

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO
DO RIO GRANDE DO NORTE**

CARLOS HENRIQUE DA CUNHA MELO

**NORMAS E METODOLOGIAS DE DESENVOLVIMENTO DE SOFTWARE
APLICADOS AO MINISTÉRIO PÚBLICO DO TRABALHO**

**NATAL
2008**

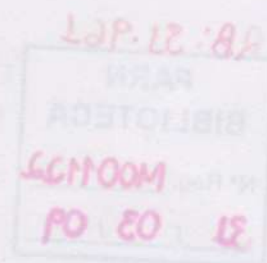
CARLOS HENRIQUE DA CUNHA MELO

**NORMAS E METODOLOGIAS DE DESENVOLVIMENTO DE SOFTWARE
APLICADOS AO MINISTÉRIO PÚBLICO DO TRABALHO**

Monografia apresentada ao Curso de Especialização em Desenvolvimento de Sistemas Corporativos da Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte, como requisito obrigatório para obtenção do grau de Especialista em Desenvolvimento de Sistemas Corporativos.

Orientador: Gleydson Lima, Msc.

Natal
2008



Catalogação na Publicação – Biblioteca da FARN
Setor de Processos Técnicos

Melo, Carlos Henrique da Cunha.

Normas e metodologias de desenvolvimento de software aplicados ao
Ministério Público do Trabalho / Carlos Henrique da Cunha Melo. – Natal:
FARN, 2008.

70f.

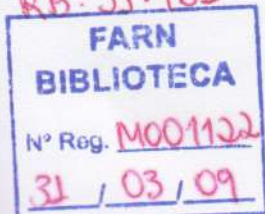
Orientador: Gleydson Lima (Msc.)

Monografia (Especialização em Desenvolvimento de Sistemas
Corporativos) – Faculdade Natalense para o Desenvolvimento do Rio
Grande do Norte.

1. Desenvolvimento de Software – Monografia. 2. Programas de
Computadores – Monografia. 3. Ministério Público do Trabalho –
Monografia. I. Título.

RN/FARN/BC

CDU 004.43



004.43
M485n
2008
ex. 01

CARLOS HENRIQUE DA CUNHA MELO

NORMAS E METODOLOGIAS DE DESENVOLVIMENTO DE SOFTWARE APLICADOS AO MINISTÉRIO PÚBLICO DO TRABALHO

Monografia apresentada ao Curso de Especialização em Desenvolvimento de
Sistemas Corporativos da Faculdade Natalense para o Desenvolvimento do Rio
Grande do Norte, aprovada em _____ de _____ de _____, pela banca
examinadora composta pelos seguintes membros:

Gleydson Lima, Msc.
Orientador

RESUMO

O Ministério Público do Trabalho possui uma estrutura de Tecnologia da Informação (TI) que está longe de ser a ideal para cumprir o seu papel dentro de uma instituição pública tão importante para a sociedade brasileira. Apesar de possuir um capacitado quadro de analistas e técnicos de informática e dispor de bons recursos de hardware, sofre com a falta de métodos, processos e métricas que norteiem as suas ações. Com o objetivo de contribuir na mudança dessa realidade, esse trabalho descreve algumas normas técnicas ligadas à área de TI, apresenta alguns dos mais modernos processos de desenvolvimento de software fazendo uma comparação entre os processos ágeis e não ágeis e apresenta também os guias e metodologias de gestão e governança de TI mais utilizados na atualidade. Dá uma visão da plataforma de desenvolvimento JEE (Java Enterprise Edition), da atuação do MPT e de uma iniciativa já em andamento na instituição: o projeto Mpt Digital.

Palavras-chave: Normas técnica de desenvolvimento de software. Processos de desenvolvimento de software. Governança de TI. Java 2 Enterprise Edition, J2EE. Ministério Público do Trabalho. MPT Digital.

ABSTRACTS

Ministério Público do Trabalho has a information technology structure that is far away from being inside to ideal to accomplish its role of an important public institution for the Brazilian society. In spite of possessing a qualified team of analysts and computer science technicians and disposal of good hardware resources, it suffers with the lack of methods, processes and metric that orientate their actions. With the objective of contributing to change this reality, this work describes some technical standards linked to the area of information technology, presents some of the most modern development processes of software making a comparison between the processes agile and not agile and also presents the guides and methodologies of management and information technology governance more used nowadays. Gives a vision for development platform JEE (Java Enterprise Edition), of the MPT action and a current job in the institution: the MPT Digital project.

Keywords: Software development standards. Software development processes. TI governance. Java 2 Enterprise Edition, J2EE. Ministério Público do Trabalho. MPT Digital.

LISTA DE FIGURAS

Figura 1 - Dimensões do RUP.....	23
Figura 2 - Ciclo do Scrum.....	32
Figura 3 - Capa da 3ª Edição do Guia PMBOK.....	35
Figura 4 - CMMI - Modelo contínuo.....	43
Figura 5 - CMMI - Modelo por estágios.....	44
Figura 6 - Os quatro domínios de processos do COBIT.....	46
Figura 7 - Os processos do ITIL.....	49
Figura 8 - Portabilidade do JAVA.....	52
Figura 9 - Arquitetura Multicamadas do J2EE.....	53

SUMÁRIO

1 INTRODUÇÃO.....	8
2 NORMAS.....	10
2.1 ISO/IEC 12207.....	10
2.2 ISO/IEC 15504.....	12
3 PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE.....	17
3.1 PROCESSOS ÁGEIS X PROCESSOS NÃO ÁGEIS.....	19
3.2 PROCESSO UNIFICADO (RUP).....	21
3.3 METODOLOGIAS ÁGEIS.....	27
3.3.1 Extreme Programming (XP).....	27
3.3.2 SCRUM.....	31
4 QUALIDADE DE SOFTWARE E GOVERNANÇA.....	35
4.1 PMBOK.....	35
4.2 CMM.....	40
4.3 CMMI.....	41
4.4 COBIT.....	45
4.5 ITIL.....	48
5 JAVA ENTERPRISE EDITION 5.....	52
6 MPT DIGITAL.....	55
6.1 MINISTÉRIO PÚBLICO DO TRABALHO.....	55
6.1.1 Ministério Público da União.....	55
6.1.2 Atuação do Ministério Público do Trabalho (MPT).....	56
6.1.3 Atuação como Órgão Interveniente.....	57
6.1.4 Atuação com Órgão Agente.....	57
6.2 CARACTERIZAÇÃO DO PROJETO MPT DIGITAL (DTI/PGT, 2008).....	58
6.2.1 Descrição.....	58
6.2.2 Objetivos.....	59
6.2.3 Justificativas.....	60
6.2.4 Metas.....	60
6.2.5. Escopo.....	61
6.2.6 Não escopo.....	62
6.2.7 Premissas.....	62

6.2.8 Restrições.....	63
6.2.9 Gerência do projeto.....	63
6.2.10 Interface com outros projetos e sistemas.....	63
7 CONCLUSÕES.....	64
8 GLOSSÁRIO.....	65
8.1. ISO/IEC.....	65
8.2 STAKEHOLDER.....	65
8.3 FRAMEWORK.....	66
8.4 FRAMEWORK DE SOFTWARE.....	66
8.5 PMI.....	66
8.6 AD HOC.....	66
8.7 GOVERNANÇA DE TI.....	67
8.8 TEST-DRIVEN DEVELOPMENT.....	67
8.9 USERS STORIES.....	67
REFERÊNCIAS.....	68

1 INTRODUÇÃO

O Ministério Público do Trabalho (MPT), um dos quatro ramos do Ministério Público da União (MPU), atua em defesa da sociedade no que se refere às relações de trabalho e é formado pela Procuradoria Geral do Trabalho (PGT), por 24 Procuradorias Regionais (PRT's) e aproximadamente 100 Sub-sedes (Ofícios) instalados nas principais regiões do país.

Para cumprir seus objetivos, como qualquer empresa ou instituição, seja ela pública ou privada, o MPT depende de uma estrutura de Tecnologia da Informação (TI) que forneça um bom suporte às suas atividades institucionais, oferecendo qualidade, rapidez e segurança no trato da informação.

A realidade atual não traduz essa necessidade, uma vez que os profissionais de TI, dispersos nas diversas regionais, além de serem em quantidade insuficiente para as necessidades do órgão, não atuam de maneira integrada, principalmente no tocante ao desenvolvimento de software. Soluções isoladas foram sendo desenvolvidas e algumas delas compartilhadas entre regionais, mas permanecendo um ambiente com múltiplas soluções, o que dificulta a disponibilização das informações de forma centralizada, gera bastante retrabalho e, sobretudo, deixa de oferecer ferramentas importantes para a melhoria contínua da atuação dos Membros da Instituição.

Iniciativas têm sido tomadas a fim de resolver a situação posta acima. A principal delas foi o lançamento do projeto "MPT Digital", que como será visto adiante neste trabalho, pretende dotar o MPT de um software corporativo, sem antes suprir as suas deficiências no tocante a organização de TI, uso de metodologias modernas de desenvolvimento de software, segurança da informação, entre outras.

Dentro do contexto descrito acima, este trabalho se coloca como uma leitura inicial para os profissionais de Tecnologia da Informação do Ministério Público do Trabalho, que estão ou estarão em breve envolvidos com as mudanças na forma com que a instituição passará a gerir a sua área de TI, principalmente no tocante ao desenvolvimento de software.

Além deste capítulo introdutório, o trabalho é composto dos seguintes:

O capítulo 2 apresenta as normas técnicas ISO/IEC 12.207 – Processos de Ciclo de Vida de Software e a ISO/IEC 15.504 – Avaliação de Processos de Software.

Em seguida, no capítulo 3, são descritos os principais processos modernos de desenvolvimento de software (RUP, XP e SCRUM), onde será vista também uma comparação

entre os processos ágeis e não ágeis.

O Capítulo 4 mostra o PMBOK, guia com as melhores práticas no gerenciamento de projetos, os modelos de maturidade de processos CMM e CCMI, que entre outras, definem níveis de maturidade e capacidade para os processos de desenvolvimento de software, a metodologia de governança COBIT e a biblioteca ITIL de melhores práticas em TI.

Na sequência o capítulo 5 faz uma síntese do JEE5 – Java Enterprise Edition, plataforma de software destinada ao uso corporativo e já definida para o desenvolvimento do “MPT Digital”.

O capítulo 6 faz uma introdução sobre o papel constitucional do Ministério Público, insere o Ministério Público do Trabalho neste contexto e reproduz as definições do projeto “MPT Digital”.

O capítulo 6, afinal, são emitidas opiniões e sugestões visando enquadrar toda a introdução teórica deste trabalho no dia a dia da Instituição.

2 NORMAS

Uma **norma técnica** é um documento, normalmente produzido por um órgão oficialmente acreditado para tal, que estabelece diretrizes e restrições acerca de um material, produto, processo ou serviço.

A engenharia de software tem se tornado uma das disciplinas mais importantes dos nossos dias devido à crescente disseminação do uso de software nos mais variados produtos e serviços. Entretanto, por ser uma área bastante recente em comparação com outras áreas da engenharia, não tem havido na produção e manutenção de software o mesmo rigor com que são tratados os projetos tradicionais de engenharia. Isso tem gerado insatisfações por parte tanto de usuários, que não vêem suas necessidades atendidas, quanto dos desenvolvedores, que não conseguem tornar seus produtos comercial e economicamente viáveis.

Desta forma, a comunidade de informática vem criando e aperfeiçoando normas para regular e orientar as atividades de produção de software.

2.1 ISO/IEC 12207

A norma ISO/IEC 12207 – Processos de Ciclo de Vida de Software foi criada com o objetivo de estabelecer uma estrutura comum para os processos de ciclo de vida de software, 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. Teve seu desenvolvimento proposto em 1988 e a primeira versão foi publicada em agosto de 1995 e em 1998 foi publicada a versão brasileira. Em 2002 e 2004 foram feitas atualizações na norma gerando as ementas 1 e 2 respectivamente.

Ela estabelece uma arquitetura de alto nível do ciclo de vida de software e 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.

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".

Seus processos são modulares, ou seja, são fortemente coesos e fracamente acoplados. Isto significa que todas as partes de um processo são fortemente relacionadas, mas a quantidade de interfaces entre os processos é mínima.

Eles 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 3 diferentes classes de processos, que são: processos fundamentais, processos de apoio e processos organizacionais.

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

- **Aquisição:** define as atividades que o requerente deve desempenhar para adquirir um sistema, desde a idealização até a entrega.
- **Fornecimento:** inicia quando se decide aceitar uma solicitação. O projeto é planejado, concebido, executado e entregue.
- **Desenvolvimento:** define as atividades do desenvolvedor. Contém as atividades análise de requisitos, projeto, codificação, integração, testes, instalação e aceitação.
- **Operação:** inicia quando um produto está pronto e deve entrar em produção. Contem atividades para testar, implantar, gerenciar e manter o produto.
- **Manutenção:** processo no qual os diversos problemas encontrados no produto quando em produção são analisados e corrigidos.

Os **processos de apoio** auxiliam outros processos. Eles são usados para garantir a qualidade. São eles:

- **Documentação:** inclui planejamento, projeto, desenvolvimento, produção e manutenção dos documentos necessários a gerentes, engenheiros e usuários do software.
- **Gerência de configuração:** inclui controle armazenamento, liberações, manipulação, distribuição e modificação de cada um dos itens que compõem o software.
- **Garantia da qualidade:** define as atividades para garantir que os processos e produtos de software estejam em conformidade com os requisitos especificados.
- **Verificação:** determina se os produtos de software de uma atividade atendem

completamente aos requisitos ou condições impostas a eles.

- Validação: determina se os produtos (sistema ou software) atendem ao uso específico proposto.
- Revisão conjunta: responsável por realizar auditorias periódicas no produto visando encontrar e solucionar inconsistências, não somente no código, mas em todo o escopo do projeto.
- Resolução de problemas: estabelece um processo para resolver problemas, permitindo documentar e publicar as soluções, facilitando seu reuso.

Os **processos organizacionais** auxiliam a organização e gerência geral dos processos e podem ser empregados fora do domínio de projetos e contratos específicos, servindo para toda a organização. São eles:

- Gerência: define as atividades de planejamento e controle para execução de todos os outros processos, utilizando os recursos adequados no tempo e custo orçado.
- Infra-estrutura: define as atividades para estabelecer e manter a infra-estrutura necessária para todos os outros processos. Pode incluir hardware, software, ferramentas, técnicas, padrões e recursos para o desenvolvimento, operação e manutenção;
- Melhoria: define as atividades para estabelecer, avaliar, medir, controlar e melhorar um processo de ciclo de vida de software.
- Treinamento: define as atividades para prover e manter o pessoal treinado.

2.2 ISO/IEC 15504

A norma ISO/IEC 15504 está sendo desenvolvida pela ISO com apoio da comunidade internacional através do projeto SPICE (Software Process Improvement and Capability Determination). Seu modelo mais recente foi publicado em fevereiro de 2008.

Esta norma define um framework para modelos de avaliação de processo, que também pode ser utilizado como referência para a melhoria de processo.

Composição da norma (volumes):

- 15504-1: Conceitos e Vocabulário
- 15504-2: Executando uma Avaliação
- 15504-3: Guia sobre Executando uma Avaliação

- 15504-4: Guia sobre Utilização do Resultado de Avaliação
- 15504-5: Um Exemplo de Modelo de Avaliação de Processo

Seu modelo de referência de processo identifica e descreve um conjunto de 48 processos considerados universais e fundamentais para a boa prática da engenharia de software, que estão organizados em 3 categorias e 9 grupos de processo:

- Processos fundamentais
 - Grupo de processo de aquisição
 - ACQ.1: Preparação da aquisição
 - ACQ.2: Seleção do fornecedor
 - ACQ.3: Acordo contratual
 - ACQ.4: Monitoramento do fornecedor
 - ACQ.5: Aceitação pelo cliente
 - Grupo de processos de fornecimento
 - SPL.1: Prospecção de fornecimento
 - SPL.2: Liberação de produto
 - SPL.3: Apoio para aceitação
 - Grupo de processos de engenharia
 - ENG.1: Elicitação de requisitos
 - ENG.2: Análise de requisitos de sistema
 - ENG.3: Projeto de arquitetura de sistema
 - ENG.4: Análise de requisitos de software
 - ENG.5: Projeto de software
 - ENG.6: Construção de software
 - ENG.7: Integração de software
 - ENG.8: Teste de software
 - ENG.9: Integração de sistema
 - ENG.10: Teste de sistema
 - ENG.11: Instalação de software
 - ENG.12: Manutenção de software e sistema
 - Grupo de processos de operação
 - OPE.1: Operação
 - OPE.2: Suporte ao cliente

- Processos organizacionais
 - Grupo de processos de gerência
 - MAN.1: Alinhamento organizacional
 - MAN.2: Gerência organizacional
 - MAN.3: Gerência de projeto
 - MAN.4: Gerência da qualidade
 - MAN.5: Gerência de riscos
 - MAN.6: Medição
 - Grupo de processos de melhoria de processo
 - PIM.1: Estabelecimento de processo
 - PIM.2: Treinamento
 - PIM.3: Gerência de conhecimento
 - Grupo de processos de recursos e infra-estrutura
 - RIN.1: Gerência de recursos humanos
 - RIN.2: Treinamento
 - RIN.3: Gerência de conhecimento
 - RIN.4: Infra-estrutura
 - Grupo de processos de reuso
 - REU.1: Gerência de ativos
 - REU.2: Gerência de programa de reuso
 - REU.3: Infra-estrutura
- Processos de apoio
 - Grupo de processos de apoio
 - SUP.1: Garantia da qualidade
 - SUP.2: Verificação
 - SUP.3: Validação
 - SUP.4: Revisão conjunta
 - SUP.5: Auditoria
 - SUP.6: Avaliação de produto
 - SUP.7: Documentação
 - SUP.8: Gerência de configuração
 - SUP.9: Gerência de resolução de problemas

- SUP.10: Gerência de solicitação de mudança

Cada processo é descrito com os seguintes elementos:

- Identificação;
- Nome;
- Propósito;
- Resultados: descreve os resultados esperados de uma implementação com sucesso deste processo;
- Práticas base: atividade que quando executada de forma consistente, contribui para o atendimento do propósito de um processo. Para cada prática base estão relacionados os resultados que a prática ajuda a alcançar;
- Produtos de trabalho: os produtos de trabalho de um processo são aqueles esperados de serem utilizados e/ou produzidos pela execução do processo. A lista de produtos de trabalho para cada processo deve ser utilizada como orientação para avaliação ou melhoria do processo.

A norma define também seis níveis de capacidade, sequenciais e cumulativos que podem ser utilizados como uma métrica para avaliar como uma organização está realizando um determinado processo e também podem ser utilizados como um guia para a melhoria.

Níveis de capacidade e seus atributos:

- Nível 0 – **Incompleto**: O processo não é executado, ou falha em cumprir seus requisitos básicos;
 - Não possui atributos
- Nível 1 – **Executado**: O propósito do processo é alcançado, mas de forma não planejada ou controlada;
 - PA 1.1: atributo de execução de processo
- Nível 2 – **Gerenciado**: O processo gera produtos de acordo com procedimentos, padrões e práticas definidas, dentro do cronograma e com os recursos esperados;
 - PA 2.1: atributo de gerência de execução
 - PA 2.2: atributo de gerência de produto de trabalho
- Nível 3 – **Estabelecido**: O processo é padrão na organização e é utilizado sempre da mesma forma;
 - PA 3.1: atributo de definição de processo
 - PA 3.2: atributo de implementação de processo

- **Nível 4 – Previsível:** O processo é executado de forma a possibilitar a avaliação do seu desempenho e produzindo indicadores adequados para melhorá-lo;
 - PA 4.1: atributo de medição de processo
 - PA 4.2: atributo de controle de processo
- **Nível 5 – Em otimização:** O desempenho do processo é continuamente melhorado para atingir as necessidades presentes e futuras da organização, sendo acompanhado continuamente.
 - PA 5.1: atributo de inovação de processo
 - PA 5.2: atributo de otimização do processo

Para se chegar à pontuação de cada processo, um valor tem que ser atribuído a cada atributo de processo, baseado nos dados validados, composta pelos seguintes quatro valores:

- N: o atributo não foi atingido pelo processo;
- P: o atributo foi atingindo apenas parcialmente pelo processo;
- L: o atributo foi atingido largamente pelo processo; e
- F: o atributo foi atingido completamente (fully) pelo processo.

Para estar em um nível de capacidade, um processo tem que ter notas “L” ou “F” nos atributos do nível e “F” em todos os atributos dos níveis anteriores.

A ISO/IEC 15504 define ainda 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 8 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 melhorias 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.

3 PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE

Processo de desenvolvimento de software pode ser definido resumidamente como uma seqüência coerente de práticas utilizadas com o fim de se produzir ou manter software. Essas práticas envolvem atividades (especificação, projeto, implementação, testes) e se utilizam de recursos (ferramentas, pessoas, métodos) para serem desenvolvidas.

Os primeiras experiências de produção de produtos de software foram levadas a cabo sem o uso de ordenações nem de padrões o que comumente conduziam a sistemas mal modelados e de difícil manutenção.

Foram surgindo então os modelos de desenvolvimento de software representados principalmente por:

- **Modelo em cascata** - Também conhecido como modelo tradicional, é caracterizado pelas fases de análise, modelagem, implementação, teste e implantação. Sua principal vantagem é a simplicidade, que leva a um fácil gerenciamento. Mas, em contra-partida tem várias desvantagens entre elas a dificuldade de se levantar todas as informações no início do projeto, o elevado tempo necessário para a análise em grandes sistemas, o surgimento de problemas nas fases finais e o crescimento exponencial dos custos para se retificar erros. Frequentemente, ao se utilizar esse modelo, o produto final não corresponde ao desejado pelo cliente.
- **Modelo em espiral** - Criado como alternativa ao modelo em cascata, no espiral o projeto é desenvolvido como uma série de pequenos ciclos, cada um finalizando com uma versão de software executável. Nele as equipes participam de ciclos inteiros (análise, design, codificação, testes) e a cada fase recebe-se um feedback do cliente, o que favorece à descoberta precoce de novos problemas e à possibilidade de incorporação de mudanças na tecnologia. È, entretanto, mais difícil de gerenciar que o modelo em cascata e em muitos projetos não há como de realizar o ciclo em virtude da ausência de protótipos a serem feitos no início.
- **Modelo iterativo e incremental** – Extensão do modelo espiral com a agregação de mais formalidade e rigor. Composto pelas fases de iniciação, elaboração, construção e transição, e em cada uma delas pode haver diversas iterações. (LIMA, 2004)

Baseados nos princípios do modelo iterativo e incremental foram desenvolvidos os principais representantes dos chamados “processos formais”, “processos controlados” ou “processos não ágeis” de desenvolvimento de software que são, entre outros, RUP, CMMI, ITIL, ISO, PMI e MPS.br, que por terem como premissa fundamental a ênfase no controle operacional, conduzirem seus trabalhos baseados em processos formais e englobarem todas as disciplinas no escopo de suas soluções, são acusados por seus críticos de serem excessivamente complicados, pesados, rígidos e burocráticos.

Foi então que a partir de um movimento global de profissionais, empresas e instituições, chamado de “manifesto ágil” (2001), surgiram os processos ágeis de desenvolvimento de software, cujos principais representantes são, entre outros, Xtreme Programming (XP), SCRUM e FDD. A ênfase na agilidade está entre suas premissas fundamentais, seus trabalhos são conduzidos tendo como base processos empíricos e centram o escopo de suas soluções na equipe de desenvolvimento. (BARTIE, 2007)

Os princípios do desenvolvimento ágil valorizam:

- Garantir a satisfação do consumidor entregando rapidamente e continuamente softwares funcionais;
- Softwares funcionais são entregues frequentemente (semanas, ao invés de meses);
- Softwares funcionais são a principal medida de progresso do projeto;
- Até mesmo mudanças tardias de escopo no projeto são bem-vindas.
- Cooperação constante entre pessoas que entendem do “negócio” e desenvolvedores;
- Projetos surgem através de indivíduos motivados, e que deve existir uma relação de confiança.
- Design do software deve prezar pela excelência técnica;
- Simplicidade
- Rápida adaptação às mudanças
- Indivíduos e iterações ao invés de processos e ferramentas
- Software funcional ao invés de documentação extensa
- Colaboração com clientes ao invés de negociação de contratos
- Responder a mudanças ao invés de seguir um plano

3.1 PROCESSOS ÁGEIS X PROCESSOS NÃO ÁGEIS

O desenvolvimento de produto final de software, iniciado em uma idéia, passa, a grosso modo, pelas fases de levantamento de requisitos, análise, projeto, construção e testes. Para se cumprir esse ciclo são envolvidas as variáveis escopo, custo, tempo e qualidade. A três primeiras podem ser negociadas, mas a qualidade, espera-se, é inegociável.

Os processos de desenvolvimento de software agirão neste contexto, podendo encontrar dois tipos de ambiente: o dinâmico, onde não se sabe exatamente onde se quer chegar e cujos requisitos podem mudar ao longo do tempo, e o ambiente estável, onde existe um contrato fixado, se sabe exatamente onde se quer chegar e se possui requisitos bem definidos. Os processos ágeis são mais indicados para o ambiente dinâmico enquanto para o ambiente estável se recomenda a utilização de processos formais.

Nos quadros abaixo vemos como cada um deles se comporta em relação às fases e variáveis envolvidas.

Processos formais (não ágeis)	Processos Ágeis
Variável: Tempo	
Definido em contrato Estimativa baseada na experiência	Pequenas iterações Pequenos releases Escopo definido até a próxima release Estimativa das atividades em baixo nível
Variável: Escopo	
Definido em contrato Segue requisitos estabelecidos no início do processo	Priorização de requisitos Requisitos transformam-se em funcionalidades até a próxima release
Variável: Custo	
Estabelecido em contrato, no início do projeto	Contrato renegociável a cada release de sistema
Variável: Qualidade	
Plano de testes Revisão de código Equipe de testes Interação com usuário a cada release	Test-driven development (8.8) Programação em par Iterações curtas Interação contínua com o usuário
Fase: Requisitos	
Casos de uso Levantamento no início do projeto	Users stories (8.9) Levantamento ao longo do projeto

Quadro 1 – Comparativo entre processos ágeis e não ágeis

Fonte: ALMEIDA (2007)

Processos formais (não ágeis)	Processos Ágeis
Fase: Análise	
Fase específica de análise Geração de artefatos de análise (MER, diagramas de seqüência, etc.)	Quebra de user stories em tarefas de implementação Programação em pares Test-driven development Posse coletiva de código
Fase: Projeto	
Geração de artefatos específicos para projeto arquitetural Análise de requisitos não-funcionais	Etapas não definidas especificamente nos processos Test-driven development Programação em pares Refatoração
Fase: Construção	
Uso de práticas comuns de desenvolvimento Programador segue especificação do sistema	Programação em pares Test-driven development Segue user stories e tarefas especificadas
Fase: Testes	
Plano de testes Execução de planos de testes após a implementação	Testes são feitos antes da implementação Testes ao longo do desenvolvimento Equipe de testes avalia os testes gerados

Quadro 1 – Comparativo entre processos ágeis e não ágeis (continuação)

Fonte: ALMEIDA (2007)

Pré-requisitos para uso de processos ágeis:

- Cliente sempre presente.
- Requisitos mudam com muita frequência
- Requisitos imprevisíveis.
- Equipe forte, bons profissionais
- Profissionais devem ser capazes de tomar decisões.
- Não adotar contrato com valor fixo

Quando não usar processos ágeis:

- Cliente ausente
- Reuniões a cada dois meses.
- Requisitos fixos, com pouca mudança no tempo.
- Equipe volátil e despreparada.
- Contratos com valor e escopo fixos.

- Submissão a programas de certificação de qualidade, como CMMI.

3.2 PROCESSO UNIFICADO (RUP)

O RUP (Rational Unified Process) é um processo de engenharia de software criado pela Rational Software Corporation, bem conhecida pelo seu investimento em orientação em objetos e criadora da Unified Modeling Language (UML), assim como de várias ferramentas que a suportam, sendo a mais conhecida o Rational Rose, e foi posteriormente adquirida pela IBM. Ele é na verdade um produto composto de material de referência na forma de páginas HTML, descrevendo toda a sua metodologia, que é apoiada por diversas ferramentas de desenvolvimento integradas e vendidas pela IBM através de seus "Rational Suites".

É um processo considerado pesado e preferencialmente aplicável a grandes equipes de desenvolvimento e a grandes projetos, porém o fato de ser amplamente customizável torna possível que seja adaptado para projetos de qualquer escala. Para a gerência do projeto, o RUP provê uma solução disciplinada de como assinalar tarefas e responsabilidades dentro de uma organização de desenvolvimento de software. Utiliza a metodologia iterativa e incremental, e sua principal meta é garantir a produção de software de alta qualidade que atenda às necessidades dos usuários e que esteja de acordo com cronogramas e orçamentos previsíveis.

O RUP é baseado nas melhores e mais modernas práticas de desenvolvimento de software:

- **Desenvolvimento baseado em iterações:** Como o desenvolvimento de um produto de software é uma grande tarefa que pode ser estendida por vários meses, possivelmente um ano ou mais, é mais prático dividir o trabalho em pedaços menores ou mini-projetos. Cada mini-projeto é uma iteração que resulta em um incremento. Iterações são passos em um fluxo de atividade e incrementos são crescimentos do produto. Cada iteração comporta um conjunto de atividades em modelagem de negócios, requisitos, análise e design, implementação, teste e implantação e cada uma delas possui um escopo e objetivos bem definidos.
- **Gerência de requisitos:** é uma abordagem sistemática para identificar, organizar, documentar e gerenciar os requisitos de um sistema, além de firmar e atualizar acordos entre a equipe e o cliente.

- **Arquitetura baseada em componentes:** cria um sistema que é facilmente extensível, intuitivo e de fácil compreensão e promove a reusabilidade de software, já que os componentes possuem interfaces e contratos bem definidos e podem ser substituídos conforme a necessidade. Arquiteturas baseadas em componentes normalmente reduzem a complexidade e o tamanho da solução, sendo mais robustas e flexíveis.
- **Modelagem visual:** um modelo é uma representação simplificada de uma solução e mostra seus elementos essenciais. A UML pode ser usada para modelagem de Casos de Uso, diagrama de classes e outros objetos.
- **Verificação contínua de qualidade:** ao longo do desenvolvimento de software, a qualidade do mesmo e de seus artefatos deve ser verificada. O RUP prevê planejamento de testes, revisões de artefatos e avaliações de qualidade do software.
- **Gerência de mudanças:** propicia um controle formal das mudanças nos requisitos, projeto, implementação e testes do sistema, além de permitir o controle de suas versões e releases.

Para se ter uma boa compreensão do RUP se faz necessário um conhecimento de alguns conceitos-chave:

- **Disciplina:** é uma coleção de atividades relacionadas a uma “área de interesse” principal. As disciplinas no RUP são: Modelagem de Negócios, Requisitos, Análise e Design, Implementação, Teste, Implantação, Configuração e Gerenciamento de Mudança, Gerenciamento de Projeto, Ambiente.
- **Fluxo de trabalho:** seqüência de atividades realizadas em um negócio que produz um resultado de valor observável para um ator individual do negócio.
- **Papel:** definição do comportamento e das responsabilidades de um indivíduo ou de um conjunto de indivíduos trabalhando em equipe. Exemplos: analista de sistemas, designer de negócio, revisor de requisitos, especificador de requisitos, revisor de código, implementador, arquiteto de software, designer (projetista).
- **Atividade:** uma unidade de trabalho que um papel pode ser solicitado a executar.
- **Artefato:** é uma informação que é produzida, modificada ou usada por um processo. Os papéis usam artefatos para executar atividades e produzem

artefatos ao executá-las.

A arquitetura geral do RUP é composta de duas dimensões:

- **Dimensão dinâmica do processo:** o eixo horizontal representa o tempo e mostra os aspectos do ciclo de vida do processo à medida que se desenvolve. É expressa em termos de fases, iterações e marcos.
- **Dimensão estática do processo:** o eixo vertical representa as disciplinas, que agrupam as atividades de maneira lógica, por natureza. É descrita em termos de componentes, disciplinas, atividades, fluxos de trabalho, artefatos e papéis do processo.

A figura abaixo mostra como a ênfase varia através do tempo. Por exemplo, nas iterações iniciais o RUP dedica mais tempo aos requisitos. Já nas iterações posteriores, gasta mais tempo com implementação.

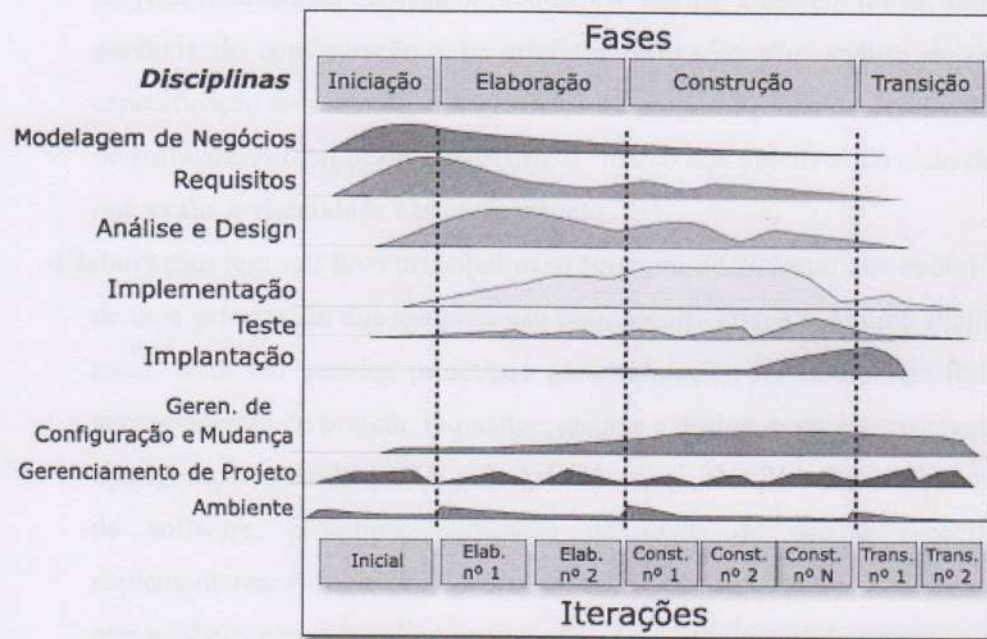


Figura 1 - Dimensões do RUP
Fonte: Rational(2003)

Na **dimensão dinâmica**, o projeto de software é dividido ao longo do tempo em quatro fases, cada uma concluída por um marco principal, ou seja, cada fase é basicamente um intervalo de tempo entre dois marcos principais. Em cada final de fase é executada uma avaliação para determinar se os objetivos da fase foram alcançados. Uma avaliação satisfatória permite que o projeto passe para a próxima fase. Cada fase é dividida em uma ou

mais iterações. Uma iteração é um ciclo de desenvolvimento completo, resultando em uma versão que cresce incrementalmente de iteração em iteração até que se transforme no produto final.

As quatro fases do RUP são:

- **Iniciação:** tem como principal meta o estabelecimento de um consenso entre todos os envolvidos sobre os objetivos do ciclo de vida do projeto. Para que seja bem sucedida é necessário que a equipe conheça bem o escopo e o que se quer do projeto. Nesta fase são levantados os casos de uso críticos do sistema, definida pelo menos uma opção de arquitetura, feitas estimativas de custo para o projeto inteiro e são previstos os **riscos** em potencial. Por risco se entende algo que possa ocorrer no ciclo do desenvolvimento e que comprometa ou elimine o sucesso do projeto. As disciplinas fortes da fase de iniciação são: gerência de projeto, modelagem de negócio, requisitos, análise e design, testes, ambiente e gerência de configuração e os artefatos utilizados são: análise de domínio, especificação de casos de uso, glossário do projeto e plano de desenvolvimento de software. A final dessa fase ocorre o “marco dos objetivos do ciclo de vida”, que avalia a viabilidade básica do projeto.
- **Elaboração:** tem seu foco principal na arquitetura do sistema, que evolui a partir de uma priorização dos casos de uso mais significativos e de uma avaliação de risco. Nela são gerados protótipos para validação. As disciplinas fortes são: gerenciamento de projeto, requisitos, análise e design, teste e gerenciamento de configuração e mudança. Os principais artefatos são: documento de arquitetura de software, protótipo, realização de casos de uso e especificações suplementares. Ao seu final ocorre o “marco da arquitetura do ciclo de vida” que estabelece uma baseline gerenciada para a arquitetura do sistema e permite o escalonamento da equipe do projeto na fase de Construção.
- **Construção:** tem como meta esclarecer os requisitos restantes e concluir o desenvolvimento do sistema com base na arquitetura da baseline. A fase de construção é de certa forma um processo de manufatura, em que a ênfase está no gerenciamento de recursos e controle de operações para otimizar custos, programações e qualidade. A qualidade da fase de elaboração é decisiva para o sucesso da construção. Os objetivos principais da fase de construção incluem:

minimizar os custos de desenvolvimento, atingir as versões úteis (alfa, beta e outros releases de teste), concluir a análise, o design, o desenvolvimento e o teste de todas as funcionalidades necessárias, decidir se o software, os locais e os usuários estão prontos para que o aplicativo seja implantado e atingir um certo paralelismo entre o trabalho das equipes de desenvolvimento. As atividades básicas desenvolvidas nessa fase são: gerenciamento de recursos, otimização de controle e processo; desenvolvimento completo do componente e teste dos critérios de avaliação definidos; avaliação dos releases do produto de acordo com os critérios de aceitação para a visão. Utiliza principalmente os artefatos: “o sistema”, plano de implantação, conjunto de testes, artefatos de treinamento (a serem utilizados na transição), modelos de casos de uso e requisitos de casos de uso. Ao seu final ocorre o “marco capacidade operacional inicial”, que determina se o produto está pronto para ser implantado em um ambiente de teste beta.

- **Avaliação:** o foco dessa fase é assegurar que o software esteja disponível para seus usuários finais. Em alguns casos, o fim do ciclo de vida atual pode coincidir com o início de outro ciclo de vida no mesmo produto, conduzindo à nova geração ou versão do produto. Seus principais objetivos são: validar o novo sistema em confronto com as expectativas do usuário; converter bancos de dados operacionais; treinar usuários e equipe de manutenção; introduzir o produto ao marketing, distribuição e equipe de vendas; desenvolver atividades de ajuste, como correção de erros, melhoria no desempenho e na usabilidade; preparar e empacotar o produto; obtenção de auto-suportabilidade do usuário; obter o consentimento dos envolvidos de que as baselines de implantação estão completas e são consistentes com os critérios de avaliação da visão. No fim na fase de transição está o marco mais importante do projeto, o “marco do release do produto”. Nesse momento, é decidido se os objetivos foram atendidos, e se outro ciclo de desenvolvimento deve ser iniciado.

Na **dimensão estática** o RUP descreve basicamente quem deve fazer o que, como e quando. Apresenta aqui as seguintes disciplinas:

- **Modelagem de negócio:** disciplina responsável pelo entendimento da estrutura e a dinâmica da organização na qual um sistema deve ser implantado, dos seus

problemas atuais e pela identificação das possibilidades de melhoria. Deve também assegurar que os clientes, usuários e desenvolvedores tenham um entendimento comum da organização e deve derivar os requisitos de sistema necessários para sua sustentação.

- **Requisitos:** responsável por estabelecer e manter concordância com os clientes e outros envolvidos sobre o que o sistema deve fazer; oferecer aos desenvolvedores do sistema uma compreensão melhor dos requisitos do sistema; definir as fronteiras do sistema; fornecer uma base para planejar o conteúdo técnico das iterações; fornecer uma base para estimar o custo e o tempo de desenvolvimento do sistema e definir uma interface de usuário para o sistema, focando nas necessidades e metas dos usuários.
- **Análise e design:** disciplina responsável por transformar os requisitos em um design do sistema a ser criado, desenvolver uma arquitetura sofisticada para o mesmo e adaptar o design para que corresponda ao ambiente de implementação, projetando-o para fins de desempenho.
- **Implementação:** a finalidade da implementação é definir a organização do código em termos de subsistemas de implementação organizados em camadas; implementar classes e objetos em termos de componentes (arquivos-fonte, binários, executáveis e outros); testar os componentes desenvolvidos como unidades e integrar os resultados produzidos por implementadores individuais (ou equipes) ao sistema executável.
- **Teste:** essa disciplina atua em vários aspectos como uma provedora de serviços para as demais. O teste enfatiza principalmente a avaliação da qualidade do produto, realizada através das seguintes práticas: localizar e documentar defeitos na qualidade do software; avisar de forma geral sobre a qualidade observada no software; validar as suposições feitas nas especificações de design e requisito através de demonstração concreta; validar as funções do software conforme projetadas e verificar se os requisitos foram implementados de forma adequada.
- **Implantação:** descreve as atividades que garantem que o produto de software será disponibilizado a seus usuários finais.
- **Gerenciamento de Configuração e Mudança:** controla as mudanças feitas nos

artefatos de um projeto e mantém a integridade entre eles. Um Sistema de Configuração de Mudança é fundamental para controlar os inúmeros artefatos produzidos pelas muitas pessoas que trabalham em um mesmo projeto. O controle ajuda a evitar confusões dispendiosas e garante que os artefatos resultantes não entrem em conflito devido a problemas tais como atualização simultânea, notificação limitada e diversidade de versões.

- **Gerenciamento de Projeto:** responsável pela elaboração dos planos e controle da execução do projeto de software. A abordagem de gerenciamento de projeto descrita pelo RUP nessa disciplina foi influenciada pelo PMBOK, e o leitor é aconselhado a consultar esse trabalho para obter uma cobertura completa das melhores práticas geralmente aceitas em gerenciamento de projeto.

- **Ambiente:** descreve as atividades para o desenvolvimento das diretrizes de suporte de um projeto. A meta das atividades dessa disciplina é oferecer à organização o ambiente de desenvolvimento de software — processos e ferramentas — que dará suporte à equipe de desenvolvimento.

3.3 METODOLOGIAS ÁGEIS

3.3.1 Extreme Programming (XP)

Extreme Programming (XP) é uma metodologia ágil de desenvolvimento de software nascida nos Estados Unidos no final da década de 1990. Tem como objetivo ajudar a criar sistemas de melhor qualidade, em menor tempo e de forma mais econômica que o habitual. É voltada para equipes pequenas e/ou médias (até 12 desenvolvedores) e tem como alvo o desenvolvimento de softwares cujos requisitos sejam vagos ou mudem constantemente.

Para alcançar seus objetivos XP se baseia em conjuntos de valores e práticas.

Valores do XP:

a) Feedback

O cliente aprende com o sistema que utiliza e com este aprendizado consegue reavaliar o produto recebido, com isso pode realimentar a equipe de desenvolvimento com as suas reais necessidades. Com o feedback, o cliente conduz o desenvolvimento do seu produto, estabelece prioridades e informa aquilo que é realmente importante.

Analogamente, há o feedback dado pelo desenvolvedor ao cliente, onde o mesmo aponta riscos, estimativas e alternativas de design. Este retorno é o aprendizado do desenvolvedor sobre o que o cliente deseja.

Com este valor, o tempo de defasagem entre as fases do projeto são extremamente pequenos, o cliente está o tempo todo em contato com a equipe de desenvolvimento, muito diferente dos modelos tradicionais, onde o cliente entrava em contato com a equipe um bom tempo depois do último feedback dado. (KUHN, 2004)

b) Comunicação

Para que o feedback entre cliente e desenvolvedor possa ser efetuado com sucesso é necessário ter uma boa comunicação entre eles. A XP prega que esta comunicação ocorra da forma mais direta e eficaz possível, oferecendo agilidade aos assuntos tratados. Recomenda-se o contato direto (face-a-face) entre cliente e desenvolvedor, para evitar qualquer tipo de especulação ou mal entendido entre as partes e para que possíveis dúvidas possam ser resolvidas de imediato.

Além de sanar as dúvidas no desenvolvimento, o cliente deverá estar disponível para a equipe, ou mesmo presente no ambiente de trabalho da empresa. Isto fará com que o cliente entenda o sistema e enriquecerá os relacionamentos pessoais, criando um elo de parceria e confiança mútua.

Algumas equipes não se adaptam bem a este valor. Este problema deve ser trabalhado em conjunto com a equipe. Enquanto não se acostumarem a falar e a trocar idéias com seus companheiros o sucesso da metodologia estará comprometido. Membros introvertidos são desaconselháveis para equipes de XP. (KUHN, 2004)

c) Simplicidade

Para que o cliente possa aprender durante o projeto e consiga dar o feedback necessário à equipe, não basta apenas uma boa comunicação, é necessário que os desenvolvedores implementem da forma mais simples possível o que o cliente deseja.

O XP prega que se deve fazer a coisa mais simples que pode funcionar. Com esta filosofia, o cliente terá a funcionalidade atendida rapidamente e da forma desejada, dando um feedback instantaneamente, evitando assim especulações. O desenvolvedor deve implementar apenas o necessário para que o cliente tenha seu pedido atendido.

O objetivo que se busca com a simplicidade é a precisão. Não se deve confundir simplicidade com facilidade. O mais simples nem sempre é o mais fácil e também não é

escrever menos código. Simplicidade significa codificar o necessário para que um requisito seja atendido e entregue ao cliente. (KUHN, 2004)

d) Coragem

Há quem defenda que a única constante em um projeto de software é a mudança. Clientes mudam de idéia com frequência, mesmo quando fecham contratos nos quais, teoricamente, assumem o compromisso de não alterar o que está na especificação. Eles mudam porque aprendem durante o projeto e descobrem problemas mais prioritários a serem solucionados ou formas mais apropriadas de resolvê-los. Embora isso seja natural, gera uma preocupação para a equipe de desenvolvimento que, de tempos em tempos, precisa alterar partes do sistema que já estavam prontas, correndo o risco de se quebrar o que já vinha funcionando.

XP não tem uma solução mágica para eliminar esse risco. Ele existe em um projeto XP, como existe em qualquer outro. O que muda é a forma de lidar com ele. Equipes XP acreditam que errar é natural e quebrar o que vinha funcionando pode acontecer eventualmente. É necessário ter coragem para lidar com esse risco, o que em XP se traduz em confiança nos seus mecanismos de proteção. As práticas são voltadas, entre outras coisas, para proteger o software de inúmeras formas. Equipes XP confiam na eficácia destas práticas e destes mecanismos de proteção e isso é o que as tornam receptivas a mudanças. Assim, ao invés de frear a criatividade do cliente e evitar mudanças, equipes XP as consideram inevitáveis e procuram se adaptar a elas com segurança e com coragem, isto é, com confiança em seus mecanismos de proteção, tais como **desenvolvimento orientado a testes, programação em par e integração contínua**. (TELES, 2008)

Práticas do XP

a) Planejamento

No XP o planejamento é feito várias vezes durante o projeto. É o momento onde o cliente solicita as funcionalidades desejadas através de histórias (user stories) e onde a equipe estima o custo de cada uma delas e planeja as releases e as iterações.

Após a definição das histórias (pequenos cartões em que o cliente deve descrever o que deseja com suas palavras e da forma mais simples possível) é necessário estimar o tempo das mesmas para que o cliente priorize o que deve ser implementado. A XP utiliza uma unidade chamada ponto, que refere-se a um dia de trabalho ideal do desenvolvedor, onde o mesmo não precisaria atender telefonemas, participar de reuniões, ou seja, estaria preocupado

apenas com a estória em questão. Quando uma estória se torna demasiadamente grande XP recomenda que seja quebrada em tarefas menores e que as mesmas não utilizem mais que alguns pontos de um desenvolvedor (4 pontos ao máximo).

O XP tem por objetivo gerar valor para o cliente ao longo do projeto, para isto o software é desenvolvido de forma incremental, onde a cada entrega (releases a cada 2 meses no máximo) o cliente possa utilizar o sistema e obter benefícios do mesmo. (KUHN, 2004)

b) Stand up meeting

O dia de trabalho de uma equipe XP sempre começa com um stand up meeting . É uma reunião rápida pela manhã, com aproximadamente 20 minutos de duração e onde seus integrantes permaneçam preferencialmente em pé, pois segundo estudos uma reunião em pé é mais rápida, evita bate-papos paralelos e faz os integrantes irem diretamente ao assunto.

A reunião tem por objetivo atualizar a equipe sobre o que foi implementado no dia anterior e trocar experiências das dificuldades enfrentadas. Neste momento também são decididas as estórias que serão implementadas no dia e em conjunto definir os responsáveis por cada uma delas. (KUHN, 2004)

c) Programação em par

O XP exige que todo e qualquer código implementado no projeto seja efetuado em dupla, chamada de programação em par. é uma técnica onde dois desenvolvedores trabalham no mesmo problema, ao mesmo tempo e no mesmo computador. Um deles é responsável pela digitação (condutor) e o outro acompa o trabalho do parceiro (navegador), revisando permanentemente o código digitado.(KUHN, 2004)

d) Desenvolvimento guiado por testes (TDD)

XP utiliza dois tipos de testes, os testes de unidade e de aceitação. O teste de unidade tem por objetivo verificar se os resultados gerados por cada classe estão corretos, já o teste de aceitação tem por objetivo verificar se a interação entre as classes que implementam uma funcionalidade (estória) atendem a necessidade de forma correta. Os testes de aceitação são descritos pelo cliente e implementados pelos desenvolvedores, facilitando ainda mais a interação entre as partes. (KUHN, 2004)

e) Releases curtos

No modelo tradicional o projeto é dividido em fases, de um modo que o cliente começará a ter benefício com o sistema muito próximo de o mesmo estar finalizado. A maior parte do investimento do projeto é feita antes mesmo de estar concluído, sem ter gerado

qualquer tipo de valor econômico ao cliente. O XP recomenda que pequenos investimentos sejam efetuados de forma gradual e que busque grandes recompensas o mais rapidamente possível. Com a prática de releases curtos, o XP pretende dar o máximo de valor econômico ao cliente em um curto espaço de tempo. (KUHN, 2004)

Papéis do XP

Em uma equipe de XP existem papéis a serem desempenhados por um ou mais desenvolvedores. Estes papéis serão listados a seguir.

a) Gerente de projeto:

Responsável pelos assuntos administrativos do projeto, mantendo um forte relacionamento com o cliente

b) Coach:

Responsável pelas questões técnicas do projeto, verificando o comportamento da equipe frente ao processo XP, sinalizando os eventuais erros cometidos.

c) Analista de teste:

Responsável em garantir a qualidade do sistema através dos testes escritos.

d) Redator técnico:

Responsável em documentar o sistema, evitando um forte trabalho dos desenvolvedores neste tipo de atividade.

e) Desenvolvedor:

Responsável pela análise, projeto e codificação do sistema. No XP não existe diferença entre analista, projetista e programador uma vez que em vários momentos do projeto o desenvolvedor estará exercendo alguma destas atividades. (KUHN, 2004)

3.3.2 SCRUM

Scrum foi concebido inicialmente como um estilo de gerenciamento de projetos em empresas de fabricação de automóveis e produtos de consumo, por Takeuchi e Nonaka em 1986. Eles perceberam que projetos usando equipes pequenas e multidisciplinares produziam os melhores resultados. Jeff Sutherland, John Scumniotales, e Jeff McKenna documentaram, conceberam e implementaram o Scrum na empresa Easel Corporation em 1993, incorporando estilos de gerenciamento observados por Takeuchi e Nonaka. Em 1995, Ken Schwaber formalizou a definição de Scrum e ajudou a implantá-lo em desenvolvimento de software em

todo o mundo. Em 2002 Ken Schwaber and Mike Cohn fundaram a Scrum Alliance.

Scrum é uma metodologia ágil para gerência de projetos. O ciclo do Scrum tem o seu progresso baseado em um série de iterações bem definidas, cada uma com duração de 2 a 4 semanas, chamadas Sprints. Antes de cada Sprint, realiza-se uma Reunião de planejamento (Sprint Planning Meeting) onde a equipe (team) de desenvolvedores tem contato com o cliente (Product Owner) para priorizar o trabalho que precisa ser feito, selecionar e estimar as tarefas que o time pode realizar dentro da Sprint. A próxima fase é a Execução da Sprint. Durante a execução da Sprint, o time controla o andamento do desenvolvimento realizando Reuniões Diárias Rápidas (Daily Meeting), não mais que 15 minutos de duração, e observando o seu progresso usando um gráfico chamado Sprint Burndown.

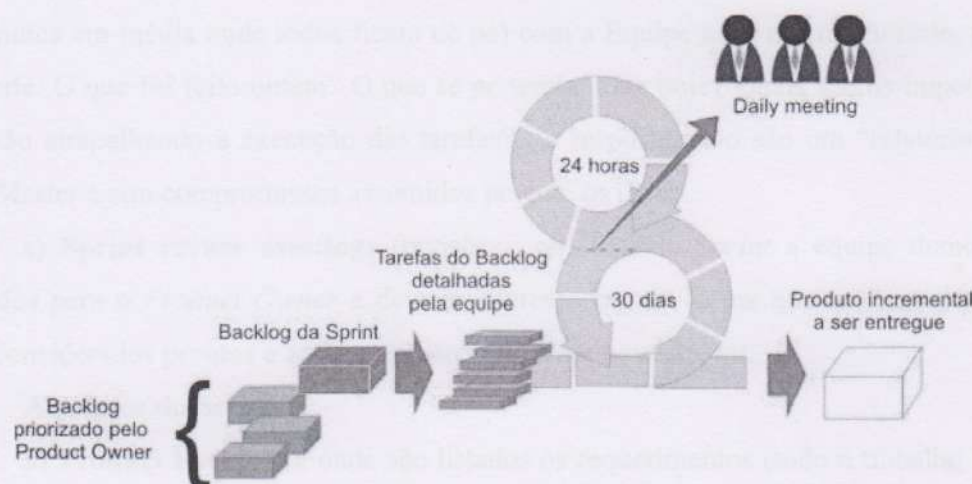


Figura 2 - Ciclo do Scrum
Fonte: PEREIRA(2007)

Ao final do Sprint, deve-se realizar uma Reunião de Revisão (Sprint Review), onde o time demonstra o produto gerado na Sprint e valida se o objetivo foi atingido. Logo em seguida, realiza-se a Reunião de Retrospectiva (Sprint Retrospective), uma reunião de lições aprendidas, com o objetivo de melhorar o processo/time e/ou produto para a próxima Sprint. Scrum torna-se ideal para projetos dinâmicos e suscetíveis a mudanças de requisitos, sejam eles novos ou apenas requisitos modificados. No entanto, para aplicá-lo, é preciso entender antes os seus papéis, cerimônias e artefatos. (PEREIRA, 2007)

Papéis do Scrum

a) **Product owner:** responsável pela visão de negócios do projeto, é ele quem define e prioriza o Product Backlog. Geralmente é o papel desempenhado pelo cliente.

b) Scrum master: é uma mistura de gerente, facilitador e mediador. Seu papel é remover obstáculos da equipe e assegurar que as práticas de *Scrum* estão sendo executadas com eficiência.

c) Team (equipe): responsável por entregar soluções, geralmente é formada por um grupo pequeno (entre 5 e 9 pessoas) e que trabalha de forma auto-gerenciada.

Cerimônias do Scrum

a) Sprint planning meeting (planejamento): todas as funcionalidades ou mudanças no produto são definidas pelo *Product Owner* no *Product Backlog*. Esta lista é priorizada para refletir a necessidade dos clientes ou demandas do mercado. Os itens do topo da lista são destacados para serem entregues no final do próximo *Sprint*.

b) Daily scrum meetings (reuniões diárias): o Scrum Master se reúne diariamente (15 minutos em média onde todos ficam de pé) com a Equipe num mesmo horário, para que se reporte: O que foi feito ontem? O que se pretende fazer hoje? Quais são os impedimentos que estão atrapalhando a execução das tarefas? As respostas não são um “relatório” para o ScrumMaster e sim compromissos assumidos perante os pares.

c) Sprint review meetings (revisões): no final do *Sprint* a equipe demonstra os resultados para o *Product Owner* e demais interessados, de forma que os itens do *Backlog* sejam considerados prontos e então se possa iniciar um novo *Sprint*.

Artefatos do Scrum

a) Product backlog: é onde são listados os requerimentos (todo o trabalho desejado no projeto) numa forma em que cada item tenha seu peso de acordo com a vontade do cliente ou usuários. É priorizado pelo *product owner* e repriorizado no início de cada *Sprint*.

b) Sprint backlog: durante o *Sprint*, os itens do *Product Backlog* que devem ser entregues são agora tratados no *Sprint Backlog*. As tarefas agora são responsabilidade da *Equipe*, que tem autonomia para decidir como elas devem ser executadas. (SANCHES, 2007)

c) Burndown chart: demonstra o trabalho a ser realizado em um *sprint* dia a dia.

As práticas do Scrum trazem vantagens ao gerenciamento dos projetos como:

- participação mais efetiva da equipe quanto à definição das atividades, gerando maior comprometimento, motivação e confiança,
- as pessoas sabem o que estão fazendo e o porquê, diferente quando o processo é imposto e não foi construído com a equipe.
- o estímulo a colaboração entre os membros da equipe deixa o time mais coeso.

- o fortalecimento do trabalho de time: cada um sabe o que o outro faz, escolhe o que vai fazer, as responsabilidades estão visíveis, existe transparência e alinhamento para atender o objetivo do projeto.
- estímulo aos relacionamentos e integração entre os membros da equipe, minimizando conflitos.
- incentivo ao compartilhamento e disseminação do conhecimento.
- maior visibilidade do desempenho da equipe e de cada membro.
- maior participação e satisfação do cliente. (PEREIRA, 2007)

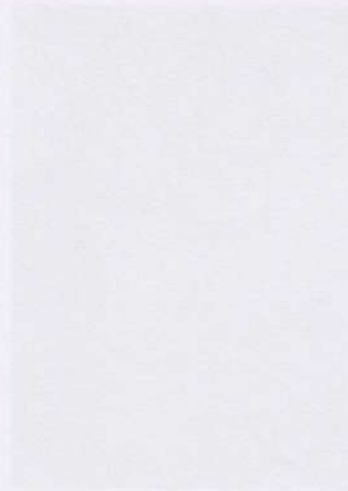


Figura 1 - Capa do Guia de Referência de Referência
Projeto PMBOK (2004)

O livro identifica e descreve o conhecimento de natureza de conhecimento de Gerenciamento de Projetos (o conhecimento) que é necessário para o sucesso de projetos de maior porte, tempo, envolvendo recursos, pessoas, processos e recursos. A natureza do livro não representa a abordagem de referência utilizada em todos os projetos, devendo ser definido o que é apropriado para cada projeto, indústria.

O PMBOK Guide também estabelece uma linguagem comum para o profissional, servindo de referência básica para qualquer um que se interesse pelo Gerenciamento de Projetos e que não deve ser encarado como um documento que contempla a totalidade

4 QUALIDADE DE SOFTWARE E GOVERNANÇA

Neste capítulo são descritos alguns dos principais modelos de qualidade e governança de software disponíveis na atualidade. Estes modelos são ferramentas que auxiliam as organizações a evoluir sua capacidade de desenvolver e manter serviços e produtos de software com qualidade.

4.1 PMBOK

O Guia PMBOK (Project Management Body of Knowledge) é uma publicação do PMI (Project Management Institute), organismo fundado em 1969 para identificar, inicialmente, práticas comuns de gerência na indústria. A sua primeira edição foi publicada em 1987, a segunda em 2000 e versão atual, de número 3, em 2004.



Figura 3 - Capa da 3ª Edição do Guia PMBOK
Fonte: PMI/SP (2004)

O livro identifica e descreve o subconjunto do universo do conhecimento de Gerenciamento de Projetos reconhecido como boas práticas em muitos projetos na maior parte do tempo, havendo consenso pelos praticantes sobre seus valores e aplicabilidade. Entretanto, a aceitação geral não representa a necessidade de aplicação uniforme em todos os projetos, devendo ser definido o que é apropriado para cada projeto / indústria.

O *PMBOK Guide* também estabelece uma linguagem comum para a profissão, servindo de referência básica para qualquer um que se interesse pelo Gerenciamento de Projetos e, como tal, não deve ser encarado como um documento que contemple a totalidade

do conhecimento de Gerenciamento de Projetos. Periodicamente serão feitas revisões e novas versões serão desenvolvidas. (PMI São Paulo)

O Guia é também utilizado pelo PMI (Project Management Institute) como uma estrutura consistente para seus programas de desenvolvimento profissional incluindo: “Certificação de Profissionais de Gerência de Projetos (PMP – Project Management Professional)” e “Credenciamento dos programas educacionais que concedem grau em Gerência de Projetos”.

A primeira parte do Guia, em seus 3 primeiros capítulos, descreve a “A Estrutura de Gerência de Projetos”, divididos em:

- Capítulo 1 – **Introdução**, onde são definidos os termos chaves e apresentada uma visão geral do resto do documento;
- Capítulo 2 - **O Contexto da Gerência de Projetos**, que descreve o ambiente no qual o projeto opera.
- Capítulo 3 - **Os Processos da Gerência de Projetos**, que apresenta uma visão geral da interação entre os diversos processos de gerência de projetos

A segunda parte, composta pelos capítulos de 4 a 12, descreve **As Áreas de Conhecimento da Gerência de Projetos**, onde são mostrados os conhecimentos e práticas em gerência de projetos em termos dos processos que as compõem. Estes processos foram organizados em nove áreas de conhecimentos como descrito abaixo:

I. Gerenciamento de Integração do Projeto

A Gerência da Integração do Projeto inclui os processos requeridos para assegurar que os diversos elementos do projeto estão adequadamente coordenados. Ela envolve fazer compensações entre objetivos e alternativas eventualmente concorrentes a fim de atingir ou superar as necessidades e expectativas. Enquanto todos os processos de gerência de projetos são de alguma maneira integrados, os processos descritos neste capítulo são por natureza integrativos.

- 1.Desenvolver o termo de abertura do projeto: geração de documento que autorizam formalmente um projeto ou uma fase do projeto.
- 2.Desenvolver a declaração do escopo preliminar do projeto: abordagem e documentação das características e limites do projeto e seus produtos e serviços associados, além dos métodos de aceitação e controle do escopo.
- 3.Desenvolver o plano de gerenciamento do projeto: definição das ações

necessárias para definir, coordenar e integrar todos os planos auxiliares em um plano de gerenciamento de projeto.

4. Orientar e gerenciar a execução do projeto: levar a cabo o projeto através da realização das atividades nele incluídas.
5. Monitorar e controlar o trabalho do projeto: monitoramento dos processos do projeto, tomando ações corretivas ou preventivas para controlar o seu desempenho.
6. Controle Integrado de Mudanças – coordenar as mudanças através do projeto inteiro.
7. Encerrar o projeto: execução das ações de encerramento previstas no plano de gerenciamento do projeto.

II. Gerenciamento do Escopo do Projeto

A Gerência do Escopo do Projeto inclui os processos requeridos para assegurar que o projeto inclua todo o trabalho necessário, e tão somente este, para complementar de forma bem sucedida o projeto. A preocupação fundamental compreende definir e controlar o que está incluído no projeto.

1. Planejamento do Escopo – desenvolver uma declaração escrita do escopo como base para decisões futuras do projeto.
2. Detalhamento do escopo – subdividir os principais subprodutos do projeto em componentes menores e mais manejáveis.
3. Criar EAP: Estrutura Analítica do Projeto – decomposição hierárquica orientada à entrega do trabalho a ser executado pela equipe do projeto para atingir os seus objetivos e criar as entregas necessárias.
4. Verificação do Escopo – formalizar a aprovação do escopo do projeto.
5. Controle de Mudanças do Escopo – controlar as mudanças do escopo do projeto.

III. Gerenciamento de Tempo do Projeto

A Gerência do Tempo do Projeto inclui os processos necessários para assegurar que o projeto será implementado no prazo previsto.

1. Definição das Atividades – identificar as atividades específicas que devem ser realizadas para produzir os diversos subprodutos do projeto.
2. Seqüenciamento das Atividades – identificar e documentar as relações de dependência entre as atividades.

3. Estimativa de recursos da atividade: determinação dos recursos (pessoas, equipamentos ou material) e respectivas quantidades a serem usadas e quando cada recurso estará disponível
4. Estimativa da Duração das Atividades - estimar a quantidade de períodos de trabalho que serão necessários para a implementação de cada atividade.
5. Desenvolvimento do Cronograma - analisar a seqüência e as durações das atividades, e os requisitos de recursos para criar o cronograma do projeto.
6. Controle do Cronograma - controlar as mudanças no cronograma do projeto.

IV. Gerenciamento de Custos do Projeto

A Gerência do Custo do Projeto inclui os processos necessários para assegurar que o projeto será concluído dentro do orçamento aprovado.

1. Estimativa dos Custos – desenvolver uma estimativa dos custos dos recursos necessários à implementação das atividades do projeto.
2. Orçamentação dos Custos – alocar as estimativas de custos globais aos itens individuais de trabalho.
3. Controle dos Custos – controlar as mudanças no orçamento do projeto.

V. Gerenciamento da Qualidade do Projeto

A Gerência da Qualidade do Projeto inclui os processos requeridos para garantir que o projeto irá satisfazer as necessidades para as quais ele foi empreendido. Isto inclui “todas as atividades da função de gerência geral que determinam as políticas de qualidade, objetivos e responsabilidades e para a implementação destes, por meio de planejamento, controle, garantia e melhoria da qualidade”.

1. Planejamento da Qualidade – identificar quais padrões de qualidade são relevantes para o projeto e determinar a forma de satisfazê-los.
2. Realizar a Garantia da Qualidade – avaliar periodicamente o desempenho geral do projeto buscando assegurar a satisfação dos padrões relevantes de qualidade.
3. Realizar o Controle da Qualidade – monitorar os resultados específicos do projeto para determinar se eles estão de acordo com os padrões de qualidade relevantes e identificar as formas para eliminar as causas de desempenhos insatisfatórios.

VI. Gerenciamento de Recursos Humanos do Projeto

A Gerência dos Recursos Humanos do Projeto inclui os processos requeridos para possibilitar o uso mais efetivo das pessoas envolvidas com o projeto. Isto inclui todos os interessados do projeto – patrocinadores, clientes, contribuintes individuais e outros.

1. Planejamento de recursos humanos – identificar, documentar e designar as funções, responsabilidades e relacionamentos dentro do projeto.
2. Contratar ou mobilizar a equipe do projeto – conseguir que os recursos humanos necessários sejam designados e alocados ao projeto
3. Desenvolver a equipe do projeto – desenvolver habilidades individuais e do grupo para aumentar o desempenho do projeto.
4. Gerenciar a equipe do projeto - acompanhamento do desempenho de membros da equipe, fornecimento de feedback, resolução de problemas e coordenação de mudanças para melhorar o desempenho.

VII. Gerenciamento das Comunicações do Projeto

A Gerência das Comunicações do Projeto inclui os processos requeridos para garantir a geração apropriada e oportuna, a coleta, a distribuição, o armazenamento e o controle básico das informações do projeto. Fornece ligações críticas entre pessoas, idéias e informações que são necessárias para o sucesso. Todos os envolvidos no projeto devem estar preparados para enviar e receber comunicações na "linguagem" do projeto e devem entender como as comunicações, que eles estão individualmente envolvidos afetam o projeto como um todo.

1. Planejamento das Comunicações – determinar as informações e comunicações necessárias para os interessados: quem necessita de qual informação, quando necessitarão dela e como isso será fornecido.
2. Distribuição das Informações – disponibilizar as informações necessárias para os interessados do projeto da maneira conveniente.
3. Relato de Desempenho – coletar e disseminar as informações de desempenho. Inclui relatórios de situação, medição de progresso e previsões.
4. Gerenciar as partes interessadas: gerenciamento das comunicações para satisfação das necessidades das partes interessadas (stakeholders) e resolução de conflitos entre elas.

VIII. Gerenciamento de Riscos do Projeto

A Gerência de Risco do Projeto inclui os processos envolvidos na identificação,

análise e resposta aos riscos do projeto. Isto inclui a maximização dos resultados de eventos positivos e minimização das consequências de eventos negativos.

1. Identificação dos Riscos – determinar quais os riscos são mais prováveis de afetar o projeto e documentar as características de cada um.
2. Quantificação dos Riscos – avaliar os riscos e suas interações no sentido de avaliar possíveis consequências.
3. Desenvolvimento das Respostas aos Riscos – definir as melhorias necessárias para o aproveitamento de oportunidades e respostas às ameaças.
4. Controle das Respostas aos Riscos – responder às mudanças nos riscos no decorrer do projeto.

IX. Gerenciamento de Aquisições do Projeto

A Gerência de Aquisições do Projeto inclui os processos necessários à obtenção de bens e serviços externos à organização executora. Para simplificação, os bens e serviços referidos como um “produto”.

1. Planejamento das Aquisições – determinar o que contratar e quando.
2. Preparação das Aquisições – documentar os requerimentos do produto e identificar os fornecedores potenciais.
3. Obtenção de Propostas – obter propostas de fornecimento conforme apropriado a cada caso (cotações de preço, cartas-convite, licitação).
4. Seleção de Fornecedores – escolher entre os possíveis fornecedores.
5. Administração dos Contratos – gerenciar o relacionamento com os fornecedores.
6. Encerramento do Contrato – completar e liquidar o contrato incluindo a resolução de qualquer item pendente. (Tradução livre do PMBOK)

4.2 CMM

Capability Maturity Model - Modelo para avaliação da maturidade dos processos de software de uma organização e para identificação das práticas chave que são requeridas para aumentar a maturidade desses processos. O CMM (também conhecido como SW-CMM) prevê cinco níveis de maturidade: **inicial**, **repetível**, **definido**, **gerenciado** e **otimizando**. O modelo foi proposto por Watts S. Humphrey, a partir das propostas de Philip B. Crosby, e vem sendo aperfeiçoado pelo Software Engineering Institute - SEI da Carnegie Mellon University. A especificação do modelo compreende um instrumental para a averiguação do nível atual da

organização e expõe os pontos de ação, ou áreas chave de processo (key process area – KPA), que servem como pontos de destaque para a promoção a um nível mais elevado de maturidade. Por exemplo, uma área de processo do nível 2 do CMM seria a garantia de qualidade de software.

A adoção do CMM se baseia em um processo gradual de aumento da maturidade da capacidade de desenvolvimento de software de uma organização. A maturidade desse processo pode ser traduzida como a probabilidade de entregar sistemas de software dentro do prazo, utilizando os recursos planejados e atendendo aos requerimentos para o qual foi projetado dentro dos padrões de qualidade desejados.

Este modelo foi descontinuado pelo SEI em dezembro de 2007 e substituído a partir de então pelo CMMI ((Capability Maturity Model Integration).

4.3 CMMI

Capability Maturity Model Integration (CMMI) é um modelo de referência que fornece orientação para o desenvolvimento de processos de softwares e tem como objetivos eliminar suas inconsistências; aumentar sua clareza e entendimento; fornecer uma terminologia comum e um estilo consistente; estabelecer regras de construção uniformes e assegurar consistência com a ISO/IEC 15504.

Ele não define como o processo deve ser implementado, mas prescreve suas características estruturais e semânticas em termos de objetivos e de grau de qualidade com que o trabalho deve ser realizado.

O CMMI possui duas representações: "contínua" ou "por estágios". Elas permitem à organização utilizar diferentes caminhos para a melhoria de acordo com seu interesse.

A **contínua** permite que uma organização selecione uma área (ou um grupo de áreas) de processo e melhore os processos relacionados. Ela usa níveis de capacidade para caracterizar melhorias relativas a uma área de processo individual.

A **estagiada** usa conjuntos pré-definidos de áreas de processo (KPA's) para definir um caminho para uma organização, caracterizado por níveis de maturidade. Cada nível contém um conjunto de áreas de processo que caracterizam diferentes comportamentos organizacionais, correspondendo à capacidade da empresa de realizar projetos grandes e complexos.

- **Representação Contínua**

Na representação contínua, o enfoque ou componentes principais são as áreas de processo. Existem metas e práticas de dois tipos: específicas a uma determinada área de processo e genéricas aplicáveis indistintamente a todas as áreas de processo. A partir da avaliação e do atendimento dessas práticas e metas é possível classificar o nível de capacidade de cada área de processo, em níveis de zero a cinco:

- **Nível 0 - Incompleto:** um processo é parcialmente realizado ou não realizado. Um ou mais objetivos específicos do processo não estão satisfeitos.
- **Nível 1 - Realizado:** um processo realizado satisfaz todos os objetivos específicos da área de processo e produz algum trabalho.
- **Nível 2 - Gerenciado:** um processo de capacidade nível 2 é um processo realizado (nível 1) que também é planejado e executado de acordo com políticas pré-definidas. Emprega pessoas hábeis com os recursos adequados para produzir saídas adequadas, envolve os stakeholders principais e é monitorado, controlado, revisto e avaliado quanto à aderência à sua descrição. A gerência do processo é relacionada com a realização de objetivos específicos estabelecidos para o processo, como custo, cronograma e qualidade.
- **Nível 3 - Definido:** um processo definido é um processo gerenciado e ajustado para o conjunto padrão de processos da organização de acordo com suas políticas de conduta. Esse conjunto é estabelecido e melhorado com o tempo e descreve os elementos fundamentais de processos que são esperados nos processos definidos.
- **Nível 4 - Gerenciado quantitativamente:** um processo neste nível é definido e controlado com a ajuda de técnicas quantitativas e estatísticas. A qualidade e o desempenho do processo são compreendidos em termos estatísticos e são geridos durante sua vida. Objetivos quantitativos para qualidade e desempenho de processos são estabelecidos e usados como critério na gerência do processo.
- **Nível 5 - Em otimização:** um processo em otimização é gerenciado quantitativamente, alterado e adaptado para atender aos objetivos de negócio atuais e projetados. Tal processo enfoca a melhoria contínua do desempenho do processo através de aprimoramentos tecnológicos inovadores e incrementais, selecionados com base em uma compreensão quantitativa de sua contribuição esperada à obtenção da melhoria de processos.

A figura 4 esquematiza como as práticas e objetivos são organizados:

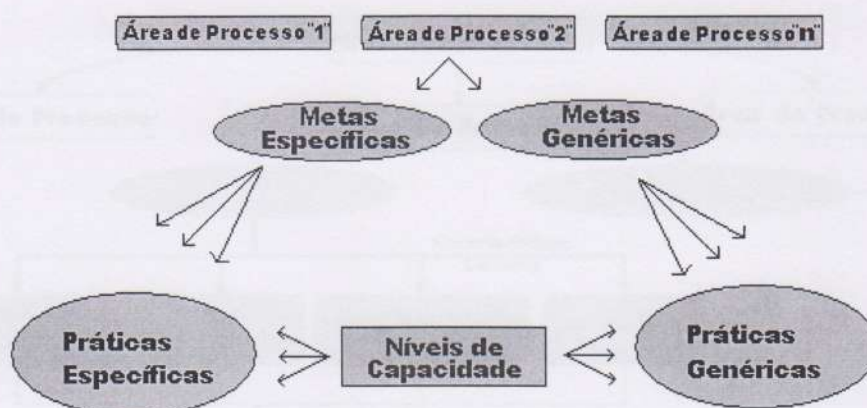


Figura 4 - CMMI - Modelo contínuo
Fonte: MANERA et al. (2007)

Uma organização pode escolher melhorar o desempenho de um único ponto problemático de um processo, ou pode trabalhar em diversas áreas fortemente ligadas aos seus objetivos. A representação contínua permite a melhoria de diferentes processos em taxas diferentes. Há dependências entre algumas áreas de processo, o que faz com que seja adequada para situações em que tais dependências são bem compreendidas e os processos que precisam ser melhorados tenham sido identificados.

• Representação por estágios

A representação em estágios oferece uma abordagem estruturada e sistemática para a melhoria de um estágio por vez. Atingir um estágio significa que uma estrutura de processo adequada foi estabelecida como base para o próximo estágio.

As áreas de processo são organizadas por níveis de maturidade (1 a 5), que definem o caminho de melhoria que uma organização deve seguir do nível inicial ao nível otimizado. Dentro de cada nível, existem áreas de processo que contêm metas, características comuns e práticas.

Na representação em níveis, as práticas são caracterizadas pelos atributos: compromisso para execução (práticas que garantem que o processo seja estabelecido e apoiado); habilidade para execução (práticas que criam condições para que o processo seja estabelecido completamente) e atividade para execução (práticas que implementam diretamente o processo); controle e verificação de implementação. A transição entre os níveis resulta em melhorias incrementais e duradouras. A figura 5 esquematiza as metas e práticas

desse modelo:



Figura 5 - CMMI - Modelo por estágios

Fonte: MANERA et al. (2007)

Os estágios de maturidade são:

- **Nível 1 - Inicial:** É o nível de maturidade CMMI mais baixo. Em geral, as organizações desse nível têm processos imprevisíveis que são pobremente controlados e reativos. Nesse nível de maturidade os processos são normalmente “ad hoc” e caóticos. A Organização geralmente não fornece um ambiente estável. Neste nível não há KPA's.
- **Nível 2 – Gerenciado:** No nível de maturidade 2 os projetos da organização têm a garantia de que os requisitos são gerenciados, planejados, executados, medidos e controlados. Quando essas práticas são adequadas, os projetos são executados e controlados de acordo com o planejado. O foco, neste nível, é o gerenciamento básico de projetos e tem as seguintes KPA's desse nível são: gerenciamento de requisitos; planejamento do projeto; controle e monitoração do projeto; gerenciamento de suprimentos; avaliação e análise; garantia da qualidade do processo; configuração do gerenciamento.
- **Nível 3 – Definido:** No nível de maturidade 3, em que todos os objetivos específicos e genéricos atribuídos para os níveis de maturidade 2 e 3 foram alcançados, os processos são melhor caracterizados e entendidos e são descritos em padrões, procedimentos, ferramentas e métodos. O foco neste nível é a padronização do processo, tendo como KPA's: requisitos de

desenvolvimento; soluções técnicas; integração de produtos; verificação; validação; foco no processo organizacional; definição do processo organizacional; treinamento organizacional; gerenciamento de projeto integrado; gerenciamento de riscos; integração da equipe de trabalho; gerenciamento integrado de suprimentos; análise de decisões; ambiente organizacional para integração.

- **Nível 4 - Quantitativamente Gerenciado:** No nível de maturidade 4, em que os objetivos específicos atribuídos para os níveis de maturidade 2, 3 e 4 e os objetivos genéricos atribuídos para os níveis de maturidade 2 e 3 foram alcançados, os processos são medidos e controlados. O foco neste nível é o gerenciamento quantitativo e possui as seguintes KPA's: performance organizacional do processo; gerenciamento quantitativo de projetos.
- **Nível 5 – Em otimização:** No nível de maturidade 5, o mais alto nível de maturidade CMMI, uma organização atingiu todos os objetivos específicos atribuídos para os níveis de maturidade 2, 3, 4 e 5, e os objetivos genéricos atribuídos para os níveis de maturidade 2 e 3. Os processos são continuamente aperfeiçoados, baseados em um entendimento quantitativo em que a variação de um processo existe devido às interações, normais e presumidas, entre os componentes desse processo. Esse nível de maturidade tem como objetivo a melhoria contínua do processo. As KPA's desse nível são: inovação organizacional e análise de causas e resoluções.

4.4 COBIT

Control Objectives for Information and Related Technology (COBIT) tem por missão explícita pesquisar, desenvolver, publicar e promover um conjunto atualizado de padrões internacionais de boas práticas referentes ao uso corporativo da TI para os gerentes e auditores de tecnologia.

A metodologia COBIT foi criada pelo Information Systems Audit and Control Association (ISACA) através do IT Governance Institute, organização independente que desenvolveu a metodologia considerada a base da governança tecnológica. O COBIT funciona como uma entidade de padronização e estabelece métodos documentados para nortear a área de tecnologia das empresas, incluindo qualidade de software, níveis de

maturidade e segurança da informação.

Os documentos do COBIT definem Governança Tecnológica como sendo "uma estrutura de relacionamentos entre processos para direcionar e controlar uma empresa de modo a atingir objetivos corporativos, através da agregação de valor e risco controlado pelo uso da tecnologia da informação e de seus processos".

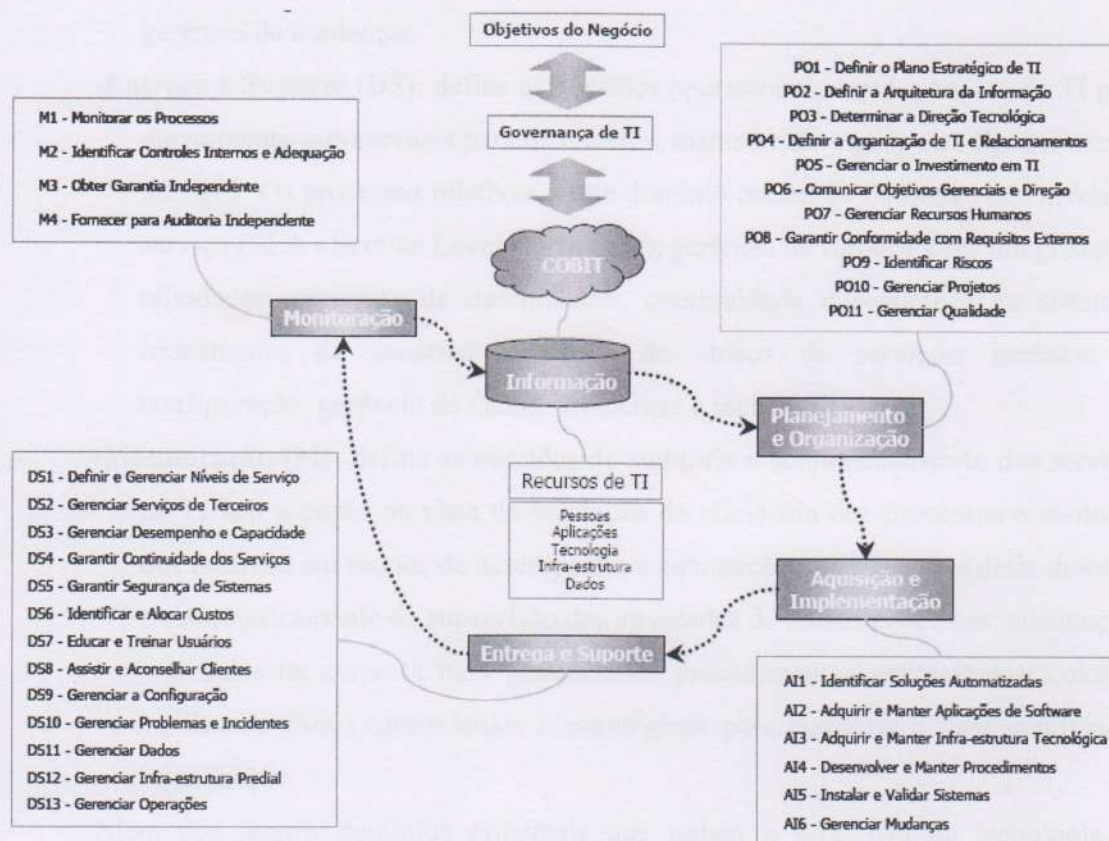


Figura 6 - Os quatro domínios de processos do COBIT
Fonte: TRAINNING (2007)

O COBIT está organizado em quatro domínios para refletir um modelo para os processos de TI. Os domínios podem ser caracterizados pelos seus processos e pelas atividades executadas em cada fase de implantação da Governança Tecnológica. Os domínios do COBIT são:

- **Planejamento e Organização (PO):** define as questões estratégicas ligadas ao uso da TI em uma organização, trata de vários processos, entre eles, a definição da estratégia de TI, arquitetura da informação, direcionamento tecnológico, investimento, riscos, gerência de projetos e da qualidade.
- **Aquisição e Implementação (AI):** define as questões de implementação da TI

conforme as diretrizes estratégicas e de projeto pré-definidos no Plano Estratégico de Informática da empresa. Possui uma série de processos como, por exemplo, identificação de soluções automatizadas a serem aplicadas ou reutilizadas na corporação, aquisição e manutenção de sistemas e de infra-estrutura, desenvolvimento e mapeamento de procedimentos nos sistemas, instalação e gerência de mudanças.

•**Entrega e Suporte (DS):** define as questões operacionais ligadas ao uso da TI para atendimento aos serviços para os clientes, manutenção e garantias ligadas a estes serviços. Os processos relativos a este domínio tratam da definição dos níveis de serviço (SLA - Service Level Agreement); gerência de fornecedores integrados às atividades; garantias de desempenho, continuidade e segurança de sistemas; treinamento de usuários; alocação de custos de serviços; gerência de configuração; gerência de dados, problemas e incidentes.

•**Monitoração (M):** define as questões de auditoria e acompanhamento dos serviços de TI sob o ponto de vista de validação da eficiência dos processos e evolução dos mesmos em termos de desempenho e automação. Os processos deste domínio tratam basicamente da supervisão das atividades de outros processos; adequações realizadas na empresa para garantia dos procedimentos operacionais; coleta e análise de dados operacionais e estratégicos para auditoria e para controle da organização.

Além dos quatro domínios principais que guiam o bom uso da tecnologia da informação na organização, existe também a questão da auditoria que permite verificar, através de um relatório de avaliação, o nível de maturidade dos processos da organização. O método de auditoria segue o modelo do CMM que estabelece os níveis:

- Inexistente: o processo de gerenciamento não foi implantado.
- Inicial: o processo é realizado sem organização, de modo não planejado.
- Repetível: o processo é repetido de modo intuitivo, depende mais das pessoas do que de um método estabelecido.
- Definido: o processo é realizado, documentado e comunicado na organização.
- Gerenciado: existem métricas de desempenho das atividades, o processo é monitorado e constantemente avaliado.
- Otimizado: as melhores práticas de mercado e automação são utilizadas para a

melhoria contínua dos processos.

O resultado do relatório identifica o grau de evolução dos processos na organização que é avaliada com base em dados confiáveis de auditoria e parâmetros de mercado. O sumário executivo do relatório traz as seguintes informações: se existe um método estabelecido para o processo; como o método é definido e estabelecido; quais os controles mínimos para a verificação do desempenho do método; como pode ser feita auditoria no método; quais as ferramentas utilizadas no método e o que avaliar no método para sua melhoria. A partir de então, a organização define as metas, isto é, os objetivos de controle a serem atingidos.

Os domínios do COBIT são integrados da seguinte forma: a informação de uma empresa é gerada/modificada pelos recursos de TI. Esta informação é requisito para o domínio de Planejamento de Organização (PO) e seus processos. Os requisitos de saída do PO são requisitos de entrada para o domínio de Aquisição e Implementação (AI), que, por sua vez, definem os requisitos de entrada para o domínio de Entrega e Suporte (DS). Finalmente, o domínio de Monitoração (M) utiliza as informações do DS nos seus processos e atividades relacionadas.

Os requisitos da informação são dados por: efetividade, eficiência, confiabilidade, integridade, disponibilidade, conformidade e confiabilidade. Já os recursos de TI são classificados como: pessoas, sistemas aplicativos, tecnologia, infraestrutura e dados. (TRAINNING, 2007)

4.5 ITIL

Information Technology Infrastructure Library (ITIL) é um conjunto de melhores práticas para a gestão de serviços em TI e para o alinhamento desta área com os negócios da empresa. Estas práticas estão especificadas em sete livros (daí o termo “biblioteca”), dentre os quais os dois primeiros são mais conhecidos: Suporte a Serviços e Entrega de Serviços. A idéia nasceu na Inglaterra, na década de 80, quando a Tecnologia da Informação ainda estava bastante afastada da área de negócios. Hoje o modelo é regulado pelo IT Service Management Forum (ITSMF), uma organização internacional dedicada a disseminar as melhores práticas em gerenciamento de serviços de TI.

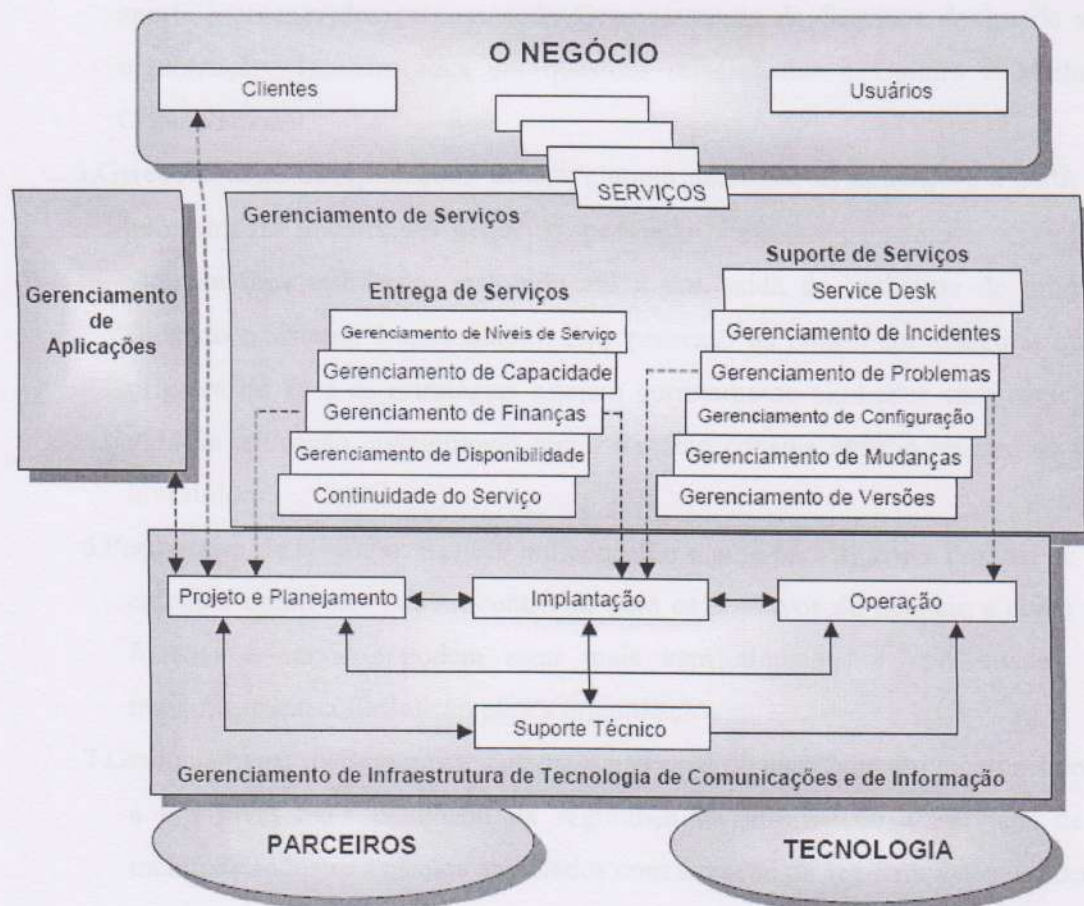


Figura 7 - Os processos do ITIL
Fonte: TRAINNING (2007)

Os sete livros que compõe a biblioteca ITIL são os seguintes:

1. Suporte a Serviços: descreve os processos associados ao suporte do dia-a-dia e atividades de manutenção associadas com a provisão de Serviços de TI.
2. Entrega de Serviços: cobre os processos necessários para o planejamento e entrega de Serviços de TI com qualidade e se preocupa ao longo do tempo com o aperfeiçoamento desta qualidade.
3. ICT - Gerenciamento da Infra-estrutura: cobre todos os aspectos do Gerenciamento da Infra-estrutura como a identificação dos requisitos do negócio, testes, instalação, entrega, e otimização das operações normais dos componentes que fazem parte dos Serviços de TI.
4. Planejamento para Implementação do Gerenciamento de Serviços: examina questões e tarefas envolvidas no planejamento, implementação e

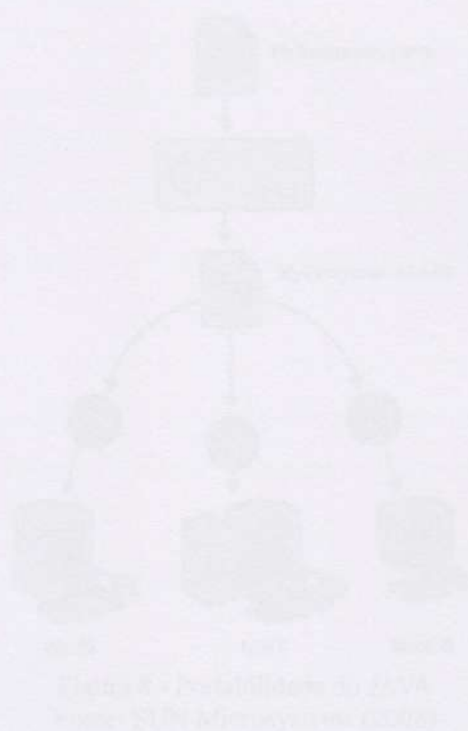
aperfeiçoamento dos processos do Gerenciamento de Serviços dentro de uma organização. Também foca em questões relacionadas à Cultura e Mudança Organizacional.

5. Gerenciamento de Aplicações: descreve como gerenciar as aplicações a partir das necessidades iniciais dos negócios, passando por todos os estágios do ciclo de vida de uma aplicação, incluindo até a sua saída do ambiente de produção (quando o sistema é aposentado). Este processo dá ênfase em assegurar que os projetos de TI e as estratégias estejam corretamente alinhados com o ciclo de vida da aplicação, assegurando que o negócio consiga obter o retorno do valor investido.
6. Perspectiva de Negócio: fornece um conselho e guia para ajudar o pessoal de TI a entender como eles podem contribuir para os objetivos do negócio e como suas funções e serviços podem estar mais bem alinhados e aproveitados para maximizar sua contribuição para a organização.
7. Gerenciamento da Segurança: detalha o processo de planejamento e gerenciamento a um nível mais detalhado da segurança da informação e Serviços de TI, incluindo todos os aspectos associados com a reação da segurança dos incidentes. Também inclui uma avaliação e gerenciamento dos riscos e vulnerabilidade, e implementação de custos justificáveis para a implementação de contra-recursos (estratégia de segurança).

Entre os principais objetivos da adoção da melhores práticas da ITIL podem ser destacados os seguintes:

- Alinhar os Serviços de TI com as necessidades atuais e futuras do negócio e seus clientes. Todos os processos da ITIL falam que a TI precisa entender os requisitos de negócio da empresa para poder planejar e prover seus serviços para atender as expectativas.
- Melhorar a qualidade dos serviços de TI. Através de um programa de melhoria contínua deve se buscar a consistência na entrega dos serviços, atendendo as necessidades de negócio.
- Reduzir custos na provisão de serviços. Este é um dos motivos chaves que levam os gestores de TI a adotarem as melhores práticas. Existem vários casos de sucesso onde houve grande redução dos custos operacionais e investimentos em TI.

- Processos mais eficientes e eficazes, buscando rapidez e resultados nos processos.
- Adoção das melhores práticas, evitando-se reinventar a roda. (TI-Exames, 2006)



Java é disponibilizado em três edições, cada uma delas sendo de uso para um tipo diferente de aplicação:

- **Java 2 Standard Edition (J2SE):** É a linguagem Java que proporciona suporte para a maioria das aplicações de propósito geral e também para as aplicações de negócios. A API de classes padrão e as APIs de classes de terceiros são fornecidas especificamente no J2SE da Sun. É com ele que a maioria das aplicações são construídas e executadas. O J2SE possui duas versões:
 - **Java Development Kit (JDK):** ou **Standard Technology Edition (STE):** fornece o conjunto de ferramentas para desenvolver, executar e testar Java.
 - **Java Runtime Edition (JRE):** versão mais leve da JDK, destinada para ambientes

5 JAVA ENTERPRISE EDITION 5

A tecnologia JAVA para desenvolvimento de sistemas computacionais, desenvolvida pela Sun Microsystems a partir da década de 1990, tem como principais e mais conhecidas características ser orientada a objetos e possuir total portabilidade, ou seja, o produto da compilação de códigos fontes java, chamados de bytecodes, podem ser executados em quaisquer combinações de hardware e sistema operacional que possuam uma JVM (java virtual machine) instalada.

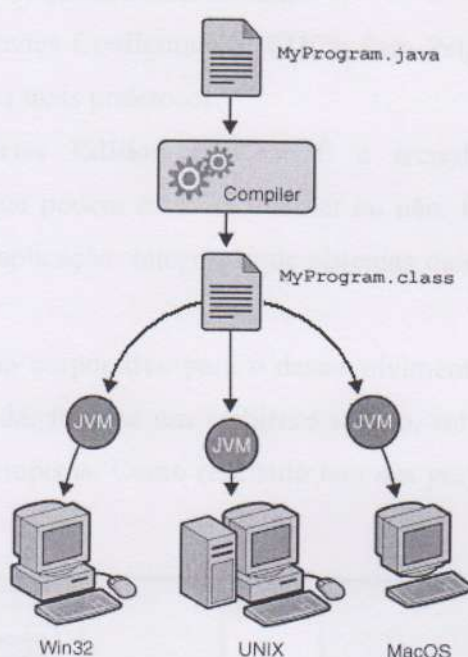


Figura 8 - Portabilidade do JAVA
Fonte: SUN Microsystems (2008)

Java é disponibilizada em três edições, cada uma delas indicada para um tipo diferente de ambiente:

- **Java 2 Standard Edition (J2SE):** É a tecnologia Java para computadores pessoais, notebooks e arquiteturas com poder de processamento e memória consideráveis. Várias APIs acompanham esta versão e tantas outras podem ser baixadas opcionalmente no site da Sun. É com elas que a maioria das aplicações são construídas e executadas. O J2SE possui duas divisões:
 - **Java Development Kit (JDK) ou Standard Development Kit (SDK):** conjunto composto de ferramentas para desenvolvimento em Java.
 - **Java Runtime Edition JRE:** versão mais leve da JDK, destinada para ambientes

de execução.

•**Java 2 Mobile Edition (J2ME):** É a tecnologia Java para dispositivos móveis com limitações de memória ou processamento. Possui APIs bem simples e leves para economizar espaço, memória e processamento. São utilizadas para sistemas em celulares, palm tops, pocket pcs, smartphones, javacards e demais dispositivos. O J2ME se divide em dois grupos de bibliotecas:

- Connected Limited Device Configuration (CLDC):** Para celulares e smartphones, que são mais limitados
- Connected Device Configuration (CDC):** Para Palmtops e Pocket pcs e alguns dispositivos mais poderosos.

•**Java 2 Enterprise Edition (J2EE):** É a tecnologia Java para aplicações corporativas que podem estar na internet ou não. É ideal para a construção de servidores de aplicação, integração de sistemas ou distribuição de serviços para terceiros.

A J2EE é o padrão corporativo para o desenvolvimento de aplicações de negócio. Sendo corretamente aplicada, fornece um ambiente seguro, robusto e escalável, através de aplicações de controle da empresa. Como resultado tem sua portabilidade e reusabilidade de código.

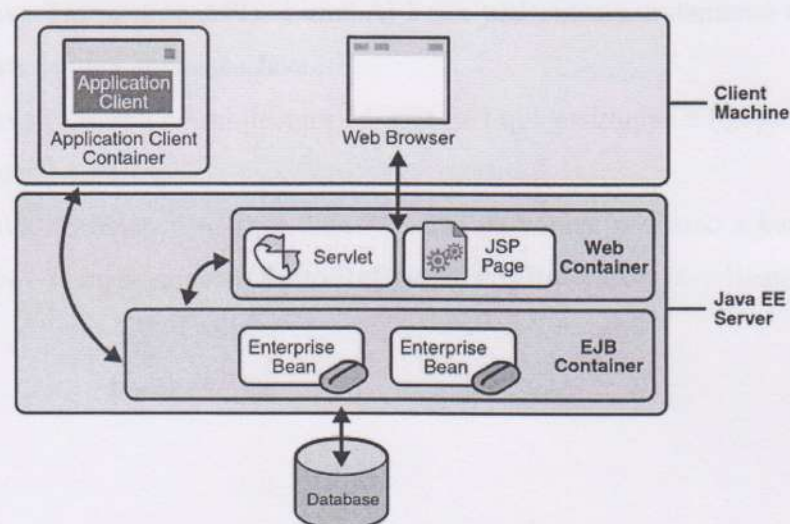


Figura 9 - Arquitetura Multicamadas do J2EE
Fonte: SUN Microsystems (2008)

O padrão J2EE é voltado para aplicações multi-camadas, baseadas em componentes que são executados em um servidor de aplicações. A plataforma Java EE é considerada um

padrão de desenvolvimento já que o fornecedor de software nesta plataforma deve seguir determinadas regras se quiser declarar os seus produtos como compatíveis com Java EE. Tais características, justificam o uso desta plataforma para o desenvolvimento de aplicações corporativas. (SILVA, 2006)

A plataforma J2EE contém uma série de especificações, cada uma com funcionalidades distintas. Entre elas, tem-se:

- Servlets, são utilizados para o desenvolvimento de aplicações Web com conteúdo dinâmico. Ele contém uma API que abstrai e disponibiliza os recursos do servidor Web de maneira simplificada para o programador.
- JSP (Java Server Pages), uma especialização do servlet que permite que conteúdo dinâmico seja facilmente desenvolvido.
- EJBs (Enterprise Java Beans), utilizados no desenvolvimento de componentes de software. Eles permitem que o programador se concentre nas necessidades do negócio do cliente, enquanto questões de infra-estrutura, segurança, disponibilidade e escalabilidade são responsabilidade do servidor de aplicações.
- JDBC (Java Database Connectivity), utilizado no acesso a bancos de dados;
- JNDI (Java Naming and Directory Interface) provê acesso a serviços de nome e de diretórios tais como DNS e LDAP.
- JTA (Java Transaction API), é uma API que padroniza o tratamento de transações dentro de uma aplicação Java.
- JCA (Java Connector Architecture), é uma API que padroniza a ligação a aplicações legadas.
- JPA (Java Persistence API), é uma API que padroniza o acesso a banco de dados através de mapeamento Objeto/Relacional dos Enterprise Java Beans.

6 MPT DIGITAL

6.1 MINISTÉRIO PÚBLICO DO TRABALHO

6.1.1 Ministério Público da União

O Ministério Público é instituição permanente, essencial à função jurisdicional do Estado, incumbindo-lhe a defesa da ordem jurídica, do regime democrático e dos interesses sociais e individuais indisponíveis (art. 127 da Constituição da República).

Compreende o Ministério Público Federal, o Ministério Público do Trabalho, o Ministério Público Militar e o Ministério Público do Distrito Federal e Territórios, abrangendo, também, os Ministérios Públicos dos Estados (art. 128, incisos I e II da Constituição da República).

Representa a sociedade político-juridicamente organizada no Estado, mas não a pessoa jurídica desse ou de seus governantes. O Ministério Público defende os interesses sociais da comunidade a que serve, salvaguardando os bens e os valores essenciais à prevalência da Cidadania e do Estado de Direito.

O Ministério Público é o fiscal da lei, sendo sua missão a de preservar a ordem democrática.

Princípios Institucionais: Unidade, indivisibilidade e independência funcional sendo-lhe assegurada a autonomia funcional e administrativa. O Ministério Público constitui um único organismo, cumprindo suas atribuições constitucionais e legais através da atuação de qualquer de seus Membros, e ainda que haja a substituição de um membro por outro, é o próprio Ministério Público que subsiste na atuação.

Composição (Lei Complementar nº 75/93)

"Art. 24 – O Ministério Público da União compreende:

I – o Ministério Público Federal;

II – o Ministério Público do Trabalho;

III – o Ministério Público Militar;

IV – o Ministério Público do Distrito Federal e Territórios.

Parágrafo Único – A estrutura básica do Ministério Público da União será organizada por regimento, nos termos da lei."

6.1.2 Atuação do Ministério Público do Trabalho (MPT)

Até 1988, o Ministério Público do Trabalho atuava apenas como Órgão Interveniente junto ao Tribunal Superior do Trabalho ou aos Tribunais Regionais do Trabalho, emitindo parecer nos processos judiciais, na condição de fiscal da lei.

A partir da nova Constituição Federal, passou a atuar também como Órgão Agente, ou de campo, na defesa dos direitos difusos, coletivos e individuais indisponíveis dos trabalhadores.

Desde 1999, elegeu cinco áreas prioritárias de atuação: erradicação do trabalho infantil e regularização do trabalho do adolescente, combate ao trabalho escravo e regularização do trabalho indígena, combate a todas as formas de discriminação no trabalho, preservação da saúde e segurança do trabalhador, e regularização dos contratos de trabalho.

Como órgão interveniente, o MPT desempenha papel de defensor da lei para intervir nos feitos judiciais em curso nos quais haja interesse público a proteger. Emite pareceres em processos de competência da Justiça do Trabalho, participa das Sessões de Julgamento e ingressa com recursos quando houver desrespeito à legislação.

O Ministério Público do Trabalho também atua como árbitro e mediador na solução de conflitos trabalhistas de natureza coletiva, envolvendo trabalhadores e empresas ou as entidades sindicais que os representam. A possibilidade está prevista no artigo 83, inciso XI da Lei Complementar 75/93 e foi regulamentada pela Resolução nº 44 do Conselho Superior do MPT. Além disso, o Ministério Público do Trabalho fiscaliza o direito de greve nas atividades essenciais.

A atuação como Órgão Agente envolve o recebimento de denúncias, a instauração de procedimentos investigatórios, inquéritos civis públicos e outras medidas administrativas ou o ajuizamento de ações judiciais, quando comprovada a irregularidade.

Importante instituto de atuação do Ministério Público do Trabalho, de natureza administrativa, é o Termo de Compromisso de Ajustamento de Conduta, que prevê multa caso seja descumprido, e que pode ser executado perante as Varas do Trabalho, por ser título executivo extrajudicial. Em termos judiciais, o MPT dispõe da Ação Civil Pública e da Ação Civil Coletiva, além da Ação Anulatória Trabalhista, que possibilita sua atuação no controle das cláusulas de acordos e convenções coletivas de trabalho.

O Ministério Público do Trabalho também orienta a sociedade por meio de audiências públicas, palestras, oficinas, reuniões setoriais e outros eventos semelhantes.

Desenvolve, ainda, ações em parceria com órgãos do Governo e entidades representativas de empregadores e trabalhadores, organizações não governamentais nacionais e internacionais e com a sociedade civil organizada, seja por meio de protocolos e convênios, seja pela participação em Conselhos e Fóruns.

6.1.3 Atuação como Órgão Interveniente

Como Órgão Interveniente, o Ministério Público do Trabalho recebe dos Tribunais os processos em que esteja envolvido interesse público, dos menores, incapazes e índios, decorrentes das relações de trabalho, ou que tenham como parte pessoa jurídica de direito público, estado estrangeiro ou organismo internacional. Devolve os processos com parecer em que avalia aspectos técnicos e o mérito da questão, agindo como fiscal da lei.

O Ministério Público do Trabalho pode manifestar-se em qualquer fase do processo trabalhista, acolhendo solicitação do Juiz ou por sua iniciativa, e recorrer das decisões da Justiça do Trabalho quando entender necessário, tanto nos processos em que for parte, como naqueles em que officiar como fiscal da lei. Além disso, cabe ao MPT pedir revisão dos Enunciados da Súmula de Jurisprudência do Tribunal Superior do Trabalho.

Os Membros do MPT participam dos julgamentos nos Tribunais Trabalhistas, sentados à direita do presidente da Sessão. Manifestam-se verbalmente sobre a matéria em debate, sempre que entenderem necessário, sendo-lhes assegurado o direito de vista dos processos em julgamento. E podem solicitar as requisições e diligências que julgarem convenientes.

Dentre as atribuições do MPT, inclui-se, ainda, a de promover ou participar da instrução e conciliação em dissídios decorrentes da paralisação de serviços de quaisquer natureza, oficiando obrigatoriamente nos processos. Deve manifestar sua concordância ou discordância em eventuais acordos firmados antes da homologação, e pode recorrer em caso de violação à lei ou à Constituição Federal. Também está garantido ao MPT o direito de promover mandado de injunção, quando a competência for da Justiça do Trabalho.

6.1.4 Atuação com Órgão Agente

As Procuradorias Regionais do MPT recebem a denúncia e distribuem a um dos Membros da Coordenadoria de Defesa dos Interesses Difusos e Coletivos (CODIN), para que

seja autuada como Procedimento Preparatório (PP) ou Procedimento Investigatório (PI), onde é feita a coleta prévia de provas. Caso haja indícios de que a denúncia é verídica, o Procedimento Preparatório transforma-se em Inquérito Civil Público (ICP).

Durante o ICP, as partes podem juntar peças de informação, arrolar testemunhas e produzir provas. Uma vez comprovada a veracidade da denúncia, o Procurador responsável pelo ICP propõe à parte infratora a assinatura de um Termo de Compromisso de Ajustamento de Conduta (TCAC), por meio do qual compromete-se a regularizar a situação, sob pena de multa em caso de descumprimento, e, se for o caso, obriga-se ao pagamento do que for devido aos prejudicados, à título de indenização. Se assinado o TCAC, o processo é arquivado, mas o Procurador continua acompanhando seu cumprimento e pode ingressar com ação de execução se o infrator voltar a cometer as irregularidades.

Se não for aceita a assinatura do TCAC, o Ministério Público do trabalho recorre à via Judicial. Quando se tratar de lesão a interesse difuso¹ ou coletivo², o MPT utiliza a Ação Civil Pública, ajuizada perante uma Vara do Trabalho. Caso o interesse violado seja de natureza individual homogênea³, cabe a Ação Civil Coletiva, com pedido de reparação do dano sofrido pelo grupo, cujo julgamento também cabe ao Juízo de Primeira Instância. Quando a denúncia referir-se à cláusula de acordo ou convenção coletiva de trabalho que viole as liberdades individuais ou coletivas ou os direitos individuais indisponíveis⁴ dos trabalhadores, o MPT utiliza a ação anulatória, ajuizada perante a Seção Especializada em Dissídios Coletivos dos TRTs ou do TST. (MPT, 2008)

6.2 CARACTERIZAÇÃO DO PROJETO MPT DIGITAL (DTI/PGT, 2008)

6.2.1 Descrição

O MPT Digital será um sistema de controle de processos dos órgãos Agente e Interveniente do Ministério Público do Trabalho, que integrará a Procuradoria Geral, as

1 Interesse difuso – não se pode determinar que são os atingidos pela lesão, no entanto, eles estão ligados por circunstâncias de fato.

2 Interesse coletivo – de natureza indivisível, transindividual, tem como titulares grupos, categorias ou classes de indivíduos ligados entre si ou com a parte contrária por uma relação jurídica básica, como os trabalhadores de uma determinada categoria.

3 Interesses individuais homogêneos - são individuais em sua essência, sendo coletivos apenas na forma em que são tutelados, pela conveniência da solução a ser dada ao caso. Ex. direito à indenização por danos pessoais.

4 Direitos individuais indisponíveis - são indisponíveis, dentre outros, o direito à vida, à liberdade, à saúde e ao trabalho.

Procuradorias Regionais e os Ofícios da Instituição. Este sistema introduzirá o processo eletrônico na rotina de trabalho do MPT.

A informatização dos processos no âmbito do MPT representa uma mudança necessária à efetividade processual. Porém, além da velocidade, publicidade e da automação proporcionadas pelo processo eletrônico, também há novos e sérios desafios que precisam ser enfrentados para a viabilização desta alternativa tecnológica, tais como assinatura digital de documentos, controle eletrônico de documentos e a utilização de ferramentas que automatizam o fluxo dos processos no âmbito do MPT.

6.2.2 Objetivos

- Prover maior celeridade à atuação institucional do MPT
- Permitir uma utilização mais racional dos recursos humanos e materiais
- Possibilitar a redução da impressão de documentos
- Permitir a consulta rápida ao andamento de processos de qualquer unidade do MPT
- Facilitar a elaboração de relatórios gerenciais
- Possibilitar a utilização de numeração única de processos para todo o MPT
- Unificar o temário da CODIN
- Uniformizar as rotinas de trabalho das diversas unidades do MPT
- Permitir a integração digital com a justiça trabalhista
- Possibilitar a extração de informação gerencial e estratégica tanto em âmbito administrativo interno quanto da atuação do MPT na sociedade
- Viabilizar a identificação de procedimentos correlatos ou de mesma natureza
- Permitir a identificação de posicionamento prévio de membros sobre matérias em andamento

6.2.3 Justificativas

O grande número de processos da Justiça do Trabalho tem criado gastos enormes com produção, armazenamento, conservação e transporte de documentos impressos. Além dos gastos, o uso do documento impresso gera lentidão no andamento dos processos e implica em determinados riscos. Entre os riscos inerentes ao documento em papel, destacam-se a possibilidade de extravio, o desgaste natural causado pelo tempo e a vulnerabilidade a desastres como incêndios e inundações.

O documento digital, por outro lado, não tem os problemas do papel relacionados a espaço físico ocupado, conservação ou transporte. Além disso, a facilidade em se criar cópias de segurança, elimina o risco de perdas irreversíveis de documentos. Desde que tratados de maneira adequada, a autenticidade, o sigilo e a integridade dos documentos também são garantidos.

A utilização do documento digital também traz mais transparência, uma vez que informações em meio eletrônico podem ser disponibilizadas para o público através da Internet.

Além disso, é importante ressaltar os ganhos advindos da integração da informação em nível nacional entre todas as PRT's e ainda a automação do fluxo de procedimentos, gerando ganhos de produtividade.

A facilidade na disponibilização de informações aos interessados e a possibilidade de auto-conhecimento institucional em relação aos procedimentos em andamento e das manifestações já realizadas. O acompanhamento dinâmico dos problemas trabalhistas na localidade em que a PRT se insere, e em nível nacional, são também outros benefícios importantes que o MPT Digital proporcionará.

6.2.4 Metas

- No âmbito do MPT como Órgão Interveniente (custos legis):
 - Integração com Justiça Trabalhista
 - Suporte informacional e operacional à manifestação em custos legis, isto é, a intervenção nos feitos judiciais em curso onde haja interesse público a proteger
 - Gestão de fluxo processual, coleta e disseminação de informações para os interessados

- No âmbito do MPT como Órgão Agente:
 - Integração com Justiça Trabalhista
 - Suporte a tratamento digital de documentos recebidos em diversos meios físicos
 - Suporte informacional e operacional para instauração de procedimentos investigatórios e inquéritos civis no âmbito da Justiça do Trabalho, para a defesa de interesses coletivos, difusos ou individuais homogêneos
 - Gestão de fluxo processual, coleta e disseminação de informação para interessados
- Gestão integrada de informações estratégicas
- Integração entre atividades finalísticas e atividades administrativas de suporte

6.2.5. Escopo

- Órgão Agente
 - Cadastramento e acompanhamento de denúncias
 - Distribuição automática
 - Criação e acompanhamento de Procedimento Preparatório digital
 - Criação e acompanhamento de Inquérito Civil digital
 - Consulta on-line de situação de procedimentos
 - Agenda de atividades de membros
 - Geração de documentos com assinatura digital
 - Protocolo digital
 - Acompanhamento de prazos legais
 - Elaboração de relatórios estatísticos e mapas de atividades
 - Publicação de Termos de Ajuste de Conduta na Internet.
- Órgão Interveniente
 - Cadastramento de processos trabalhistas
 - Distribuição automática
 - Criação de guias
 - Criação e acompanhamento de pareceres digitais
 - Geração de documentos com assinatura digital

- Criação de banco de pareceres
- Elaboração de relatórios estatísticos e mapas de atividades
- Infra Estrutura de Tecnologia da Informação
 - Implantação da virtualização de processos (processo digital) – LEI Nº 11.419, de 19/12/2006
 - Implantação de ferramenta para Fluxo eletrônico de processos (Workflow)
 - Implantação de ferramenta para manuseio e disponibilização dos processos em meio digital (GED)

6.2.6 Não escopo

- Migração dos dados dos sistemas atualmente existentes
- Digitalização de documentos arquivados

6.2.7 Premissas

- Planejamento estratégico direcionador
- Adoção de Políticas e Diretrizes de Tecnologia da informação no âmbito do MPT
- Conhecimento e racionalização de fluxos de atividades, advindos de:
 - definições do Conselho Superior do MPT
 - modelagem de processos
- Política de Informação que aborde responsabilidades, manuseio e guarda de documentos e processos no formato digital
- Edição de normas que regulem o processo digital no MPT
- Existência de fornecimento confiável de energia elétrica na PGT
- Existência de equipamentos de reserva para o caso de falhas nas máquinas que hospedarão o sistema
- Treinamento em segurança da informação para os servidores do MPT
- Aumento do quadro da Divisão de Desenvolvimento de Sistemas em pelo menos mais quatro Analistas de Informática
- Mapeamento e gestão de processos (fluxo de atividades e informações necessárias) para digitalização do fluxo de trabalho

6.2.8 Restrições

- A parte do sistema referente ao Órgão Agente do MPT obedecerá à Resolução nº 69 de 12/12/2007
- A parte do sistema referente ao Órgão Interveniente deverá ser compatível com o modelo de processo digital adotado pela Justiça do Trabalho e com a instrução normativa nº 28 do TST
- O sistema se guiará, no que couber, pelo disposto na Lei 11419/2006
- O sistema será hospedado na PGT
- O sistema funcionará na plataforma Web e estará disponível para acesso, mediante senha pessoal, de qualquer computador conectado à internet
- O sistema deverá funcionar com os navegadores Mozilla Firefox e Internet Explorer
- Os documentos gerados pelo sistema deverão ser compatíveis com software livre
- O sistema deverá estar disponível ininterruptamente, no regime de 24/7

6.2.9 Gerência do projeto

A gerência do projeto será feita pela Divisão de Desenvolvimento de Sistemas do Departamento de Tecnologia da Informação.

6.2.10 Interface com outros projetos e sistemas

- Sistema do Departamento de Recursos Humanos (MentorRH) – Busca de informações referentes a férias e afastamentos de procuradores, para fins de distribuição automática, entre outros.
- Sistema de Diárias e Passagens (DIP) - Busca de informações referentes viagens de procuradores a serviço, para fins de distribuição automática, entre outros.
- Sistema Integrado de Protocolização e Fluxo de Documentos na Justiça do Trabalho (e-DOC) – Para envio de petições eletrônicas.

7 CONCLUSÕES

Como pode ser visto nos dados apresentados no decorrer deste trabalho, a área de TI do Ministério Público do Trabalho, encontra-se em um momento de profunda transição. De uma organização que, em um contexto de intensas mudanças por que tem passado a Tecnologia da Informação, atua de maneira reativa, apenas dando respostas às demandas geradas por essas mudanças, passará a atuar de maneira planejada, obedecendo a normas consagradas e seguindo metodologias modernas e bem aceitas, a fim de melhor cumprir a sua missão.

Com relação ao projeto “MPT Digital”, que já possui a definição de uso da plataforma de desenvolvimento JEE5, resta a definição de que processo de desenvolvimento de software será utilizado. Se a metodologia formal do Processo Unificado (RUP) ou um dos processos ágeis citados. Propõe-se aqui uma solução mista, onde a fase de iniciação utilize as disciplinas descritas no RUP, uma vez que os requisitos são bem definidos e não deverão sofrer grandes alterações ao longo do projeto, passando então a utilizar nas fases seguintes os “Sprints” definidos no SCRUM.

E para que as mudanças ora em curso venham a ter caráter definitivo dentro da instituição podendo ser constantemente avaliadas e reestruturadas é recomendável que seja criado um grupo que, recebendo treinamento em gestão e governança de TI, possa assumir esse importante papel. Aqui se sugere a opção pelo COBIT, cujo modelo de quatro domínios se mostra bastante adaptável a uma instituição pública nos moldes do MPT, além do fato de que o Tribunal de Contas da União (Acórdão nº 1603/2008) usa os processos desse modelo como parâmetros para as suas ações de fiscalização nas áreas de TI dos órgãos/entidades da Administração Pública Federal.

8 GLOSSÁRIO

8.1. ISO/IEC

Fundada em 23 de Fevereiro de 1947, em Genebra, na Suíça, a ISO aprova normas internacionais em todos os campos técnicos, exceto na eletricidade e eletrônica, cuja responsabilidade é da International Electrotechnical Commission (IEC), fundada em 1906.

No Brasil é representada pela Associação Brasileira de Normas Técnicas (ABNT).

Embora popularmente se acredite que a expressão "ISO" é um acrônimo de "International Standards Organization", na realidade o nome originou-se da palavra grega "ἴσος" ("isos"), que significa igualdade. Evita-se com isso que a organização possua diferentes acrônimos em diferentes idiomas, já que em inglês, o acrônimo seria IOS ("International Organization for Standardization"), em francês OIN ("Organisation internationale de normalisation"), e assim por diante. A escolha do nome "ISO" reflete assim o objetivo da organização, ou seja, a padronização entre as diversas culturas.

8.2 STAKEHOLDER

Stakeholder ou, em português, parte interessada ou interveniente, refere-se a todos os envolvidos em um processo, por exemplo, clientes, colaboradores, investidores, fornecedores, comunidade, etc. O processo em questão pode ser de caráter temporário (como um projeto) ou duradouro (como o negócio de uma empresa ou a missão de uma organização sem fins lucrativos). O sucesso de qualquer empreendimento depende da participação de suas partes interessadas e por isso é necessário assegurar que suas expectativas e necessidades sejam conhecidas e consideradas pelos gestores. De modo geral, essas expectativas envolvem satisfação de necessidades, compensação financeira e comportamento ético. Cada interveniente ou grupo de intervenientes apresenta/representa um determinado tipo de interesse no processo. O envolvimento de todos os intervenientes não maximiza obrigatoriamente o processo, mas permite achar um equilíbrio de forças e minimizar riscos e impactos negativos na execução desse processo. Uma organização que pretende ter uma existência estável e duradoura deve atender simultaneamente as necessidades de todas as suas partes interessadas. Para fazer isso ela precisa gerar valor, isto é, a aplicação dos recursos usados deve gerar um benefício maior do que seu custo total. O termo "stakeholders" foi

criado para designar todas as pessoas, instituições ou empresas que, de alguma maneira, são influenciadas pelas ações de uma organização.

O termo "Stakeholder" foi utilizado pela primeira vez por R. Edward Freeman[1] no livro: "Strategic Management: A Stakeholder Approach", (Pitman, 1984), para referir-se àqueles que possam afetar ou são afetados pelas atividades de uma empresa. Estes grupos ou indivíduos são o público interessado ("stakeholders"), que segundo Freeman deve ser considerado como um elemento essencial na planificação estratégica de negócios.

8.3 FRAMEWORK

Conjunto de conceitos usado para resolver um problema de um domínio específico.

8.4 FRAMEWORK DE SOFTWARE

Framework ou **arcabouço** é uma estrutura de suporte definida em que um projeto de software pode ser organizado e desenvolvido. Pode incluir programas de suporte, bibliotecas de código, linguagens de script e outros softwares para auxiliar no desenvolvimento e unir diferentes componentes de um projeto de software. São projetados com a intenção de facilitar o desenvolvimento de software, habilitando designers e programadores a gastarem mais tempo determinando as exigências do software do que com detalhes de baixo nível do sistema.

8.5 PMI

Estabelecido em 1969 e sediado na Filadélfia, Pensilvânia EUA, o Project Management Institute (PMI®) é a principal associação mundial sem fins lucrativos em Gerenciamento de Projetos, atualmente com mais de 250.000 associados em 170 países. Dentre as suas publicações destaca-se o Guia PMBOK (Project Management Body of Knowledge).

8.6 AD HOC

Ad hoc é uma expressão latina que quer dizer "com este objetivo". Geralmente significa uma solução designada para um problema ou tarefa **específicos**, que não pode ser aplicada em outros casos.

Em engenharia de software, o termo *ad hoc* é utilizado para designar ciclos completos de construção de softwares que não foram devidamente projetados, para assegurar que este esteja de acordo com os padrões e especificações de prazo, qualidade e custo. Geralmente está focado na necessidade específica de algum usuário ou situação, sem uma visão ampla e racionalizada.

8.7 GOVERNANÇA DE TI

“Governança de TI é um conjunto de práticas, padrões e relacionamentos estruturados, assumidos por executivos, gestores, técnicos e usuários de TI de uma organização, com a finalidade de garantir controles efetivos, ampliar os processos de segurança, minimizar os riscos, ampliar o desempenho, otimizar a aplicação de recursos, reduzir os custos, suportar as melhores decisões e conseqüentemente alinhar TI aos negócios.” (Prof. João R. Peres – FGV)

8.8 TEST-DRIVEN DEVELOPMENT

Test-Driven Development ou Desenvolvimento Guiado por Testes, ou simplesmente TDD, consiste numa técnica de desenvolvimento de software onde primeiro são criados os testes e somente depois é escrito o código necessário para passar por eles.

Baseado na premissa de que quando se implementam testes unitários depois do código estar pronto, há a tendência de se implementar testes de baixa qualidade, pois inconscientemente se escreve testes para rodar no código produzido, quando o correto é o contrário, o código é que deve passar no teste previamente implementado. (ALENCAR, 2008)

8.9 USERS STORIES

Ferramenta muito utilizada nos processos ágeis de desenvolvimento de software que consiste em descrever uma funcionalidade de um sistema utilizando a linguagem de quem conhece a realidade do problema e/ou solução, ou seja, o cliente.

REFERÊNCIAS

ALENCAR, Roberto. **Test Driven Development (TDD) - O Desenvolvimento Guiado por Testes**, ago 2008. Disponível em: <<http://www.javafree.org/javabb/file/d.jbb?fileId=973>> Acesso em: 3 nov. 2008.

ALMEIDA, Rodrigo Rebouças de. **Processos de desenvolvimento de software** – Slides. 2007. Disponível em: <<http://www.rodriigor.com>>

BALBO, Luciano de Oliveira. **Uma Abordagem Correlacional dos Modelos CobiT / ITIL e da Norma ISO 17799 para o tema Segurança da Informação**. São Paulo, 2007. Disponível em: <<http://www.pece.org.br/cursos/TI/monografias/MBA-MONO-LucianoBalbo.pdf>> Acesso em 6 nov. 2008.

BARTIE, Alexandre. Agilidade ou controle operacional? Os dois. - **Engenharia de Software Magazine**, ano 1, 2007. Disponível em: <<http://www.devmedia.com.br/esm>>

COHN, Mike. **Uma Introdução ao SCRUM**. www.mountangoatsoftware.com. Tradução e adaptação Cesar Brod www.brod.com.br

COSTA, Luciana. **O Cobit como Modelo de Gestão da TI**, fev 2007. Disponível em: http://imasters.uol.com.br/artigo/5388/gerencia/o_cobit_como_modelo_de_gestao_da_ti. Acesso em out. 2008

DTI/PGT. Diretoria de Tecnologia da Informação/ Procuradoria Geral do Trabalho. **Projeto MPT Digital**. 2008

FAGUNDES, Eduardo Mayer. **COBIT - Um kit de ferramentas para a excelência na gestão de TI**. 2004. Disponível em: http://www.efagundes.com/Artigos/Arquivos_pdf/cobit.pdf. Acesso em: 15 out. 2008

JENDROCK, Eric et al. **The Java EE 5 Tutorial**. Out. 2008. Disponível em <http://docs.sun.com>

KUHN, Giovane Roslindo e PAMPLONA, Vitor Fernando. **Apresentando XP. Encante seus clientes com Extreme Programming**. 2004. Disponível em <http://www.javafree.org/content/view.jf?idContent=5>. Acesso em 4 nov. 2008.

LIMA, Geovane Nogueira. **ISO/IEC 15504**. Recife, nov. 2006. Disponível em: http://www.cin.ufpe.br/~processos/TAES3/slides-2006.2/ISO15504_V02.ppt. Acesso em: 14 out. 2008

LIMA, Gleydson. Apresentação: **Processos de engenharia de Software**. J2EEBrasil, 2004. <http://www.j2eebrasil.com.br>

MANERA, Aline Fátima et al. - **Modelo de Qualidade – CMMI**. Universidade Federal de São Carlos – Departamento de Computação, set 2007. Disponível em:

www.comp.ufscar.br/~bruno_abrahao/Artigos/CMMI.pdf . Acesso em 17 out. 2008

MPT - Ministério Público do Trabalho. **O que é o MPT?**. Disponível em <http://www.pgt.mpt.gov.br>. Acesso em 10 nov. 2008.

NEVES, Patrícia Ferrão. **CE 230 – Qualidade, Confiabilidade e Segurança em Software**. ITA Instituto Tecnológico da Aeronáutica. Disponível em: <http://br.geocities.com/patriciagn/norma/NormaISO15504.ppt> Acesso em: 14 out. 2008

PEREIRA, Paulo; TORREÃO, Paula; MARÇAL, Ana Sofia. Entendendo Scrum para Gerenciar Projetos de Forma Ágil. **Revista MundoPM**, 2007. Disponível em http://www.cesar.org.br/files/file/SCRUM_MundoPM-Abril-Maio-2007.pdf. Acesso em 5 nov. 2008.

PMI/MG. **Tradução livre do PMBOK 2000**. Minas Gerais, 2002. Disponível em: www.pming.org.br. Acesso em out. 2008.

PMI/SP. **PMBOK GUIDE**. São Paulo, 2004. Disponível em: www.pmis.org.br/pmbok.asp Acesso em: out. 2008.

RICCIARDI, Gianni -**Otimização da estrutura dos processos de TI – COBIT, ITIL e demais conceitos, metodologias e ferramentas**. Disponível em: <http://www.isaca.org.br/novoportal/modules/wfsection/article.php?articleid=9>

SANCHES, Ivan. **Scrum em 2 minutos**. Fev. 2007. Disponível em: <http://dojofloripa.wordpress.com> Acesso em: 5 nov. 2008.

SCRUM ALLIANCE. **What is Scrum?** Disponível em: <http://www.scrumalliance.org> Acesso em: 5 nov. 2008

SILVA, Luiz Augusto Matos da. **Plataforma Java J2EE: a atual pedida no cenário corporativo**. 2006. Disponível em http://www.luizmatos.eti.br/artigos/Plataforma_Java.pdf. Acesso em: 10 nov. 2008

SUN Microsystems. **Tutoriais JAVA**. Disponíveis em: <http://java.sun.com/docs/books/tutorial/> Acesso em: 11 nov. 2008.

TAMAKI, Paulo Augusto Oyamada. **Uma extensão do RUP com ênfase no gerenciamento de projetos do PMBOK baseada em process patterns**. São Paulo, 2007. Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para o título de mestre em engenharia.

TELES, Vinícius Manhães. **EXTREME PROGRAMMING**. Improve It, 2008. Disponível em <http://www.improveit.com.br/xp>. Acesso em 5 nov. 2008

TI-Exames (<http://www.tiexames.com.br>). **Fundamentos em gerenciamento de serviços de TI baseado na ITIL V2**. 2006. Disponível em: http://www3.solar.com.br/~asergio/apostila_ITIL.pdf Acesso em: 6 nov. 2008.

TRAINNING. **Governança de TI: comparativo entre COBIT e ITIL**. Set. 2007.

Disponível em: <http://www.trainning.com.br/download/Apostila_ITIL_Cobit.pdf> Acesso em 6 nov. 2008