operação: possui o propósito de operar o produto no seu ambiente e prover suporte aos usuários;
- Manutenção: possui o propósito de modificar o produto de software e depois dar liberação para o uso.

2.1.1.2 PROCESSOS DE APOIO

Os processos de apoio auxiliam outro processo. Eles são usados para garantir a qualidade, mas não são fundamentais. São eles:

- Documentação: possui o propósito de prover, manter um registro de informações de software;
- Gerência de configuração: possui o propósito de estabelecer e manter a integridade de todos os produtos de trabalho (artefato) de um processo do projeto;
- Garantia da qualidade: possui o propósito de prover garantia de que os produtos e processos estão em conformidade com o requisitos (padrões/normas) pré-definidos;
- Verificação: possui o propósito de confirmar que os produtos e/ou serviços refletem os requisitos especificados;
- Validação: possui o propósito de confirmar que os requisitos para o uso específico de um produto e/ou serviço são atendidos;
- Revisão conjunta: possui o propósito de manter o entendimento (gerencial comum com os stakeholders);
- Auditoria: possui o propósito de determinar independentemente a conformidade dos produtos e processos contra os requisitos definidos;

- Resolução de problema: possui o propósito de assegurar que todos os problemas levantados sejam analisados e resolvidos;
- Usabilidade;
- Contrato.

2.1.1.3 PROCESSOS ORGANIZACIONAIS

Os processos organizacionais auxiliam a organização e gerência geral dos processos servindo para toda a organização. São eles:

- Gerência: possui o propósito de organizar, monitorar e controlar a iniciação e o desempenho dos processos;
- Infra-estrutura: possui o propósito de manter uma infra-estrutura estável e confiável;
- Melhoria;
- Recursos humanos: possui o propósito de prover e manter recursos humanos adequados mantendo as suas capacitações consistentes com o negócio;
- Gestão de ativos: possui o propósito de gerenciar a vida dos ativos (reusáveis) desde a concepção até a desativação;
- Gestão de programa de reuso: possui o propósito de planejar, estabelecer, controlar, monitorar os programas de reuso;
- Engenharia de domínio: possui o propósito de desenvolver e manter modelos de domínio, arquiteturas e ativos deste domínio.

2.1.1.4 PROCESSOS DE ADAPTAÇÃO

Os processos são adaptáveis a:

- Projeto;
- Organização;
- Cultura;
- Modelo de ciclo de vida, métodos e técnicas, e linguagens.

2.2 ISO / IEC 15504

Teve início de seu desenvolvimento em 1993 pela ISO em conjunto com a comunidade internacional através do projeto SPICE (*Software Process Improvement and Capability Determination*) com base nos modelos já existentes como ISO 9000 e CMM, sendo oficialmente publicada pela ISO em outubro de 2003.

Trata-se da evolução da ISO/IEC 12207 possuindo níveis de capacidade para cada processo, como demonstrado na Figura 3. Define um modelo bi-dimensional que tem por objetivo a realização de avaliações de processos de software com o foco da melhoria dos processos, gerando um perfil dos processos, identificando os pontos fracos e fortes, que serão utilizados para a elaboração de um plano de melhorias.

A ISO/IEC 15504 define também um guia para a orientação da melhoria de processo, tendo como referência um modelo de processo e como uma das etapas a realização de uma avaliação de processo. Este guia sugere oito etapas sequenciais, que inicia com a identificação de estímulos para a melhoria e o exame das necessidades da organização.

Em seguida existem ciclos de melhoria, nos quais um conjunto de melhoria são identificadas, uma avaliação das práticas correntes em relação à melhoria é realizada, um planejamento da melhoria é feito, seguido pela implementação, confirmação, manutenção e acompanhamento da melhoria.

2.2.1 CATEGORIAS E PROCESSOS

A norma define detalhes de cada um dos processos. Para cada um deles existe uma definição mais detalhada, uma lista dos resultados da sua implementação bem sucedida e uma descrição detalhada de cada uma das práticas básicas.

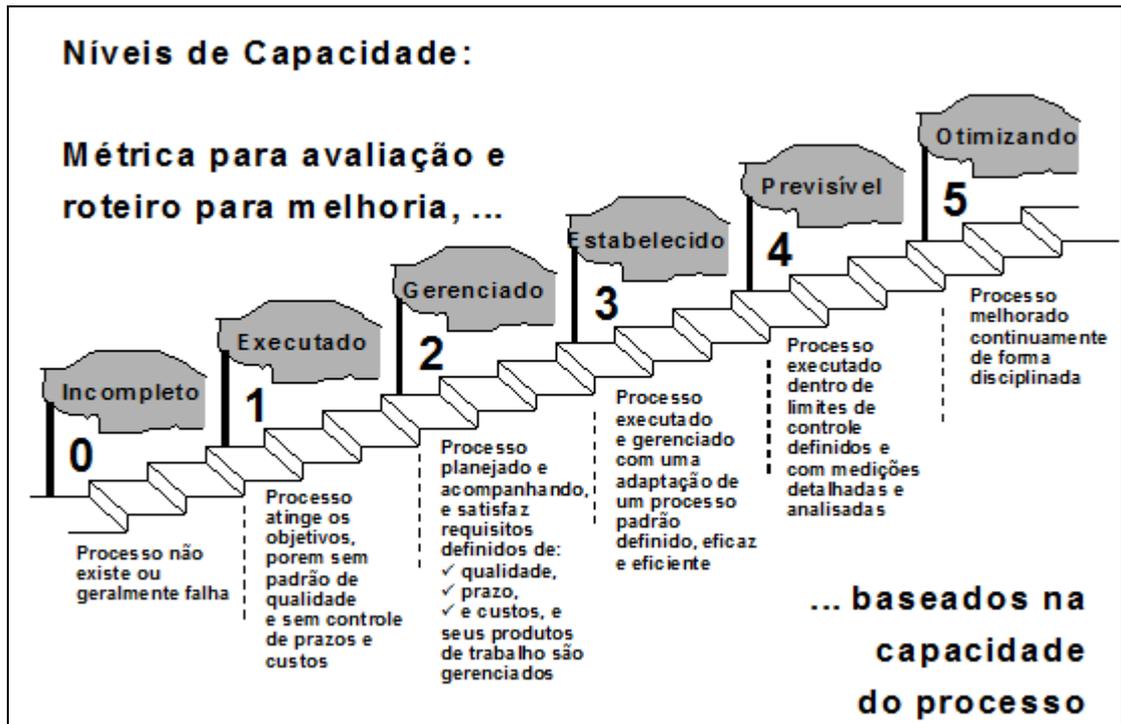


Figura 3: Modelo SPICE - Níveis de capacidade

2.2.2 MEDIDA DA CAPABILIDADE DE UM PROCESSO

Baseada em um conjunto de atributos de processo (Process Attributes — PA), ilustrados na Figura 4, são usados para se determinar se um processo alcançou um determinado nível.

- Cada atributo mede um aspecto particular da capacidade de um processo;
- Cada nível tem associado um conjunto de atributos de processo que devem ser atendidos;
- O atendimento aos atributos pode ser medido em uma escala percentual, fornece uma visão mais detalhada de aspectos específicos da capacidade de um processo;

Atributos de Processo	Níveis de Capabilidade Nomes dos Atributos de Processo
	Nível 0: Processo Incompleto
	Nível 1: Processo Executado
PA 1.1	atributo de execução de processo
	Nível 2: Processo Gerenciado
PA 2.1	atributo de gestão de execução
PA 2.2	atributo de gestão de produto de trabalho
	Nível 3: Processo Estabelecido
PA 3.1	atributo de definição de processo
PA 3.2	atributo de recursos de processo
	Nível 4: Processo Estabelecido
PA 4.1	atributo de definição de processo
PA 4.2	atributo de recursos de processo
	Nível 5: Processo em Melhoria
PA 5.1	atributo de mudança de processo
PA 5.2	atributo de melhoria contínua

Figura 4: Atributos e níveis de Processo

2.2.2.1 NÍVEL 0: PROCESSO INCOMPLETO

Há uma falha geral em realizar o objetivo do processo. Não existem produtos de trabalho nem saídas do processo facilmente identificáveis.

- O processo não é implementado ou não consegue gerar os produtos de trabalho Esperados;
- Existe pouca ou nenhuma evidência de qualquer tipo de sucesso sistemático;
- Não há atributos de processo neste nível;

2.2.2.2 NÍVEL 1: PROCESSO EXECUTADO (REALIZADO)

O objetivo do processo em geral é atingido, embora não necessariamente de forma planejada e controlada. Há um consenso na organização de que as ações devem ser realizadas e quando são necessárias. Existem produtos de trabalho para o processo e eles são utilizados para atestar o atendimento dos objetivos.

- O processo consegue alcançar os objetivos de alguma maneira e gerar os produtos de trabalhos esperados;
- Há um atributo neste nível:
 - PA 1.1: Atributo de execução de processo:
- O processo consegue transformar produtos de trabalho de entrada em produtos de trabalho de saída e os resultados esperados do processo são alcançados;

2.2.3 NÍVEL 2: PROCESSO GERENCIADO

O processo produz os produtos de trabalho com qualidade aceitável e dentro do prazo. Isto é feito de forma planejada e controlada. Os produtos de trabalho estão de acordo com padrões e requisitos.

- O processo, além de executado, é feito de maneira gerenciada: planejado, controlado, acompanhado, verificado e corrigido;
- Há dois atributos neste nível:
 - PA 2.1: Atributo de execução de processo:
 - Este atributo mede até que ponto o processo é gerenciado para produzir os produtos de trabalho que satisfazem aos seus objetivos;
 - Os objetivos do processo em termos de qualidade, prazo e uso de recursos estão identificados
 - A atribuição de responsabilidades pelos produtos de trabalho está feita;
 - A execução do processo é gerenciada;
 - PA 2.2: Atributo de gestão dos produtos de trabalho:
 - Este atributo mede até que ponto os produtos de trabalho são documentados, controlados e verificados;
 - Os requisitos dos produtos de trabalho estão documentados;
 - Os requisitos para a documentação e controle dos produtos de trabalho estão definidos;
 - As dependências entre produtos de trabalho estão definidas;

- Os produtos de trabalho têm as suas mudanças controladas e são distribuídos em baselines;
- Os produtos são verificados;

2.2.3.1 NÍVEL 3: PROCESSO ESTABELECIDO

O processo é realizado e gerenciado usando um processo definido, baseado em princípios de Engenharia de Software. As pessoas que implementam o processo usam processos aprovados, que são versões adaptadas do processo padrão documentado.

- O processo agora, além de executado e gerenciado é definido com base em princípios de engenharia de software;
- Há dois atributos neste nível:
 - PA 3.1: Atributo de definição de processo:
 - Este atributo mede até que ponto o processo é definido com base em um processo padronizado;
 - Existe um processo padronizado na organização a partir do qual adaptações podem ser feitas;
 - O processo é executado de acordo com um processo padronizado ou adaptado de acordo com orientações para adaptações;
 - Dados históricos do processo são coletados para auxiliar na compreensão do comportamento estatístico do processo;
 - A experiência acumulada na utilização do processo definido é usada para refiná-lo;
 - PA 3.2: Atributo de recursos de processo:
 - Este atributo mede até que ponto o processo faz uso de recursos humanos e materiais para ser executado de maneira bem sucedida;
 - Os papéis, responsabilidades e competências necessárias para a execução do processo são identificadas e documentadas;
 - A infra-estrutura necessária para a execução do processo é identificada e documentada;
 - Os recursos necessários são alocados e utilizados na execução do processo;

2.2.3.2 NÍVEL 4: PROCESSO PREVISÍVEL

O processo é realizado de forma consistente, dentro dos limites de controle, para atingir os objetivos. Medidas da realização do processo são coletadas e analisadas. Isto leva a um entendimento quantitativo da capacitação do processo a uma habilidade de prever a realização.

- O processo agora, além de executado, gerenciado e definido é executado dentro de limites quantitativos bem definidos;

- Há dois atributos neste nível:

- PA 4.1: Atributo de medida:

- Este atributo mede até que ponto métricas e objetivos de processo/produto são usados para assegurar que a execução do processo é efetiva no sentido de alcançar os objetivos de negócio da empresa;

- Métricas são identificadas e coletadas;

- As tendências observadas são analisadas;

- A capacidade de processo é medida;

- PA 4.2: Atributo de controle de processo:

- Este atributo mede até que ponto o processo é controlado por intermédio da coleta, análise e uso de medidas para servir de base para ações corretivas, quando necessário;

- Técnicas de medida são estabelecidas;

- Características de produto e processo são medidas e usadas como insumo para permitir o controle do processo dentro de limites de variabilidade;

- O processo é gerenciado de forma quantitativa;

2.2.3.3 NÍVEL 5: PROCESSO EM OTIMIZAÇÃO

A realização do processo é otimizada para atender às necessidades atuais e futuras do negócio. O processo atinge seus objetivos de negócio e consegue ser repetido. São estabelecidos objetivos quantitativos de eficácia e eficiência para o processo, segundo os

objetivos da organização. A monitoração constante do processo segundo estes objetivos é conseguida obtendo feedback quantitativo e o melhoramento é conseguido pela análise dos resultados. A otimização do processo envolve o uso piloto de idéias e tecnologias inovadoras, além da mudança de processos ineficientes para atingir os objetivos definidos.

- O processo agora, além de executado, gerenciado, definido e executado dentro de limites quantitativos pode ser mudado e evoluído de maneira dinâmica e sob controle;

- Há dois atributos neste nível:

- PA 5.1: Atributo de mudança de processo:

- Este atributo mede até que ponto mudanças na definição, gerência e execução do processo são controladas;

- O impacto de mudanças é avaliado contra os objetivos do processo e de seus produtos;

- A implementação de mudanças é feita de modo a permitir que as perturbações sejam controladas e prevenidas;

- A eficácia das mudanças é medida contra o planejado;

- PA 5.2: Atributo de melhoria contínua:

- Este atributo mede até que ponto as mudanças de processo contribuem para a melhoria contínua;

- As metas de melhoria de processo são estabelecidas à luz dos objetivos de negócio;

- As possíveis fontes de risco e problemas são identificadas, assim como as possíveis oportunidades de melhoria;

- Uma estratégia de melhoria contínua é estabelecida;

3 CMMI (*CAPABILITY MATURITY MODEL*)

É uma metodologia criada pela SEI (Software Engineering Institute) para ser um guia destinado a melhorar os processos organizacionais e a habilidade desses em gerenciar o desenvolvimento, a aquisição e a manutenção de produtos e serviços. O CMMI organiza as práticas, que já são consideradas efetivas, em uma estrutura que visa auxiliar a organização a estabelecer prioridades para melhoria e também fornece um guia para a implementação dessas melhorias. (CMU/SEI, 2006).

O primeiro passo a ser dado é a identificação – através de um método definido pelo SEI (SCAMPI – SEI Members of the Assessment Method Integrated Team, 2001) e conduzido por um avaliador credenciado – do estágio em que a empresa se encontra no presente; uma vez que este denota um nível de maturidade a ser alcançado pelas empresas, visando ajudá-las no desenvolvimento e manutenção dos projetos de software, como também melhorar a capacidade de seus processos.

Após a verificação do estágio da empresa, verifica-se qual a próxima etapa a ser alcançada e quais as competências que devem ser adquiridas neste processo. Esta fase é importante, pois permite alcançar o sucesso e, conseqüentemente, melhoria na qualidade dos serviços e produtos fornecidos pela área de tecnologia da Empresa.

3.1 ESTÁGIOS

O CMMI está dividido em cinco estágios conforme as Figura 6 e 7:

- Realização – Estágio inicial;
- Gerenciado – Gerenciamento de requisitos, planejamento de projeto, monitoramento e controle de projeto, gerenciamento de fornecedores, medição e análise, garantia da qualidade do processo e do produto, gerenciamento de configuração;
- Definido – Desenvolvimento de requisitos, solução técnica, integração do produto, verificação e validação, foco no processo organizacional, definição do

processo organizacional, treinamento organizacional, gerenciamento de riscos, gerenciamento integrado do projeto, análise da decisão e resolução;

- Quantitativamente – Gerenciamento quantitativo do projeto, performance do processo organizacional;
- Otimização – Análise causal e resolução, inovação organizacional e implantação.

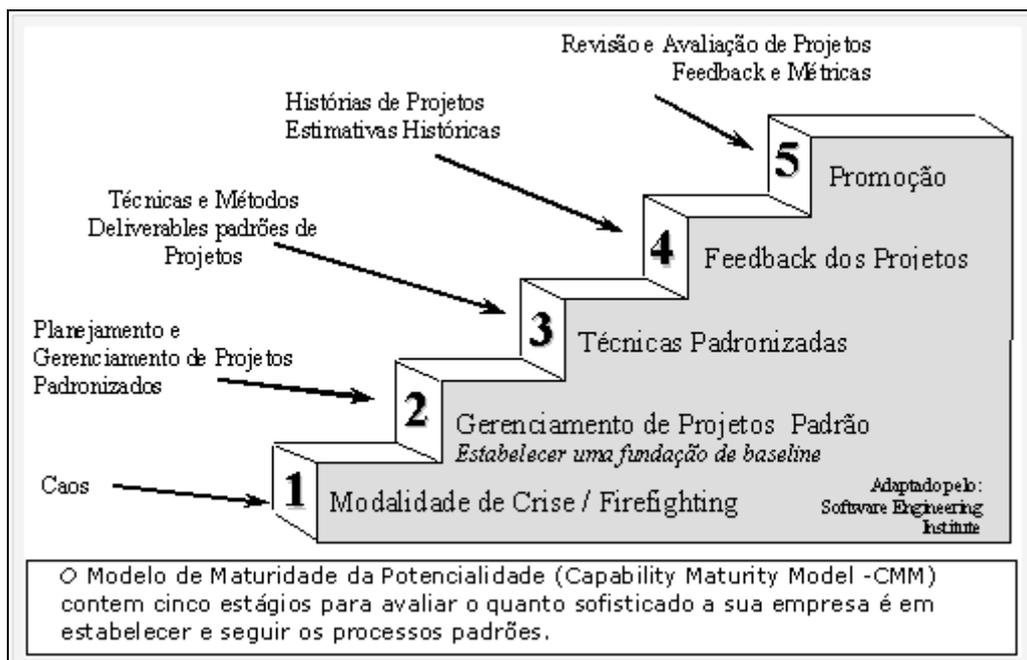


Figura 5: Características das fases do Modelo CMMI.

Fonte: <http://www.dcc.unicamp.br/~cortes/mc726/cap5a.pdf>

	Level	Capability	Result
5	Optimizing Continuous Process Improvement	Organizational Innovation & Deployment Causal Analysis & Resolution	Productivity & Quality
4	Quantitatively Managed Quantitative Management	Quantitative Process Management Software Quality Management	
3	Defined Process Standardization	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Product Management Risk Management Integrated Teaming Integrated Supplier Management Decision Analysis & Resolution Organizational Environment for Integration	
2	Managed Basic Project Management	Requirements Management Project Planning Project Monitoring & Control Supplier Agreement Management Measurement & Analysis Product & Process Quality Assurance Configuration Management	
1	Initial Heroic Efforts	Design Develop Integrate Test	
			Risk & Waste

Figura 6: Modelo CMMI-Dev – Níveis e resultados

Fonte: <http://www.thoughtcapital.us/OperationalExcellence/CMMi.html>

A seguir seguem os detalhes de cada estágio e suas respectivas competências.

3.1.1 REALIZADO

Estágio inicial, completa falta de planejamento e controle dos processos. Os funcionários estão focados basicamente em atividades corretivas que surgem a todo o momento. Nenhum processo está definido.

3.1.2 GERÊNCIADO

São estabelecidos processos básicos de gerenciamento de projeto para planejar e acompanhar custos, prazos e funcionalidades. Compromissos são firmados e gerenciados. A disciplina de processo permite repetir sucessos de projetos anteriores em aplicações similares. Tipicamente, possui gerenciamento de projetos estabelecido; alguns procedimentos técnicos escritos; acompanhamento de qualidade; gerência de configuração inicial; atividades básicas de medição e análise. O sucesso depende basicamente do gerenciamento do projeto.

- Gerenciamento de Requisitos - REQM (*Requirements Management*);
- Planejamento de Projeto - PP (*Project Planning*);
- Acompanhamento e Controle de Projeto - PMC (*Project Monitoring and Control*);
- Gerenciamento de Acordo com Fornecedor - SAM (*Supplier Agreement Management*);
- Medição e Análise - MA (*Measurement and Analysis*);
- Garantia da Qualidade de Processo e Produto - PPQA (*Process and Product Quality Assurance*);
- Gerência de Configuração - CM (*Configuration Management*);

3.1.3 DEFINIDO

Atividades de gerenciamento básico e as de Engenharia de Software são documentadas, padronizadas e integradas em processos-padrão. Todos os projetos de desenvolvimento ou manutenção de softwares utilizam uma versão de um desses processos adaptada às características específicas de cada projeto. Possui processos gerências e técnicos bem definidos, possibilidade de avaliação do processo; ferramentas e metodologias padronizadas; medições iniciais de desempenho; inspeções e auditorias rotineiras; testes padronizados; gerência de configuração; evolução controlada dos processos técnicos e gerenciais.

- Desenvolvimento de Requisitos - RD (*Requirements Development*);
- Solução Técnica - TS (*Technical Solution*);

- Integração de Produto - PI (*Product Integration*);
- Verificação - VER (*Verification*);
- Validação - VAL (*Validation*);
- Foco de Processo Organizacional - OPF (*Organizational Process Focus*);
- Definição de Processo Organizacional - OPD (*Organizational Process Definition*);
- Treinamento Organizacional - OT (*Organizational Training*);
- Gerenciamento Integrado de Projeto - IPM (*Integrated Project Management*);
- Gerenciamento de Riscos - RSKM (*Risk Management*);
- Análise de Decisão e Resolução - DAR (*Decision Analysis and Resolution*);

3.1.4 QUANTITATIVAMENTE GERENCIADO

Métricas detalhadas do processo de software e da qualidade do produto são coletadas. Tanto o processo como o produto de software são quantitativamente compreendidos, avaliados e controlados. Relatórios estatísticos são gerados. Tipicamente, encontra-se estabelecido e em uso rotineiro um programa de medições, a qualidade é planejada por um grupo dedicado, sendo rotineiramente avaliada e aprimorada.

- Desempenho de Processo Organizacional - OPP (*Organizational Process Performance*)
- Gerenciamento Quantitativo de Projeto - QPM (*Quantitative Project Management*)

3.1.5 OTIMIZAÇÃO

A melhoria contínua do processo é estabelecida por meio de sua avaliação quantitativa e da implantação planejada e controlada de tecnologias e idéias inovadoras. Projetos-piloto são realizados para a absorção e internalização de novas tecnologias. Tipicamente, um alto nível de qualidade e de satisfação dos clientes é alcançado rotineiramente, com grande foco na melhoria contínua.

Nesta representação a maturidade é medida por um conjunto de processos. Assim é necessário que todos os processos atinjam nível de maturidade dois para que a empresa seja certificada com nível dois. Se quase todos os processos forem nível três, mas apenas um deles estiver no nível dois a empresa não irá conseguir obter o nível de maturidade três.

Esta representação é indicada quando a empresa já utiliza algum modelo de maturidade por estágios, quando deseja utilizar o nível de maturidade alcançado para comparação com outras empresas ou quando pretende usar o nível de conhecimento obtido por outros para sua área de atuação.

- Inovação Organizacional e Implantação - OID (*Organizational Innovation and Deployment*);
- Análise Causal e Resolução - CAR (*Causal Analysis and Resolution*);

3.2 PONTOS A SEREM TRABALHADOS PARA A OBTENÇÃO DO CMMI

No nível 2 é preciso trabalhar pontos como Gestão de Requisitos, de Acordo com Fornecedores e de Configuração, Planejamento e Monitoramento de Projetos, Medição e Análise.

No nível 3, os quesitos a serem trabalhados são, entre outros, Solução Técnica, Integração do Produto, Verificação e Validação, Definição de Processos Organizacionais, Gestão de Riscos, Análise de Decisões e Resolução.

Já nos níveis 4 e 5, respectivamente, deve-se trabalhar pontos como Performance de Processos Organizacionais e Gestão Quantitativa de Projetos, Inovação e Análise de Causas e Resolução.

3.3 ETAPAS DAS AVALIAÇÕES PARA O CMMI

A primeira etapa de avaliação para o CMMI é o treinamento da equipe de avaliação, que poderá ser composta somente por profissionais da consultoria ou da consultoria e dos clientes.

A segunda etapa é o planejamento da avaliação, onde diversos aspectos são contemplados, como logística e estabelecimento de expectativas.

A terceira etapa é a execução da avaliação, quando o diagnóstico propriamente dito é realizado. Tipicamente ocorrem de cinco a dez dias consecutivos de visitas ao cliente, quando o planejamento da avaliação é posto em prática. Entrevistas, revisão de documentos e atividades de consenso da equipe de avaliação são realizadas nesta etapa.

Por último vem a reportagem dos resultados, que ocorre no último dia de avaliação, em uma sessão onde comparecem todos os participantes e eventuais convidados. Ela é seguida por uma sessão executiva, onde são discutidos os principais aspectos levantados durante a avaliação.

4 MPS.BR (MELHORIA DO PROCESSO DE SOFTWARE)

Entre os principais problemas que inibem organizações, especialmente PMEs, de adotar modelos de processos de software, tais como CMMI, ISO/IEC 12207 e ISO/IEC 15504, estão o alto investimento (de implementação, manutenção e avaliação) e a dificuldade em convencer as organizações dos potenciais benefícios de investimentos em melhoria de processos.

Para superar estes problemas, são necessárias abordagens diferentes para mudar o cenário de melhoria de processos em organizações de software, especialmente em PMEs. Neste contexto, a SOFTEX decidiu realizar uma iniciativa nacional, denominada Programa MPS.BR, para melhorar os processos de software de organizações brasileiras, e, como consequência, disseminar boas práticas da engenharia de software e tornar as organizações mais competitivas no mercado local e global. Para alcançar este objetivo, foi importante obter consenso dos interesses da indústria de software nacional e documentá-lo na forma de um modelo de processos de software que represente os interesses da comunidade brasileira de software.

O principal objetivo do programa MPS.BR é desenvolver e disseminar um modelo de processos de software (o modelo MPS) visando estabelecer um caminho economicamente viável para que organizações, incluindo as PMEs, alcancem os benefícios da melhoria de processos e da utilização de boas práticas da engenharia de software em um intervalo de tempo razoável. Embora o foco da iniciativa seja PMEs, o modelo é adequado também para apoiar a melhoria de processos em grandes organizações. O programa MPS.BR teve início em dezembro de 2003, é coordenado pela SOFTEX e patrocinado pelo Ministério de Ciência e Tecnologia (MCT), pela Financiadora de Estudos e Projetos (FINEP), pelo Serviço Brasileiro de Apoio às Micro e Pequenas Organizações (SEBRAE) e pelo Banco Interamericano de Desenvolvimento (BID). (SOFTEX, 2009).

Para gerenciar o programa MPS.BR, uma estrutura organizacional foi definida e responsabilidades foram atribuídas a pesquisadores e profissionais de engenharia de software. As unidades desta estrutura organizacional são as seguintes:

- Unidade de Execução do Programa MPS.BR: responsável por definir estratégias e gerenciar as atividades do programa. Esta equipe é coordenada, atualmente, pela SOFTEX;
- Equipe Técnica do Modelo MPS: responsável pela criação e aprimoramento contínuo do modelo MPS; e capacitação de pessoas por meio de cursos, provas e workshops do MPS.BR. Esta equipe é coordenada pela COPPE/UFRJ;
- Fórum de Credenciamento e Controle do MPS: responsável por emitir parecer que subsidie a decisão da Sociedade SOFTEX sobre o credenciamento de Instituições Implementadoras e Instituições Avaliadoras (IA); e avaliar e controlar resultados de implementações e avaliações MPS; e assegurar que organizações avaliadas segundo o modelo MPS realizam suas atividades dentro dos limites éticos e de usabilidade esperados. Esta equipe é composta por representantes do governo brasileiro, da indústria e da academia;

4.1 MODELO MPS

Um dos requisitos para desenvolver o modelo MPS é que ele deveria incorporar tanto práticas internacionalmente reconhecidas para implementação e avaliação de processos de engenharia de software quanto atender às necessidades de negócio da indústria de software brasileira. Assim, as normas ISO/IEC 12207 e ISO/IEC 15504 foram usadas como base técnica para definir os componentes do modelo MPS. Adicionalmente, tendo em vista a importância do modelo CMMI para organizações brasileiras que atuam em mercados internacionais, a equipe técnica do modelo MPS considerou o CMMI como um complemento técnico para a definição dos processos do modelo MPS. O modelo MPS possui três componentes principais: Modelo de referência MPS (MR-MPS); Método de avaliação MPS (MA-MPS); Modelo de negócios MPS (MN-MPS). (SOFTEX, 2009).

A discussão do método de avaliação e do modelo de negócios está fora do escopo deste trabalho. A seguir será fornecida uma visão geral do modelo de referência, que contempla os processos de engenharia de software.

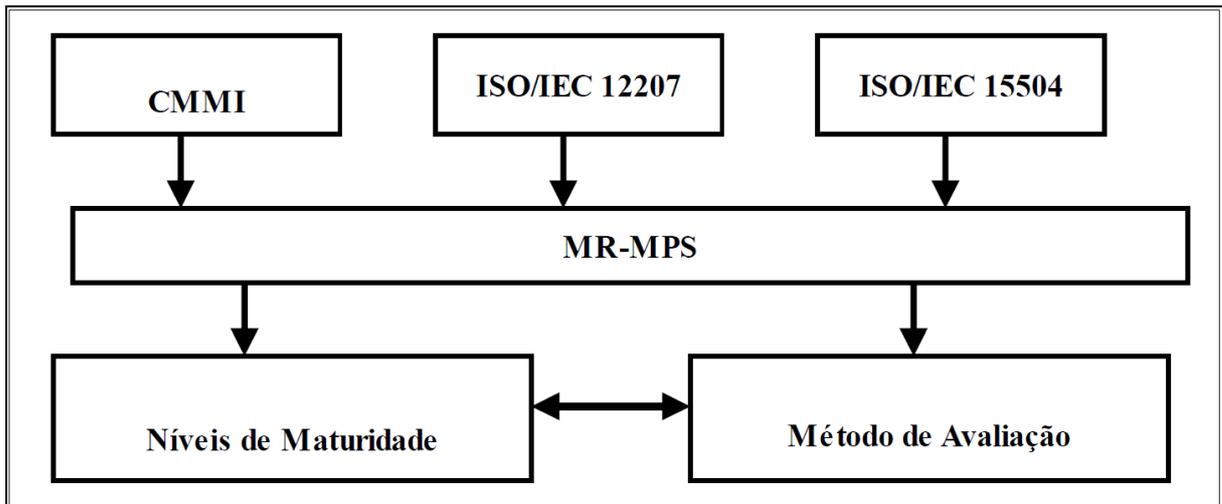


Figura 7: Estrutura do Modelo de Referência MR-MPS

Fonte: (SOFTEX 2009)

4.2 MODELO DE REFERÊNCIA MR-MPS

O modelo de referência MPS (MR-MPS) é documentado no Guia Geral do MPS, disponível em português e espanhol no site da SOFTEX. O Guia Geral do MPS provê uma definição geral do modelo MPS. O MR-MPS está em conformidade com a norma ISO/IEC 15504, satisfazendo os requisitos para modelo de referência de processos definidos na ISO/IEC 15504. Cada processo definido no MR-MPS tem um conjunto de resultados necessário e suficiente para alcançar o propósito do processo.

Os processos do MR-MPS são uma adaptação dos processos da norma ISO/IEC 12207 e suas emendas e são, adicionalmente, compatíveis com as áreas de processo do CMMI-DEV. A última atualização do modelo MPS leva em consideração a nova versão da norma ISO/IEC 12207.(SOFTEX, 2009)

O MR-MPS busca atender à necessidade de implantar os princípios de engenharia de software de forma adequada às necessidades de negócio das organizações brasileiras e define sete níveis de maturidade de processos para organizações que produzem software:

- A (Em Otimização);
- B (Gerenciado Quantitativamente);
- C (Definido);

- D (Largamente Definido);
- E (Parcialmente Definido);
- F (Gerenciado);
- G (Parcialmente Gerenciado);

Cada um dos níveis de maturidade (do nível G – primeiro estágio de maturidade ao nível A - mais maduro) apresenta cumulativamente um conjunto de processos e atributos de processos que indicam onde a unidade organizacional tem que investir esforço para melhoria, de forma a atender aos seus objetivos de negócio e ao MR-MPS. Assim, os níveis de maturidade são definidos em duas dimensões: a dimensão de capacidade de processos e a dimensão de processos. A dimensão de capacidade de processos do MR-MPS é constituída de um framework de medição. A capacidade de processos é definida em uma escala ordinal que representa capacidade crescente do processo. Esta escala vai desde não atingir o propósito básico do processo até atingir precisamente objetivos de negócio atuais para o processo. Dentro do framework a medida da capacidade é baseada em um conjunto de nove atributos de processo (AP), em total conformidade com a ISO/IEC 15504:

- AP 1.1 (o processo é executado);
- AP 2.1 (o processo é gerenciado);
- AP 2.2 (os produtos de trabalho do processo são gerenciados);
- AP 3.1 (o processo é definido);
- AP 3.2 (o processo está implementado);
- AP 4.1 (o processo é medido);
- AP 4.2 (o processo é controlado);
- AP 5.1 (o processo é objeto de inovações);
- AP 5.2 (o processo é otimizado continuamente);

Cada AP contém um conjunto de resultados de atributo de processo (RAP) utilizados para avaliar a implementação de um AP.

A dimensão de processo do MR-MPS, por sua vez, é constituída do conjunto de processos de engenharia de software que deve ser avaliado para o nível de maturidade pretendido.

O nível de maturidade G (Parcialmente Gerenciado) é composto pelos processos mais críticos de gerência. Ao implementar os processos deste nível a organização pode focar os esforços de melhoria em estabelecer mecanismos mais adequados para o planejamento de projetos, monitoração e controle, e para gerenciar requisitos ao longo do ciclo de vida do produto.

De modo a melhorar o controle dos projetos, a organização deve implementar processos de apoio para o desenvolvimento de software. Estes processos constituem o próximo nível do MR-MPS, o F (Gerenciado). Os processos deste nível focam em assegurar a qualidade dos produtos e do processo, obtendo indicadores quantitativos do desempenho de processos e gerenciando as configurações dos produtos.

A implementação dos níveis G e F é um passo significativo em uma organização de desenvolvimento de software. Entretanto, no nível F, a organização pode ainda ser fortemente dependente do conhecimento e desempenho individual dos profissionais. Por este motivo, grandes benefícios podem ser obtidos pela institucionalização de processos padrão da organização utilizados nos diferentes projetos. O nível E (Parcialmente Definido) possui o conjunto de processos que apóiam a institucionalização e melhoria de processos padrão para guiar projetos de software. Neste nível o grupo de processos de engenharia de software tem um papel fundamental.

Com a infra-estrutura para execução e melhoria de processos estabelecida na organização, o próximo passo é focar na melhoria de processos de engenharia de software mais técnicos. Estes são os processos de engenharia, agrupados no nível de maturidade D (Largamente Definido). É neste nível que questões mais técnicas do desenvolvimento do produto são tratadas, tais como o estabelecimento de métodos para desenvolvimento de requisitos, definição de arquiteturas e do projeto do software, estratégias para a integração do produto, validação e verificação (incluindo revisões por pares e testes).

O nível C é composto por processos de engenharia de software complementares à gerência de projetos. Estes processos são relacionados à gerência de riscos e ao apoio a situações de tomada de decisão. Adicionalmente, o processo Desenvolvimento para Reutilização foi incorporado a fim de possibilitar o estabelecimento de um programa de reutilização para desenvolver ativos e artefatos através da engenharia de domínio, em consonância com as definições da ISO/IEC 12207.

Por fim, os níveis B e A são os de alta maturidade focando na melhoria contínua de processos. Os atributos de processo do nível B visam o entendimento quantitativo dos produtos e processos de software e o tratamento de causas de variações para estabilizar

processos. Os atributos de processo do nível A, por sua vez, focam em capacidades competitivas crescentes através da implementação de inovações tecnológicas e da resolução de causas de defeitos e outros tipos de problemas. (SOFTEX, 2009)

Podem ser feitas correspondências entre os níveis de maturidade MR-MPS e CMMI. A avaliação dos processos e atributos de processo do MR-MPS para os níveis F, C, B e A contempla a verificação do alcance dos objetivos específicos e genéricos de uma avaliação do CMMI níveis 2, 3, 4 e 5, respectivamente, como mostra a Figura 9:

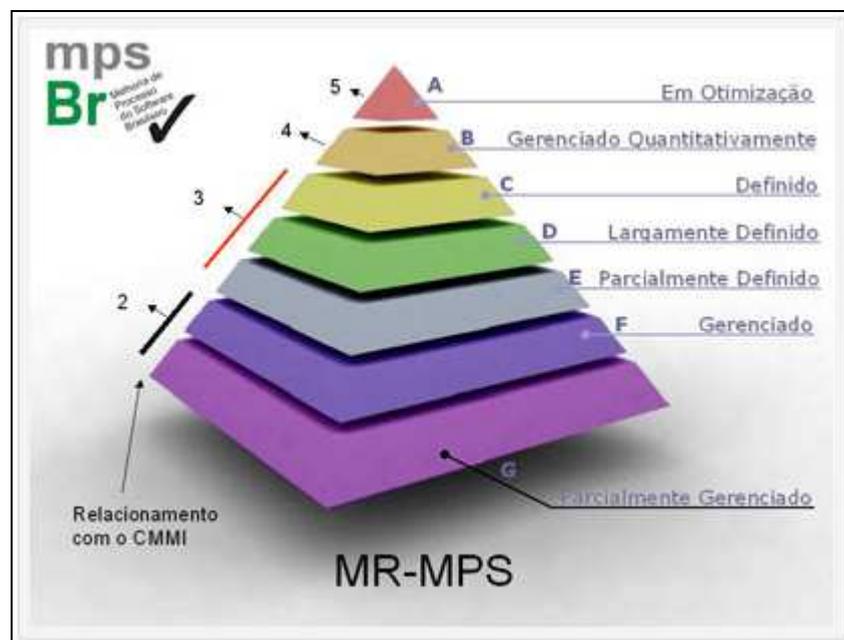


Figura 8: Comparativo entre as fase dos modelos MPS.BR e CMMI-Dev
 Fonte: <http://www.assembla.com/spaces/procsw/wiki>

A estrutura diferente dos níveis de maturidade no MR-MPS tem dois motivos, prover um caminho para o aumento da maturidade ao reduzir o número de processos a serem implementados nos primeiros níveis de maturidade e permitir a visibilidade dos resultados da melhoria de processos em um intervalo de tempo menor. Vale ainda ressaltar que algumas áreas de processo do CMMI aparecem como atributos de processo no MRMPS, tendo em vista a natureza destes processos, de capacitar os demais. Existem ainda processos no MR-MPS que não estão presentes no CMMI, como Gerência de Portfólio de Projetos, Gerência de Recursos Humanos (com gerência do conhecimento), Gerência de Reutilização e Desenvolvimento para Reutilização.

Além do Guia Geral, existem outros onze guias disponíveis (conforme a Figura 10) em português e espanhol que apóiam as organizações na implementação dos processos específicos do MRMPS:

- Sete guias de implementação, um para cada nível do modelo, dos quais estarei utilizando o primeiro para descrever o nível G do modelo, foco deste trabalho;
- Três guias voltados respectivamente para organizações que adquirem software, para fábricas de software e para fábricas de teste que querem implementar o MR-MPS;
- Um guia de aquisição apoiando a implementação do processo de aquisição do MR-MPS.;

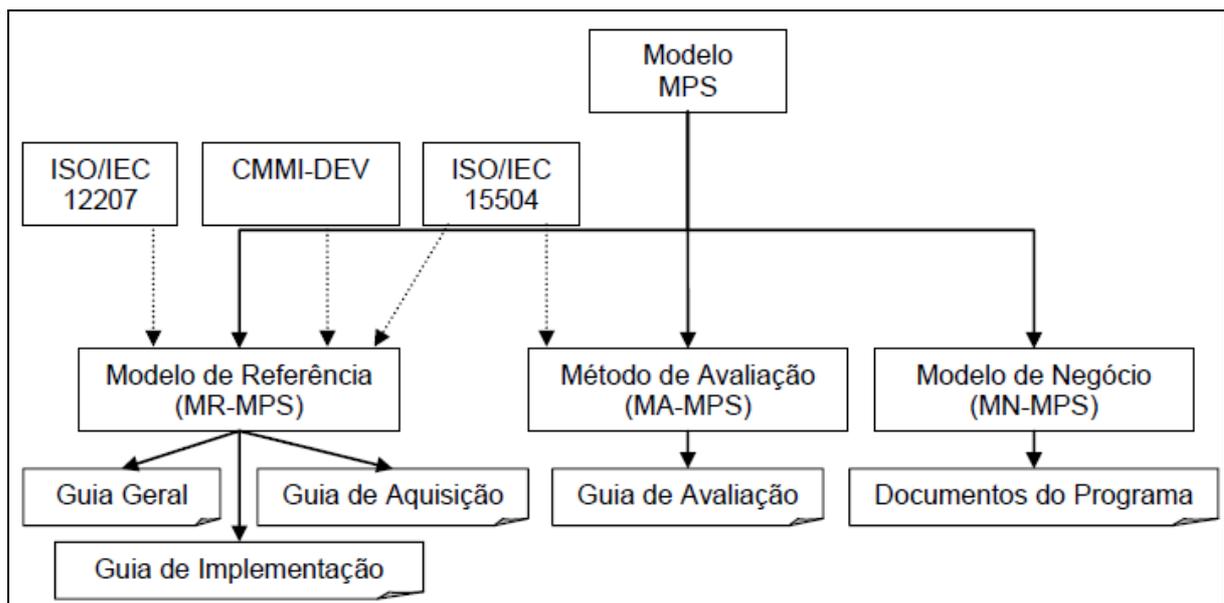


Figura 9: Modelo e fundamentos de Guias MPS.

Fonte: (SOFTEX, 2009)

4.3 NÍVEL DE MATURIDADE: G – PARCIALMENTE GERÊNCIADO

O nível G é o primeiro nível de maturidade do MR-MPS. Sua implementação deve ser executada com cautela por estabelecer o início dos trabalhos em implantação de melhoria dos processos de software na organização. Ao final da implantação deste nível a organização deve ser capaz de gerenciar parcialmente seus projetos de desenvolvimento de software. Dois

pontos são desafiadores na implantação do nível G: A mudança de cultura organizacional, orientando a definição e melhoria dos processos de desenvolvimento de software e a definição do conceito acerca do que é “projeto” para a organização. No nível G, o projeto pode usar os seus próprios padrões e procedimentos, não sendo necessário que se tenha padrões em nível organizacional. Caso a organização possuir processos já definidos e os projetos necessitem adaptar os processos existentes, esse fato deverá ser declarado durante o planejamento do projeto. Essas adaptações podem incluir alteração em processos, atividades, ferramentas, técnicas, procedimentos, padrões, medidas, dentre outras. Diversas organizações de software trabalham com evolução de produtos e precisam adequar as suas formas de trabalhar para se tornarem organizações orientadas a projetos. Ser orientada a projetos significa redefinir algumas operações (atividades de rotina) já em andamento, como projeto, estabelecendo objetivos, prazos e escopo para sua execução.

4.3.1 GERÊNCIA DE PROJETOS (GPR)

O PMI (*Project Management Institute*), um dos mais conceituados e reconhecidos institutos na área de gerenciamento de projetos, é responsável pela publicação e atualização do PMBOK (*Project Management Body of Knowledge*). O PMBOK é um guia em gerência de projetos. Ele agrupa o conhecimento em gerência de projetos que é amplamente reconhecido como as boas práticas deste tipo de gerenciamento.

O gerenciamento de projeto na visão do PMBOK é a aplicação de conhecimento, habilidades, ferramentas e técnicas às atividades do projeto, a fim de atender aos seus requisitos. Gerenciar projeto envolve identificar as necessidades, estabelecer objetivos claros e viáveis e balancear as demandas conflitantes em termos de qualidade, escopo, tempo e custo. Um processo de gerenciamento de projeto identifica, estabelece, coordena e produz um produto, de acordo com seus requisitos.

No MPS.BR, a definição de projeto é “Um empreendimento realizado para criar um produto. O projeto se caracteriza por temporalidade e resultado, produto único e elaboração progressiva” (SOFTEX, 2009). A temporalidade na definição de projeto significa que todos os projetos possuem um início e um fim bem definidos e estabelecidos. O fim do projeto é atingido quando os objetivos do projeto tiverem sido alcançados, quando se tornar claro que os objetivos não serão ou não poderão ser alcançados ou ainda quando o projeto for cancelado.

O IEEE (Institute of Electrical and Electronics Engineers), em seu Glossário Padrão de Terminologias da Engenharia de Software (IEEE, 1990), diz que a gerência de projetos de software pode ser definida como a aplicação de planejamento, coordenação, medição, monitoramento, controle e divulgação de relatórios, com o intuito de garantir que o desenvolvimento e a manutenção de software sejam sistemáticos, disciplinados e qualificados. E, segundo a norma internacional ISO/IEC 12207, o propósito da gerência de projetos é identificar, estabelecer, coordenar e monitorar as atividades, tarefas e recursos que um projeto necessita para produzir um produto, no contexto dos requisitos e restrições do projeto (ISO/IEC, 2008).

4.3.1.1 RESULTADOS ESPERADOS

O modelo de referência MR-MPS define os resultados esperados com a gerência de projetos através do uso da sigla GPR. A seguir a definição de cada resultado esperado:

GPR1 - O escopo do trabalho para o projeto é definido

O escopo do projeto define todo o trabalho necessário, e somente ele, para entregar um produto que satisfaça as necessidades, características e funções especificadas para o projeto, de forma a concluí-lo com sucesso. O escopo é o ponto de partida para o planejamento do projeto. A definição do escopo deve estabelecer o que está e o que não está incluído no projeto. Para isso, o escopo em geral contém a definição do objetivo e da motivação, os limites e restrições, todos os produtos que serão entregues e os outros produtos gerados pelo projeto, entre outras informações. (SOFTEX, 2009)

O escopo pode ser representado por meio de uma Estrutura Analítica do Projeto (EAP) também conhecida como WBS (Work Breakdown Structure). A EAP fornece um esquema para identificação e organização das unidades lógicas de trabalho a serem gerenciadas, que são chamadas de “pacotes de trabalho” (work packages). Este resultado também pode ser implementado por meio de um Documento de Visão ou outro documento que defina, claramente, o escopo do trabalho. (PMI, 2008)

GPR2 - As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados

O escopo do projeto, identificado na forma dos seus principais produtos de trabalho e das tarefas do projeto, deve agora ser decomposto em componentes menores, mais facilmente gerenciáveis e possíveis de serem dimensionados. Uma estrutura de decomposição do trabalho apropriada deve ser estabelecida. Esta estrutura de decomposição pode ser a EAP do projeto ou estrutura equivalente.

O tamanho é a dimensão das funcionalidades sob o ponto de vista do usuário. São contadas tabelas internas e externas ao sistema, classes, objetos, relatórios, telas, consultas a banco de dados, cálculos, transações e atores dos casos de uso, linhas de código etc. Uma técnica bastante utilizada para medir o tamanho do software é a técnica de Análise de Pontos por Função, que visa estabelecer uma medida de tamanho do software em Pontos por Função. No entanto, o uso de uma técnica deste tipo não é exigido no nível G do MPS.BR, porém será obrigatória a partir do nível E. No nível G, a estimativa de escopo, produtos e tarefas pode ser feita baseada na complexidade, no número de requisitos ou no uso da EAP juntamente com dados históricos e a experiência em projetos anteriores. Uma organização pode também aplicar técnicas de estimativas próprias que se mostraram eficientes e adequadas às necessidades e características da empresa.

GPR3 - O modelo e as fases do ciclo de vida do projeto são definidos

O ciclo de vida de um projeto consiste de fases e atividades que são definidas de acordo com o escopo dos requisitos, as estimativas para os recursos e a natureza do projeto, visando oferecer maior controle gerencial. O ciclo de vida de projeto define um conjunto de fases, que por sua vez geram produtos de trabalho necessários para o desenvolvimento de fases posteriores.

É importante conhecer alguns modelos de ciclo de vida e em que situações são aplicáveis. Os principais modelos de ciclo de vida podem ser agrupados em três categorias principais: modelos sequenciais ou cascata, modelos incrementais e modelos evolutivos (ISO/IEC, 1998). Cada um destes modelos pode ser utilizado na sua forma original ou eles podem ser combinados para criar outro modelo de ciclo de vida.

A organização pode preestabelecer que todos os projetos tenham o mesmo ciclo de vida, ou definir um modelo de ciclo de vida para cada projeto. A determinação das fases do ciclo de vida do projeto possibilita períodos planejados de avaliação e de tomada de decisões, nos quais compromissos significativos são realizados com relação aos recursos, abordagem técnica, reavaliação de escopo e custo do projeto.

GPR4 - (Até o nível F) O esforço e o custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas

As estimativas de esforço e custo são baseadas nos resultados de análises utilizando modelos e dados históricos aplicados ao tamanho, atividades e outros parâmetros de planejamento. As estimativas de esforço e custo tipicamente consideram: o escopo, produtos de trabalho e as tarefas estimadas para o projeto; os riscos; as mudanças já previstas; o ciclo de vida escolhido para o projeto; viagens previstas; nível de competência da equipe do projeto e outros.

Empresas implementando o nível G do MR-MPS geralmente não possuem bases de dados históricas. Entretanto, para alcançar níveis superiores de maturidade é preciso que essa base seja construída e os dados obtidos pelos projetos executados.

GPR5 - O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos

As dependências entre tarefas são estabelecidas e potenciais gargalos são identificados utilizando métodos apropriados por exemplo, análise de caminho crítico (PMI, 2008). Os gargalos são resolvidos quando possível e o cronograma das atividades com início, duração e término é estabelecido. Recursos requeridos são refletidos nos custos estimados. Uma forma de se definir o cronograma é utilizando a EAP e as estimativas de esforço e custo (GPR4), considerando as dependências entre as tarefas e os marcos e pontos de controle – eventos que são considerados significativos no âmbito do projeto. É importante ter-se o cuidado de manter a coerência entre ciclo de vida, EAP, estimativas e cronogramas. O orçamento do projeto é estabelecido com base no cronograma e na estimativa de custos. Este resultado é importante

porque o cronograma e o orçamento são instrumentos fundamentais para o acompanhamento do dia-a-dia do projeto.

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

Para facilitar a identificação dos riscos, é interessante elaborar uma lista de riscos mais comuns a ser examinada pelo gerente do projeto e equipe do projeto para identificar quais destes são potenciais riscos para o projeto em questão. A análise da probabilidade de ocorrência e da gravidade dos problemas decorrentes de sua ocorrência ajuda a definir a prioridade dos riscos. Os riscos identificados devem ser registrados, bem como o acompanhamento dos seus estados e ações tomadas. Uma planilha de riscos, contendo dados como identificador, descrição, probabilidade, impacto e prioridades no seu tratamento, pode ser utilizada para identificação dos riscos, monitoração dos riscos identificados e atualização da lista de riscos do projeto à medida que novos riscos forem sendo identificados. (SOFTEX, 2009)

GPR7 - Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo

O planejamento de recursos humanos determina funções, responsabilidades e relações hierárquicas do projeto. As funções do projeto podem ser designadas para pessoas ou grupos, os quais podem ser internos ou externos à organização. O planejamento de recursos humanos inclui informações de como e quando o recurso será envolvido no projeto, critérios para sua liberação, competência necessária para a execução das atividades, mapa de competências da equipe e identificação de necessidades de treinamento. Caso uma pessoa seja alocada ao projeto sem ter as competências necessárias, o risco pode ser minimizado, por exemplo, com ações de treinamento e mentoring ou supervisionando-se o trabalho da pessoa por um membro melhor capacitado. (SOFTEX, 2009)

GPR8 - Os recursos e o ambiente de trabalho necessários para executar o projeto são planejados

Este resultado faz referência à necessidade de se planejar, com base na EAP (ou estrutura equivalente), as tarefas e previstos os recursos e o ambiente necessários, incluindo, por exemplo, equipamentos, ferramentas, serviços, componentes, viagens e requisitos de processo (processos especiais para o projeto). Todos os recursos precisam ser explicitamente planejados, mesmo os já considerados como existentes e disponíveis ou que serão compartilhados com outros projetos, uma vez que se trata da sua alocação para uso. Estes itens podem, por exemplo, estar registrados no plano do projeto. Caso não haja necessidade de nenhum recurso a ser adquirido para o projeto deve-se registrar o fato de que a questão foi examinada. (SOFXTEX, 2009)

GPR9 - Os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição.

Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança. Os dados do projeto são as várias formas de documentação exigidas para sua execução, por exemplo: relatórios; dados informais; estudos e análises; atas de reuniões; documentação; lições aprendidas; artefatos gerados; itens de ação; e indicadores. Os dados podem estar em qualquer formato e existir em qualquer meio, como: impressos ou desenhados em diversos materiais; fotografias; meio eletrônico; e multimídia. (SOFTEX, 2009)

A confidencialidade das informações deve ser tratada com cuidado. É recomendável, portanto, explicitar a existência ou não de dados confidenciais. Se a organização tem um critério padrão para execução dessas atividades, isto deve ser registrado no plano do projeto ou em outro documento.

GPR10 - Um plano geral para a execução do projeto é estabelecido com a integração de planos específicos

O objetivo deste resultado esperado é garantir que todos os planos que afetam o projeto estejam integrados e que a dependência entre estes planos tenha sido identificada e levada em consideração durante o planejamento, conciliando o trabalho a ser realizado aos

recursos e condições existentes. A realização do planejamento do projeto é garantida pelos resultados esperados no escopo do nível G do MR-MPS do processo Gerência de Projetos (GPR), que prevê, dentre outros, a criação do cronograma de atividades, o planejamento de recursos humanos, custos, riscos, dados etc. A reunião destes documentos é entendida como sendo o Plano de Projeto. As tarefas do processo definido para o projeto podem, também, fazer parte deste Plano do Projeto. (SOFTEX, 2009)

O monitoramento efetivo do projeto dependerá de uma organização adequada destas informações de planejamento. Ao longo do projeto, elas deverão ser comparadas aos dados obtidos durante sua execução, em busca de uma maior visibilidade do andamento do projeto. Quando necessário, o planejamento deverá ser revisto.

GPR11 - A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada. Se necessário, ajustes são realizados

O estudo de viabilidade considera o escopo do projeto e examina aspectos técnicos (requisitos e recursos), financeiros (capacidade da organização) e humanos (disponibilidade de pessoas com a capacitação necessária). Podem-se considerar também os objetivos de negócio da organização. Muitas vezes é preferível não iniciar ou parar um projeto já iniciado do que prosseguir com um projeto inviável. O prosseguimento pode levar a perdas maiores, tanto para o fornecedor como para o cliente. À medida que o projeto evolui, a viabilidade de sucesso pode ser reavaliada com mais precisão. As mudanças de requisitos são eventos que podem levar à necessidade de reavaliar a viabilidade do projeto. Em marcos do projeto e mesmo durante as atividades de acompanhamento, pode ser necessária a confirmação da viabilidade de continuidade do projeto. (SOFXTEX, 2009)

GPR12 - O Plano do Projeto é revisado com todos os interessados e o compromisso com ele é obtido

Para obter compromissos dos interessados relevantes, é importante revisar o planejamento com eles e conciliar as diferenças existentes entre os recursos estimados e disponíveis. Negociações devem ser realizadas quando existirem conflitos entre as diversas variáveis do projeto, como requisitos, custos e prazos. Por exemplo: o escopo pode sofrer

redução para que as metas de prazos e custos sejam cumpridas ou, ao contrário, aumenta-se o orçamento do projeto para que os requisitos sejam atendidos na íntegra, dentro da meta de prazo (PMI, 2008).

Algumas organizações costumam realizar uma reunião de início do projeto (kick off) que pode ser utilizada para resolver os conflitos e obter o comprometimento. Este resultado esperado está de certa forma associado ao GPR11, pois a realização da análise de viabilidade pode resultar em ações para solução de conflitos. A solução dos conflitos e estabelecimento de compromissos é fundamental para que o projeto possa efetivamente contar com os recursos planejados, para atingir as metas definidas.

GPR13 - O projeto é gerenciado utilizando-se o Plano do Projeto e outros planos que afetam o projeto e os resultados são documentados

A aderência aos diversos planos deve ser avaliada continuamente durante todo o ciclo de vida do projeto. Os resultados e os critérios de conclusão de cada tarefa são analisados, as entregas são avaliadas em relação às suas características (por meio de revisões e auditorias, por exemplo), a aderência ao cronograma e o dispêndio de esforços são examinados, bem como o uso dos recursos. Análises devem ser realizadas e decisões serem tomadas considerando-se as variações dos dados e desvios entre resultados e valores atuais e esperados. (SOFTEX, 2009)

GPR14 - O envolvimento das partes interessadas no projeto é gerenciado

Devem ser identificados os interessados relevantes no projeto, em que fases eles são importantes e como eles serão envolvidos (comunicações, revisões em marcos de projeto, comprometimentos, entre outros). Uma vez identificado e planejado o envolvimento, este deverá ser seguido. Os interessados no projeto podem incluir o cliente e o usuário (ou seus representantes), a direção da organização e os membros da equipe do projeto. Em projetos pequenos, estas atividades podem ser simplificadas devido ao pequeno número de interessados e a pouca comunicação necessária em função do curto prazo.

A comunicação envolve, por exemplo, questões relativas a prazos, custos, recursos, comprometimentos e também requisitos, pois estes afetam as outras variáveis. Um plano de gerenciamento das comunicações pode cobrir este resultado esperado (PMI, 2008).

Além da comunicação, é necessário verificar se os compromissos assumidos pelas partes interessadas estão sendo cumpridos ou negociados, sejam eles internos ou externos, visando identificar aqueles que não foram satisfeitos ou que possuem um grande risco de não serem satisfeitos.

GPR15 - Revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento

Revisões em marcos do projeto não devem ser confundidas com o acompanhamento descrito em GPR13, que é o acompanhamento do dia-a-dia do projeto. Os marcos do projeto precisam ser previamente definidos ao se realizar o planejamento do projeto. Este resultado é importante porque as revisões em marcos são oportunidades para verificar o andamento de todo o projeto, independente do acompanhamento do dia-a-dia. Em projetos grandes essas revisões são fundamentais, questionando, inclusive, a viabilidade de continuidade do projeto.

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

As atividades de revisão em marcos (GPR15) e de monitoramento (GPR13) do projeto possibilitam a identificação de problemas que estejam ocorrendo nos projetos. Estes problemas devem ser analisados e registrados, por exemplo, por meio de ferramentas específicas, planilhas ou outros tipos de mecanismos de gerenciamento de problemas. Para completar o trabalho de monitoramento do projeto, os problemas precisam ser corrigidos e gerenciados até a sua resolução, com base em planos de ações, estabelecidos especificamente para resolver os problemas levantados e registrados (GPR17). Caso não se consiga resolver os problemas neste nível, deve-se escalar a resolução das ações a níveis superiores de gerência. (SOFTEX, 2009)

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

Como resultado do acompanhamento do projeto (GPR13) e das revisões em marcos (GPR15), problemas são identificados, analisados e registrados (GPR16). Ações corretivas devem ser estabelecidas para resolver problemas que possam impedir o projeto de atingir seus objetivos se não forem resolvidos de forma adequada. O controle dos problemas levantados, as ações tomadas, os responsáveis pelas ações e os resultados devem ser registrados.

Os problemas identificados provêm a base para a tomada de ações corretivas. Quando apropriado, e quando o impacto e os riscos associados são identificados e gerenciados, as mudanças podem ser realizadas no projeto.

4.3.2 GERÊNCIA DE REQUISITOS (GRE)

O propósito do processo Gerência de Requisitos é gerenciar os requisitos do produto e dos componentes do produto do projeto e identificar inconsistências entre os requisitos, os planos do projeto e os produtos de trabalho do projeto. O principal objetivo da Gerência de Requisitos é controlar a evolução dos requisitos. O processo Gerência de Requisitos (GRE) gerência todos os requisitos recebidos ou gerados pelo projeto, incluindo requisitos funcionais e não-funcionais, bem como os requisitos impostos ao projeto pela organização. (SOFTEX, 2009)

Quando um projeto recebe requisitos de um fornecedor de requisitos – pessoa autorizada a participar de sua definição e a solicitar modificação –, estes devem ser revisados para resolver questões e prevenir o mau entendimento, antes que os requisitos sejam incorporados ao escopo do projeto.

Outras atribuições do processo Gerência de Requisitos são documentar as mudanças nos requisitos e suas justificativas, bem como manter a rastreabilidade bidirecional entre os requisitos e produtos de trabalho em geral e identificar inconsistências entre os requisitos, os planos do projeto e os produtos de trabalho do projeto.

Uma boa comunicação com os fornecedores de requisitos é fundamental para assegurar um bom entendimento das necessidades do cliente e dos requisitos do projeto e, aumentar as chances de sucesso do projeto. Segundo Dorfmann e Thayer (1990), requisito de software representa a capacidade requerida pelo usuário que deve ser encontrada ou possuída por um determinado produto ou componente de produto para resolver um problema ou alcançar um objetivo ou para satisfazer a um contrato, a um padrão, a uma especificação ou a outros documentos formalmente impostos. A gerência de requisitos envolve identificar os requisitos do produto e dos componentes do produto do projeto, bem como estabelecer e manter um acordo entre o cliente e a equipe de projeto sobre esses requisitos. Também é objetivo da gerência de requisitos controlar e tratar as mudanças nos requisitos ao longo do desenvolvimento.

Para apoiar o processo de mudança de requisito, é fundamental definir e manter a rastreabilidade dos requisitos. Quando os requisitos são bem gerenciados, a rastreabilidade pode ser estabelecida, desde um requisito fonte, passando por todos os níveis de composição do produto. Tal rastreabilidade bidirecional auxilia a determinar se todos os requisitos foram completamente tratados e se todos os requisitos de mais baixo nível podem ser rastreados para uma fonte válida (SEI, 2006). A rastreabilidade bidirecional deve acontecer tanto de forma horizontal quanto vertical. A rastreabilidade horizontal estabelece a dependência entre os requisitos ou produtos de trabalho em um mesmo nível, por exemplo, rastreabilidade dos requisitos entre si ou rastreabilidade entre códigos de unidades dependentes. A rastreabilidade vertical estabelece uma rastreabilidade bidirecional desde um requisito fonte, passando pelos seus requisitos de mais baixo nível, até o nível de decomposição mais baixo do produto, por exemplo, códigos de unidade ou módulos do software. Esse mecanismo de rastreabilidade vertical é essencial para a realização da análise de impacto de mudanças de requisitos, por exemplo, para identificar de que forma uma mudança de requisito impacta nos planos do projeto, bem como os códigos de unidade ou módulos do software que necessitam ser modificados.

4.3.2.1 RESULTADOS ESPERADOS

O modelo de referência MR-MPS utiliza a sigla GRE para definir os resultados esperados no processo de gerência de requisitos, sendo eles os seguintes:

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

O objetivo deste resultado é garantir que os requisitos estejam claramente definidos a partir do entendimento dos requisitos realizado junto aos fornecedores. Informações sobre esses fornecedores podem ser identificadas no plano do projeto, bem como informações sobre como será a comunicação com eles. Essas comunicações devem ser registradas formalmente em atas, e-mails, ferramentas de comunicação ou outros meios.

Após a identificação dos requisitos do produto e dos componentes do produto do projeto, é necessário garantir que os requisitos propostos atendem às necessidades e expectativas do cliente e dos usuários. Os requisitos identificados devem ser avaliados com base em um conjunto de critérios objetivos, previamente estabelecidos. Alguns exemplos de critérios são: possuir identificação única; estar claro e apropriadamente declarado; não ser ambíguo; ser relevante; ser completo; estar consistente com os demais requisitos; ser implementável, testável e rastreável (IEEE, 1998). O uso de um checklist para apoiar esta atividade pode ser útil por poder possibilitar que a organização compreenda melhor quais os problemas que têm ocorrido em relação à especificação de requisitos.

A avaliação dos requisitos, um registro de aceite dos requisitos deve ser obtido pelos fornecedores de requisitos. Esse registro pode ser tratado como um marco do projeto a partir do qual mudanças nos requisitos devem ser tratadas formalmente para minimizar o impacto dessas mudanças no projeto em termos de escopo, estimativas e cronograma, bem como compromissos já estabelecidos.

GRE2 - O comprometimento da equipe técnica com os requisitos aprovados é obtido

Após a aprovação dos requisitos, um comprometimento formal da equipe técnica com os requisitos aprovados deve ser obtido e registrado, por exemplo, na forma de ata de reunião, e-mail ou algum outro mecanismo. Mudanças de requisitos aprovados pelos fornecedores de requisitos podem afetar compromissos já estabelecidos pela equipe técnica. Nestes casos, um novo comprometimento da equipe técnica com os requisitos modificados deve ser obtido e registrado após os requisitos modificados terem sido novamente aprovados a partir de critérios estabelecidos. (SOFTEX, 2009)

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

Este resultado indica a necessidade de se estabelecer um mecanismo que permita rastrear a dependência entre os requisitos e os produtos de trabalho. Ter definida a rastreabilidade facilita a avaliação do impacto das mudanças de requisitos que possam ocorrer, por exemplo, nas estimativas do escopo, nos produtos de trabalho ou nas tarefas do projeto descritas no cronograma.

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

A consistência entre os requisitos e os produtos de trabalho do projeto deve ser avaliada e os problemas identificados devem ser corrigidos. Este resultado sugere, portanto, a realização de revisões ou de algum mecanismo equivalente para identificar inconsistências entre os requisitos e os demais elementos do projeto como, por exemplo, planos, atividades e produtos de trabalho. As inconsistências identificadas devem ser registradas e ações corretivas executadas a fim de resolvê-las.

Quando há mudanças nos requisitos, é importante examinar se os demais artefatos estão consistentes com as alterações realizadas como, por exemplo: verificar se a planilha de estimativas está contemplando todos os requisitos e mudanças. (SOFTEX, 2009)

GRE5 - Mudanças nos requisitos são gerenciadas ao longo do projeto

Durante o projeto, os requisitos podem mudar por uma série de motivos. Desta forma, requisitos adicionais podem ser incorporados no projeto, requisitos podem ser retirados do projeto e mudanças podem ser feitas nos requisitos já existentes. Devido às mudanças, os requisitos podem ter que ser revistos, conforme definido no GRE4. As necessidades de mudanças devem ser registradas e um histórico das decisões sobre os requisitos deve estar disponível. Estas decisões são tomadas por meio da realização de análises de impacto da

mudança no projeto e podem incluir aspectos como: influência em outros requisitos, expectativa dos interessados, esforço, cronograma, riscos e custo. É importante destacar que o mecanismo de rastreabilidade bidirecional instituído é um importante mecanismo para facilitar a análise de impacto.

4.3.3 ATRIBUTOS DE PROCESSO DO NÍVEL G

De acordo com o Guia Geral do MR-MPS, “a capacidade do processo é representada por um conjunto de atributos de processo descrito em termos de resultados esperados. A capacidade do processo expressa o grau de refinamento e institucionalização com que o processo é executado na organização/unidade organizacional. No MR-MPS, à medida que a organização/unidade organizacional evolui nos níveis de maturidade, um maior nível de capacidade para desempenhar o processo deve ser atingido” (SOFTEX, 2009).

Vale, ainda, ressaltar que “Os níveis são acumulativos, ou seja, se a organização está no nível F, esta possui o nível de capacidade do nível F que inclui os atributos de processo dos níveis G e F para todos os processos relacionados no nível de maturidade F (que também inclui os processos do nível G)” (SOFTEX, 2009). No que se refere aos atributos de processo, para atingir o nível G do MR-MPS, uma organização deve atender aos resultados esperados de RAP 1 a RAP 10. Numa avaliação, segundo o MA-MPS (SOFTEX, 2009), é exigido, para se considerar um processo “SATISFEITO” no nível G, que o atributo de processo AP 1.1 seja caracterizado como T (Totalmente implementado) e que o atributo de processo AP 2.1 seja caracterizado como T (Totalmente implementado) ou L (Largamente implementado). É importante destacar que, a partir do nível E, as exigências são diferentes, conforme descrito no Guia de Avaliação (SOFTEX, 2009). A seguir, os atributos de processo AP 1.1 e AP 2.1, conforme aplicáveis no nível G.

4.3.3.1 AP 1.1 - O PROCESSO É EXECUTADO

Este atributo é uma medida do quanto o processo atinge o seu propósito e está diretamente relacionado ao atendimento do propósito do processo. Assim o seguinte resultado esperado:

RAP 1 - O processo atinge seus resultados definidos

Este resultado esperado busca garantir que o processo transforma produtos de trabalho de entrada identificáveis em produtos de trabalho de saída, também identificáveis, permitindo, assim, atingir o propósito do processo. Ou seja, este resultado implica diretamente na geração dos principais produtos requeridos pelos resultados dos processos. (SOFTEX, 2009)

4.3.4 AP 2.1 - O PROCESSO É GERENCIADO

Este atributo é uma medida do quanto à execução do processo é gerenciada e está relacionado à gerência dos processos. A implementação deste atributo de processo implica no planejamento da execução do processo, atribuindo responsabilidade e autoridade para sua execução, bem como fornecendo recursos adequados. Envolve também o monitoramento e controle da execução dos processos, tomando ações corretivas, quando necessárias. Relacionados a este atributo de processo estão definidos os seguintes resultados esperados:

RAP 2 - Existe uma política organizacional estabelecida e mantida para o processo

Este resultado visa à definição de uma política contendo as diretrizes de como a organização planeja e implementa os seus processos, bem como informações sobre as expectativas organizacionais para a execução dos processos e a indicação de como devem ser atendidos os aspectos mais importantes de cada processo. Isso pode incluir princípios básicos e definições gerais de como executar os processos, incluindo aspectos de responsabilidades, tempos e instrumentos. A política não deve ser uma reprodução de textos do MR-MPS, mas sim, como a organização enxerga seus processos. Um documento genérico pode existir definindo quem tem autoridade, delegada pela gerência de alto nível, para aprovar cada tipo de documento.

Normalmente, as políticas são definidas e aprovadas pela gerência de alto nível, não havendo a obrigatoriedade de serem rotuladas exatamente de “políticas”. Uma vez definidas, as políticas devem ser publicadas e divulgadas aos interessados em sua execução. Tal publicação pode ser realizada, por exemplo, na Intranet da organização. Em geral, a

divulgação da política pela alta gerência ajuda a enfatizar a importância dos processos, facilitando sua institucionalização.(SOFTEX, 2009)

RAP 3 - A execução do processo é planejada

Este resultado visa à realização de um plano para a execução do processo. Este planejamento deve incluir recursos, responsabilidades e tempo, bem como as atividades de controle e monitoramento da execução do processo. Deve ser estabelecido e documentado um plano para a execução do processo, o que inclui sua própria descrição.

RAP 4 - (Para o nível G) A execução do processo é monitorada e ajustes são realizados

Este resultado só se aplica ao nível G e visa monitorar a execução dos processos conforme o que foi planejado e assegurar que ações corretivas sejam tomadas sempre que houver desvios significativos em relação ao planejado. Desta forma, revisões das atividades, estado e resultados dos processos devem ser realizadas e podem ocorrer tanto periodicamente ou motivadas por algum evento. Durante o monitoramento dos processos, questões poderão ser identificadas, para as quais ações corretivas deverão ser tomadas e acompanhadas até o seu encerramento.

RAP 5 - (Até o nível F) As informações e os recursos necessários para a execução do processo são identificados e disponibilizados

Este resultado visa assegurar que as informações e os recursos necessários para executar o processo serão identificados previamente e que estarão disponíveis quando forem necessários. Incluem recursos financeiros, condições físicas adequadas, pessoal e ferramentas apropriadas (incluindo processos e modelos de documentos predefinidos). Estas informações e recursos podem estar estabelecidos na própria descrição do processo ou podem, também, estar presentes em planos específicos para os processos nos níveis da organização e do projeto.

RAP 6 - (Até o nível F) As responsabilidades e a autoridade para executar o processo são definidas, atribuídas e comunicadas

Este resultado visa assegurar que as responsabilidades e a autoridade para executar o processo estão claramente definidas e bem compreendidas.

Deve-se assegurar, também, que as responsabilidades e a autoridade para executar o processo foram atribuídas explicitamente e comunicadas a todas as partes interessadas, por exemplo, patrocinador, implementadores etc.

RAP 7 - (Até o nível F) As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência

Este resultado visa assegurar que as pessoas tenham as habilidades, conhecimentos e experiências necessários para executar ou apoiar o processo. Deve-se assegurar que as pessoas tenham o conhecimento em relação ao seu papel no processo: conhecimento completo para aqueles que vão realizar as atividades do processo e conhecimento genérico para os que vão interagir com o processo. Conhecimento e habilidades não se restringem aos documentos de processo, mas podem incluir trabalho em grupo, liderança, análise e solução de problemas. Quando se julgar necessário, um treinamento apropriado deve ser fornecido para as pessoas que executarão os processos. Os treinamentos podem ser de diferentes tipos, por exemplo: treinamento autodirecionado; instrução programada autodefinida; treinamento formal dentro do trabalho; mentoring; treinamento formal em salas de aula.

RAP 8 - A comunicação entre as partes interessadas no processo é gerenciada de forma a garantir o seu envolvimento

O objetivo deste resultado é identificar as partes interessadas no processo, planejar e manter o seu envolvimento. Os interessados podem ser envolvidos tipicamente em atividades tais como: planejamento; coordenação; revisão; e definição dos requisitos para a execução do processo. É importante gerenciar a interface entre as partes interessadas de forma a assegurar a comunicação.

RAP 9 - (Até o nível F) Os resultados do processo são revistos com a gerência de alto nível para fornecer visibilidade sobre a sua situação na organização

O objetivo deste resultado é fornecer visibilidade à alta gerência com relação ao estado da execução dos processos, considerando sua adequação, operação com recursos apropriados e alcance dos resultados esperados. Um dos métodos de monitoração de processo é a revisão,

junto à gerência de alto nível, de seu estado, atividades realizadas e resultados alcançados. As revisões devem ocorrer periodicamente ou, então, motivadas por algum evento e não necessitam ser presenciais. Desta forma, o andamento da implantação dos processos, tendências e problemas são relatados e tratados em níveis apropriados.

RAP 10 - (Para o nível G) O processo planejado para o projeto é executado

O objetivo deste resultado é garantir que o projeto é conduzido a partir da execução do seu processo planejado. Deve-se garantir que existem registros de execução das atividades do processo com base no seu planejamento. Esses registros devem ser mantidos e revistos periodicamente para garantir que o processo planejado está sendo seguido para atingir os objetivos do projeto.

4.4 ESTATÍSTICAS DO MODELO MPS.BR

A Figura 10 demonstra que em 2009 houve um grande salto nas avaliações, chegando próximo à 80 avaliações. Isto mostra que as empresas de TI estão começando aos poucos a dar valor em qualidade e melhoria de processos de software.

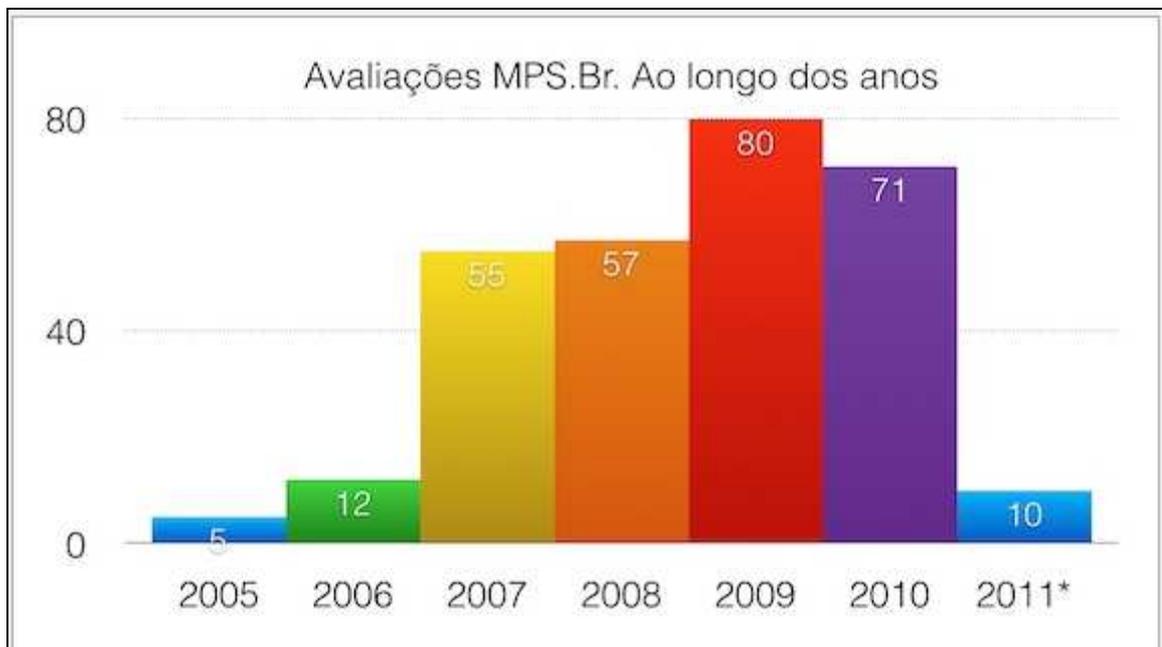


Figura 10: Avaliações MPS.BR ao longo dos anos (Fevereiro de 2011)

Fonte: <http://www.blogcmmi.com.br/avaliacao/lista-de-empresas-mps-br-no-brasil>

A Figura 11 apresenta uma pesquisa realizada pela própria Softex, sobre a variação de desempenho em determinadas questões de empresas que realizaram um avanço de nível de maturidade do modelo MPS.BR.

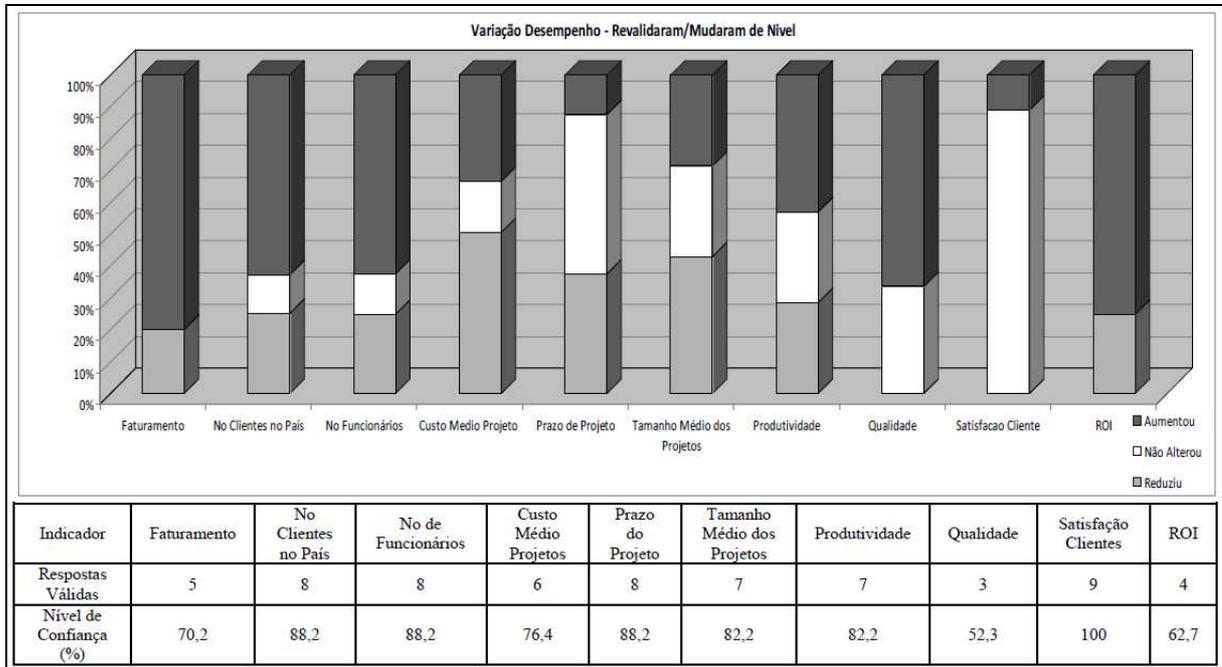


Figura 11: Variação de desempenho das organizações com MPS que Revalidaram/Mudaram de Nível.

5 IMPLEMENTAÇÃO DO MPS.BR NÍVEL G EM UMA EMPRESA DE PEQUENO PORTE

Como mencionado anteriormente a idéia do trabalho veio apartir da necessidade da organização de processos internos de uma empresa de pequeno porte. Com o passar do tempo ficou visível a necessidade de uma reestruturação organizacional, sendo assim escolhido o modelo MP.BR e a implantação e certificação do nível de maturidade G.

Este trabalho não tem como objetivo detalhar características da implantação e da avaliação realizada, como prazos e custos. O que será descrito apartir daqui é como a estrutura organizacional da empresa foi alterada através da definição de processos realizados na fase de implatanção, e como ficarão definidos apartir do projeto piloto.

5.1 PORQUE O MODELO MPS.BR?

Por basear-se em normas internacionalmente reconhecidas como a ISO / IEC 12107 e a norma ISO / IEC 15504, juntamente com conceitos do modelo CMMI-Dev, ter valores mais acessíveis e possuir fases iniciais menos morosas, o modelo MPS.BR vem sendo cada vez mais escolhido entre as empresas nacionais de pequeno e médio porte, começando a ter número interessantes no que se diz respeito a melhoria de processos internos e melhoria na qualidade dos produtos.

É um modelo que tem como objetivo organizar e criar novos processos sem que a empresa tenha que ser completamente reestruturada, ou seja, o modelo não descreve como fazer e sim o que deve ser feito. Esse também é um dos principais motivos pelo qual pequenas empresas tendem a adotar o modelo, a facilidade de adaptação, sem que seus antigos processos e metodos precisem ser remodelados de uma hora pra outra.

5.2 PLANO DE PROJETO (CERTIFICAÇÃO MPS.BR NÍVEL G)

O Projeto aborda:

1. Definição em acordo com o modelo sugerido pelo Programa MPS.BR.

- 1.1. Técnicas aprimoradas de gerenciamento de Projetos
- 1.2. Técnicas aprimoradas de coleta e gerenciamento de Requisitos.
- 1.3. Processos de Projetos, desde sua negociação ate sua finalização, contando com o ciclo de vida básico.
- 1.4. Técnicas de comunicação interna (equipe de colaboradores) e externa (empresa e envolvidos não colaboradores).
- 1.5. Estratégias necessárias a rastreabilidade.
- 1.6. Procedimentos padrões para atividades determinadas nos processos da empresa.
- 1.7. Técnicas de treinamento para equipe de colaboradores.
- 1.8. Documentação necessária para evidenciação da aderência ao modelo sugerido pelo Programa MPS.BR.
- 1.9. Diretrizes da empresa (política interna e políticas externas).
- 1.10. Ferramentas utilizáveis para o auxilio da criação, utilização, armazenamento e visualização dos processos de projetos a serem realizados pela empresa.
- 1.11. Aperfeiçoamentos de ferramentas já em utilização para que atendam as necessidades do modelo sugerido.

2. Validação.

- 2.1. Criação de modelos de documentos (Guias).
- 2.2. Documentação das diretrizes da empresa, posterior a validação das mesmas junto a consultoria.
- 2.3. Publicação de todo o material necessário aos processos da empresa, posterior a validação do mesmo junto a consultoria.
- 2.4. Criação de treinamento especifico para utilização e aderência de todos os colaboradores aos novos processos da empresa.

3. Treinamento

- 3.1. Apresentação geral dos novos processos para a equipe de colaboradores, dividindo esta apresentação em grupos distintos a serem definidos de acordo com a matriz de perfis.
 - 3.1.1. Serão formuladas com uma visão geral para embasamento.

3.1.2. Contemplarão explicações necessárias aos novos procedimentos para o perfil em questão.

3.1.3. Disponibilização de treinamento em mídia acessível aos colaboradores, para eventual consulta posterior.

3.2. Disponibilização de procedimentos padrões de processos (POPs), que serão criados nas etapas de definição e validação, para treinamento de novos colaboradores.

4. Avaliação

Durante a duração do projeto Rumo ao MPS.BR ocorrerão a Simulação de Avaliação presidida pelo órgão responsável, e também a Avaliação Inicial e a Avaliação Final, presididas por órgãos associados ao Programa MPS, órgãos estes que serão definidos pelos coordenadores do Programa.

4.1. Acompanhamento nas Avaliações por avaliador interno capacitado através do curso introdutório ao Modelo C1.

4.1.1. Apresentação do material de evidencia solicitado pelo avaliador principal.

4.1.2. Explicação do processo evolutivo da empresa junto ao modelo adotado.

4.1.3. Explicação de conteúdos, em caso de dúvidas dos avaliadores.

4.2. Ajustes solicitados pela equipe avaliadora para próxima avaliação (Se necessários).

5. Finalização do Projeto.

5.1. Emissão de relatório de Lições Aprendidas para arquivamento em histórico a ser mantido pela empresa.

5.2. Emissão de Relatório Final do Projeto.

5.3. Estudo de viabilidade para busca de Nível de maturidade F ou maior.

O projeto foi dividido em cinco fases principais para controle:

Marco ou entrega	Objetivo esperado	Data prevista
Marco Fase 1: Definição	Definir padrões base	28/01/2011
Marco Fase 2: Validação e Treinamento	Emissão do laudo da consultoria atinge 50%	17/03/2011
Entrega de Documentação	Guias, templates, diretrizes	21/04/2011
Marco Fase 3: Institucionalização	Emissão de laudo da consultoria atinge 100%	14/10/2011
Projetos utilizando o novo processo	No mínimo 1 projeto já finalizado, e 1 em andamento	Mês 09 ano 2011
Marco Fase 4: Avaliações	Avaliação Inicial, Avaliação Final	Meses 11 e 12 de 2011
Entrega de alterações pedidas pela equipe da Avaliação Inicial	Aprovação na Avaliação Final	Mês 12 ano 2011
Marco Fase Final: Finalização	Relatórios de finalização	Janeiro 2012

Tabela 1: Exemplo de cronograma do Projeto Rumo ao MPS.BR

5.3 CRONOGRAMA E EAP

O projeto possui um cronograma estipulado a partir de atividades previstas para seu desenvolvimento, construído com uma ferramenta de controle de cronograma e atualizado junto ao repositório semanalmente. O Cronograma pode ser visualizado no repositório padrão do projeto.

A EAP do projeto é dividida em oito atividades principais, sendo estas:

1. Gerenciamento, que ocorre durante todos os processos deste projeto até sua finalização.
2. Pesquisa, primeira parte, sendo resumida a pesquisa de ferramentas e assuntos relativos ao projeto.
3. Definição, etapa que define o futuro fluxo da empresa em Gerência de Projetos e Gerência de Requisitos.

4. Validação, época na qual serão desenvolvidos os modelos de templates, guias e documentos de política interna, validados junto a consultoria e comitados no repositório.
5. Treinamento, onde ocorrerão os treinamentos internos quanto as alterações realizadas nos processos que afetam o fluxo da empresa.
6. Institucionalização, está será a fase tida como decisiva, nela ocorrerá a institucionalização de todas as definições obtidas anteriormente, nesta fase será lançado o Plano Piloto, e os Projetos passarão a utilizar a política adotada para fluxo segundo o Modelo.
7. Avaliação, quando ocorrerão as avaliações da empresa segundo o modelo MPS.
8. Finalização, fase que finaliza o Projeto com relatórios de lições aprendidas e arquivamento de dados relevantes a Histórico.

5.4 DEFINIÇÃO DE CARGOS E PERFÍS

O primeiro passo a ser tomado pela organização é a definição de sua estrutura. Sendo assim, os setores e os cargos devem definidos juntamente com as responsabilidades. Cada funcionário deve possuir um documento contendo detalhes sobre seu perfil profissional, para possíveis mudanças de cargo ou necessidade da empresa, também deve saber detalhadamente sobre o seu papel na empresa, sendo que os cargos devem possuir um título e uma descrição completa, suas funções e responsabilidades. Isso inclui todos os funcionários da empresa, estagiários, efetivos, gerentes, diretores e proprietários da organização.

Sendo assim os cargos foram definidos, juntamente com a descrição de habilidades e qualificações necessárias para cada função. Segue abaixo alguns exemplos, excluindo habilidades e qualificações por serem características próprias da empresa:

5.4.1 ANALISTA DE NEGÓCIO

Responsável por efetuar o levantamento e documentação dos processos e requisitos necessários do cliente, identificando e avaliando os processos de negócio, adaptando as configurações do sistema para atendê-lo da melhor forma possível. Acompanhar e validar com o cliente o encerramento das etapas do projeto, reportar sugestões de melhoria no sistema e no processo com base em dificuldades encontradas durante a execução de sua atividade.

5.4.2 ANALISTA DE SISTEMAS

Responsável por analisar regras de negócio e funções desejadas, entregando as especificações, descrevendo documentos de análise para facilitar a implementação pelos desenvolvedores, além de analisar a viabilidade técnica, impacto e gerenciar a rastreabilidade dos requisitos.

5.4.3 ANALISTA DE SUPORTE

Responsável por recepcionar as solicitações de clientes, analisar, resolver ou encaminhar para solução, sanar dúvidas de clientes que são relacionadas ao sistema, registrar informações sobre o atendimento.

5.4.4 ANALISTA DE TESTES

Responsável por coordenar as atividades de testes, executar testes e revisar os requisitos de sistemas com o objetivo de validar se o sistema desenvolvido atende os requisitos solicitados pelo cliente, bem como verificar se a correção ou adequação desenvolvida está se integrando adequadamente ao sistema já em produção.

5.4.5 TESTER

É responsável por executar testes e revisar os requisitos de sistemas com o objetivo de validar se o sistema desenvolvido atende os requisitos solicitados pelo cliente, bem como verificar se a correção ou adequação desenvolvida está se integrando adequadamente ao sistema já em produção.

5.4.6 DESENVOLVEDOR DE SISTEMAS JAVA

Responsável por elaborar codificação para o case fornecido, valendo-se de métodos e técnicas adequadas aos equipamentos e aplicações a que se destinam, prestar manutenção corretiva aos cases do sistema.

5.4.7 DESENVOLVEDOR DE SISTEMAS RESPONSÁVEL POR RELEASE

É responsável por criar e controlar versões estáveis do produto de software do projeto, preparar ambiente de atualização, realizar upload de arquivos para servidores de clientes e realizar as atualizações.

5.4.8 GERENTE DE PROJETO

É responsável por realizar as atividades de gerência dos projetos, garantindo o planejamento, execução e finalização dos mesmos, tomando atitudes eficazes quando o desempenho do projeto desviar significativamente do planejamento realizado.

5.5 DESENVOLVIMENTO DE GUIAS

Uma das atividades definidas pelo modelo MPS.BR é a criação de guias referentes a processos, tarefas e de qualquer tipo de atividade executada na empresa. É obrigatória a existência destes guias para uma definição clara de tarefas e para facilitar a compreensão de novos funcionários. Esses guias devem ser disponibilizados para conhecimento funcional interno da organização, de maneira que possam ser visualizados e até modificados, desde que a pessoa possua permissão para tal. O controle destes documentos fica a critério da organização, sendo que na empresa em questão foi definido o uso de um repositório para controlar arquivos dessa natureza.

Alguns exemplos de guias desenvolvidos serão listados abaixo:

- Guia de dados;

- Guia de ciclo de vida de projetos;
- Guia de controle de versões e atualização de projetos;
- Guia de gerenciamento de projetos;
- Guia de gerenciamento de requisitos;
- Guia de estimativas;
- Guia de suporte;
- Guia de uso do Director (Sistema de controle de requisições);

5.6 CONTROLE DE REQUISIÇÕES E SUPORTE: DIRECTOR

Toda empresa desenvolvedora de software possui ou ao menos deveria possuir um sistema para controle de requisições. A empresa em questão optou por desenvolver seu próprio sistema internamente, tendo assim uma ferramenta personalizada especialmente para atender as necessidades deste quesito.

O Director é um sistema voltado para cadastro e controle de requisições, controle de suporte e gerência de estimativas de projetos. Possui um sistema de cadastro e controle de projetos, clientes, cases, requisições com suas características (previsão, controle de status, tempo gasto para realização de cada atividade do processo, responsável, iterações, controle de dados, prioridade, tipo etc...), controle de releases, atualizações de projetos e clientes, e um modelo de 'WBS' próprio para uma visão mais ampla dos projetos, produtos e seus relacionamentos.

5.6.1 DETALHES DO CADASTRO DE UMA REQUISIÇÃO

Projeto: Toda a requisição é associada a um sistema desenvolvido pela empresa, neste item é selecionado o sistema ao qual a requisição está sendo associada. Em casos onde não há um sistema envolvido é utilizado o sistema Atendimento GERAL.

Case: Campo não obrigatório, a requisição pode ser associada a um case pré-cadastrado, facilitando assim sua rastreabilidade.

Cliente: Os clientes estão pré-cadastrados no sistema Director, com a nomenclatura seguindo nome do funcionário + nome da instituição. A requisição deve ser associada ao

solicitante, pois ao final ele será avisado da conclusão da sua solicitação. Para requisições que tratam de quesitos internos existe um cliente interno.

Responsável: Este campo corresponde ao colaborador que receberá a requisição inicialmente e irá gerenciá-la. Ao abrir uma nova requisição a mesma deve ser enviada para triagem.

Prioridade: A prioridade é definida como Baixa, Média, Alta, Urgente e Crítica. Onde a prioridade crítica é utilizada somente para questões que não podem esperar a próxima atualização, que devem ser desenvolvidas para atualização pontual.

Tipo: Toda requisição deve ser associada a um tipo. Deve-se dar atenção ao tipo e prioridade de uma requisição. Bugs devem ser cadastrados como urgentes para situações corriqueiras, e devem ter prioridade crítica para situações que comprometem o funcionamento de algum módulo do sistema. Lembrando que relatórios, tanto com tipo de Relatório ou Personalização não devem possuir a prioridade crítica. O tipo Suporte deve ser utilizado unicamente para requisições associadas a suporte, como entrar em contato com cliente para solucionar alguma questão. Migração é o tipo de requisição a ser utilizado para migração de dados.

Título: Todas as requisições devem possuir um Título, este deve ser escrito em caixa baixa, com a inicial maiúscula.

Descrição: O corpo da requisição deve ser descrito de forma clara a todos os colaboradores, deve conter todos os dados vistos junto ao cliente ou no teste. Neste item deve ser utilizada escrita formal, e a ortografia deve ser observada, pois quesitos aqui descritos podem ser utilizados em emails ou documentos posteriores.

Quando a requisição tratar de correção de bugs ou relatórios, todos os parâmetros utilizados em teste devem ser informados. Se possível utilize ID dos parâmetros. Exemplo: Edital de Vestibular ID = 18 Descrição = Vestibular de Verão.

Nos casos de urgência deve ser informada a data para a qual o cliente necessita a funcionalidade no sistema. Exemplo: O vestibular ocorrerá no dia 02/03/11, e o cliente necessita requisição concluída para dia 01/03/11.

Ao cadastrar uma requisição o colaborador também conta com o material de apoio, sendo imagens, documentos etc..., este material de apoio é encaminhado após salvar a requisição. Deve ser feita a pesquisa da requisição cadastrada e feito um encaminhamento de anexo.

As Figuras 12 e 13 ilustram as telas de cadastro e consulta de requisições do Director.

The screenshot shows a web application window titled "Cadastro de Requisição". The window has a standard Windows-style title bar with a close button. Below the title bar, there is a "Sobre" (About) section with an information icon and the text "Informe os dados da requisição...". The main area is labeled "Dados da Requisição" and contains several input fields: "Código:" (text), "Projeto:" (text with a dropdown arrow), "Case:" (text), "Cliente:" (text), "Responsável:" (text with a dropdown arrow), "Prioridade:" (text with a dropdown arrow), "Tipo:" (text with a dropdown arrow), "Horas Previstas:" (text with a dropdown arrow), and "Data Prev. Conclusão:" (date picker). Below these fields is a "Titulo:" label and a text input field. A rich text editor toolbar is visible, containing icons for undo, redo, bold, italic, underline, strikethrough, text color, background color, bulleted list, numbered list, indent, outdent, link, unlink, and source code. Below the toolbar are two dropdown menus, one showing "(none)" and the other showing "(default)". At the bottom of the window, there is a row of buttons: "Inserir", "Editar", "Salvar", "Cancelar", "Apagar", and "Voltar".

Figura 12: Director – Tela de cadastro de requisição

Figura 13: Director – Tela de consulta de requisição

5.6.2 CADASTRO DE DADOS DA ANÁLISE DE SISTEMAS

Os dados da análise são cadastrados pelo analista de sistemas no Sistema Director após o cadastro da descrição de requisição, pois é necessária uma requisição cadastrada para inserção dos dados analíticos da requisição.

Após termos uma descrição cadastrada e salva, criamos então: Atualização, Atores e Condições, Fluxo, Exceções, Protótipos. Fazemos esta alteração através do item Especificações, conforme mostra a Figura 14.

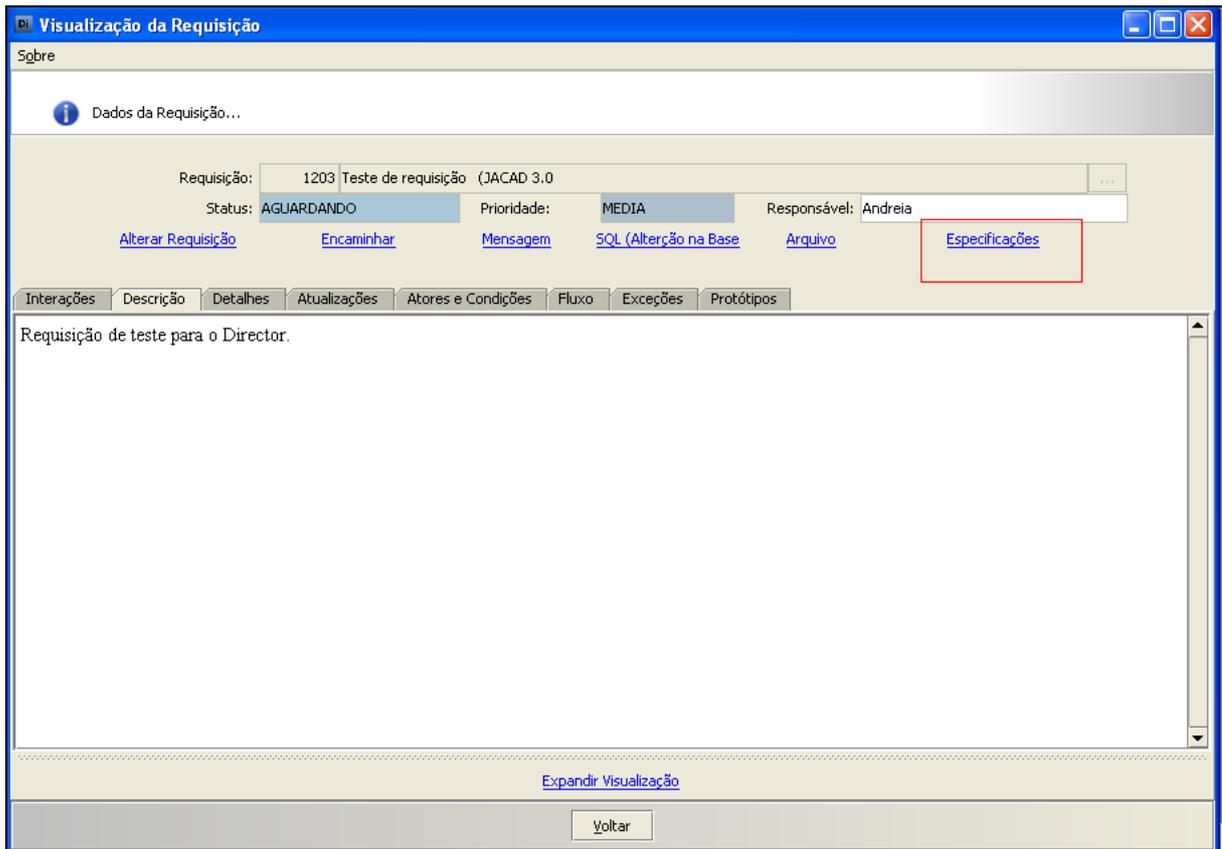


Figura 14: Director – Tela de Cadastro de dados de análise

5.6.3 CADASTRO DE ATENDIMENTOS - SUPORTE

Projeto: Relativo ao sistema a qual o atendimento foi prestado, estes estão Pré-cadastrados.

Funcionário Cliente: Nome do funcionário ao qual foi prestado atendimento, nomenclatura: nome do funcionário + nome da instituição. Os funcionários clientes são pré-cadastrados no sistema.

Forma de Atendimento: Canal utilizado para atendimento. As formas de atendimentos já estão cadastradas no sistema. Exemplo: Skype, MSN, E-mail

Início e Fim: Campos destinados ao registro de intervalo de data e/ou hora onde ocorreu o atendimento.

Tempo de Duração: Campo onde o sistema apresenta a duração do atendimento.

Contabilizar como horas trabalhadas: CheckBox para contabilização ou não do tempo utilizado no atendimento.

A Figura 15 demonstra a tela de cadastro de atendimentos do Director:

Figura 15: Director – Tela de cadastro de Atendimento/Suporte

5.6.4 COMO CADASTRAR UM ATENDIMENTO - SUPORTE

Um atendimento deve ser cadastrado informando os parâmetros solicitados pelo cabeçalho do mesmo. O início de um atendimento se dá a partir do momento que o colaborador recebe o contato de um cliente e passa a atendê-lo. Todo o período de tempo utilizado em verificações para o atendimento deve ser contabilizado no registro do atendimento. Ao clicar no botão Registrar Horário o sistema registra automaticamente o dia e horário do término do atendimento, disponibilizando a duração do mesmo.

O item Contabilizar como horas trabalhadas, contabiliza o atendimento ao cliente, como um atendimento de suporte, portanto quando a questão vista tratar de bugs o mesmo não deve ser selecionado. Este item deve ser selecionado em todos os casos onde são solicitados auxílios e personalizações.

No campo descrição o colaborador deve informar a descrição do atendimento de forma resumida, informando requisições cadastradas para o atendimento. Nos casos onde o atendimento se deu de forma escrita em canais de troca de mensagem instantânea, sugerimos que a conversa seja colada no campo descrição para registro efetivo da mesma.

O campo descrição deve ser informado com escrita formal, observando-se a ortografia, pois periodicamente são impressos relatórios que contam com a descrição do atendimento.

5.7 CICLO DE VIDA DOS PROJETOS

A empresa em questão já possui seus principais produtos em atividade, sendo assim o modelo de ciclo de vida dos projetos foi escolhido para que melhor atende-se a realidade atual da empresa. Como os produtos ainda estão em fases de aperfeiçoamento, e pelo fato de existirem constantes personalizações para clientes, foi escolhido o modelo de ciclo de vida iterativo e incremental.

O modelo de ciclo de vida iterativo e incremental foi proposto como uma resposta aos problemas encontrados no modelo em cascata. Um processo de desenvolvimento segundo essa abordagem divide o desenvolvimento de um produto de software em ciclos, conforme ilustrado nas Figuras 16 e 17. Em cada ciclo de desenvolvimento, podem ser identificadas as fases de análise, projeto, implementação e testes.

Cada um dos ciclos considera um subconjunto de requisitos. Os requisitos são desenvolvidos uma vez que sejam alocados a um ciclo de desenvolvimento. No próximo ciclo, outro subconjunto dos requisitos é considerado para ser desenvolvido, o que produz um novo incremento do sistema que contém extensões e refinamentos sobre o incremento anterior. Assim, o desenvolvimento evolui em versões, através da construção incremental e iterativa de novas funcionalidades até que o sistema completo esteja construído. Note que apenas uma parte dos requisitos é considerada em cada ciclo de desenvolvimento. Na verdade, um modelo de ciclo de vida iterativo e incremental pode ser visto como uma generalização da abordagem em cascata: o software é desenvolvido em incrementos e cada incremento é desenvolvido em cascata. (BEZERRA, 2002).

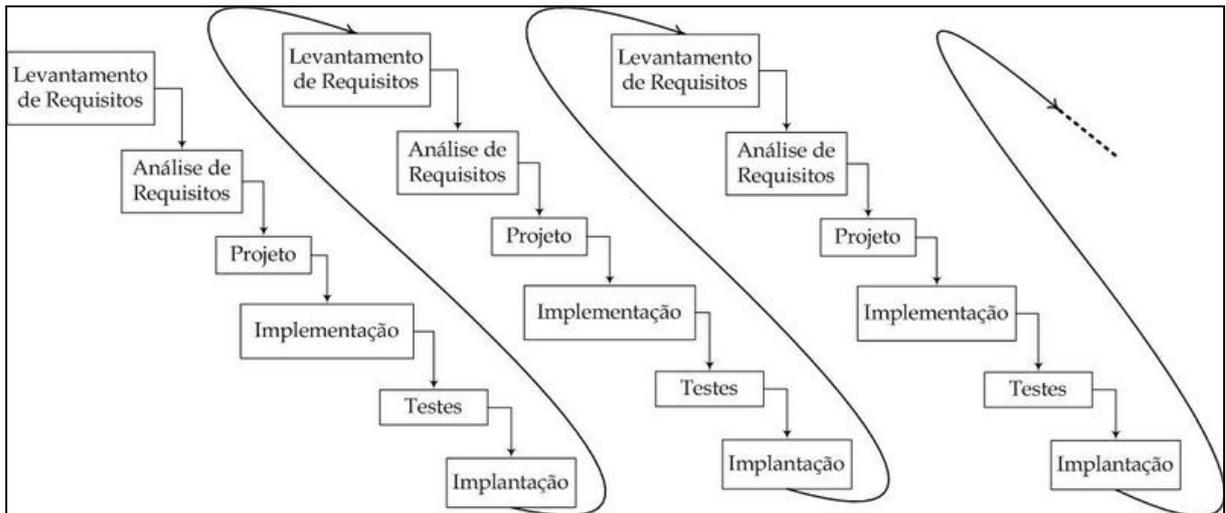


Figura 16: Figura Ciclo de vida Iterativo e Incremental
 Fonte: (<http://www.devmedia.com.br/space.asp?id=172629>)

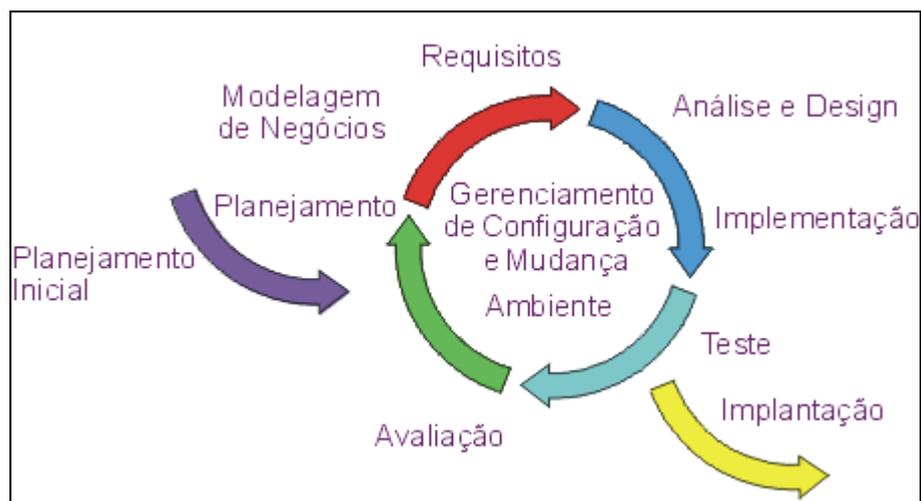


Figura 17: Ciclo de vida Iterativo e Incremental 2
 Fonte: <http://leonwgn.wordpress.com/tag/metodologia/>

A abordagem incremental e iterativa somente é possível se existir um mecanismo para dividir os requisitos do sistema em partes, para que cada parte seja alocada a um ciclo de desenvolvimento. Essa alocação é realizada em função do grau de importância atribuído a cada requisito.

A noção de Processo Iterativo corresponde à ideia de “melhorar (ou refinar) pouco - a - pouco” o sistema (iterações). Em cada iteração a equipe de desenvolvimento identifica e especifica os requisitos relevantes, cria um projeto utilizando a arquitetura escolhida como guia, implementa o projeto em componentes e verifica se esses componentes satisfazem os

requisitos. Se uma iteração atinge os seus objectivos, o desenvolvimento prossegue com a próxima iteração, caso contrário a equipa deve rever as suas decisões e tentar uma nova abordagem.

Portanto, o âmbito do sistema não é alterado, mas o seu detalhe vai aumentando em iterações sucessivas. Um excelente exemplo de aplicação do processo iterativo encontra-se no trabalho artístico, em que o resultado final de uma obra sofre inúmeras iterações. (BEZERRA, 2002).

A noção de processo incremental corresponde à ideia de “aumentar (alargar) pouco-a-pouco” o âmbito do sistema. Uma boa imagem para este atributo é a de uma mansão que foi construída por sucessivos incrementos a partir de uma primeira casa com apenas duas divisões.

Um incremento não é necessariamente a adição do código executável correspondente aos casos de uso que pertencem à iteração em andamento. Especialmente nas primeiras fases do ciclo de desenvolvimento, os desenvolvedores podem substituir um projeto superficial por um mais detalhado ou sofisticado. Em fases avançadas os incrementos são tipicamente aditivos. (BEZERRA, 2002).

5.7.1 VANTAGENS

- Redução dos riscos envolvendo custos a um único incremento. Se os desenvolvedores precisarem repetir a iteração, a organização perde somente o esforço mal direcionado de uma iteração, não o valor de um produto inteiro;
- Redução do risco de lançar o projeto no mercado fora da data planejada. Identificando os riscos numa fase inicial o esforço despendido para gerenciá-los ocorre cedo, quando as pessoas estão sob menos pressão do que numa fase final de projeto;
- Aceleração do tempo de desenvolvimento do projeto como um todo, porque os desenvolvedores trabalham de maneira mais eficiente quando buscam resultados de escopo pequeno e claro;

- Reconhecimento de uma realidade freqüentemente ignorada: as necessidades dos usuários e os requisitos correspondentes não podem ser totalmente definidos no início do processo. Eles são tipicamente refinados em sucessivas iterações. Este modelo de operação facilita a adaptação a mudanças de requisitos;

5.7.2 DESVANTANGES

- Dificuldade de gerenciamento. Isso ocorre porque as fases de do ciclo podem estar ocorrendo de forma simultânea;
- O usuário pode se entusiasmar excessivamente com a primeira versão do sistema e pensar que tal versão já corresponde ao sistema como um todo;

Como todo modelo esta sujeito a riscos de projeto:

- O projeto pode não satisfazer aos requisitos do usuário;
- A verba do projeto pode acabar;
- O sistema de software pode não ser adaptável, manutenível ou extensível;
- O sistema de software pode ser entregue ao usuário tarde demais;

5.8 CONTROLE DE VERSÕES E ATUALIZAÇÕES

O controle de versões dos projetos da empresa em questão é realizado através do programa *SUBVERSION* (SVN), capaz de controlar cada modificação nos arquivos referentes aos projetos, sendo que o tempo de uma versão para outra é determinado por projeto. O padrão é que seja realizada uma atualização a cada quinze dias, sendo que para cada atualização é gerada uma nova versão do produto (*release*) a qual será disponibilizada aos clientes após a homologação.

Após a fase de testes, as requisições estão prontas para a criação de uma nova versão do projeto. O responsável pela atualização do projeto deve verificar as requisições de

atualização para a nova versão, e utilizar o ambiente (*workspace*) de controle de versões para gerar a nova versão com as modificações necessárias. Para isso é necessário verificar os arquivos alterados nas requisições, e fazer atualização dos mesmos através da versão de desenvolvimento (*branch*) do projeto. A necessidade de se ter um workspace de atualização para o projet se da devido o fato de que pode haver alterações que não dizem respeito à atualização desejada, e pelo mesmo motivo a nova versão não pode ser gerada a partir de uma cópia do workspace de um desenvolvedor.

Após realizar a atualização dos arquivos da cópia de desenvolvimento do projeto do *workspace* de atualização, deve-se gerar uma nova versão estável do projeto, conhecida como TAG. Essas versões ficam armazenadas no SVN no diretório contendo o nome do projeto, dentro ta pasta TAG. A nova versão deve conter um número predecessor, ex: Sistema2.0.4.

Após a criação da nova versão do projeto, deve-se fazer um '*checkout*' do projeto na versão TAG, verificar sua estrutura para evitar erros, compilar o projeto e encaminhar todos os arquivos para que o setor de testes realize um teste geral, utilizando as principais funcionalidades do sistema e caminhos críticos para certificar que as novas alterações não afetaram partes distintas do mesmo. Ao finalizar esta tarefa, o setor de testes confirma as alterações em uma lista e informa que a atualização já pode ser realizada.

É importante manter o registro de quais arquivos foram enviados aos servidores na atualização de maneira organizada. Para isso, o responsável pela atualização deve possuir/criar uma pasta para armazenar tais dados. O padrão utilizado até o momento é a utilização de uma pasta 'Atualização' no diretório raiz do computador utilizado, dentro dela criar sub-pastas contendo a data em que cada atualização foi realizada.

Dentro da pasta com a data da atualização a ser realizada, deve-se criar sub-pastas referentes a projetos, e dentro das sub-pastas deve-se colocar os arquivos para a atualização. Ainda pode-se criar mais um nível de sub-pastas para diferenciar os tipos de arquivos ex: compilados, relatórios.

As alterações nas bases de dados dos sistemas também devem ficar armazenadas em um arquivo de texto dentro da pasta da atualização, com um nome comum como 'SQL'.

Cada cliente possui um tipo de servidor/conexão, e por este motivo são necessárias varias formas de atualização. Para cliente que utilizam servidores LINUX, a atualização se dá através da utilização com SSH (*Secure Shell Client*). Para servidores WINDOWS pode se utilizar o upload dos arquivos através de um cliente, ou através de uma VPN (*virtual private network*), conexão remota ou *TEAM VIEWER*.

A atualização deve ser realizada dentro da data prevista e acordada com os clientes, e geralmente fora de horário de expediente ou de pico.

5.9 REQUISITOS E RASTREABILIDADE

Conforme determinado pelo modelo MPS.BR, a gerência de requisitos é uma das tarefas mais importantes para empresa. É a partir dos requisitos que todo fluxo de trabalho será realizado, por isso deve-se tomar muito cuidado na hora da captação e entendimento, assim como na análise dos requisitos. Na empresa em questão foram verificados os seguintes fornecedores de requisitos:

5.9.1 FORNECEDORES DE REQUISITOS

- Canal de suporte e atendimento:

O suporte atende diariamente os clientes da empresa, registra suas reclamações e sugestões, e os ajuda com a utilização do sistema. É comum, que por este canal os clientes solicitem alterações, relatem bugs e façam sugestões ao sistema que já possuem. Estes quesitos que são absorvidos e registrados pelo canal de suporte da empresa podem tornar-se requisitos para projetos de manutenção.

Nestes casos o responsável pelo atendimento que registra a requisição é tido como o fornecedor do requisito. Em explicação a esta afirmação temos o principio que o responsável pelo suporte conhece tanto o sistema do qual o requisito diz respeito quanto à solicitação do cliente da empresa que utiliza o sistema, podendo ser responsável assim pelo requisito que realizará o solicitado pelo cliente atendido. É o responsável pelo atendimento em específico que responde como fornecedor do requisito registrado por este, quando necessário o fornecedor interno tem abertura para consulta do cliente solicitante, para decisões como, por exemplo, a alteração do requisito por inviabilidade técnica.

A triagem das requisições que farão parte do projeto é realizada antes da abertura do projeto, pois define a idéia da concepção do projeto. Esta triagem é realizada pelo analista do negocio responsável pela triagem de requisições.

- Por atendimento personalizado:

Ao elaborar personalizações exclusivas a um cliente o atendimento para levantar de requisitos é tido de forma personalizada, via canais ou in locos, neste caso o fornecedor de requisitos é externo, sendo ele um representante da instituição que solicita os quesitos.

O responsável pelo levantar destes requisitos pode ser então o analista de negócios. É importante lembrar que o responsável pelo levantar de requisitos neste caso não pode ser o gerente de projeto ou analista de sistemas que analisará a definição dos requisitos, pois estes estão assumindo papel de gerência de requisitos e projeto administrando e gerenciando os requisitos, por tanto não é plausível a idéia que o responsável por estabelecer os requisitos gerencie os mesmos.

O fornecedor de requisitos para o projeto é identificado no documento Termo de abertura. Se novo fornecedor for adicionado ao projeto através do controle de mudanças, o registro da sua adesão é evidenciado através do documento preenchido e aprovado 'Solicitação de alteração' e a documentação afetada, como exemplo Plano de projeto etc.. é alterada.

5.9.2 RASTREABILIDADE

A rastreabilidade ajuda a entender o relacionamento entre os produtos de trabalho, quer sejam eles especificações de requisitos, código, arquitetura, testes e vários outros e nos ajuda a garantir a integridade entre estes elementos.

O objetivo é realizar o rastreamento a partir do topo com o levantamento dos requisitos até a verificação da realização do teste na requisição, ou começar a partir do código fonte e chegar até o requisito que o originou.

O fluxo inicia a partir do cadastro de requisitos no sistema Director que será utilizado para realizar a análise de rastreabilidade, um ou mais requisitos podem ser associados a um case (Funcionalidade dentro do sistema) que serão documentados pelo analista de sistemas. Tendo os cases prontos são cadastradas requisições que contem as especificações do desenvolvimento no Director para serem enviadas ao desenvolvimento e posteriormente para teste. Os cases criados pelo analista são associados a classes no código fonte. Este por sua vez, possui uma anotação nas classes identificando para qual case o código fonte faz

referência. Os cases podem se relacionar entre eles. Todo este fluxo deve ser fiscalizado pelo analista de sistema.

Tendo todos os relacionamentos entre requisitos, cases, requisições e código fonte corretamente, pode ser analisado através da ferramenta Director utilizando de consultas e relatórios os resultados do controle da rastreabilidade em todos os seus tipos: vertical, horizontal e bi-direcional.

5.9.3 RECURSOS

1. Director (Software para registro e controle de requisições e atividades desenvolvidas na empresa);
2. Participantes;
 1. Analista de Negócio: Realiza o levantamento dos requisitos no cliente e pode realizar o cadastramento dos requisitos no Director;
 2. Suporte: Realiza o cadastro da requisição no Director contendo os requisitos solicitados pelo cliente de forma resumida;
 3. Analista de Sistema: Obtem os requisitos informados pelo analista de negócio, realiza a criação dos cases no Director, associa os cases aos requisitos levantados e realiza a criação de requisições com as especificações de desenvolvimento para serem encaminhadas para a equipe e realiza o controle da rastreabilidade através do Director e Relatórios;
 4. Gerente de Projetos: Realiza o encaminhamento das requisições criadas pelo analista de sistema para a equipe de desenvolvimento;
 5. Desenvolvedor: Realiza o desenvolvimento conforme a requisição criada pelo analista de sistema;
 6. Testador: Realiza o teste da requisição criada pelo analista de sistema;

5.9.4 ESTRATÉGIA

1. Conforme ilustrado na Figura 18, o processo pode ser iniciado através do analista de negócio cadastrando os requisitos no Director ou pelo suporte através do cadastro de uma requisição no Director contendo uma descrição macro da solicitação do cliente;

2. Detalhes do processo:
 - a. O analista de negocio ou o suporte cria uma requisição com a descrição macro dos requisitos levantados;
 - b. Após o aceite dos requisitos levantados pelo analista de negocio a requisição criada deve ser encaminhada para o analista de sistema que irá realizar a criação ou associação dos requisitos com os seus devidos cases;
 - c. O analista de sistema realiza a criação das novas requisições conforme os requisitos passados contendo as especificações para o desenvolvimento, define as classes e métodos (código fonte) que serão criadas e associa elas ao case através do código de identificação do case e realiza o encaminhamento da requisição para o gerente de projetos distribuir para a equipe;
 - d. Ao cadastrar os requisitos o analista de sistema deve informar a qual atualização o requisito pertence;
 - e. O analista ainda deve realizar a associação entre cases no sentido horizontal para definir o impacto de um case no outro;
 - f. O gerente de projetos encaminha as requisições por sua vez para o desenvolvimento;
 - g. O desenvolvedor ao criar uma classe conforme a especificação de desenvolvimento deve adicionar uma documentação no código fonte com a referência para a numeração (Código) do case;
 - h. Após o desenvolvimento a requisição é encaminhada para teste, o teste é associado à requisição através do encaminhamento, e então o processo é finalizado.

Observa-se aqui que todos os encaminhamentos são associados à requisição, o que inclui um histórico de execução desde o cadastro da requisição com os requisitos até termo a requisição homologada, o cliente comunicado e a requisição tida como concluída.

3. Após o processo finalizado, o analista de sistemas tem acesso através do Director à consulta de rastreabilidade e ao relatório de rastreabilidade que auxiliam o analista na tomada de decisões e controle do desenvolvimento das atividades do projeto.

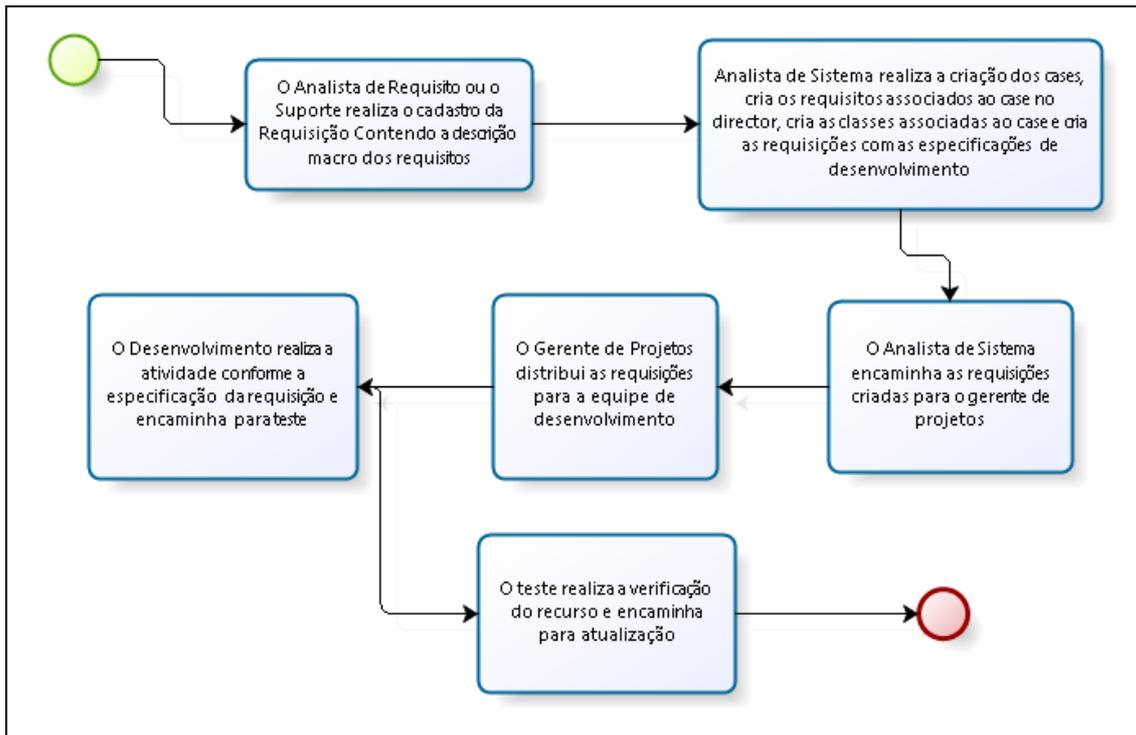


Figura 18: Diagrama do processo de rastreabilidade

5.10 DEFINIÇÃO DOS PRINCIPAIS PROCESSOS (FLUXO DE TRABALHO)

Após a definição de todas as etapas citadas anteriormente, pode-se visualizar concretamente o fluxo de trabalho da empresa. A ordem e a definição dos processos que serão comentados assegurar, foram definidos através das necessidades da empresa, juntamente com as boas práticas descritas pelo modelo MPS.

Os processos podem ser alterados a qualquer momento, desde que seja detectada a necessidade. Mas é extremamente importante a definição dos mesmos para o início do projeto piloto do programa MPS.BR. É a partir dele que toda mudança organizacional passará a ser medida e avaliada.

5.10.1 PLANEJAMENTO

Quase todas as atividades que desempenhamos precisam ser planejadas e gerenciadas. Atividades de engenharia requerem um planejamento cuidadoso. Normalmente, os produtos são complexos, com muitas partes envolvidas, e as atividades são realizadas por um grupo muito grande de pessoas.

Na engenharia de software não é diferente. Um projeto de engenharia de software deve estar baseado em um planejamento para que se possa gerenciar adequadamente a sua realização.

Planejar é estimar quais as atividades deverão ser realizadas; quem deverá realizá-las; quando devem ser realizadas e finalizadas; e quanto elas deverão custar. Tudo isto requer a elaboração de estimativas em relação ao número e à dimensão dos artefatos, do número de pessoas necessárias, dos prazos e dos custos.

Para isto, a atividade de planejamento deverá resultar:

- Na realização de estimativas;
- Na elaboração da estrutura de divisão do trabalho (WBS – Work Breakdown Structure);
- Na definição da equipe e demais recursos;
- Na alocação de pessoa-atividade;
- Na elaboração do cronograma;
- Na elaboração do orçamento;

Além disso, a análise de riscos e as revisões periódicas do plano são fundamentais para garantir que ele seja cumprido.

O planejamento está diretamente ligado ao processo de software. Um processo de software é resultado do planejamento associado a um modelo de processo (método). O modelo de processo indica quais atividades devem ser realizadas e qual a dependência entre elas. Por exemplo, no modelo iterativo incremental, deve-se especificar os requisitos, desenhar a arquitetura, fazer o design detalhado, implementar as unidades, integrar as unidades e, finalmente, entregar o software. O planejamento deve indicar quem faz estas

atividades, quanto tempo é necessário para eles, quando elas serão realizadas e quanto custará cada um delas.

Sem um planejamento adequado, os desenvolvedores não saberão o que fazer e não terão datas para iniciar ou terminar o trabalho. Sem estimativas, o responsável pelo planejamento não terá como dimensionar o tamanho e a quantidade dos artefatos a serem produzidos e o esforço necessário para produzi-los. Por fim, sem um orçamento, não se terá noção quanto o software irá custar e se haverá recursos para o desenvolvimento.

A Figura 19 ilustra as etapas do planejamento da empresa em questão:

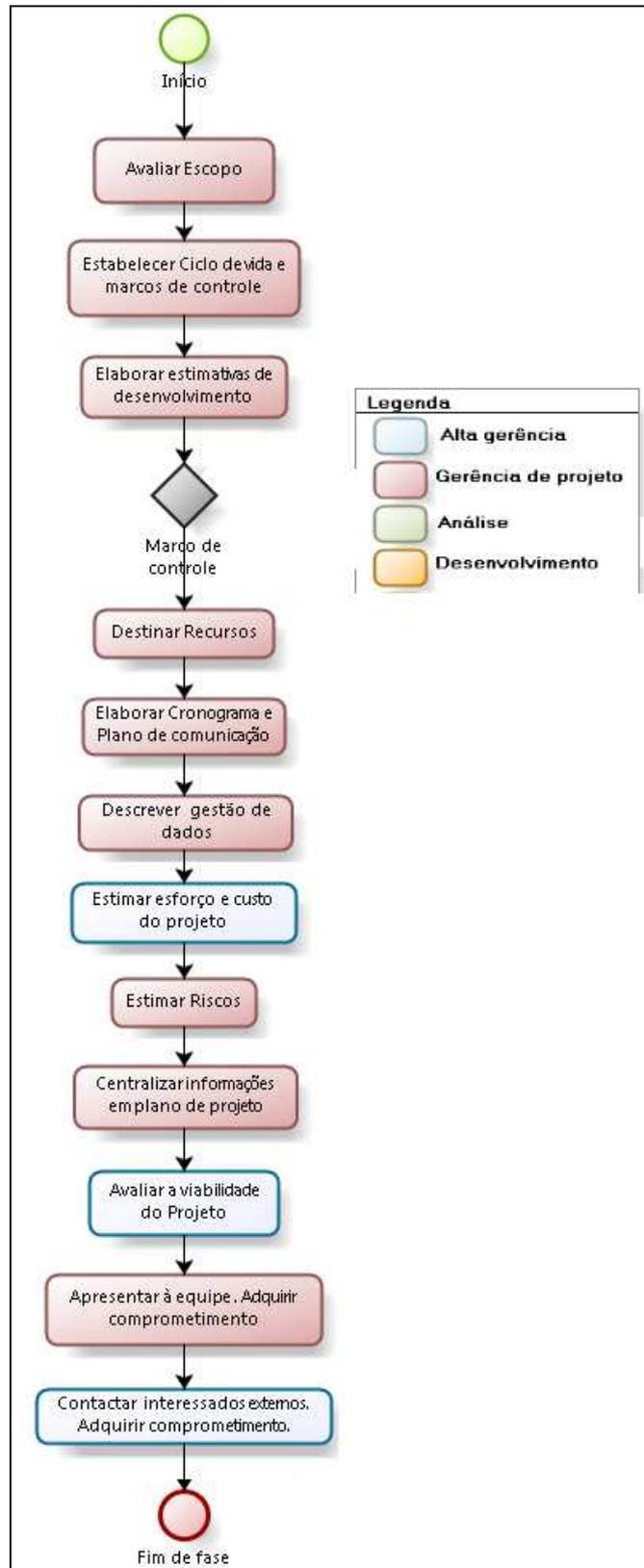


Figura 19: Processos – Planejamento

5.10.2 CONSTRUÇÃO

A construção enfoca mais a codificação e a depuração, mas também depende de outras atividades. Por se tratar de uma atividade complexa, certamente irá tomar a maior parte do desenvolvimento de um projeto de software. Segundo Mcconnel (2002), ela pode tomar até 80% de um projeto, e isso pode significar que outras atividades serão deixadas de lado. Qualquer atividade esquecida em um projeto de software resulta em problemas futuros de manutenção, e a manutenção é a fase mais longa do ciclo de vida de um software. Quando atividades como documentação são deixadas de lado para priorizar codificação, o conhecimento acaba ficando na cabeça dos desenvolvedores, que em outro momento podem não fazer mais parte da equipe.

Por esses motivos, foi determinado, como parte desta etapa, que a análise e documentação de cada requisito do projeto sejam realizadas de maneira em que possa ser facilmente encontrada e compreendida quando necessário. E é nesse processo que também estará envolvida a questão de rastreabilidade. Nesse momento se vê a importância de se conseguir encontrar um determinado código através de uma requisição e vice-versa. Ou até mesmo ter uma descrição mais explícita de uma determinada funcionalidade do sistema.

O processo de construção está ilustrado na Figura 20:

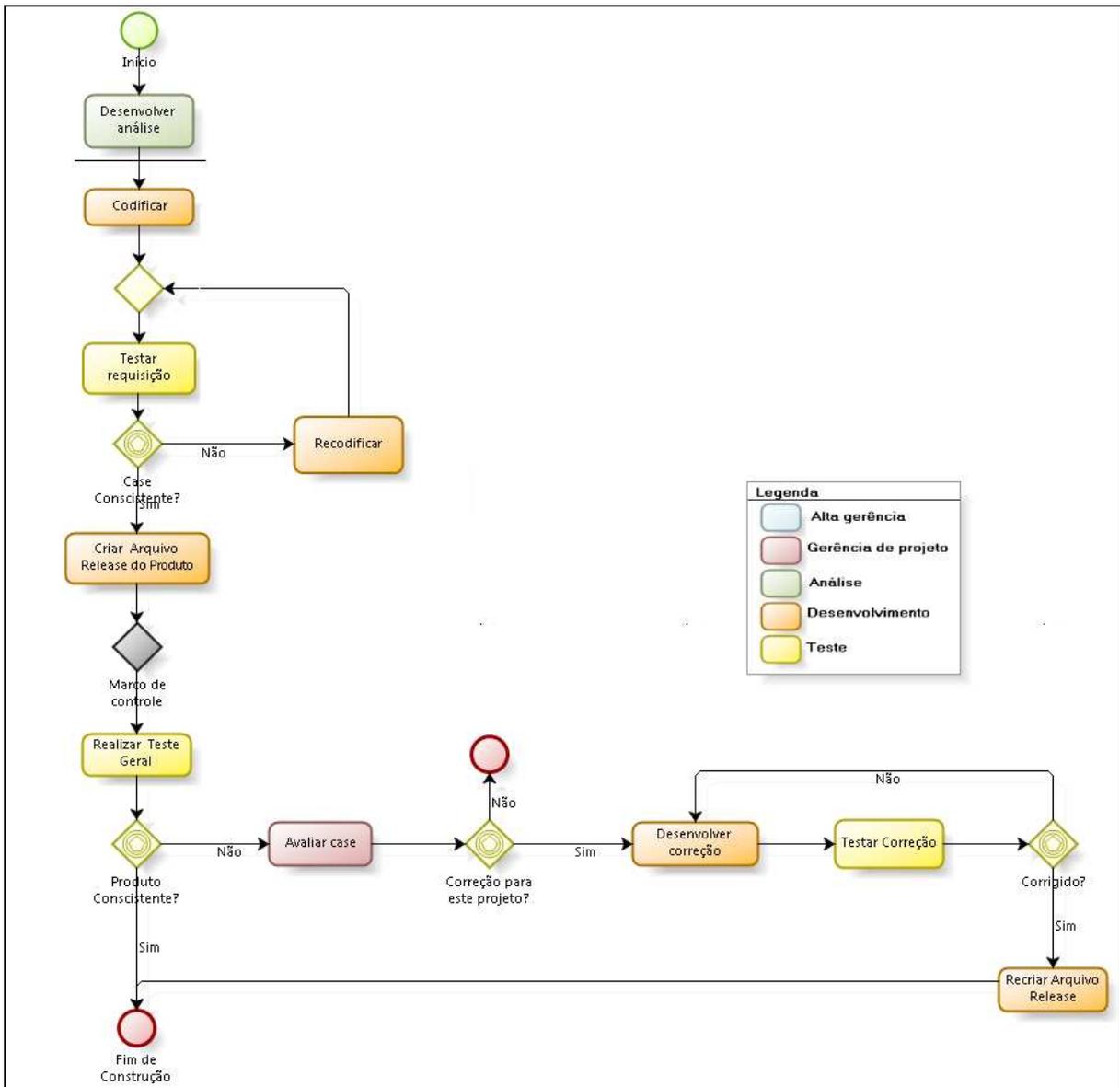


Figura 20: Processos – Construção

5.10.3 IMPLANTAÇÃO

A fase de implantação é de extrema importância, e requer atenção na realização das tarefas que dizem respeito. Nesse momento a nova versão do sistema deve ser instalada nos clientes, mas vários detalhes devem ser levados em consideração.

É importante realizar um backup dos arquivos antigos referentes ao sistema, e da base de dados que poderá sofrer modificações, sendo assim o responsável por esse processo estará preparado para qualquer imprevisto, com a capacidade de reativar o sistema na sua versão anterior da maneira mais rápida possível.

A fase de homologação também deve ser feita com muita atenção, pois diz respeito a qualidade do produto de software ou sistema, o qual os clientes irão ter acesso em próximo momento. Na fase de homologação podem ser detectados problemas que não foram possíveis de serem previstos em um ambiente de testes, como por exemplo, uma arquitetura de hardware diferente ou o próprio sistema operacional, ou até mesmo um fluxo de rede inapropriado.

Após a homologação e validação dos requisitos já em ambiente de produção, é importante ter um meio definido para a comunicação com os clientes ou usuários do sistema, informando das novas modificações ou solicitações realizadas pelos mesmos, juntamente com um acompanhamento e validação de requisitos funcionais, e nível de satisfação dos mesmos. Abaixo a Figura 21 ilustra o processo de implantação:

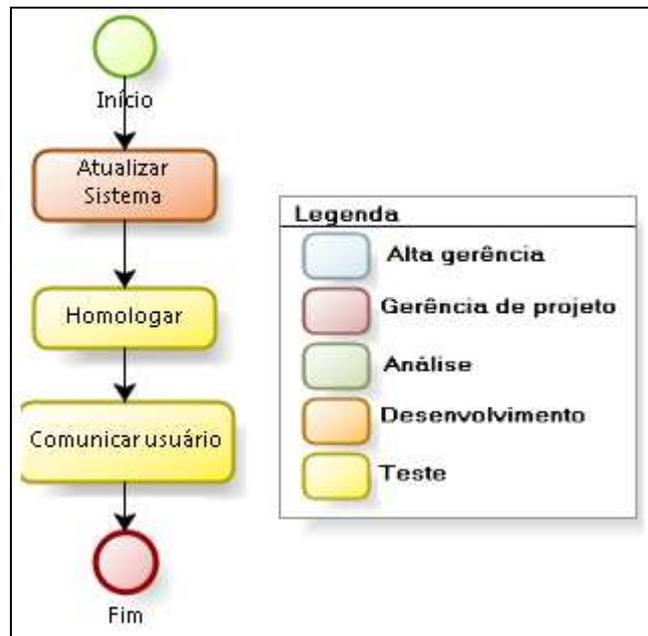


Figura 21: Processos - Implantação

5.10.4 FINALIZAÇÃO

A finalização é uma fase que nem sempre é levada em consideração, mas que estabelece um marco muito importante do projeto, e se tratando de um marco, pode-se obter dados estatísticos através do mesmo ou com o decorrer do tempo.

Esta etapa deve envolver a comunicação com todos os envolvidos do projeto, patrocinadores, funcionários, clientes etc. Deve ser relatada primeiramente a conclusão ou

falha do mesmo, juntamente com o sucesso ou nível de objetivo atingido sobre o planejamento inicial, assim como definir expectativas para futuros trabalhos. Também podem ser emitidos relatórios para futuras estimativa. O processo de finalização está ilustrado na Figura 22:

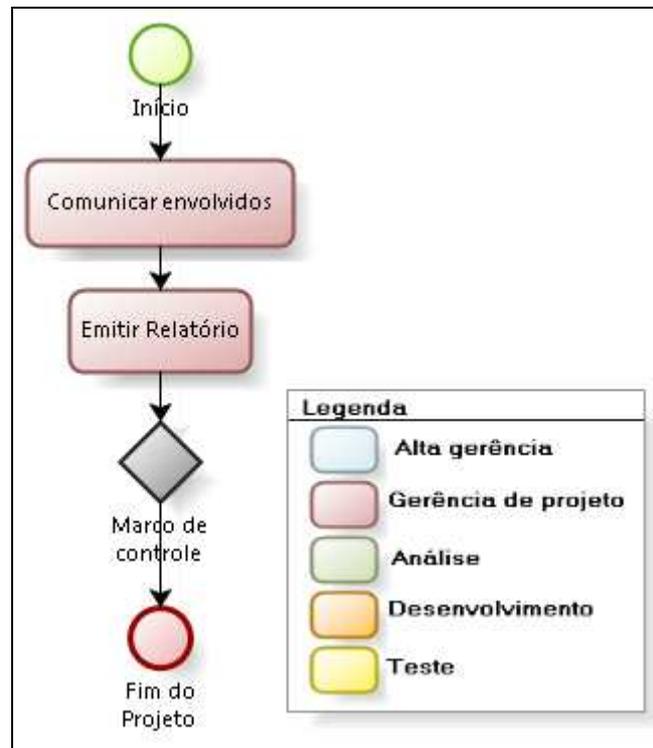


Figura 22: Processos – Finalização

5.10.5 MUDANÇAS

Nenhum planejamento irá conseguir prever todas as alterações que poderão ocorrer. Quanto maior o projeto, mais você terá que lidar com mudanças. Esta é uma das razões porque a metodologia MPS considera que os processos de definição e planejamento não têm que ser perfeitos. O gerente e a equipe do projeto necessitam fazer o seu melhor considerando o que sabem no momento. Após essa etapa, é necessário gerenciar as mudanças que podem vir a ocorrer.

Existem vários tipos de mudanças que podem ocorrer num projeto, uma delas é a mudança de escopo e normalmente há dois problemas que surgem com mais frequência:

- A equipe não investiu tempo suficiente definindo o projeto e/ou

- Houve falha no gerenciamento do Escopo.

Mesmo se o gerente do projeto desenvolver um bom trabalho de definição do escopo, a parte mais difícil será a de gerenciar o projeto dentro do escopo acordado. A finalidade do gerenciamento das mudanças do escopo é proteger a viabilidade dos documentos aprovados do "Termo de Abertura do Projeto" os "Requisitos do Negócio". Em outras palavras, o documento Termo de Abertura do Projeto define o escopo geral do projeto e os Requisitos do Negócio definem as entregas do projeto em maiores detalhes. A equipe do projeto compromete-se com uma data final e um orçamento com base nestas definições gerais e detalhadas do escopo. Se as entregas mudarem durante o projeto (normalmente isto significa que o cliente deseja itens adicionais), então as estimativas de custo, esforço e da duração podem não ser mais válidas.

Se for definida a necessidade de incluir trabalho adicional no escopo do projeto, então o gerente do projeto terá o direito de esperar que o orçamento e os prazos finais correntes também sejam modificados (normalmente aumentados) para refletir este trabalho adicional. Estas novas estimativas sobre o custo, o esforço e a duração do projeto, agora se transformam nas suas metas aprovadas.

Poucos clientes podem prever e expressar todos os requisitos no início do projeto. Conseqüentemente, geralmente há mudanças que necessitam ser introduzidas durante o ciclo de vida do projeto. Estas mudanças podem ser muito necessárias para a solução ou para o produto do projeto, e poderá haver razões válidas para que elas sejam incluídas. O gerente e a equipe do projeto devem reconhecer quando estas mudanças forem solicitadas e, então, seguir um processo pré-definido de gerenciamento das mudanças do escopo.

Outros tipos de mudanças também podem ser levadas em consideração, por exemplo, digamos que um membro da equipe se demite e tem que ser substituído. Isto não seria um exemplo de mudança do escopo do projeto, mas sim uma mudança genérica. Neste caso, poderá ser necessário documentar o fato de que houve uma mudança de recursos, determinar o impacto dessa mudança, definir um plano para lidar com essa mudança, etc. Em muitos aspectos, é seguido um processo similar ao processo de gerenciamento de requisições de mudança do escopo.

Uma das principais diferenças entre o gerenciamento de mudanças genéricas e o gerenciamento de mudanças do escopo é que, se uma mudança do escopo é requisitada e

aprovada, o orçamento e cronograma serão alterados em conformidade. O mesmo não deverá acontecer em mudanças que não estão relacionadas ao escopo.

Como os projetos da empresa em questão são pequenos, fáceis para definir e normalmente são concluídos em um curto tempo, tipicamente eles não têm muitas requisições de mudança do escopo e quando as mesmas são necessárias, normalmente as mudanças são pequenas por natureza. Ficaram então definidos os seguintes processos para gerenciá-las.

- Identificar a requisição de mudança do escopo. As mudanças do escopo podem ser levantadas por qualquer parte interessada do projeto, incluindo a equipe do projeto, os clientes, o patrocinador, etc. Elas devem ser enviadas por escrito ao gerente do projeto através de papel, e-mail, etc. Não requer nenhuma formalidade;
- O gerente do projeto determina se a requisição é de fato uma mudança do escopo. Caso seja, o restante deste processo será executado;
- O gerente do projeto determina qual o impacto que a mudança do escopo terá no projeto em termos de custo, esforço e duração. Se houver múltiplas opções viáveis, o gerente do projeto também deverá determinar os seus impactos;
- (Opcional) Executar as mudanças do escopo que não tem impacto negativo no orçamento ou no cronograma do projeto.
Se a requisição de mudança puder ser acomodada dentro do custo, do esforço e da duração acordada originalmente no projeto, o gerente do projeto e o gerente do cliente tem a flexibilidade para aprovar a mudança. Mas, o patrocinador do projeto deverá concordar antecipadamente com este processo e delegar autoridade. Normalmente o patrocinador estabelece um nível ou um limite de autorização, em termos de custo e de esforço, que o gerente do projeto e o gerente do cliente podem utilizar para autorizar e realizar as mudanças do escopo. A finalidade desse processo de autorização é para evitar que sejam submetidas ao patrocinador muitas mudanças pequenas para a aprovação;

- Apresentar a requisição de mudança do escopo ao patrocinador.
Se a requisição não foi aprovada como no passo 4 acima, os documentos de análise, o impacto e as alternativas apropriadas deverão ser encaminhadas ao patrocinador do projeto para resolução. Se o Patrocinador não aprovar a requisição, a mesma deverá ser encerrada e arquivada;
- Adicionar ao cronograma todas as atividades relativas a requisição de mudança do escopo. Se a requisição de mudança do escopo for aprovada, as atividades apropriadas deverão ser adicionadas ao cronograma para garantir que a mudança será executada;
- Comunicar a requisição de mudança do escopo através do relatório de andamento do projeto. A requisição, a situação atual (status) e a resolução devem ser documentadas no Relatório de Andamento (status) do Projeto;

A Figura 23 ilustra o processo de mudanças:

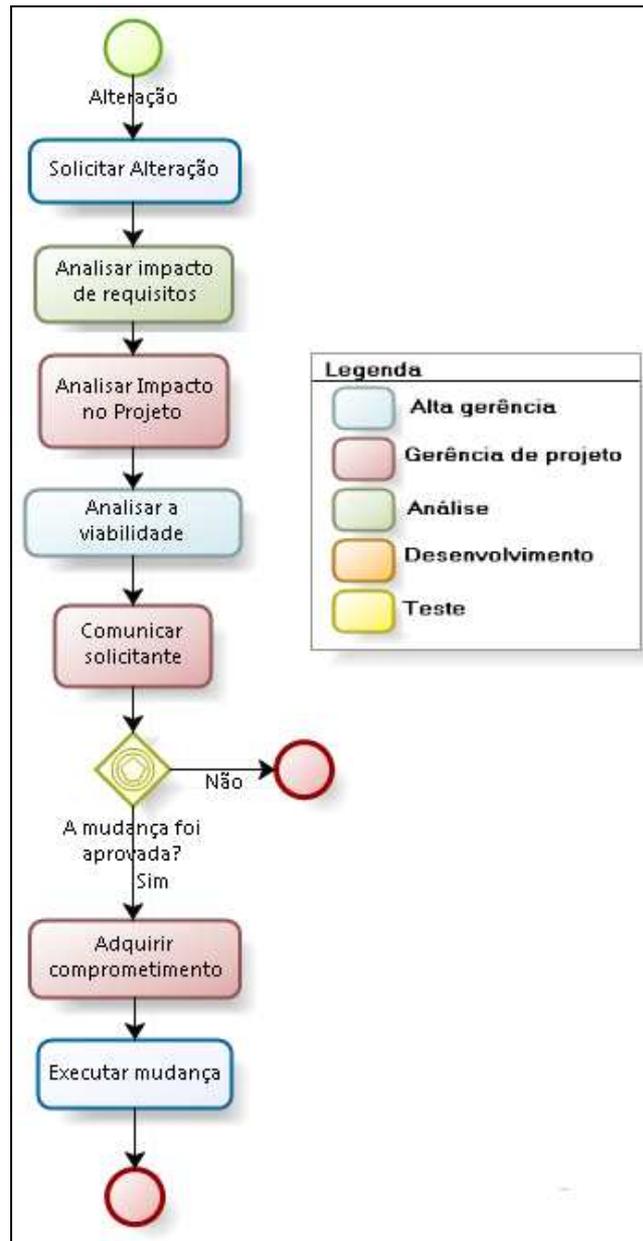


Figura 23: Processos - Mudanças

5.10.6 MONITORAMENTO

Como o próprio nome sugere, este processo compreende o acompanhamento de todas as fases e processos pelo gerente de projetos, juntamente com a possibilidade de se tomar medidas tanto para prevenção de riscos como para correção de problemas, conforme ilustrado na Figura 24.

As tomadas de decisões que poderão resultar em uma mudança de escopo, devem ser realizadas sempre que possível em marcos do projeto, para que um melhor

acompanhamento de todas as partes interessadas sem comprometer tarefas já em andamento. Nessas etapas, reuniões podem ser agendadas com as partes interessadas para discutir sobre possíveis mudanças.

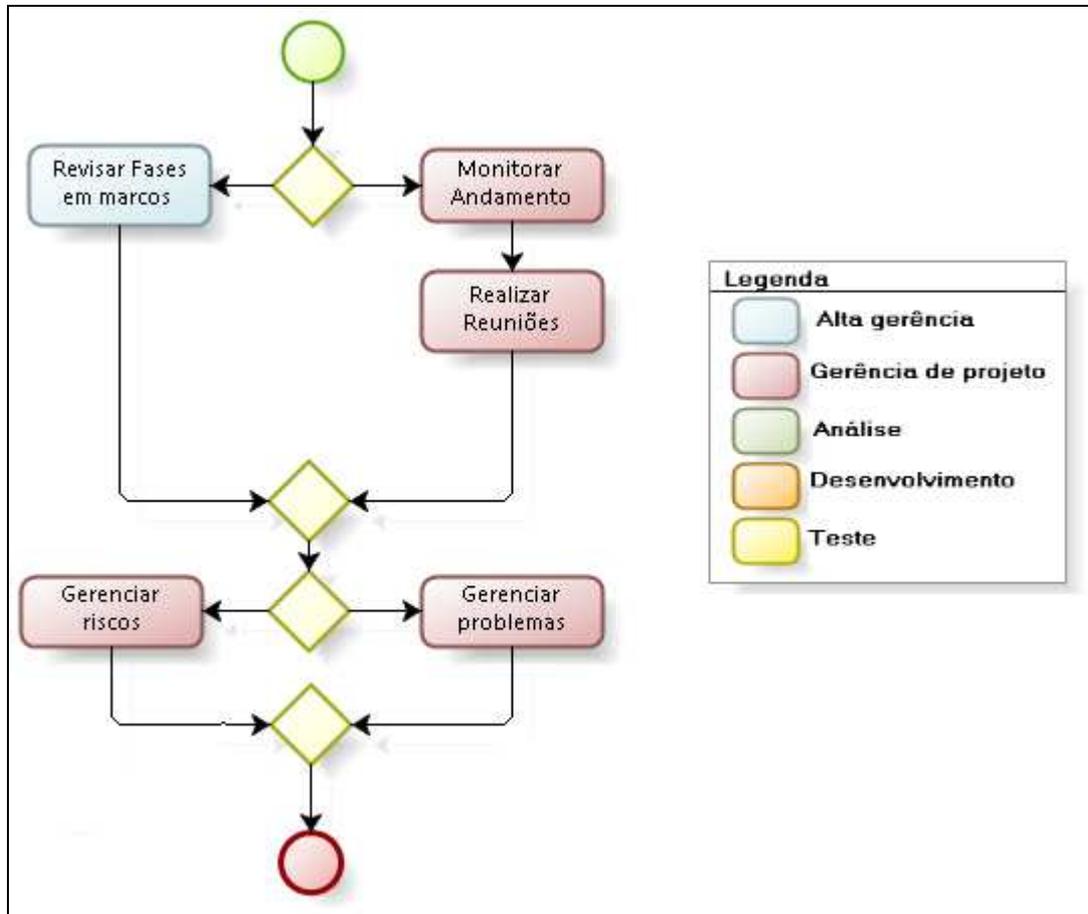


Figura 24: Processos - Monitoramento

6 CONSIDERAÇÕES FINAIS

6.1 CONCLUSÃO

Além de possuir vantagens como custos e prazos reduzidos, o modelo MPS.BR se mostrou bastante flexível em respeito a mudança ou melhoria dos processos internos já executados pela empresa, assim como na implementação de novos processos.

Mesmo possuindo um grande fluxo de trabalho, a empresa não encontrou grandes problemas em deixar de ser uma organização focada na realização de tarefas e passar a trabalhar com a realização de projetos, um dos pontos principais da implantação do nível de maturidade G do modelo, juntamente com a gerência de requisitos.

Em aproximadamente um semestre, foi possível definir os processos, estabelecer padrões, realizar treinamentos de equipe, documentar todas as mudanças assim como os processos e tarefas da organização, e passar a aplicar os novos conceitos no fluxo de trabalho da empresa de maneira graduativa.

6.2 TRABALHOS FUTUROS/CONTINUAÇÃO DO TRABALHO

Podem ser realizados estudos aplicados aos outros níveis de maturidade do MPS.BR, assim como o levantamento de dados e características de outras empresas de portes diferentes que estejam em processo de avaliação, ou até mesmo realizar um estudo de caso sobre algum dos níveis de maturidade do modelo CMMI-Dev aplicados em empresas de pequeno e médio porte no Brasil.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ABNT. ISO/IEC 12207 – Tecnologia Da Informação – Processos De Ciclo De Vida De Software. Rio De Janeiro: ABNT, 1996.

BELLOQUIN, ÁTILA. Qualidade De Software: Um Compromisso Da Empresa Inteira. Infochoose. N. 9, Abril 1999 - Ano I. Disponível Na Internet: <<http://www.choose.com.br/artigos/html/textos/dm0697.htm>> Acessado Em 12 Abril De 2011.

BEVAN, NIGEL. Making Software Usable: Business Gains. Software In Focus. Issue 11 March 2000. Disponível Na Internet: <<http://www.cse.dcu.ie/swinfocus>>. Acessado Em 12 Abril De 2011.

ESPIG, S. R. Iso / Iec 12207. Disponível Na Internet: <<http://www.slideshare.net/espig/iso-iec-12207-297652>>. Acessado Em 17 Maio De 2011.

FIORINI, SOELI T., ET AL. Engenharia De Software Com Cmm, Rio De Janeiro, ED. BRASPORT, 1998.

IEEE STD. IEEE Standard Glossary of Software Engineering Terminology, IEEE, PISCATAWAY, NJ, 1997.

JÚNIOR, JOSÉ BARRETO. Qualidade De Software. Disponível Na Internet: <<http://www.barreto.com.br/frameset.htm>>. Acessado Em 08 Abril De 2011.

MACHADO, CRISTINA F. Definindo Processos Do Ciclo De Vida De Software Usando A Norma Nbr Iso/Iec 12207 E Suas Ementas 1 E 2. UFLA/FAEPE, 2006.

MACHADO, CRISTINA ÂNGELA FILIPAK IN WEBER, KIVAL CHAVES, ET AL. Qualidade E Produtividade Em Software, SÃO PAULO, ED. MAKRON BOOKS, 2001.

FILHO, WILSON DE PÁDUA PAULA, Engenharia De Software, Rio De Janeiro, ED. LTC, 2003.

PRESSMAN, ROGER S., Engenharia De Software, São Paulo, ED. MAKRON BOOKS, 1995.

PRESSMAN, ROGER S., Engenharia De Software, Rio De Janeiro, ED. MCGRAW-HILL, 2002.

REZENDE, DENIS ALCIDES, Engenharia De Software E Sistemas De Informações, Rio De Janeiro, ED. BRASPORT, 1999.

ROCHA, MALDONADO, WEBER. Qualidade de Software: Teoria e Prática, Orgs. Prentice-Hall, São Paulo, 2001.

SEI, SOFTWARE ENGINEERING INSTITUTE, Carnegie Melon University, Disponível Na Internet: <www.sei.cmu.edu>. Acessado Em 25 Abril De 2011.

SOMMERVILLE, IAN, Engenharia De Software, São Paulo, ED. PEARSON EDUCATION, 2003.