**Prof. Kechi - Gerência, Qualidade e Tecnologia de Software**

* 1 trabalho + 2 provas
  + Trabalho = Plano de Projeto, geralmente do Projeto de Formatura
* Há entregas (parciais) do projeto
  + "Montagem" do plano durante as aulas
* P1: 9/OUT
  + Entrega da primeira parte do projeto
* P2: 04/DEZ
  + Prova com peso 2
  + Entrega da última parte do projeto (revisão)

Trabalho em grupo (mesmo que o de PF)

[kechi.hirama@poli.usp.br](mailto:kechi.hirama@poli.usp.br)

Sala C2-38

3ªs tarde

5ªs manhã

6ªs tarde

09/10 P1 – prova sem consulta

04/12 P2 – prova com consulta

T – em grupo, objetivo : Plano de projeto

Bibliografia:

Pressman, R. Eng Soft., 3ª edição, 1995

Sommerville, I Eng Soft., 8ª edição, 2007

PMI, Project Management Body of Knowledge, PMBOK, 3ª edição 2004

[www.pmi.org](http://www.pmi.org)

[www.swebok.org](http://www.swebok.org)

[www.sei.cmu.edu](http://www.sei.cmu.edu)

Periódicos:

IEEE

ACM of the communications

## Aula 1

## Qualidade

Eficácia, defeitos, atender especificação, padronização, manutenabilidade.

Visão do usuário: eficiência, pode ser produtivo, de fácil uso.

Visão do cliente: custo, prazo.

Desenvolvedor:

Gerente: saber se a estimativa de antes foi mantida (preço, tempo).

* Afeta os custos de desenvolvimento.
* Afeta os prazos.
* Afeta a satisfação.

Satisfação do cliente

Qualidade – conformidade aos requisitos.

Custo – quanto mais baixo melhor.

Prazo – entrega dentro do prazo.

Defeitos – disponibilidade confiabilidade.

Mitos da qualidade

As falhas de SW são inevitáveis.

Os testes garantem (conduzem) a qualidade, testes só servem para encontrar defeitos.

A qualidade custa dinheiro. (tem que ir melhorando ao longo de vários projetos)

## TQM. Total Quality Management

Planejamento da qualidade (apoio da alta gerência)

Responsável:

Diretrizes da organização

Políticas da organização

### Resolução de problemas

PDCA - ciclo de melhorias. Planejar, Fazer, Verificar, Agir

### Gerenciamento de processos

Grupo de processos – pessoas convidadas a trabalhar pensando em melhorar processos.

TQM – forma uma base histórica para melhoria contínua (da organização).

## Busca da Qualidade de SW

Software artesanal

Software profissional

Padronizado

Documentado

Boa relação custo x benefício

# Aula 2

## Qualidade de Software

### Obstáculos

* Cultura da organização
  + A imposição de uma nova cultura no início há rejeições dos outros.
* Complexidade das soluções
* Clientes/usuário não identificados
  + O cliente tem que ser claramente identificado.
  + Para criar a solução mais adequada para o que ele necessita.
* Custos e prazos mal definidos.
  + Tem que ter conhecimento de projetos passados.
  + Depois se começa a ir aceitando mais as estimativas.
* Planejamento inexistente
  + Não deve apenas mandar os caras trabalhar.
  + Roteiro, planejamento, não necessariamente o cronograma.
  + Cronograma é o resultado de um bom planejamento.
  + Atividade de se estruturar as atividades que vão resultar no produto final.
* Base histórica inexistente
  + Registro de informações de projetos passados que é referência para novos projetos.
  + Tanto sucessos quanto fracassos são interessantes.
* Processos inexistentes
  + Processos organizacionais.
  + Cada um tem que saber qual é o seu papel dentro da organização.

### Engenharia de Software

Métodos

Técnicas

Ciclos de vida

Gerenciamento

Garantia e controle de qualidade

Testes, aonde se vê que conseguiu chegar ao objetivo que queria.

### Produtividade X Qualidade

Tem que fazer produtos de qualidade.

Tem que entregar no prazo (tem que ser produtivo).



Técnicas, métodos, ferramentas, etc.

### Produto

Conformidade com necessidades do cliente.

Conformidade com padrões.

Conformidade com boas práticas.

Conformidade com prazos e custos.

### Processo

Atividades definidas. Todas as pessoas envolvidas reconhecem a atividade, sabem como fazer.

Tem que passar por repetição.

Gerenciado. Alguém tem cuidar do processo. Tem que definir ele. Ele precisa ser monitorado.

Dados de processo.

### Modelos e Padrões de Qualidade (Normas)

ISO/IEC 9126 – Como classificar um produto quanto a qualidade. – Para produto.

ISO/IEC 9001/90003 (diretriz de como aplicar a 9001 para software) – Para processo.

ISO 12207 – Ciclo de vida.

CMM (Processo) – CMMI

ISO/IEC 15504 (avaliação / certificação)

MPS-Br – Melhoria de Processos de Software brasileiro.

### Processo

Conjunto de atividades organizadas para determinado fim.

Artefatos (entradas e saídas)

Regras (método, técnica e padrões)

Recursos de forma em geral (materiais, humanos)

### Processos de desenvolvimento

Cascata

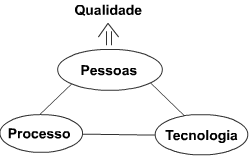
Espiral

RUP

Reuso (alguém cria componente reusáveis)

XP

### Fator Crítico de Sucesso (FCS) para Qualidade



### Projetos de TI (Tecnologia da Informação)

(Standish Group 2003 – Chaos Report)

Número de projetos: 13.522

34% com sucesso

15% cancelado

51% com problemas

(52% requisitos implementados)

(43% fora do custo)

(82% fora do prazo)

(Valores 2002)

Gastos: US$ 255 bi

Perdas: US$ 38 bi

Custo adicional: US$ 17 bi

Desperdício: U$ 55 bi

# Aula 3

## Tecnologias emergentes (“Novas”)

### Engenharia de proteção

Apóia o desenvolvimento e manutenção com intuito de resistir aos ataques maliciosos com intenção de danificar os sistemas e/ou seus dados.

### Proteção de sistemas

Aplicações

Engenharia de software

Infra-estrutura

Gerenciamento de sistemas

### Ameaças de proteção

Confidencialidade

Integridade – os dados não podem ser danificados

Disponibilidade

### Controles de proteção

Controle para que os ataques sejam mal sucedidos.

Controle para detectar e repelir os ataques.

Controle para recuperação do sistema.

### Arquitetura para proteção

Projeto da arquitetura

Boas práticas

Projeto de implantação

Proteção

Distribuição

Conflitos potenciais

### Capacidade de sobrevivência

Habilidade de continuar a fornecer os serviços enquanto o sistema está sob ataque.

Resistência

Reconhecimento

Recuperação

### Atividades de projeto

Compreensão do sistema

Identificação dos serviços críticos

Simulação dos ataques

Análise da capacidade de sobrevivência

### SOA – Service Oriented Architecture

Domínio do problema (negócio)­­­­

|Gap Semântico| - Programação estruturada e orientada a objeto tentaram estreitar o gap

Domínio da solução (computação)

|  |  |  |  |
| --- | --- | --- | --- |
| Serviço ao cliente | Gestão de encomendas | Gestão de contas |  |
| Várias camadas, com funções A, B   |  | | --- | | A | | B | |  | | |  | | --- | | A | | C | |  | | |  | | --- | | C | | D | |  | | Aplicações monolíticas |
| Repositório 1 | Repositório 2 | Repositório 3 | Repositório de dados |

As aplicações podem acessar repositórios de outras áreas.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | A | B | C | D | |

### Serviço X Componente

Serviço é um “componente” de software reutilizável, fracamente acoplado.

Serviço é distribuído.

Serviço é acessado pelos programas.

Existe um consumidor, existe um provedor e um registro.

Registro – Consumidor – Provedor

SOA (Service Oriented Architecture):

* Domínio do problema x Domínio da solução: Gap semântico entre o negócio e a computação.
* SOA x Aplicações Monolíticas: Serviços x Componentes. Exemplos: Serviço ao cliente, gestão de encomendas, gestão de compras.
* Serviço: “Componente” de software reutilizável e fracamente acoplável. É distribuído e acessado pelos programas.



* Engenharia de SOA: Desenvolvimento para reuso (Eng. Domínio) e desenvolvimento com reuso (Eng. Aplicação).
* Benefícios:
* Serviços podem ser providos localmente ou terceirizados por fornecedores externos.
* Serviços são independentes de linguagem.
* Serviços preservam investimentos em legados.
* Serviços facilitam a computação inter-organização.
* Entidades e Empresas: W3C, Oasis, SOA Alliance, IBM, Microsoft.
* Testes: Serviços são caixas-pretas. Testes unitários (caixa-branca) ficam comprometidos.

### Desenvolvimento Orientado a Aspectos

Desenvolvimento que usa a abstração chamada aspecto.

Requisitos funcionais.

Requisitos não-funcionais.

Usado em conjunto com a abordagem OO.

Aspecto é uma sobreposição do OO.

#### Separação de assuntos

Um aspecto implementa um assunto.

Assuntos primários ou centrais – requisitos funcionais

Assuntos secundários – requisitos não funcionais

1 2 3 Requisitos funcionais

Requisito não funcional de proteção

Requisito de desempenho

#### Arquitetura do sistema orientado a aspectos

Assuntos

Primários

Requisito não funcional 2

Requisito não funcional 1

Requisito não funcional 4

Requisito não funcional 3

### Gestão de Projetos

#### Conceitos gerais

Gerente de projeto (PM) – pessoa designada pela organização executora para atingir os objetivos do projeto. [PMBOK]

Projeto – um esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo. [PMBOK]

Planejamento – inclui as atividades de realizar estimativas de atributos de trabalho e tarefas, de determinar recursos (materiais e humanos), de negociar os compromissos, de produzir um cronograma e analisar os riscos do projeto. [CMMI]

Acompanhamento – o desempenho do projeto deve ser monitorado e medido regularmente. [PMBOK]

Plano de projeto – É um documento formal, aprovado e usado para guiar a execução e o controle do projeto. [PMBOK]

#### Ciclo de vida de projeto

Define o trabalho a ser realizado

Quais são as regras a serem seguidas

Como as entregas são verificadas, testadas

Quem são os envolvidos

Inicial – escopo

Intermediárias – plano, progresso (acompanhamento)

Final – Aprovação, lições aprendidas

#### Partes interessadas no projeto

Cliente / usuário

Equipe de desenvolvimento

Área de qualidade

Gerência a qual o gerente se reporta

PMBOK – coletânea de boas práticas

#### Processo de gerenciamento de projeto

Iniciação

Planejamento

Execução

Monitoração e controle

Fechamento

#### Áreas de conhecimento – PMBOK

Gerenciamento de

Integral

Escopo

Tempo

Custos

Qualidade

RH

Comunicações

Riscos

Aquisições

# Aula 4

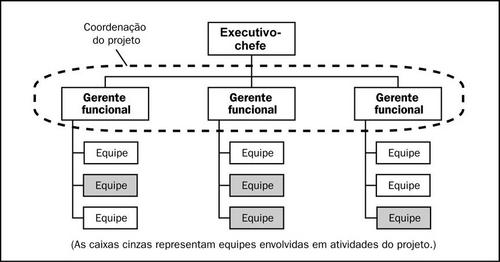
## Formação de equipes de projeto

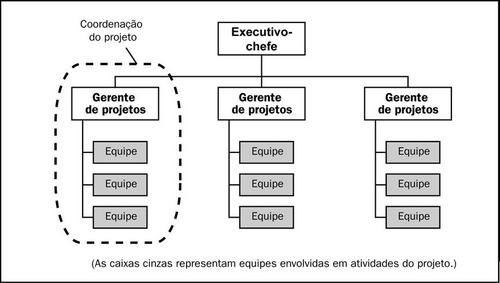
### Estrutura organizacional

Coordenações de projeto:

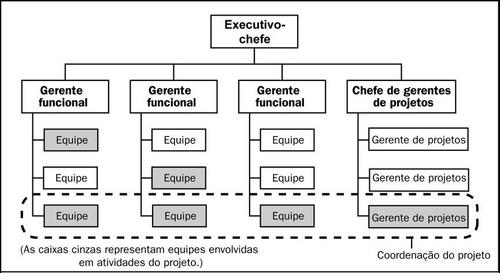
Estrutura Organizacional (PMBOK):

* Funcional (por departamento): Projeto é dividido e atribuído a áreas funcionais da organização, com a coordenação feita pelos níveis mais altos e funcionais de gerência.

* Por projeto: Recursos alocados especificamente para o projeto.



* Matricial: Equipes compostas por pessoas de diversas especialidades são reunidas com o objetivo de realizar tarefas com características temporárias.



### Estimativas

Tamanho

Funcionalidade

Esforço

Custo

Prazo

Produtos de trabalho

### Estimativa de tamanho

Linhas de código (fonte) – forma mais tradicional

Número de página de documentação

20 documentos com 15 páginas, 10 diagramas

### Estimativa de funcionalidade

Quantas funções o projeto vai ter.

Pontos de função (APF) – (Pressman)

Precisa conhecer o ambiente externo que vai estimular essas funções.

A ISO padronizou o cálculo.

Baseado na visão do usuário.

Independente de tecnologia.

Usuário ↔ Sistema ↔ BD (externo)

Certificado de especialista em APF.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Parâmetros | Quantidade | Pesos simples | Pesos médios | Pesos complexos | Contagem |
| N° de entradas do usuário | 1 (10 campos) | 1x3 | 0x4 | 0x6 | 3 |
| N° de saídas do usuário | 1 (8 campos) | 1x4 | 0x5 | 0x7 | 4 |
| N° de consultas de usuário | 1 (6campos) | 1x3 | 0x4 | 0x6 | 3 |
| N° de arquivos lógicos internos | 1 | 1x7 | 0x10 | 0x15 | 7 |
| N° de arquivos lógicos externos | 1 | 1x5 | 0x7 | 0x10 | 5 |
|  | | | | PFB= | 22 |

Pontos de caso de uso (UCP)

### Características gerais do sistema

Cada uma das características vai possui um nível de influência que vai de 0 a 5.

1. Comunicação de dados [3]
2. Funções distribuídas [0]
3. Desempenho [0]
4. Configuração do equipamento [1]
5. Volume de dados [1]
6. Entrada de dados on-line [3]
7. Interface com usuário [5]
8. Atualização on-line [1]
9. Processamento complexo [0]
10. Reusabilidade – vai ser desenvolvido pensando no reuso [0]
11. Facilidade de implantação [0]
12. Facilidade operacional – vai ser fácil de ser operado [2]
13. Múltiplos locais [0]
14. Facilidade de manutenção [0]

Soma-se um FA (fator de ajuste) = \_16\_

PF = PFB[0.65 + 0.01FA]

+35% PFB

PFB

-35% PFB

Mas precisa transformar em esforço.

Para uma única pessoa. (126 horas para uma pessoa)

### Ciclo de vida de Projeto

Análise (20%)

Projeto (15%)

Codificação (25%)

Testes (40%)

Se 1 caso de uso é aproximadamente 5 classes

# Aula 5

Plano de Projeto

Escopo funcional -> documento de requisitos

Escopo de projeto -> plano de projeto. É o limite das coisas. Planejar quais os artefatos que se deve produzir ao longo do projeto.

Diagramas, relatórios são artefatos de projeto.

Projeto -> WBS -> EAP (Estrutura analítica do projeto)

WBS para software tem que ter o sistema e o primeiro nível é fases do projeto.

Sistema

Gerenciamento

Plano do projeto

Análise

Especificação funcional

Modelo estático

Modelo dinâmico

Plano de testes

Projeto

Especificação de projeto

Diagrama de arquitetura

Plano de testes de integração

Implantação

Testes

Relatório de testes

Manual de implantação

Manual do usuário

Estimativas em Homem/hora

Restrições: Curto, prazo, linguagem, prazo.

# Aula 6

## 5. Organização do Projeto

### 5.1 Organograma

Diagrama em blocos:

Gerente

Analista

Projetista

Programador

### 5.2 Papéis e Responsabilidade

Gerente ... responsabilidade

Analista

...

Programador

Cliente

Analista de qualidade

Analista de configuração

### 6. Equipe

### 6.1 Alocação de pessoal

Função

Conhecimentos específicos

Quantidade

% Alocação

Período

### 6.2 Equipe do projeto

Nome

Função

Período

E-mail

Telefone

### 7. Recursos

Colocar não aplicado onde não tem

### 7.1 Ferramentas de SW

Licenças

### 7.2 Equipamentos

Mesmo não sendo utilizado

### 8. Acompanhamento de Projeto

### 81. Indicadores

Progresso – andamento conforme cronograma

### 8.2 Atividades de acompanhamento

Fase 1 -> Fase 2 -> Fase 3 -> Fase 4-> Produto

Semanal

Quinzenal

Mensal

Fim de fase

### 9. Cronograma e Orçamentos

Cronograma

Item, atividade, duração, início, fim, recursos, gráfico

Orçamento

Consolidação de todos os recursos

Equipe

Analista R$

Arquiteto R$

Programador R$

Recursos

Ferramentas R$

Equipamentos R$

Material consumo R$

Terceiros R$

Viagens R$

Treinamento R$

Total

### Risco de projeto

Riscos são incertezas que existem em todos os projetos.

O risco é um evento ou condição incerta que, se ocorrer, provocará um efeito **positivo** (oportunidades) ou **negativo** (ameaças) nos objetos do projeto.

### Categorias de risco

Técnica – requisitos, tecnologia, complexidade, interface, desempenho

Externa – subcontratações (terceirizações), fornecedores, regulamentos, mercado, cliente (negócio)

Organizacional

Recursos

Materiais

Humanos

Financiamento

Priorização – plano estratégico

Gerenciamento – estimativas erradas, controle, comunicação, stakeholders

### Processo de gerenciamento de riscos

Identificar risco -> riscos potenciais

Analisar risco -> riscos priorizados

Planejar risco -> ações planejadas

Acompanhar risco -> cenários de riscos

Resolver risco -> correção do desvio

### Risco

Probabilidade x Impacto = exposição

Probabilidade, Impacto

Escalas cardinais (0, 1, 0.2, 0.3, ...)

Escalas ordinais (baixa, média, alta)

### Matriz de exposição ao risco

É bom ter uma única matriz.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Probabilidade | Alto |  |  |  |
| Médio |  |  |  |
| Baixo |  |  |  |
|  | Baixo | Médio | Alto |
|  | Impacto | | | |

### Plano de ação

Preventiva (reduzir a probabilidade)

Minimização (reduzir o impacto)

Contingência (ação de limitação)

# Aula 7

# Gerenciamento de riscos – exercício

Deseja-se desenvolver um sistema de matrícula de alunos.

- Funções

* Cadastrar alunos
* Elaborar horários
* Elaborar histórico escolar
* Cadastrar notas
* Emitir certificados

Cenário organizacional

* Serão criadas +10 novas disciplinas
* Serão contratados +20 novos professores
* Será criado 1 departamento novo
* Deve-se atender nova diretriz do MEC

Premissas e restrições

* Desenvolvimento em Web
* Infra-estrutura nova será implantada

Prazo

* Dezembro / 08

Preencher tabela conforme seção 10 do Plano de Projeto e montar uma matriz de exposição de riscos. Levar em conta os tipos de riscos (técnico, externo, organizacional e gerenciamento de projeto).

Probabilidade (Baixa – B, Média – M, Alta – A)

Impacto (Baixo – B, Médio – B, Alto – A)

Ações

Evitar – E

Transferir – T

Mitigar – M , ocorreu e precisa tratar, evitar maiores estragos

Aceitar – A

## Resolução:

Esta seção descreve a análise de riscos do projeto. Apresenta a descrição do risco, probabilidade, impacto, exposição ao risco e as ações requeridas.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Categoria** | **Descrição do Risco** | **Probabilidade** | **Impacto** | **Exposição ao Risco** | **Ação Requerida**  **(ETMA)** |
| 0 | Técnico | A equipe de implementação não domina desenvolvimento WEB | B | A | BA | E |
| 1 | Infra-estrutura nova não pode ser implantada | B | A | BA | M |
| 2 | Externa | Mudanças de requisitos funcionais e não-funcionais | A | A | AA | A |
| 3 | Mudanças de escopo | A | A | AA | E |
| 4 | Necessidade de terceirizar parte do sistema. | B | B | BB | E |
| 5 | Interface não agrada cliente | M | M | MM | M |
| 6 | Indefinição do escopo e requisitos pelo cliente | A | A | AA | M |
| 7 | Implementação difícil ou custosa de diretrizes do MEC. | A | M | AM | M |
| 8 | Organizacional  Material | Não há infra-estrutura adequada para desenvolvimento | B | A | BA | A |
| 9 | Organizacional  Humanos | Contratação de novos programadores | M | M | MM | E / M |
| 10 | Desligamento de funcionário chave. | M | A | MA | E / M |
| 11 | Organizacional  Financiamento | Corta de investimento | A | A | AA | T |
| 12 | Organizacional  Priorização | Prioridade alocada não suficiente | M | M | MM | M |
| 13 | Gerenciamento | Estimativas erradas | A | A | AA | M |
| 14 | Falha de comunicação | A | M | AM | E |
| 15 | Processos internos falhos | M | M | MM | M |

Matriz de resposta aos riscos.

|  |  |  |  |
| --- | --- | --- | --- |
| A |  |  | 13 |
| M |  | 12, 15 | 14 |
| B |  |  |  |
|  | B | M | A |

# Aula 8

Planejamento de Projeto – Resumo

1. Escopo de projeto
   1. Funcional – funções
   2. De Projeto – WBS (Estrutura analítica do projeto)
2. Ciclo de vida do projeto
   1. Fases do projeto (“produtos e as entregas”)
3. Tarefas (técnicas, acompanhamento)
4. Esforço (HH, H.mês, H.ano)
5. Recursos (Humanos e Materias)
6. Orçamento ≠ Preço
7. Cronograma
8. Riscos (2 riscos p/ categoria)
   1. Plano de projeto – Aprovado (stakeholders -> interessados)

...

Kick-off do projeto

Resumão com todos os stakeholder para dar início ao projeto

# Aula 9

## Plano de Projeto

WBS

Organograma

Cronograma

Orçamento

# Aula 10

Gerência de Pessoal (fonte: Sommerville)

Seleção de pessoal – de que forma organiza as pessoas para formar as equipes de projeto

Informações que ajudam a decidir as pessoas que vão compor:

Formação e currículos dos candidatos

Entrevista com candidatos

Testes

Recomendações de outras pessoas

Candidato generalista X especialista

Fatores de seleção

Experiência no domínio da aplicação

Experiência na plataforma

Experiência na linguagem de programação

Habilidade de resolver problemas – colocar testes

Formação educacional

Capacidade de comunicação

Capacidade de adaptação

Atitude – difícil avaliação

Personalidade – difícil avaliação

Motivação de pessoas

Necessidades pessoais

Necessidades de auto-estima

Necessidades sociais

Necessidades de segurança (desconfortável quanto a empresa ou projeto)

Necessidades fisiológicas

Tipos de profissionais

Orientado a tarefas

Auto-orientados

Orientado a interações

Gerenciamento de grupos

Composição (habilidade, experiência, personalidade)

Coesão (valores de grupos)

Comunicação (entre os membros)

Organização (papel do grupo no projeto)

Ambientes (locais adequados)

Fatores críticos no gerenciamento de pessoal

Consistência – tratamento uniforme

Respeito – independente das suas habilidades

Inclusão – ouvir as pessoas

Honestidade – ser honesto com todos

Gerenciamento da qualidade (fonte: Sommerville)

Atividades

Garantia da qualidade (procedimentos e padrões) – pode definir uma vez e depois só melhorar

Planejamento da qualidade – depende de projeto a projeto (selecionar procedimentos e padrões do projeto)

Controle da qualidade – depende de projeto a projeto (assegurar que procedimentos e padrões estão sendo seguidos)

Verificação (independente)

Processo de desenvolvimento de software

Processo de gerenciamento de qualidade

Estabelecimento de procedimentos e padrões

Plano de qualidade

Vários relatórios de qualidade ao longo do tempo

## Qualidade do processo X produto



Gerenciamento da qualidade (fonte: Sommeville)

Padrões de produto

Estrutura de documento de requisitos (Ex. IEEE 830)

Estilos de programação (Ex. Java)

Formato de plano de projeto

ISO 9126 (software)

Padrão de processo

Processo de desenvolvimento (Cascata, RUP, XP)

Processo de revisão de projeto

...

Controle de qualidade

Auditorias / revisões de projeto (produto e processo)

Ferramentas de software (comparação de padrões X produto)

Medições de qualidade

Número de não conformidades (processo, produto)

Número de defeitos em documentos

Número de desvios de projeto

Etc

Gerenciamento de configuração (fonte: Sommerville)

Itens de configuração (IC) ≠ artefato

Especificação de sistema (documento, modelos, diagramas, etc)

Plano de projeto (documentos, cronograma, planos de qualidade, configuração,...)

Especificação de requisitos de software (documentos, diagrama, etc...)

Plano de testes (especificação casos de testes, estratégia de testes)

Baseline

Marco de referência de desenvolvimento de software caracterizado pela entrega e/ ou aprovação dos IC.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Processo de desenvolvimento de software |  | IC1 | IC2 | ... | ICn |  | IC11 | ... | IC1n |  |  |
| Processo de gerenciamento de configuração | Plano de configuração | Versão1 | Versão2 | ... | Versãon | **Baseline1** | Versão11 | ... | Versão1n | Baseline2 | Release |

Release -> produto verificado e aprovado por usuários

## Plano de configuração

Define os IC (identificação, armazenamento, restrições de acesso)

Define responsável

Define regras de controle (mudanças)

Especificação ferramentas (Ex. source safe, clear case, etc)

Define registros (logs)

## Visão Integrada



# Aula 11

## Plano de Projeto

## 11. Garantia de qualidade

Planejamento: Todos os artefatos indicados no WBS passarão pelo controle de qualidade, incluindo-se o próprio Plano de Projeto.

Controle: Através de “auditoria”, conforme indicados no cronograma.

Responsável: “indicar um responsável” (obs: não pode ser o gerente)

## 12. Gerenciamento de configuração

Todos os artefatos indicados no WBS passarão por controle de configuração

Indicar ferramenta de controle (se for aplicável).

Alternativamente, onde será armazenado? Quem pode acessar?

Responsável: indicar um responsável

Incluir “baselines” no cronograma (pode ser indicado nos milestones) – fim de fases. Fecha um conjunto de artefatos.

## 13. Critério de aceitação do projeto

Como o projeto será aceito?

Basear-se nos requisitos, restrições e artefatos do projeto.

## 14. Referências

[Autor][Nome][Versão][Ano]

## Anexos

Se aplicável.

# Aula 12

## Comentários gerais sobre o trabalho

Aprovação do plano 🡪 v. lista de interessados

Introdução 🡪 público alvo v. lista de interessados (usuário ?, não é público alvo)

Interessados no projeto

WBS x Cronograma

Restrições (de projeto) imposição de ambiente

Organograma

Analista de qualidade – só audita

Cronograma

Detalhar as atividades (no mínimo) aquelas que geram outros artefatos

Riscos – Inserir coluna “categoria de riscos”. Descrever as ações (mitigar, transferir, aceitar ... o quê?)

Garantia de qualidade “auditorias” 🡪 inserir no cronograma, “artefatos” com fases

Gerenciamento de configuração 🡪 “baselines”, inserir no cronograma

Critério de aceitação do projeto. Quem aceita? V. lista de interessados.

O que se espera?

## Seqüência

No final da disciplina se espera quatro documentos:

Plano de projeto, versão preliminar e final

Auditoria de qualidade preliminar e a final

Erro – humano (nós cometemos erros)

gera

Defeito – Artefato

Latentes – são feitos testes para transformar defeitos latentes em defeitos detectados

Detectados

gera

Falha – mau funcionamento que o sistema exterioriza

## Defeitos

Requisitos inconsistentes

Requisito negócio inconsistente com requisito funcional

Formatos de datas

dd/mm/aaaa => mm/dd/aaaa

## Verificação e validação (V&V)

Processo de V&V – deve ser aplicado em cada fase do ciclo de vida do projeto.

Objetivo do V&V – estabelecer confiança de que SW é adequado ao seu propósito. Não significa que o software seja completamente livre de defeitos.

V&V abrange muitas atividades de SQA (Software Quality Assurance)

Inspeções ou revisões – inspeção é feito por um grupo interno à organização.

Auditorias – é feito por um grupo externo à organização.

Simulações e testes em geral

Verificação: Estamos construindo certo produto?

Validação: Estamos construindo o produto certo?

## V Model

A cada mudança de pedaços pode-se utilizar V&V. Deve-se verificar se o padrão é bem definido.

## Técnicas estáticas de V&V

Inspeções, revisões, walkthrough

## Técnicas dinâmicas de V&V

Testes estruturais (caixa branca) e funcionais(caixa preta)

Sistema 🡪 caixa preta

Validação 🡪 caixa preta

Integração 🡪 caixa preta

Unitário 🡪 caixa branca

## Teste

Execução de programa para achar defeitos

## Depuração

Diagnóstico da natureza do defeito 🡪 correção 🡪 solução

## Regra do teste

Um bom teste é aquele que tem uma elevada probabilidade de encontrar um defeito ainda não descoberto.

Um teste bem sucedido é aquele que revela um defeito ainda não descoberto.

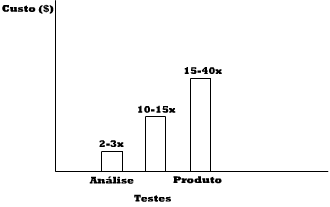
# Aula 13

## Verificação e Validação (V&V)

Técnicas Estáticas – Inspeções

Técnicas Dinâmicas – Testes

Objetivo: Garantir e controlar a qualidade do produto e processo de desenvolvimento



## Técnicas Estáticas de V&V

Revisões, walkthrough

Avaliação informal

Aplicadas em pontos diferentes do ciclo de vida de projeto

Procuram defeitos

Inspeções

Avaliação formal

Aplicadas em códigos de programas

Procuram defeitos

Hoje em dia, os itens acima são chamados de Inspeção (Formal) (de artefatos e atividades em geral)

## Tamanho de defeitos

Pequeno – podem ser corrigidos rapidamente e não causam mau funcionamento do software. Exemplo: digitação; omissão de texto

Grande – defeitos em especificações que causam mau funcionamento. Exemplo: ausência de função, problema de interface entre funções

Muito sério – podem exigir um total / parcial reprojeto do SW.

## Tipos de defeitos

Incorreção

Ausência

Extra (nova funcionalidade)

Onde o defeito pode ocorrer?

Programas (código)

Arquitetura do SW

Requisitos

Especificações

Documentos em geral

## Natureza do defeito

Pode ser identificado

Pode ser descrito

Pode se contado

## Processo de inspeção (autor: Fagan (IBM))

Planejamento

Visão geral

Preparação individual

Reunião de inspeção

Retrabalho

Acompanhamento

## Papéis envolvidos

Moderdor-chefe

Moderador (1) – José da Silva (Gerente de Projeto)

Relator (1)

Leitor (1)

Inspetores (3 – 4)

Autores (1 - +)

## Planejamento

Seleção de pessoal (inspetores)

Perdi pedaço da matéria

## Visão geral

Moderador, autor, inspetores

O autor faz uma apresentação do material a ser inspecionado

## Prepararação individual (inspetores)

Cada inspetor tem um prazo para examinar o material

## Reunião de inspeção (moderador 🡪 autores)

Duração até 2 horas

Objetividade 🡪 procurar, definir (defeitos)

Não sugerir soluções

## Retrabalho (autor)

Modificação do material inspecionado

## Acompanhamento (moderador)

Decisão de uma nova / ou não inspeção

## Registros de Inspeção

Lista de defeitos (checklist)

Relatório de inspeção

O que foi inspecionado

Quem participou

Quais foram as descobertas e decisões

## Decisões de inspeção

Participantes aceitam o produto inspecionado sem modificações adicionas

Participantes rejeitam o produto inspecionado devido a problemas graves

Participantes aceitam provisoriamente (defeitos menores foram encontrados)

## Técnicas de inspeção

Ad-hoc – sem roteiro sistemático

Checklist – lista de questões previamente estruturadas para dirigir a busca de defeitos. Para cada artefato tem um checklist

Cenário – baseado em perspectivas dos envolvidos no produto (autores)

## Verificação e Validação (V&V) – Resumo

**Importante:**

Inspeções e testes são complementares

Ambas devem ser usadas durante o projeto

Inspeções e testes verificam a conformidade, com especificações e não com requisitos reais do cliente.

Inspeções não verificam requisitos não funcionais.

## Pré-condições para uma boa inspeção

Material revisado deve estar disponível

Os inspetores devem estar familiarizados com o processo de inspeção

Checklist deve estar disponível

Gerência deve aceitar um aumento de custos no início do projeto

A gerência não deve usar o processo de inspeção para avaliar pessoal (quem cometeu os erros?)

# Aula 14

## Técnicas de inspeção

Ad-hoc – não sistemático, baseado somente nos conhecimentos e experiência dos inspetores.

Checklist – lista de questões que servem de roteiro para os inspetores.

Cenários – Focado em alguma perspectiva de temas ou assuntos específicos do artefato inspecionado.

|  |  |
| --- | --- |
| ERS | Checklist |
| Descreve funções | Todas as funções foram adequadamente descritas? |
| Descrição de interfaces | Todas as interfaces foram adequadamente definidas? |
| Descrição do modelo de dados | Todas as entidades de dados foram relacionadas? |
| Implementação de protótipo | O cliente/usuário validou o protótipo? |
| Desempenho | O desempenho requerido pode ser atingido levando-se em conta todas as funções e artefatos? |
| Critérios de validação | Os critérios de validação permitem que o sistema seja concluído? |

## Questões gerais

O documento segue a última versão do padrão?

O documento contém todos os itens desse padrão?

O documento inspecionado é a sua última versão?

## Roteiro de Inspeção

Atribuições do moderador (tipicamente gerente de projeto)

Reparação

Agendamento (com inspetores, autores)

Realização (aproximadamente 2 horas por inspeção)

Encerramento

## Atribuição dos inspetores

Esteja preparado

Seja cooperativo

Observe a linguagem

Levante questões, não os resolva

Evite discussões de estilo

Não avalie os autores

# Diretrizes para Inspeção

Inspecione o produto (não o autor)

Fixe e mantenha uma agenda – tentar ser produtivo

Limite o debate e a reputação

Enuncie os problemas, sem resolvê-los

Anote os problemas encontrados

Limite o número de participantes (ideal aproximadamente 5)

Aloque recursos adequados para inspeção

# Aula 15

|  |
| --- |
| Apresentação dos trabalhos (aproximadamente 20 minutos)   * O projeto? * Plano do projeto   + WBS   + Organograma   + Orçamento   + Riscos   + Cronograma * Status atual do projeto   Argüição (aproximadamente 10-15 minutos)  Entre os presentes principalmente os grupos que não apresentarão trabalhos |

## V&V (técnicas estáticas)

Desenvolvimento de sistemas

Gerenciamento

Análise

Projeto

...

Gerenciamento de qualidade

Garantia de qualidade

Planejamento

Controle

Técnicas

Inspeção

Ad hoc

Checklist

Cenários

Auditoria

|  |  |  |
| --- | --- | --- |
| Plano do Projeto | Controle de qualidade (Checklist) | |
|  |  | Conformidade S/N |
|  | O documento segue a última versão do padrão? |  |
|  | O documento contém histórico de revisões atualizado? |  |
|  | O documento contém todos os itens do padrão |  |
| Introdução | O público alvo foi adequadamente identificado? |  |
| Todas as convenções, termos e abreviaturas importantes foram discutidos? |  |
| A visão geral do documento está clara? |  |
| Interessados no projeto | Todos os interessados no projeto e seu envolvimento estão identificados e descritos? |  |
| Objetivo do Projeto | Os objetivos principais do projeto estão claramente descritos e resumidos? |  |
| Escopo | O WBS apresenta todos os artefatos do projeto? |  |
| O WBS está alinhado com o cronograma? |  |

# Aula 16

## Normas e modelos de qualidade

Norma 🡪 ISO, IEEE, ...

Modelos 🡪 CMMI (SEI/CMV), MPSBR (SofTex)

ISO 9126 🡪 Produto

ISSO 12207 🡪 Organização

## ISO/ IEC 9126

Aplicável na especificação de produtos de software e na avaliação ao longo do ciclo de vida.

#### Características do ISO/IEC 9126

Funcionalidade

Confiabilidade

Usabilidade

Eficiência

Manutenabilidade

Portabilidade

#### Funcionalidade

Evidencia um conjunto de funções e suas propriedades

Associa necessidades:

Explicitas: requisitos do usuário

Implícitas: custo e prazo

#### Confiabilidade

Capacidade de o software manter o seu nível de desempenho sob condições estabelecidas durante um período de tempo estabelecido.

Depende dos defeitos embutidos no software.

#### Usabilidade

Evidencia o esforço necessário para o uso do software.

Esforço necessário:

Explicito / direto: operadores

Implícito / indireto: que dependem do software (caixa eletrônico)

#### Eficiência

Evidencia a relação entre desempenho x uso dos recursos (hardware, serviços, operação, apoio)

#### Manutenabilidade

Evidencia o esforço necessário para fazer modificações no software

Manutenções podem ser:

Corretivas – bugs

Melhoria – evolução

Adaptativas – ambiente

Preventivas – antecipar futuras manutenções

#### Portabilidade

Evidencia a capacidade de o software ser transferido de um ambiente para outro 🡪 hardware e software

ISO/IEC 9126 🡪 Modelo de avaliação



## Valor medido e nível de pontuação

|  |  |  |
| --- | --- | --- |
|  | Excelente | Satisfatório |
| Valor medido 🡪 | Bom |
|  | Regular |
|  | Inaceitável | Insatisfatório |

## ISO/ IEC 12207

ISO/IEC 12207 : 1995

Amd1: (adendo 1) – 2002

Amd2: (adendo 2) – 2004

ISO/IEC 12207 (Amd 2)

Processos fundamentais

Processos de apoio

Processos organizacionais

Processos de adaptação

#### Processos fundamentais

Aquisição

Fornecimento

Desenvolvimento

Operação

Manutenção

#### Processos de apoio

Documentação

Gerência de configuração

Garantia de qualidade

Verificação

Validação

Revisão conjunta

Auditoria

Resolução de problema

#### Processos organizacionais

Gerência

Infra-estrutura

Recursos humanos (qual o perfil e treinamento adequados?)

Gerência de ativos (processo, templates, etc)

Gerência de programa de reuso (detectar oportunidades de reuso dentro da organização)

Engenharia de domínio

Exemplo:

LTS 🡪 Ambiente de maturidade

|  |  |  |
| --- | --- | --- |
| Modelo de referência | | |
| Ensino, pesquisa, extensão 🡺  ISO/IEC 12207 🡺 | Gerência  Desenvolvimento  Garantia de qualidade | Nível conceitual |
| ISO/IEC 12207 🡺 | Desenvolvimento  Engenharia de sistema  Engenharia de softwares  Diagramas entre os dois. | Nível lógico  (Ambiente de maturidade) |
| Necessidade de negócio 🡺 | Eng SW 1, Lab Eng SW 2  Fábrica de SW | Nível físico  (Ambiente físico) |

# Aula 17

## Modelos de qualidade

CMM

Fonte: [WWW.sei.cmu.edu](http://WWW.sei.cmu.edu)

## Maturidade de processos

Definido

Gerenciado – o processo definido não está largado

Medido

C ontrolado

Efetivo – as pessoas atingidas de fato utilizam

Institucionalizado – toda organização de fato utilizam os processos

## Uma organização no Caos

Não segue um processo de software

Gerentes apagam incêndio – constantemente o gerente é chamado para conduzir o projeto

Cronogramas e metas são excedidos

Não avaliam a qualidade

Não avaliam o processo

As atividades de revisão e testes são abreviados ou cortados quando o prazo não pode ser atendido.

## CMM – Capacity Maturity Model

SEI – Software Engineering Institute – CMU – USA

DoD - Department of Defense - cliente 1986

CMM

1991 v. 1.0

1993 v. 1.1

CMMI

2002 v. 1.1

2006 v. 1.2

## CMM

Não é uma ferramenta

Não é metodologia

Não são processos prontos

É um guia 🡪 ganhar o controle dos processos, evoluir em direção a uma excelência em cultura de engenharia de desenvolvimento e manutenção de software.

## CMM no Brasil

NEC – Nível 2 (1998), N3 (2001)

Credicard – N2 (1999) 🡪 virou Orbital

Motorola – N3 (2001)

Xerox – N3 (2000)

EDS – N4 (2004)

BankBoston – N2 (2002)

## CMMI – CMM Integration

SW CMM

SE CMM - Systems Engineering CMM

People CMM –voltado para RH

PSP – Personal Software Process (indivíduos)

TSP – Team Software Process (equipes)

Aquisition

## CMMI Capacibility Maturity Model Integration

CMMI-DEV – para desenvolvimento de TI

v. 1.2 (2006)

CMMI-AQV – para aquisições de TI

v. 1.2 (2007)

CMMI-SRV – para serviços de TI

v. 1.2 (TBD)

#### CMMI-DEV

Cobre as atividades de desenvolvimento e manutenção de produtos e serviços de TI.

As práticas cobrem:

Gerenciamento de projeto

Gerenciamento de processo

Engenharia de sistemas

Engenharia de Software

Engenharia de Hardware

#### Representações do CMMI

|  |  |
| --- | --- |
| Por estágios | Prevê uma seqüência de melhorias  Permite a comparação entre organizações  Permite uma comparação com padrões da ISO |
| Níveis de Maturidade | |
| Continua | Permite escolher processos  Permite uma equiparação com níveis de maturidade  Aderente à ISO |
|  | |

## Por estágios

## Contínua

## 

Nível de capacidade não está ligado com os processos, mas possui ligações com Metas Genéricas e Práticas Genéricas.

|  |  |  |
| --- | --- | --- |
| **Níveis** | **Maturidade** | **Capacidade** |
| 0 | Não disponível | Incompleto |
| 1 | Inicial | Executado |
| 2 | Gerenciado | Gerenciado |
| 3 | Definido | Definido |
| 4 | Quantitativamente gerenciado | Quantitativamente gerenciado |
| 5 | Otimização | Otimização |

## Perfil das organizações com CMMI (04/2002 até 06/2007)

1427 empresas

2410 organizações

67% fora EUA

10338 projetos de TI (inclui hardware e software)

## Desempenho do CMMI

25 organizações

18 são N5

|  |  |
| --- | --- |
|  | **Médio** |
| Custo | 20% |
| Cronograma | 37% |
| Produtividade | 62% |
| Qualidade | 50% |
| Satisfação | 14% |
| ROI (retorno sobre investimento) | 47:1 |

## CMMI no Brasil

IBM – N5 (2005)

ADS – N5 (2005)

Accenture Delivery – N3 (2005)

Microsiga – N2 (2005) – N3 (2007)

GM – N2 (2005)

Gedas – N2 (2005)

SERASA – N2 (2006)

Inicial – Processo imprevisível pobremente controlado e reativo.

Gerenciado – Processo baseado em projetos e freqüentemente reativo.

Definido – Processo padronizado e Pró-ativo.

Quantitativamente gerenciado – Processo medido e controlado

Otimização – Foco em melhoria

#### Características dos Níveis

Não podem ser pulados

Um nível provê uma base necessária para implementação dos processos

Os níveis mais altos podem ser executados, mas corre-se o risco de não ser consistente quando em crises

# Aula 18

## CMMI – DEV (v. 1.2) – Áreas de Processo

|  |  |  |  |
| --- | --- | --- | --- |
| **Gerenciamento de Processo** | **Gerenciamento de Projeto** | **Engenharia** | **Apoio** |
| - Foco de Processo Organizacional [OPF]  - Definição de Processo Organizacional [OPD]  - Treinamento Organizacional [OT]  - Desempenho de Processo Organizacional [OPP]  - Inovação Organizacional e Implantação [IO] | - Planejamento de Projeto [PP]  - Manutenção e Controle de Projeto [PMC]  - Gerenciamento de Acordos com Fornecedores [SAM]  - Gerenciamento de Projeto Integrado [IPM]  - Gerenciamento de Riscos [RSKM]  - Gerenciamento de Projeto Quantitativo [QPM] | - Gerenciamento de Requisitos [REQM]  - Desenvolvimento de Requisitos [RD]  - Solução Técnica [TS]  - Integração do Produto [PI]  - Verificação [VER]  - Validação [VAL] | - Gerenciamento de Configuração [CM]  - Garantia de Qualidade de Produção e Processos [PPQA]  - Medição e análise [MA]  - Análise causal e resolução [CAR]  - Análise de Decisão e Resolução [DAR] |

|  |  |  |
| --- | --- | --- |
| Maturidade | Áreas de Processo | Capacidade 🡪  1 2 3 4 5 |
| Nível 5 | CAR, OID | ---------------------------------------------------> |
| Nível 4 | OPP, QPM | ---------------------------------------------------> |
| Nível 3 | OPF, OPD, OT, IPM, RSKM, RD, TS, PI, VER, VAL, DAR | °°°°°°°°°°°°°°°°°°°°°°°°°°°°--------------------> |
| Nível 2 | REQM, PP, PMC, SAM, CM, MA, PPQA | °°°°°°°°°°°°°°°°°°°°°°°°°°°°--------------------> |
| Nível 1 |  |  |

Os processos são acumulativos nos níveis.

Essa regra de equivalência só vale até o nível 3, quando todos estiverem no nível 5 não será equivalente ao nível 5 dos níveis de maturidade.

## Nível 2 de Maturidade

#### Gerenciamento de Requisitos – REQM

* Metas específicas
* Metas genéricas

REQM

Engenharia

Requisitos correntes

Requisitos do produto

## REQM

#### Meta específica

Gerenciar requisitos

#### Práticas específicas

Obter um entendimento dos requisitos

Obter compromisso dos requisitos

Gerenciar mudanças dos requisitos

Manter rastreabilidade bidirecional de requisitos

Identificar inconsistências entre produtos de trabalho e requisitos

Garantindo essas práticas específicas se garante a meta específica.

Figura K1

## PP

#### Metas específicas – Práticas específicas

Estabelecer estimativas – escopo, tamanho, ciclo de vida, esforço e custo

Desenvolvimento do Plano de Projeto – orçamento, cronograma, riscos recursos, habilidades, envolvimento dos interessados, plano de projeto

Obter compromisso para o plano – Riscos de plano, reconciliação trabalho x recursos, comprometimento

## PMC

#### Metas específicas

Monitorar projeto contra Plano

Gerenciar ações corretivas até resolução

## SAM

Servidor, rede, SO, desenvolvimento

#### Metas específicas – Práticas específicas

Estabelecer acordos com fornecedor (estabelecimento de regras) – tipo de aquisição, seleção de fornecedores, acordos (contratos)

Satisfazer acordos com fornecedor (execução) – execução, monitoração, avaliação dos produtos, aceitação, transferência dos produtos.

## MA

#### Metas específicas – Práticas específicas

Alinhar atividades de medição e análise – objetivos da medição, medidas, coleta, análise.

Fornecer resultados de medição – dados de medição, análise, armazenamento Resultados

## CM

#### Metas específicas – Práticas específicas

Estabelecer baselines – itens de configuração, plano de configuração, baselines

Acompanhar e controlar mudanças – solicitações de mudança, controle de mudanças

Estabelecer integridade – registros, auditorias (externa ao projeto)

## PPQA

#### Metas específicas – Práticas específicas

Avaliar objetivamente processos e produtos – Avaliação de processo e produto de uma forma objetiva

Fornecer visão objetiva – Comunicação de não conformidades, registros de qualidade.

## Nível 2 - CMMI

REQM

PP

PMC

SAM

MA

CM

PPQA

# Aula 19

## CMMI

Processo efetivo 🡪 Práticas genéricas 🡪 institucionalização 🡪 Maturidade

|  |  |
| --- | --- |
| Meta Genérica | Nível 2 – Processo gerenciado |
| Práticas genéricas |  |

## Práticas genéricas 🡪 se referem a processo

1. Estabelecer uma política organizacional

Organização

1. Planejar o processo
2. Fornecer recursos
   1. Ambiente
   2. Pessoas capacitadas
   3. ...
3. Designar responsabilidades
4. Treinar pessoas
5. Gerenciar configuração
   1. Fluxograma
   2. Templates, modelos
6. Identificar e envolver stakeholders relevantes
7. Monitorar e controlar o processo
8. Avaliar objetivamente a aderência 🡪 PPQA
9. Revisar status com a alta gerência – reuniões periódicas da gerência de projetos com a alta gerência

Práticas específicas e genéricas 🡪 Somatórios de todas as evidências (documentos e registros)

## MPS.BR

Melhoria do processo de software Brasileiro.

#### Motivações:

Como aumentar significativamente o número de empresas níveis 4 e 5 no Brasil?

Como melhorar radicalmente os processos de software no Brasil, com foco nas empresas (micros, pequenas e médias) a um custo acessível?

## Projeto MPS.BR

Projeto de qualificação de profissionais Nível 4 e 5 do CMMI

Projeto MPS.BR – Nível 2 e 3 CMMI

Custo subsidiado.

Período 2004 – 200...

Modelo de referência

Modelo de negócio

(Softex 🡪 MCT)

## Modelo de Negócio



## Modelo de Referência



## MPS-BR V.1.1

(O número é em relação ao CMMI)

A Inovação e Análise e Resolução 5

B Desempenho e Gerência Quantitativa 4

C Análise Decisão e Gerência Riscos

D Desenvolv. Requisitos, solução técnica, integração, verificação e validação. 3

E Treinamento, avaliação de processo, definição e adaptação e gerência de projetos

F Medição Gerenciamento Configuração, Aquisição e GQ 2

G Gerência de requisitos e gerência de projeto

## Experiência de Implementação do MPS-BR (Nível G)

Modelo de negócio: Cooperado

Instituições implementadoras:

ITS (Instituto de Tecnologia de Software)

Fundação Vanzolini

Período: 02/07 a 04/08

Empresas (7):

4 São Paulo (3)

1 ABC

1 Alphaville

1 Tupã (1)

Menor 10 pessoas (Web).

Maior 100 pessoas, área médica, consórcio de veículos.

## Projeto ITS / Vanzolini – MPS-BR (Nível G)

|  |  |  |
| --- | --- | --- |
| 0 |  |  |
| 50% | Processos Definidos (gerência requisitos, gerência projetos)  1 projeto de SW iniciado | SOFTEX/BID $$$ |
| 100% | 50% +  1 projeto concluído  1 projeto iniciado | SOFTEX/BID $$$ |
| Avaliação Final |  | $ empresa |

Quem não conseguiu teve que devolver o dinheiro.

Implementadora

(treinar e apoiar)

Empresa

Avaliadora

(Nível G)

MPS Nível G

Tem duração de 3 anos.