

# Qualidade de Software

Prof. Pedro Henrique Dias Valle

# Qualidade de Software

- Qualidade é um termo que pode ter diferentes interpretações.
- Existem muitas definições de qualidade de software propostas na literatura, sob diferentes pontos de vistas.

# Qualidade de Software

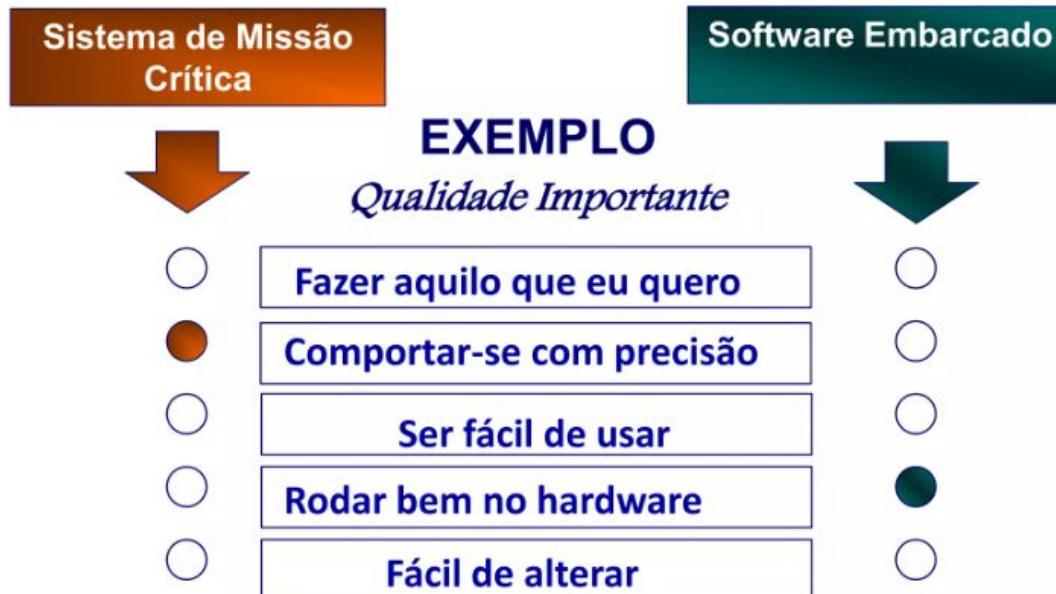
- Definição:
  - “Um produto de software apresenta qualidade dependendo do grau de satisfação das necessidades dos clientes sob todos os aspectos do produto” [Sanders, 1994].

# Qualidade de Software

- Definição:

“Qualidade é a totalidade de características e critérios de um produto ou serviço que exercem suas habilidades para satisfazer às necessidades declaradas ou envolvidas” [ISO 9126, 1994].

# Qualidade depende do tipo de aplicação



# Qualidade depende do tipo de aplicação

Software para Folha de Pagamento      Software Interativo com o usuário



## EXEMPLO

*Qualidade Importante*

- 
- 
- 
- 
- 

**Fazer aquilo que eu quero**

**Se comportar com precisão**

**Ser fácil de usar**

**Rodar bem no hardware**

**Fácil de alterar**



Software Interativo com o usuário



- 
- 
- 
- 
-

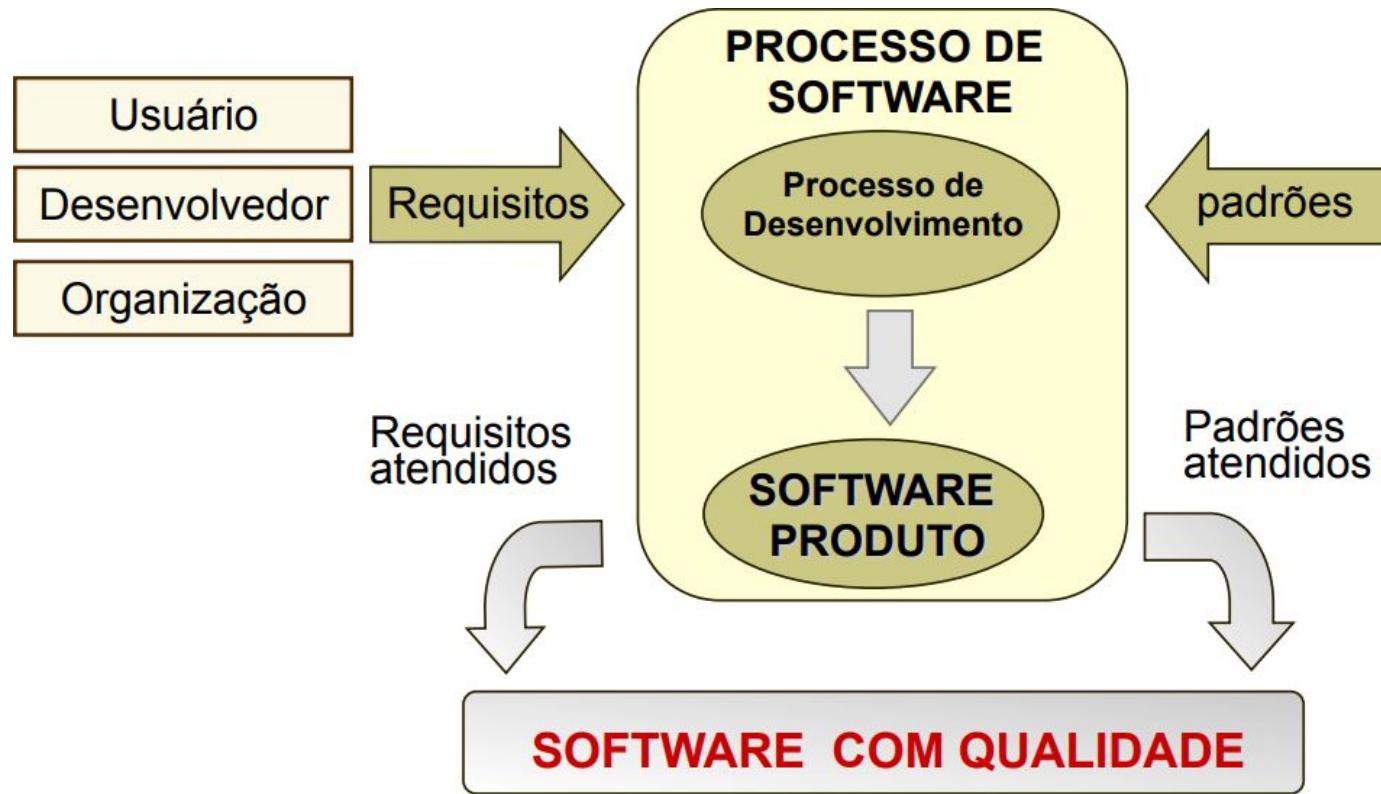
# Aspectos Importantes para Qualidade de Software

- Requisitos de software:
  - Métrica fundamental para medir a qualidade do software
    - A falta de conformidade aos requisitos significa falta de qualidade
- Padrões:
  - Definem um conjunto de critérios que orientam o desenvolvimento de software
    - Se os critérios não forem seguidos, o resultado seguramente será a falta de qualidade

# Aspectos Importantes para Qualidade de Software

- Requisitos implícitos:
  - Conjunto de requisitos que geralmente não são mencionados na especificação. Ex: “boa manutenibilidade”
    - Se o software atende aos requisitos explícitos, mas falha nos requisitos implícitos, a qualidade é suspeita.
- Qualidade de software (na visão gerencial):
  - O software é considerado de qualidade desde que possa ser desenvolvido dentro do prazo e do orçamento especificados.

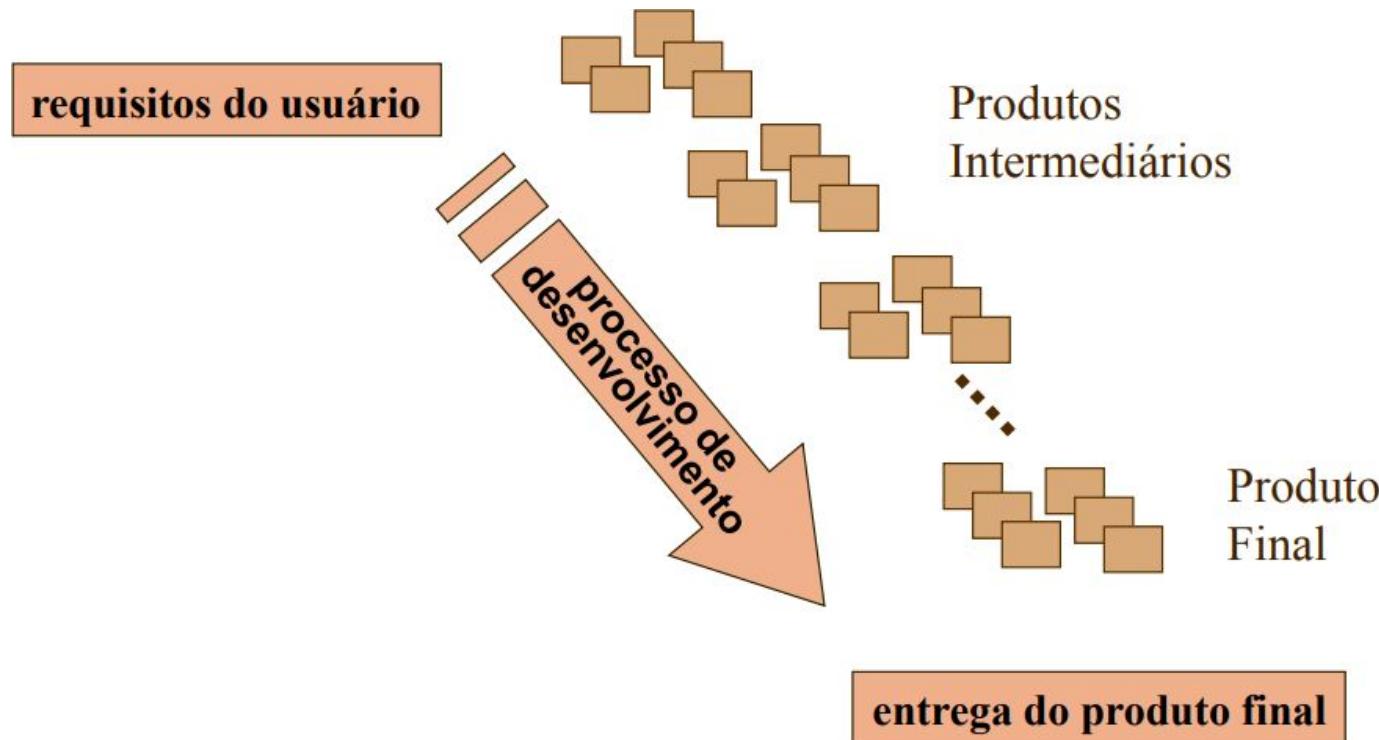
# Qualidade de Software



# Incorporação de Qualidade

- Raramente qualidade pode ser incorporada ao produto final, após o processo de desenvolvimento ter terminado.
- Dos requisitos do usuário à entrega do produto final, existe um processo de desenvolvimento que é complexo e frequentemente envolve uma série de estágios que podem comprometer a qualidade do produto final.

# Incorporação de Qualidade: Qualidade dos Produtos Intermediários



# Incorporação de Qualidade: Qualidade do Produto Final

- Cada produto intermediário tem certos atributos de qualidade que afetam a qualidade do produto intermediário da próxima fase e assim, afetam a qualidade do produto final.

# **Qualidade: Objetivo do Processo de Desenvolvimento**

- A qualidade do produto de software é um objetivo do processo de desenvolvimento.
- Assim, ao desenvolver-se um produto, deve-se ter previamente estabelecidas, como perspectiva, as características de qualidade que se desejam alcançar.

# Atividades de Padronização

- Qualidade do Produto
  - Genéricos
    - ISO/IEC 25010
  - Outros modelos de qualidade
    - Para domínios de aplicação específicos
- Qualidade do Processo
  - CMMI
  - ISO 12207
  - Outros específicos

# ISO/IEC 25010

- Representa a padronização mundial para a qualidade de produto de software
- Baseada em "três" níveis: Características, Sub-características e Métricas
- Cada característica é refinada em um conjunto de subcaracterísticas e cada subcaracterística é avaliada por um conjunto de métricas da norma ISO/IEC 25023 (2016)

# ISO/IEC 25010

Functional Suitability	Reliability	Performance efficiency	Usability	Maintainability	Security	Compatibility	Portability
Functional completeness	Maturity	Time behaviour	Appropriateness Recognisability	Modularity	Confidentiality	Co-existence	Adaptability
Functional correctness	Availability	Resource utilization	Learnability	Reusability	Integrity	Interoperability	Installability
Functional appropriateness	Fault tolerance	Capacity	Operability	Analysability	Non-repudiation		Replaceability
	Recoverability		User error protection	Modifiability	Accountability		
			User interface aesthetics	Testability	Authenticity		
			Accessibility				

# ISO/IEC 25010

ISO/IEC 25010-2023  
AI system product quality

<https://standards.itech.ai/catalog/standard/ist/60d098d2-de78-4aac-881a-c54799d8bcc8/iso-iec-25010-2023>

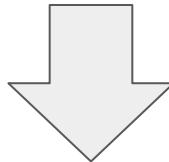
Functional suitability	Performance efficiency	Compatibility	Usability	Reliability	Security	Maintainability	Portability
Functional completeness Functional correctness <sup>m</sup> Functional appropriateness Functional adaptability <sup>a</sup>	Time behaviour Resource utilisation Capacity	Co-existence Interoperability	Appropriateness recognisability Learnability Operability User error protection User interface aesthetics Accessibility User controllability <sup>a</sup> Transparency <sup>a</sup>	Maturity Availability Fault tolerance Recoverability Robustness <sup>a</sup>	Confidentiality Integrity Non-repudiation Accountability Authenticity Intervenability <sup>a</sup>	Modularity Reusability Analysability Modifiability Testability	Installability Replaceability Adaptability

<sup>a</sup> New sub-characteristics.

<sup>m</sup> Modified sub-characteristics.

# Qualidade de Processo de Software

- Uma das maiores dificuldades encontradas pelas empresas de software é o gerenciamento de seus processos de software.



Modelos de Processo de Software

# Qualidade de Processo de Software

- Um modelo de processo procura descrever formalmente e de maneira organizada todas as atividades que devem ser seguidas para a obtenção segura de um produto de software.
- É importante escolher um modelo apropriado às metas da organização e saber o grau em que esse modelo será implementado.
- Modelos CMMI e ISO/IEC 12207 são relacionados a processo de software.

# Qualidade de Processo de Software

## Processo imaturo:

- Ad hoc: processo improvisado por profissionais e gerências.
- Não é rigorosamente seguido e o cumprimento não é controlado.
- Altamente dependente dos profissionais atuais.
- Baixa visão do progresso e da qualidade.

# Qualidade de Processo de Software

## Processo imaturo (cont.):

- A qualidade do produto pode ficar comprometida para que prazos sejam cumpridos.
- Arriscado do ponto de vista do uso de nova tecnologia.
- Custos de manutenção excessivos.
- Qualidade difícil de se prever.

# Qualidade de Processo de Software

## Processo maduro:

- Coerente com as linhas de ação, o trabalho é efetivamente concluído.
- Definido, documentado e melhorando constantemente:
  - Compreendido
  - Utilizado
  - Vivo e ativo

# Qualidade de Processo de Software

## Processo maduro (cont.):

- Com o apoio visível da alta administração e outras gerências.
- Bem controlado - fidelidade ao processo é objeto de auditoria e de controle.
- São utilizadas medições do produto e do processo.
- Uso disciplinado da tecnologia.

# Maturidade do Processo de Software

- A qualidade do processo de software pode ser analisada através do nível de maturidade do processo.
- A maturidade dos processos de software de uma organização influencia na sua capacidade de atingir metas de custo, qualidade e cronograma.

# Valor da Melhoria de Processo

- A melhoria do processo de software oferece um retorno no investimento que pode ser medido - quando é medido.
- Benefícios adicionais são intangíveis e não podem ser quantificados facilmente.
- O CMMI é um modelo útil para orientação no processo de melhoria.

# Capability Maturity Model (CMMI)

- Modelo de Maturidade de Processo de Software que auxilia as empresas a melhorar seus processos de software.
- 1986 - Software Engineering Institute (SEI) por solicitação do DoD
  - 1991 - 1a versão do CMM
  - 2002 - 1a versão do CMMI
  - 2010 - CMMI Versão
  - 2018 - CMMI 2.0
  - 2023 - CMMI 3.0
  - <http://cmmiinstitute.com/>

# Capability Maturity Model (CMMI)

- Primeiro ponto importante a se observar no modelo CMMI:
  - É um modelo que se refere ao processo pelo qual o produto é construído.
- CMMI provê modelos para três áreas de interesse:
  - Desenvolvimento (CMMI for Development)
  - Aquisição (CMMI for Acquisition)
  - Serviços (CMMI for Services)

# Capability Maturity Model (CMMI)

- Supõe-se que uma organização que possui um processo maduro tem maiores probabilidades de produzir bons produtos consistentemente do que uma outra cujo processo é imaturo e caótico.

# Capability Maturity Model (CMMI)

- CMMI é um modelo de referência para a qualidade de processo de produção de software.
- Através de um processo formal de avaliação, uma organização é classificada em um “nível de maturidade” que varia de um a cinco.

# Capability Maturity Model (CMMI)

- O nível de maturidade indica em que medida os processos daquela organização são maduros.
- Quanto maior o nível de maturidade, melhores e mais maduros são os processos.

# Níveis de Maturidade do CMMI



# Níveis de Maturidade do CMMI

- Níveis do CMM são organizados de modo a estabelecer as prioridades na condução de programas de melhoria do processo de software.
- Cada Nível é considerado como pré-requisito para o nível seguinte.
- Não é possível o “salto” de níveis.
- Cada nível apoia-se nas competências que a organização desenvolveu no nível imediatamente inferior.

# Como melhorar o processo de software?

- Realizando um grupo de atividades correlatas, denominadas ÁREAS-CHAVE DE PROCESSO - (Key process areas - KPA) que, quando efetuadas coletivamente, alcançam um conjunto de metas consideradas importantes na implementação da competência do processo.

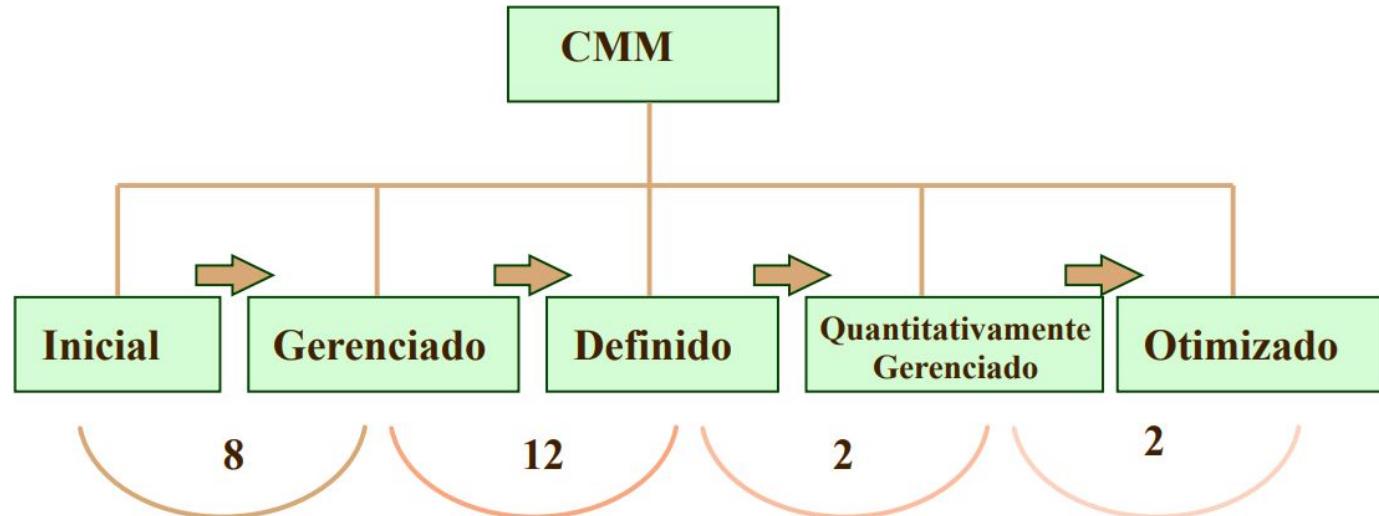
# Níveis de Maturidade do CMMI

- Para o nível de maturidade 2:
  - Gerenciamento de Requisitos (REQM)
  - Planejamento de Projeto (PP)
  - Monitoramento e Controle de Projeto (PMC)
  - Gerenciamento de Acordos com Fornecedores (SAM)
  - Medição e Análise (MA)
  - Garantia da Qualidade do Processo e do Produto (PPQA)
  - Gerenciamento de Configuração (CM)

# Níveis de Maturidade do CMMI

- Para o nível de maturidade 3:
  - Requer 12 áreas-chaves de processo
- Para o nível de maturidade 4:
  - Requer 2 áreas-chaves de processo
- Para o nível de maturidade 5:
  - Requer 2 áreas-chaves de processo

# Exemplo: Áreas-Chave de Processo

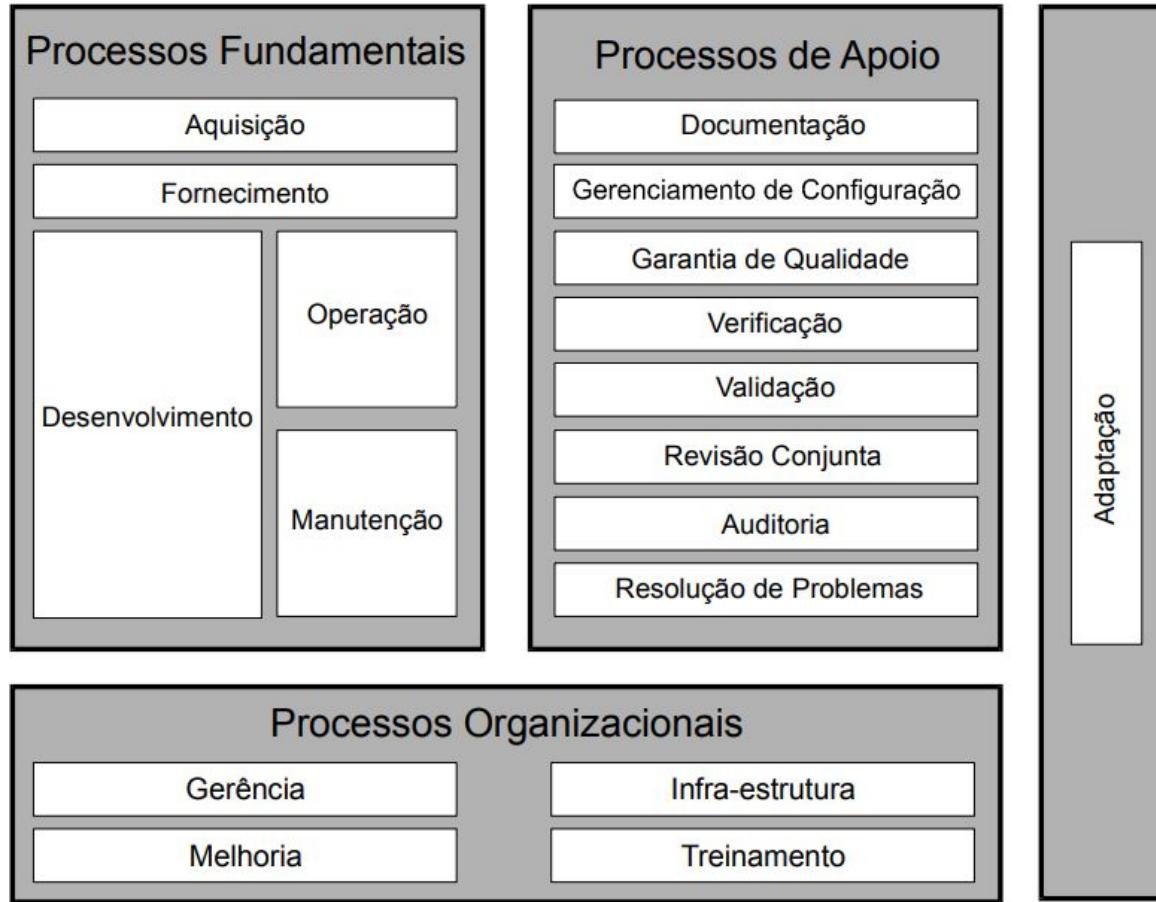


TOTAL  
24 Áreas-chave de Processo

# ISO/IEC 12207

- ISO/IEC 12207: 2008 – Software and Systems Engineering – Software Life Cycle Processes (revisto em 2013).
- Cobre todo o ciclo de vida de desenvolvimento do software desde a concepção das ideias iniciais sobre o software até ser retirado de execução.

# ISO/IEC 12207: Estrutura



# ISO/IEC 12207: Processos

- Processos Fundamentais: englobam as atividades referentes ao contrato entre o cliente e o fornecedor, ao desenvolvimento, à operação e à manutenção do software.
- Processos de Apoio: auxiliam outros processos como parte integrante dos mesmos, com propósitos distintos contribuindo para a qualidade do projeto de software.

# ISO/IEC 12207: Processos

- Processos Organizacionais: estabelecem e implementam uma estrutura que forneça uma base aos processos e ao pessoal envolvido no desenvolvimento de software.
- Processo de Adaptação: define as atividades necessárias para adaptar a norma a uma organização ou projeto específico.

# ISO/IEC 12207: Atividades dos Processos Fundamentais

Processo	Atividades
Aquisição	Iniciação Preparação do Pedido de Proposta Preparação e Atualização do Controle Monitoramento do Fornecedor Aceitação e Conclusão
Fornecimento	Iniciação Preparação da Resposta Contrato Planejamento Execução e Controle Revisão e Avaliação Entrega e Conclusão
Desenvolvimento	Implementação de um Processo Análise de Requisitos do Sistema Projeto Arquitetural do Sistema Análise dos Requisitos do Software Projeto Arquitetural do Software Projeto Detalhado do Software Codificação e Teste do Software

# ISO/IEC 12207: Atividades dos Processos Fundamentais

Processo	Atividades
Desenvolvimento	Integração do Software Teste de Qualificação do Software Integração do Sistema Teste e Qualificação do Sistema Instalação do Software Suporte à Aceitação do Software
Operação	Implementação do Processo Teste Operacional Operação do Sistema Suporte ao Usuário
Manutenção	Implementação do Processo Análise do Problema e da Modificação Implementação da Modificação Revisão/Aceitação da Manutenção Migração Descontinuação do Software

# ISO/IEC 12207: Atividades dos Processos Fundamentais

Processo	Atividades
Documentação	Implementação do Processo Projeto e Desenvolvimento Produção Manutenção
Gerência de Configuração	Implementação do Processo Identificação da Configuração Controle da Configuração Relato da Situação Avaliação da Configuração Gerência da Liberação e Entrega
Garantia da Qualidade	Implementação do Processo Garantia do Produto Garantia do Processo Sistema de Garantia da Qualidade
Verificação	Implementação do Processo Verificação
Validação	Implementação do Processo Validação
Revisão Conjunta	Implementação do Processo Revisões do Gerenciamento do Projeto Revisões Técnica
Auditória	Implementação do Processo Auditória
Resolução de Problemas	Implementação do Processo Resolução de Problema

# ISO/IEC 12207: Atividades dos Processos Fundamentais

Processo	Atividades
Gerência	Iniciação e definição do escopo Planejamento Execução e Controle Revisão e Avaliação Conclusão
Infra-estrutura	Implementação do Processo Estabelecimento da Infra-estrutura Manutenção da Infra-estrutura
Melhoria	Estabelecimento do Processo Avaliação do Processo Melhoria do Processo
Treinamento	Implementação do Processo Desenvolvimento de Material para Treinamento Implementação do Plano de Treinamento

# Listas de Empresas CMMI

- [http://www.isdbrasil.com.br/maturidade\\_mercado\\_nacional.php](http://www.isdbrasil.com.br/maturidade_mercado_nacional.php)

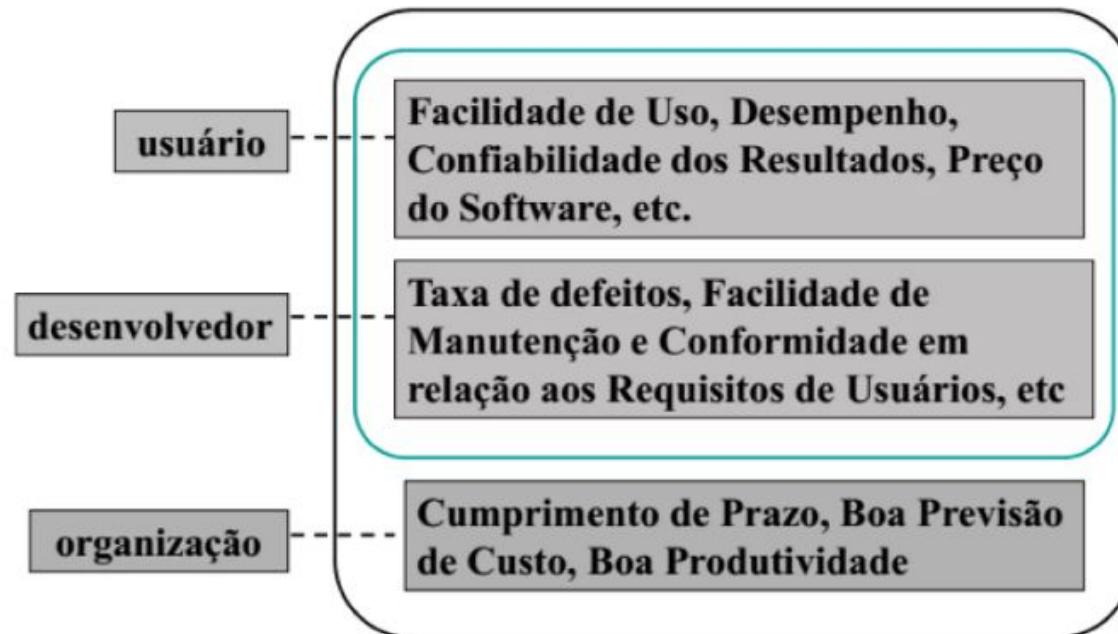
## Nível 4

Empresa	Estado	Modelo	Nível de Maturidade
C.P.M Braxis S/A (Capgemini)	BA	CMMI®	4

## Nível 5

Empresa	Estado	Modelo	Nível de Maturidade
IBM	RJ	CMMI®	5
C.P.M Braxis	BA	CMMI®	5
ACCENTURE	SP	CMMI®	5
Instituto Atlântico	CE	CMMI®	5

# Visão de Qualidade de Software



# Padrões de Qualidade

- ISO 9126 - Qualidade de produto de software
- ISO 12207 - Qualidade do processo de software
- ISO 27000 - Segurança da informação
- IEEE 829 - Documentação de testes
- IEEE 1028 - Revisão de software
- IEEE 1044 - Classificação de incidentes
- CMM (Capability Maturity Model)
- SPICE (Soft. Process Improvement & Capability dErtermination)

# Qualidade de Software

Prof. Pedro Henrique Dias Valle