

## Aula 9

Professor:

Geraldo Xexéo, D.Sc.  
DCC/IM/UFRJ  
PESC/COPPE/UFRJ

Conteúdo:

**Requisitos**

# O Problema da Pedra

# O Problema da Pedra

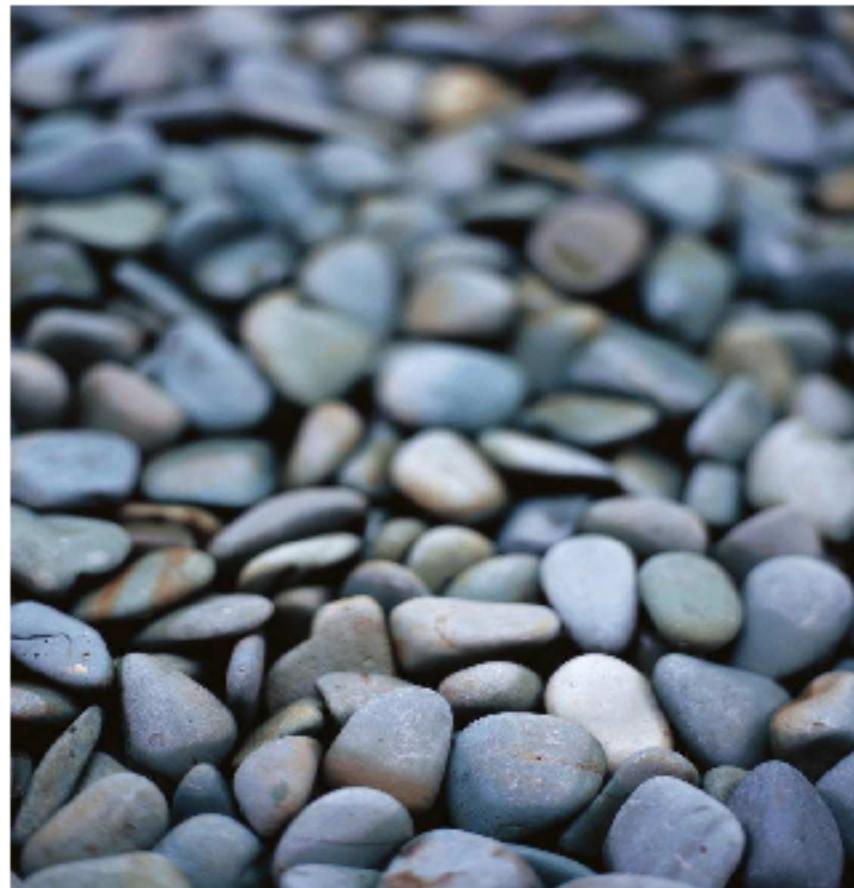
→ Me traz uma pedra?

# O Problema da Pedra

- Me traz uma pedra?
- Pequena?

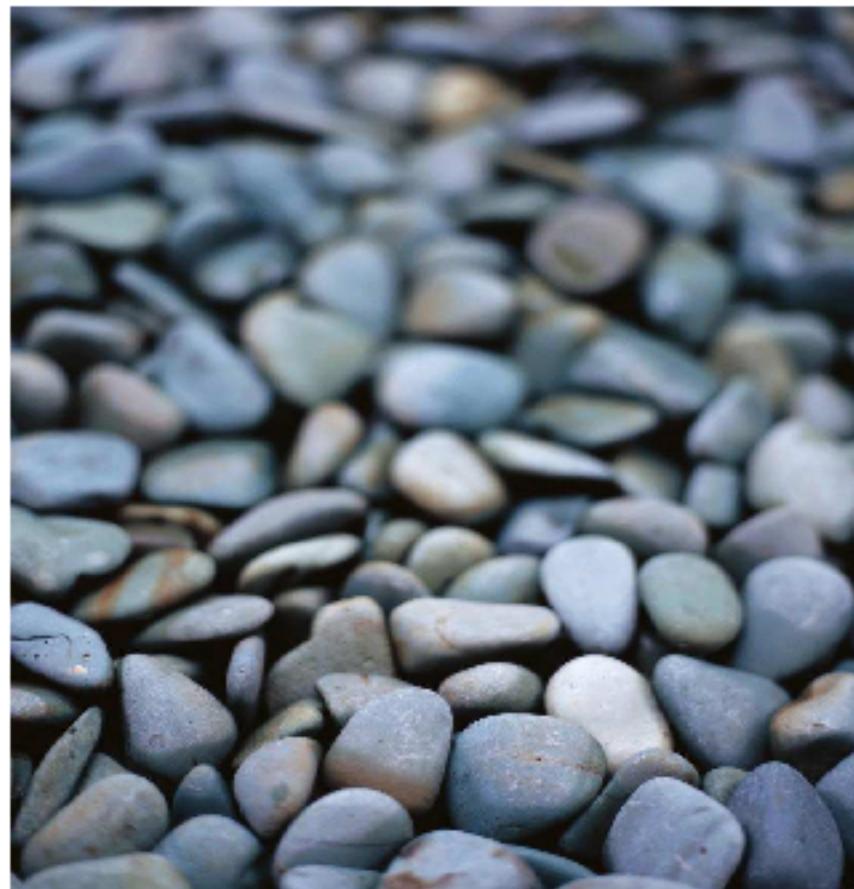
# O Problema da Pedra

- Me traz uma pedra?
- Pequena?



# O Problema da Pedra

- Me traz uma pedra?
- Pequena?
- Parece de vidro?



# O Problema da Pedra

- Me traz uma pedra?
- Pequena?
- Parece de vidro?



# O Problema da Pedra

- Me traz uma pedra?
- Pequena?
- Parece de vidro?
- Azul?



# O Problema da Pedra

- Me traz uma pedra?
- Pequena?
- Parece de vidro?
- Azul?
- Parece uma bola?



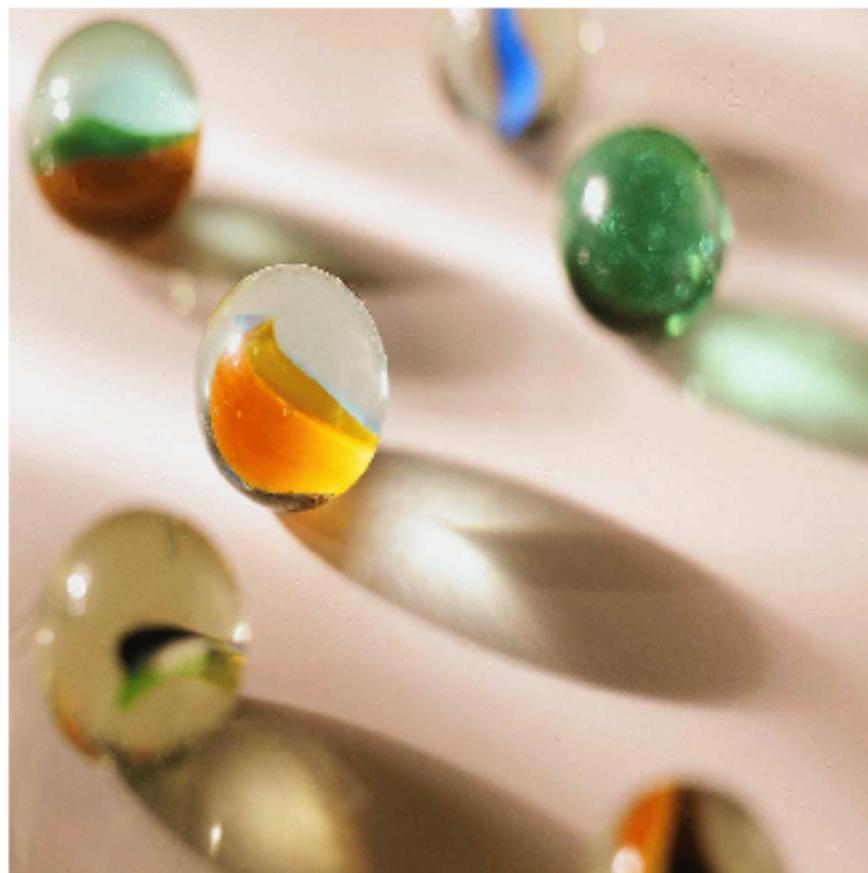
# O Problema da Pedra

- Me traz uma pedra?
- Pequena?
- Parece de vidro?
- Azul?
- Parece uma bola?



# O Problema da Pedra

- Me traz uma pedra?
- Pequena?
- Parece de vidro?
- Azul?
- Parece uma bola?
- Uma bola de gude!



## O que vimos nessa conversa

- ➡ A criança quer uma coisa
- ➡ A criança não sabe exatamente o que é essa coisa
  - Ao ver ela reconhece que é o que ela quer, ou não.
- ➡ O pai interpreta o pedido da criança
- ➡ O pai tenta trazer o pedido na forma mais exata possível

# No Mundo do Software

- ➡ Acontece a mesma coisa
- ➡ Os usuários e interessados têm desejos, querem ver algo acontecendo
- ➡ Esse algo tem que ser bem descrito, caso contrário o usuário receberá algo que não deseja
- ➡ Requisito é uma exigência que se faz ao software

## Exemplo de Requisito

- ➡ O Sistema deverá permitir a busca do preço de um produto dado o código de barra.
- ➡ O Sistema deverá emitir o histórico escolar de cada aluno.

# Qual o Segredo?

- Para atender o usuário e os interessados é importante definir os verdadeiros requisitos do software
- Existem muitas formas de definir esses requisitos

Professor:

*Geraldo Xexéo*

Conteúdo:

**O Que é um  
Requisito**

## Requisito (Padrão IEEE)

- é uma sentença identificando uma capacidade, uma característica física ou um fator de qualidade que limita um produto ou um processo
- IEEE Std 1220-1994,

# Exemplo de Requisito

- ➡ O Sistema de Informação Acadêmica (SAI) deve permitir que um aluno faça sua matrícula a cada semestre.
  - Requisito amplo
  - Provavelmente com sub-requisitos
- ➡ O SIA deve permitir que o coordenador cancele a matrícula de um aluno em uma cadeira a qualquer momento.
  - Mais específico

## Requisito - Uso Geral

- Requisito: é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir seus objetivos
- Não precisa ser uma sentença, mas pode ser outro tipo de descrição do que o usuário quer

# Requisito como Sentença

- O Sistema deverá permitir que um passageiro faça a reserva de sua cadeira até 4 horas antes do voo.

# Outros Requisitos

- ➡ Interpretação ampla da palavra
- ➡ Textos livres, sentenças, modelos ou partes do modelo

# Categorias de Requisitos

- ➡ Requisitos que devem ser totalmente satisfeitos
- ➡ Requisitos que são altamente desejáveis, mas não necessários
- ➡ Requisitos que são possíveis, mas poderiam ser eliminados

# Tipos de Requisito

# Tipos de Requisito

**Usuário**

# Tipos de Requisito



**Usuário**

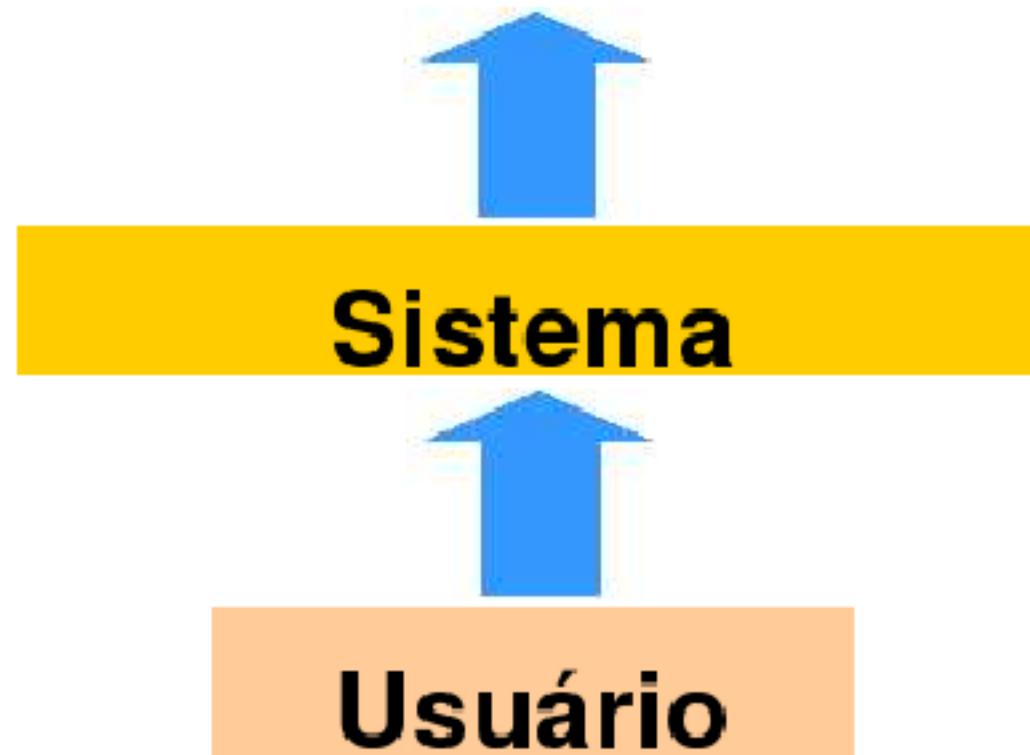
# Tipos de Requisito

**Sistema**

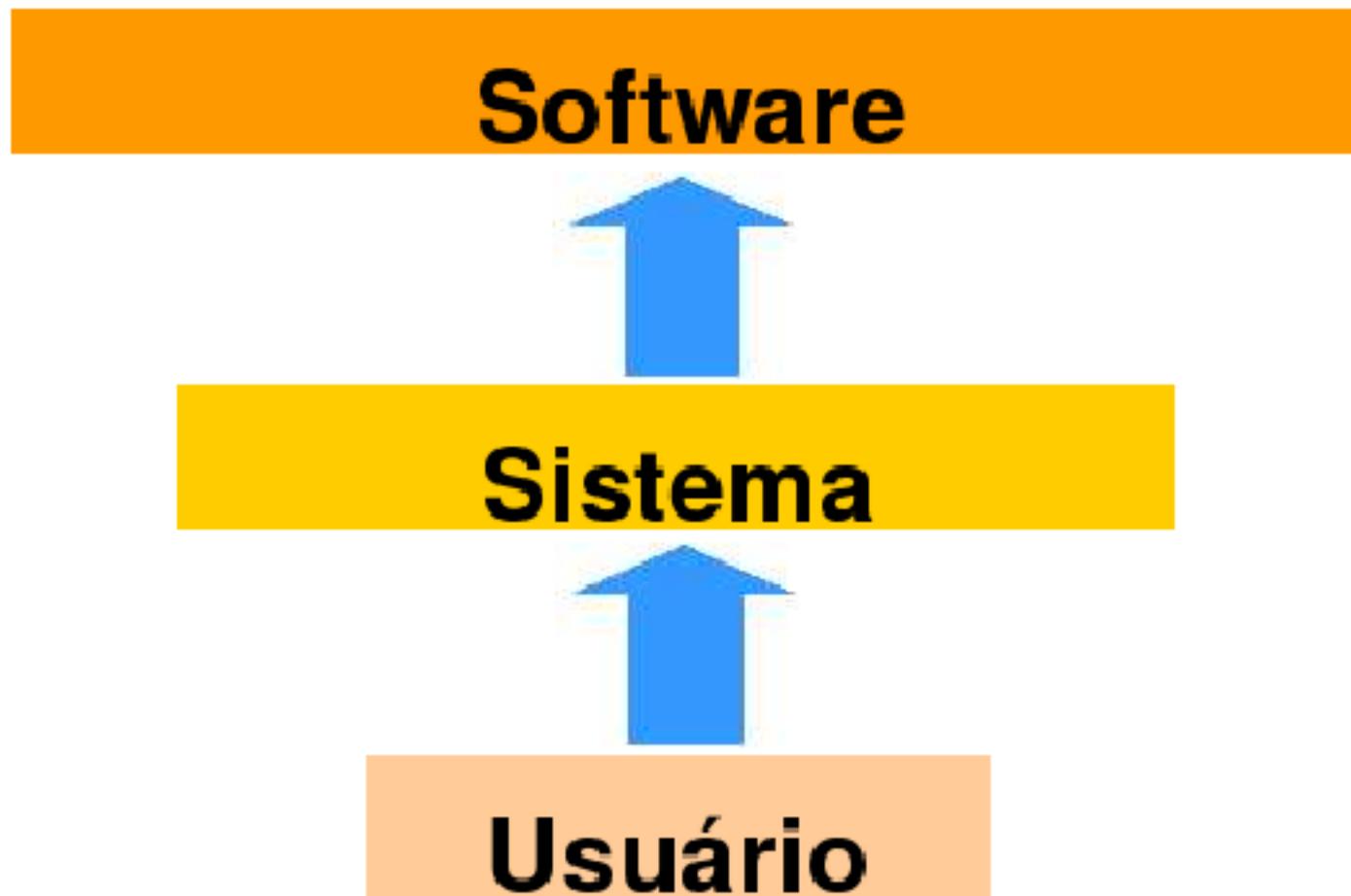


**Usuário**

# Tipos de Requisito



# Tipos de Requisito



# Requisito do Usuário

- É algum comportamento ou característica que o usuário deseja do software ou o sistema como um todo;
- o que o usuário quer.
- São escritos pelo próprio usuário ou levantados por um analista de sistemas que consulta o usuário.

## Requisito do Usuário

- O sistema deverá permitir a impressão de uma segunda via do recibo

# Requisito do Sistema

- É algum comportamento ou característica exigido do sistema como um todo, incluindo hardware e software.
- O comportamento desejado do sistema.
- São normalmente levantados por engenheiros ou analistas de sistemas, refinando os requisitos dos usuários e os transformando em termos de engenharia.

# Requisito do Sistema

- O sistema deverá funcionar utilizando a rede telefônica fixa



# Requisito do Software

- É algum comportamento ou característica que é exigido do software.
- Normalmente levantado por analistas de sistemas.

# Requisito do Software

- O sistema deverá utilizar o banco de dados Oracle

# Documentos de requisitos

- ➔ Definição dos requisitos: listagem completa de tudo que o cliente espera que o sistema proposto faça
- ➔ Especificação dos requisitos: redefine os requisitos em termos técnicos apropriados para o desenvolvimento do projeto do sistema
- ➔ Gerência de configuração: correspondência direta entre os dois requisitos

# Dependência de Requisitos

- ➡ Requisitos podem depender uns dos outros
  - Necessidades de negócio
  - Necessidades de implementação
- ➡ Cada dependência fora da ordem de prioridade do usuário aumenta o risco do projeto

# Prioridade de Requisitos

## → Fatores

- Diminuir o custo da implementação
- Valor para o comprador
- Tempo para implementar
- Facilidade técnica de implementar
- Facilidade do negócio para implementar
- Valor para o negócio
- Obrigação por alguma autoridade externa

# Mudança

- Requisitos inevitavelmente mudam durante o desenvolvimento e após.
- Não se deve esperar definir-los completamente, mas gerenciar sua mudança.
- A única constante é a mudança

# Gerência de configuração

- ➡ Conjunto de procedimentos que controlam
  - os requisitos que definem o que o sistema deverá fazer
  - os módulos de projeto gerados a partir dos requisitos
  - o código do programa que implementa o projeto
  - os testes que verificam a funcionalidade do sistema
  - os documentos que descrevem o sistema

Professor:

*Geraldo Xexéo*

Conteúdo:

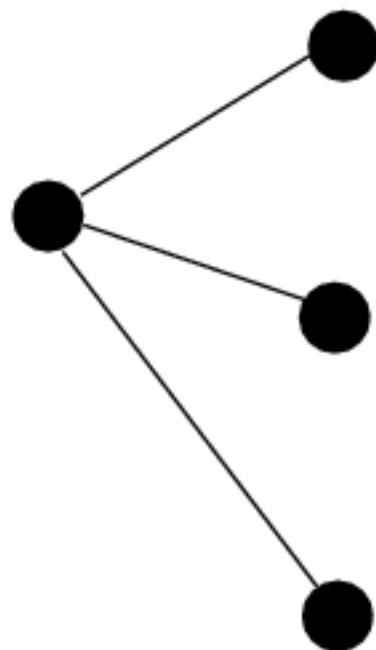
**Impactos dos  
Requisitos**

**Objetivo    Requisitos    Análise    Projeto    Implementação    V&V**

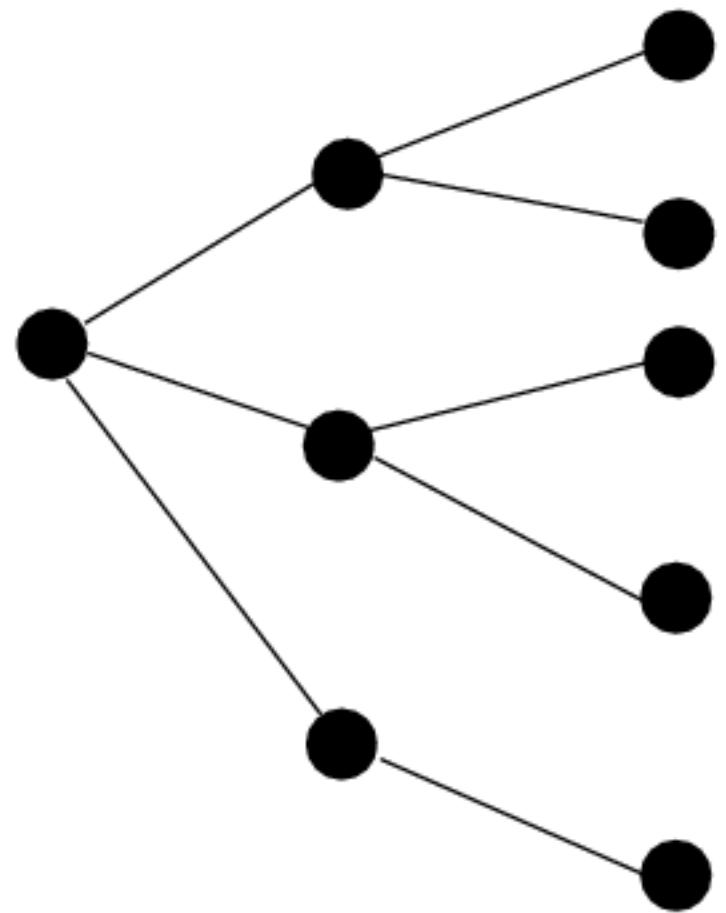
**Objetivo    Requisitos    Análise    Projeto    Implementação    V&V**



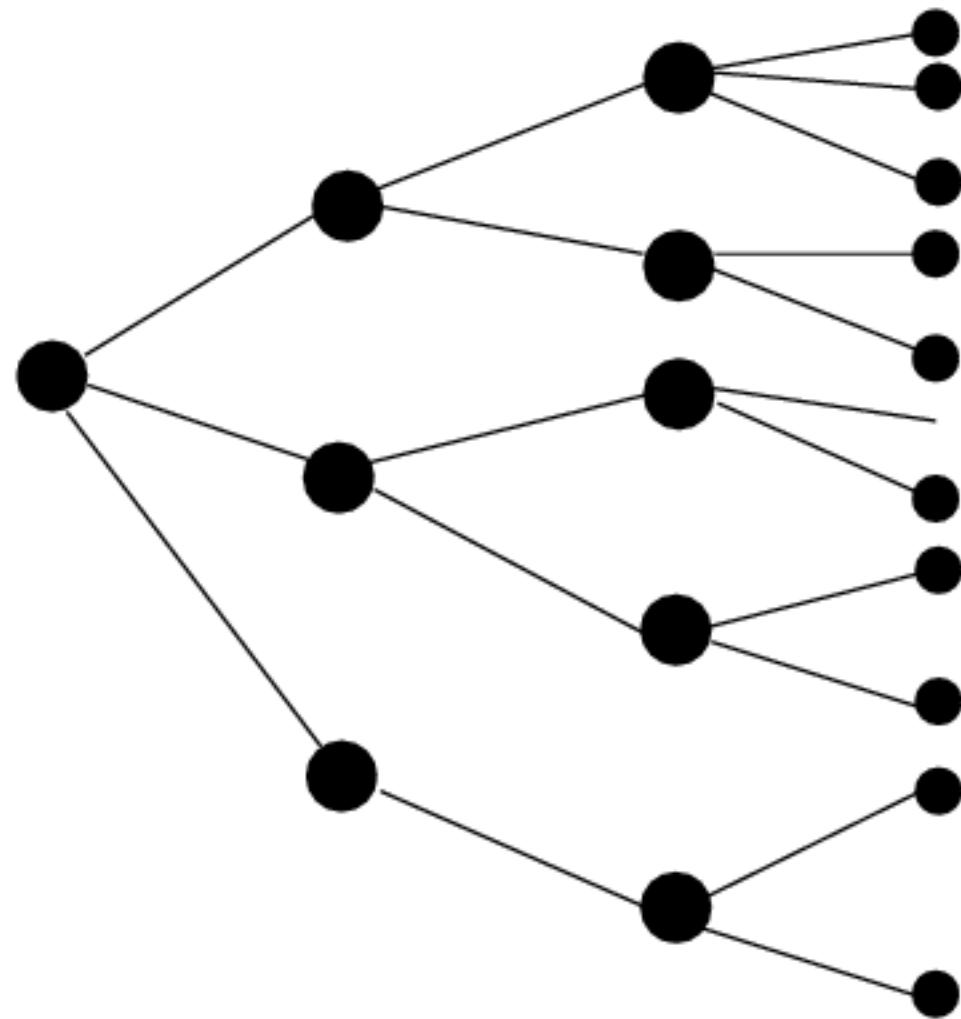
**Objetivo**   **Requisitos**   **Análise**   **Projeto**   **Implementação**   **V&V**



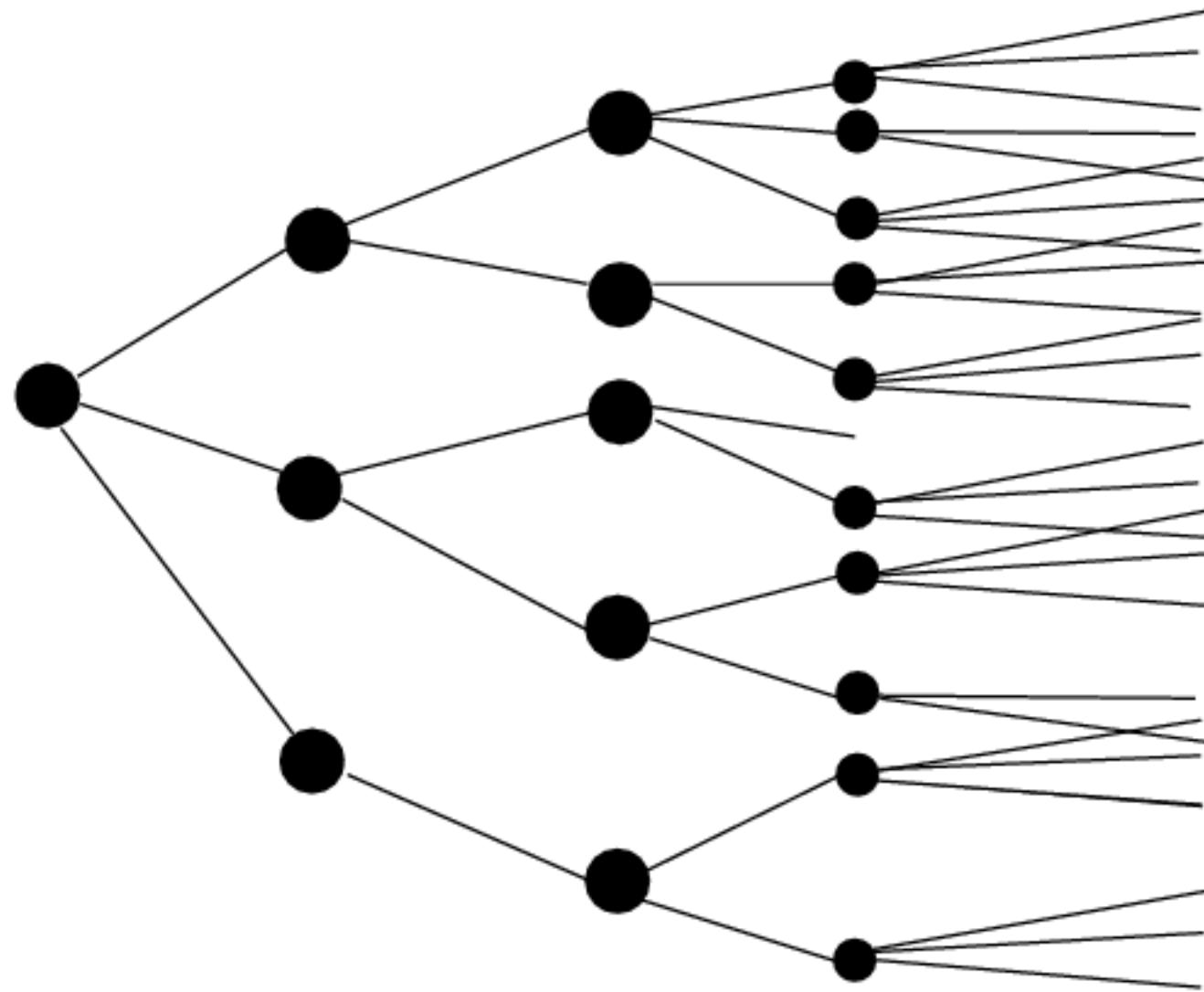
**Objetivo**   **Requisitos**   **Análise**   **Projeto**   **Implementação**   **V&V**



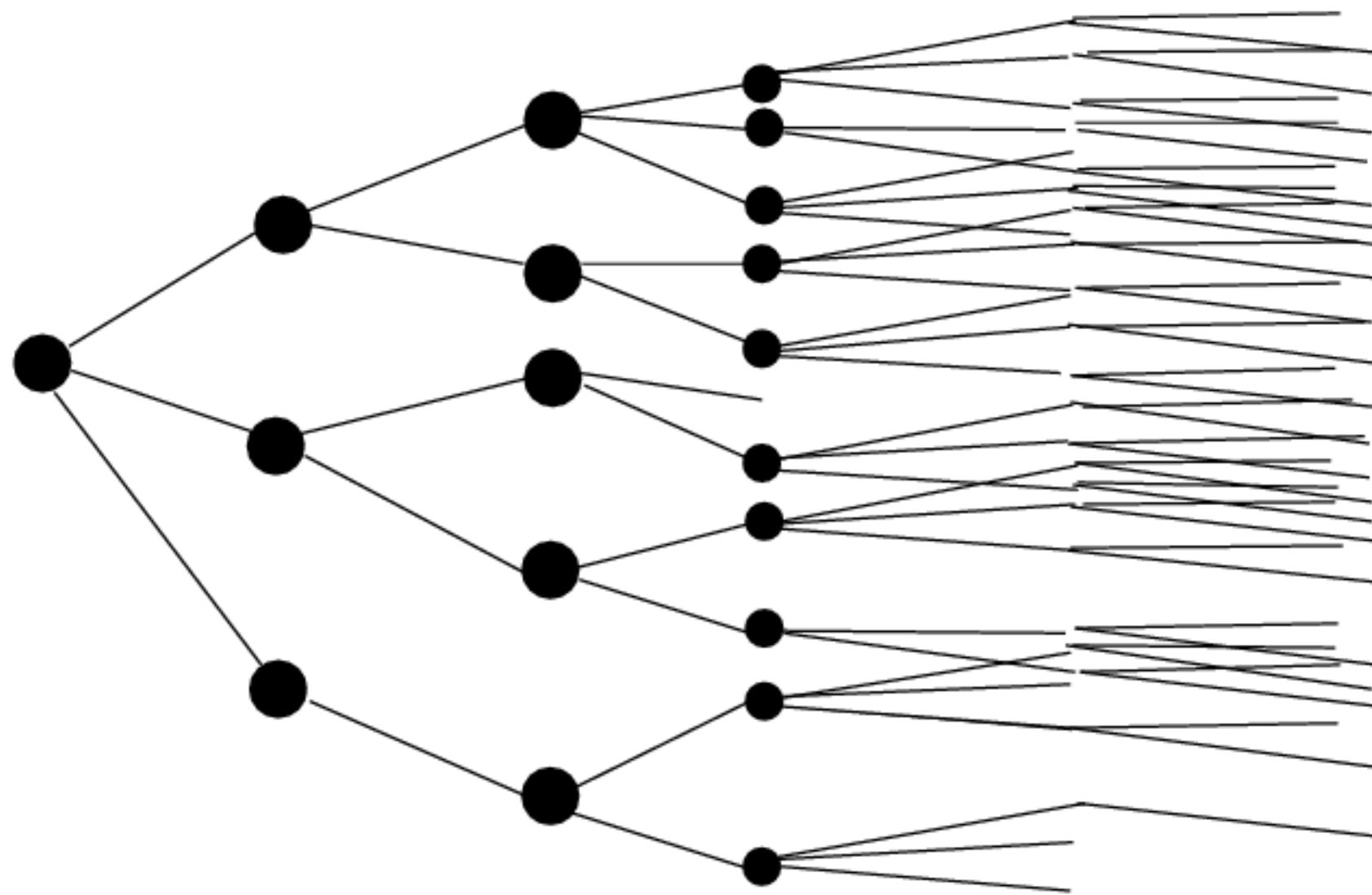
## Objetivo Requisitos Análise Projeto Implementação V&V



## Objetivo Requisitos Análise Projeto Implementação V&V



## Objetivo Requisitos Análise Projeto Implementação V&V



# Custo Relativo de Correção

Requisito	1
Projeto	5
Codificação	10
Teste de Unidade	20
Teste de Aceitação	50
Manutenção	200

# Resumo de Defeitos

Origem do Defeito	Defeitos Potenciais	Eficácia da Correção	Defeitos Entregues
Requisitos	1.00	77%	0.23
Projeto	1.25	85%	0.19
Codificação	1.75	95%	0.09
Documentação	0.60	80%	0.12
Correções	0.40	70%	0.12
Erradas			
Total	5.00	85%	0.75

Fonte: Jones [1994].

# Custo de Erros de Requisitos

- São o tipo mais comum de erro
- São o tipo mais caro para consertar
- 25% a 40% do projeto

# Taxa Mensal de Mudança

<b>Tipo de Software</b>	<b>Taxa Mensal de Mudança de Requisitos</b>
Sistemas de Informação	1.5%
Sistemas Militares	2.0%
Software Básico	2.0%
Sistemas Comerciais	3.5%

Professor:

*Geraldo Xexéo*

Conteúdo:

**Bons Requisitos**

# Bons Requisitos

## → Necessário

- se retirado, haverá uma deficiência no sistema
  - não atenderá plenamente as expectativas do usuário.
- Não devem existir requisitos do tipo "seria interessante ter".
  - O requisito é necessário ou é dispensável.
- Deve ser levado em conta que cada requisito aumenta a complexidade e o custo do projeto
  - não podem ser introduzidos de forma espúria.

# Bons Requisitos

## → Conciso

- Cada requisito define apenas um requisito que deve ser feito e apenas o que deve ser feito
- Claro e simples.
- Conciso
  - não inclui explicações, motivações, definições ou descrições do seu uso.
    - Estes textos podem ser mantidos em outros documentos, apontados pelo requisito, de preferência sob o controle de um sistema de gerência de requisitos

# Bons Requisitos

## → Independente da Implementação

- Define o que deve ser feito, mas não como.
- Não reflete um projeto ou uma implementação.
- Não descreve uma operação.
  - Requisitos de interface são uma exceção a essa regra

# Bons Requisitos

## → Alcançável

- Realizável a um custo definido por uma ou mais partes do sistema

## → Completo

- Não precisa ser explicado ou aumentado, garantindo capacidade suficiente do sistema.

## → Rastreável

- Podemos detectar sua origem

# Bons Requisitos

- ➡ Consistente
  - Não contradizendo ou mesmo duplicando outro requisito
  - Utilizando os termos da mesma forma que outros requisitos
- ➡ Não-ambíguo
  - Possuindo uma e apenas uma interpretação.
  - Nesse caso é muito importante prestar atenção na linguagem sendo utilizada.

# Bons Requisitos

## → Verificável

- Não sendo vago ou geral
- Sendo quantificado
  - de uma maneira que permita a verificação de uma das seguintes formas: inspeção, análise, demonstração ou teste.

## → Ordenável

- por estabilidade ou importância (ou ambos).

## → Aceito pelos usuários e desenvolvedores.

# Requisitos Falsos e Verdadeiros

- Um requisito é **verdadeiro** quando o sistema deve cumpri-lo qualquer que seja a tecnologia de implementação escolhida.
- Se um sistema pode cumprir sua finalidade sem que um requisito seja implementado, então esse requisito é **falso**.

# Requisitos Falsos

- ➡ Requisitos falsos aparecem de várias formas dentro do sistema:
  - por cópia de sistemas antigos,
  - por hábito do analista,
  - por pensarmos na tecnologia antes do tempo.

# Requisitos Falsos Tecnológicos

→ Incluídos pelo analista por influência de alguma tecnologia

- incluídos pelo passado, quando incluímos na especificação algo que existe na implementação existente, mas que não é necessário ao funcionamento do sistema;
- incluídos por antecipação, quando incluímos algo na especificação em função de alguma tecnologia escolhida para a implementação;

# Requisitos Falsos Arbitrários

→ Incluídos arbitrariamente pelos analistas

- Incluídos por influência da ferramenta de modelagem, quando incluímos na especificação algo desnecessário para fazer o sistema, mas necessário por alguma característica da ferramenta de modelagem, e
- incluídos por preciosismo, quando incluímos uma função espúria no sistema, i.e., uma função que foi não solicitada pelo usuário.

# Validando Requisitos

## → Técnicas Manuais

- Leitura
- Referência Cruzada Manual
- Entrevistas
- Revisões
- Listas de Verificação
- Modelos manuais
- Cenários
- Provas Matemáticas

## → Técnicas

- ### Automatizadas
- Referência Cruzada Automática
  - Protótipos

# Revisão dos requisitos

- Rever metas e objetivos estabelecidos para o sistema
- Comparar requisitos metas o objetivos
- Descrever o ambiente operacional
- Examinar
  - interfaces
  - fluxo de informações
  - funções
- Verificar omissões, imperfeições e inconsistências
- Documentar riscos
- Discutir sobre como o sistema será testado

Professor:

*Geraldo Xexéo*

Conteúdo:

**Tipos de Requisitos**

# Requisito Funcional

- ➔ Representa algo que o sistema deve fazer,
- ➔ Uma função esperada do sistema que agregue algum valor a seus usuários.
- ➔ Exemplos típicos:
  - a emissão de relatórios
  - a realização e manutenção de cadastros

# Categorias de Funções

## → Evidente

- deve ser executada com conhecimento do usuário

## → Oculta

- deve ser executada mas não necessariamente visível para os usuários

## → Enfeite/Decoração

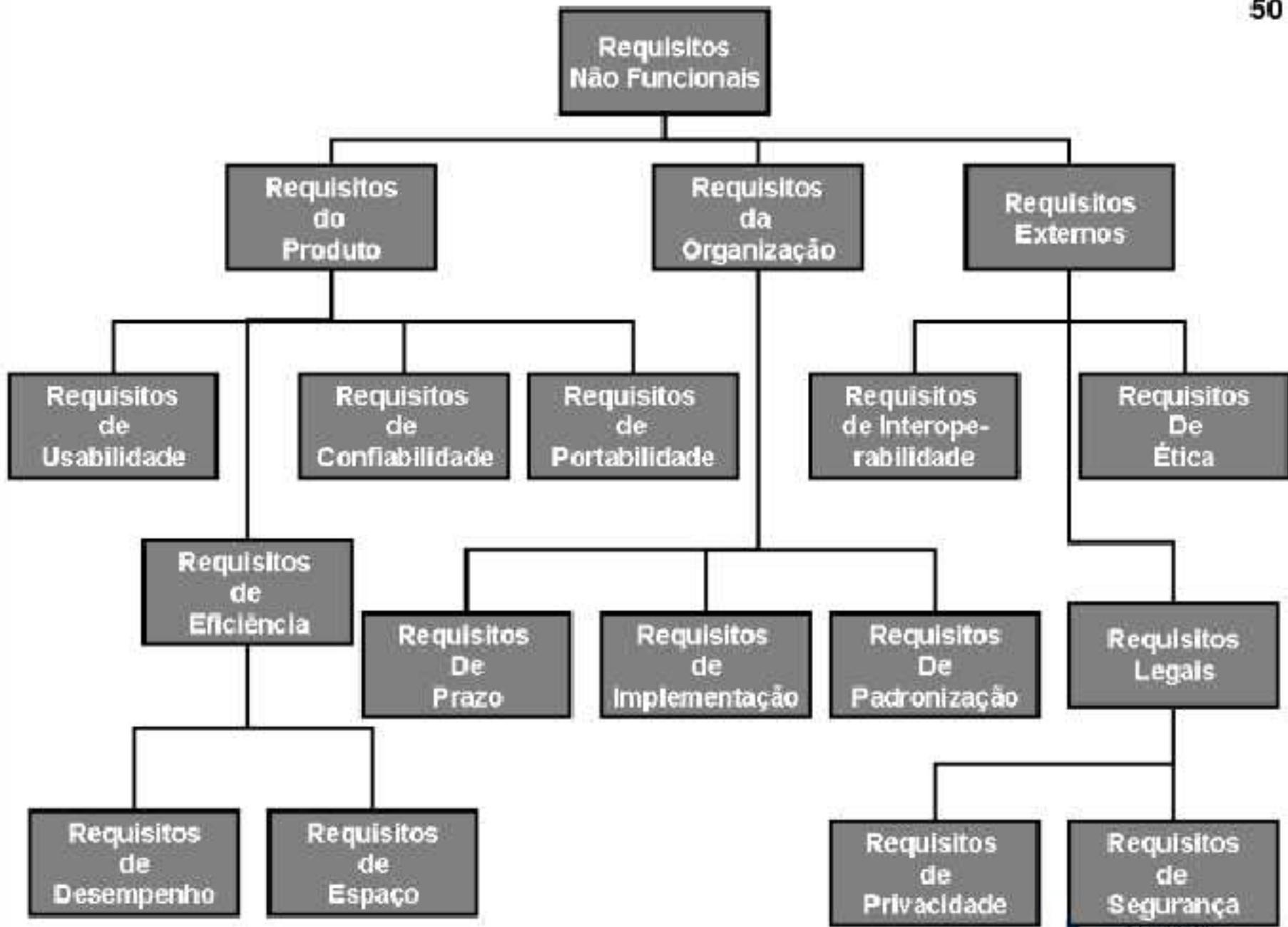
- opcional, sua adição não afeta significativamente o custo ou outras funções

# Requisito Não Funcional

- Fala da forma como os requisitos funcionais devem ser alcançados.
- Define propriedades e restrições do sistema.
  - Muitos requisitos não funcionais são também requisitos de qualidade, como exigências de desempenho e robustez.
  - Outros são restrições ou exigências de uso de uma ou outra tecnologia.

# Tipos de requisitos

- Ambiente físico
- Interfaces
- Usuários e fatores humanos
- Funcionalidade
- Documentação
- Dados
- Recursos
- Segurança
- Garantia de qualidade
- Muitos outros



# Requisito Não-Funcional

- ➡ Muitas vezes chamados de "requisitos de qualidade" ou "atributos de qualidade"
  - Qualidade fala de atendimento a expectativas,  
requisito fala de atendimento a especificações
- ➡ Difíceis de tratar formalmente

Professor:

*Geraldo Xexéo*

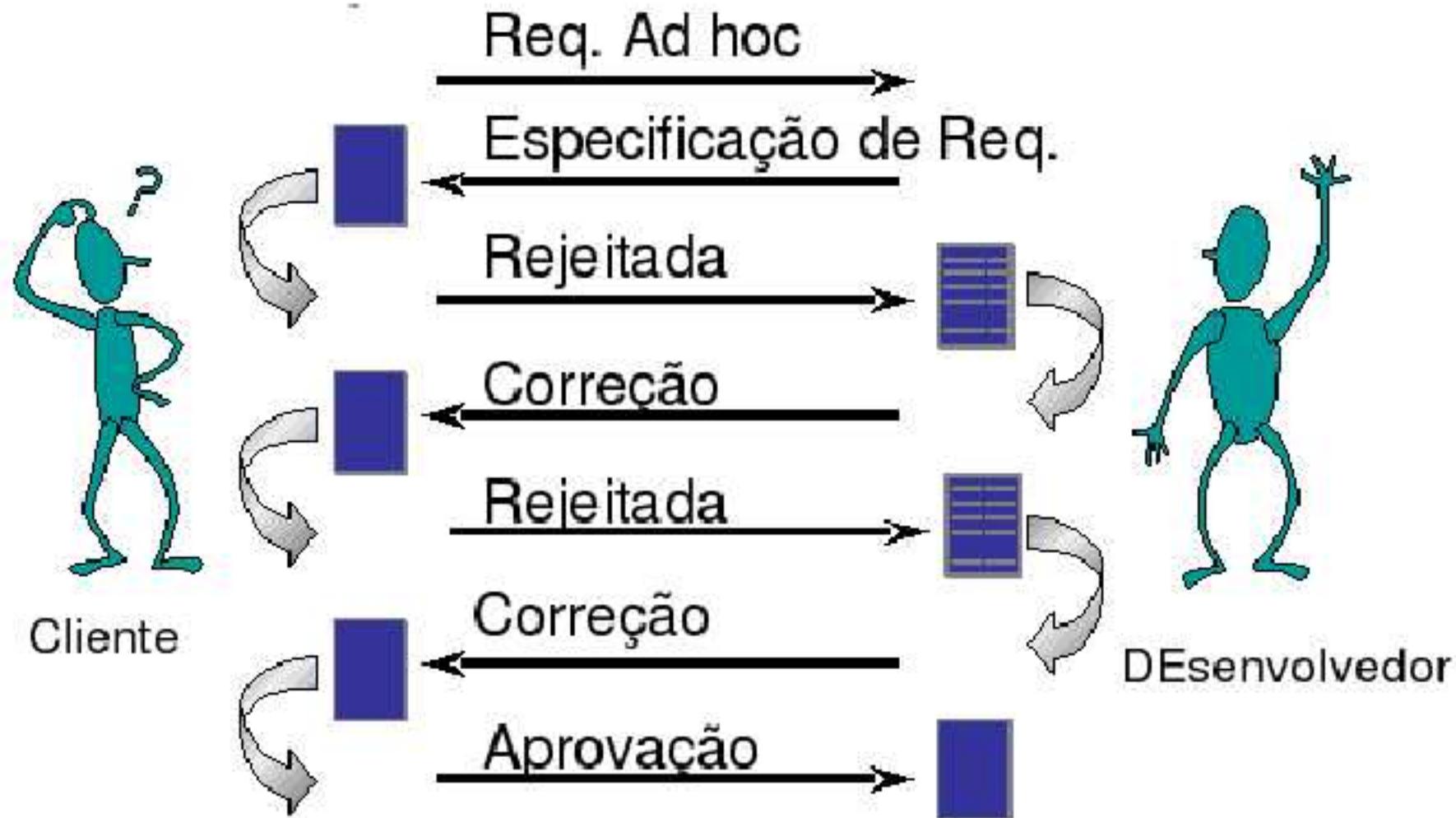
Conteúdo:

**Elicitando os  
Requisitos**

# Elicitação de Requisitos

- Nome que indica de forma mais completa o Levantamento de Requisitos
- Atividade contínua no projeto

# Como o processo acontece



# Como Fazer?

- ➔ Elicitar requisitos de fontes individuais
- ➔ Garantir que as necessidades de todos os usuários são consistentes e realizáveis
- ➔ Validar que os requisitos encontrados refletem adequadamente as necessidades do usuário

# Como Fazer?

- Identificar "partes" relevantes que são fontes de requisitos
  - Usuários finais, sistemas de interface, fatores ambientais
- Levantar as "listas de desejo" de cada parte relevante
  - Provavelmente contém ambigüidades, inconsistências, requisitos impossíveis e não-testáveis, estando ainda, incompleta
- Documentar e refinar a "lista de desejos" para cada parte, até que seja auto-consistente
  - Normalmente mantendo um alto-grau de abstração e a linguagem do usuário
- Integrar as listas de todas as partes relevantes (pontos de vista)
  - Resolver os conflitos
  - Verificar viabilidade dos requisitos
- Determinar os requisitos não-funcionais

# Problemas possíveis

## → Os Stakeholders

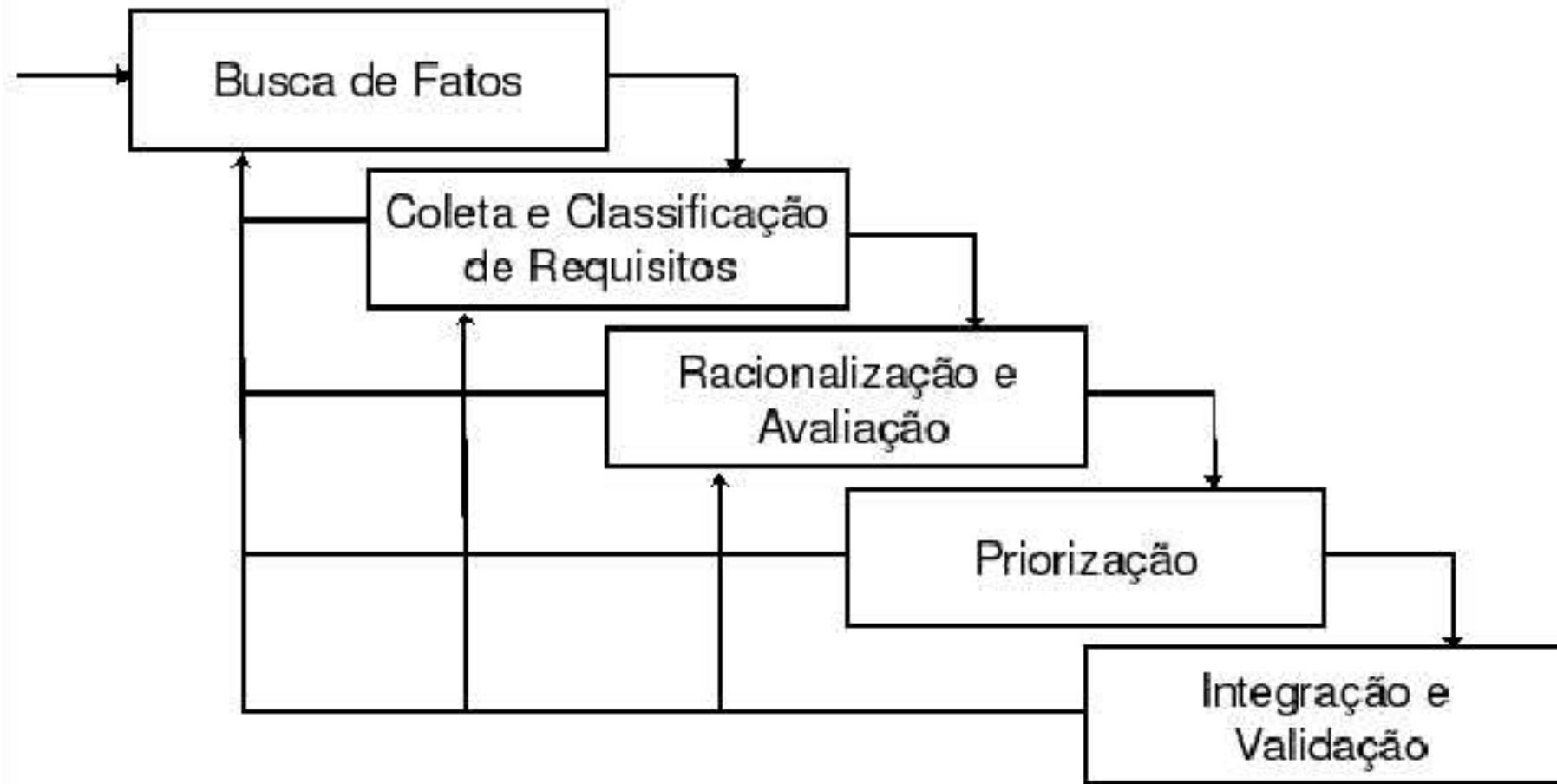
- Podem saber o que querem, mas não como articular suas idéias
- Podem não saber o que querem
- Podem pensar que sabem o que querem até receber o que pediram

## → Os Analistas

- Podem pensar que entendem o problema melhor que os usuários

## → Todos acreditam que os outros são motivados por política ou dinheiro

# Modelo de Elicitação



# Busca de Fatos (Usuário)

- Identificar grupos relevantes através de múltiplos níveis da organização.
- Determinar os contextos operacional e do problema, incluindo a definição dos modos operacionais, objetivos e cenários de missão como apropriados.
- Identificar sistemas similares.
- Realizar análise de contexto

# Busca de Fatos (Desenvolvedor)

- Identificar especialistas do domínio da aplicação e de desenvolvimento.
- Identificar modelo de domínio e modelo de arquitetura.
- Conduzir pesquisas tecnológicas, para mais tarde fazer estudo de viabilidade e análise de risco.
- Identificar custos e restrições à implementação impostas pelo patrocinador.

# Coleta e Classificação (Usuários)

→ Levantar a lista de desejos de cada grupo.

## Coleta e Classificação (Des.)

- Classificar a lista de acordo com funcionais, não funcionais, restrições de ambiente, restrições de projeto e ainda de acordo com as partições definidas pelo modelo de domínio e pelo paradigma de desenvolvimento.

# Racionalização e Avaliação (U)

- Responder questões da forma "Por que você precisa de X?", a partir de raciocínio abstrato. Isso auxilia a transformar o raciocínio das questões sobre "como?" para as questões sobre "o quê?".

## Racionalização e Avaliação (D)

- Realizar uma análise de riscos, investigando técnicas, custos, prazos e incluindo análise de custos e benefícios e viabilidade baseado na disponibilidade da tecnologia.

# Priorização (U)

→ Determinar funcionalidades críticas para a missão.

# Priorização (D)

- Priorizar requisitos baseados em custo e dependência. Estudar como o sistema pode ser implementado de forma incremental, investigando modelos arquiteturais apropriados

# Integração e Validação (U)

- Resolver a maior quantidade possível de pontos em aberto.
- Validar que os requisitos estão concordando com os objetivos iniciais.
- Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação).

# Integração e Validação (D)

→ Resolver conflitos e verificar consistência

# Técnicas de Elicitação

→ Entrevistas

- Estruturadas
- Não estruturadas

→ JAD

→ CRC

→ Volare

→ Muitas outras

# Documentando Requisitos

- Número identificador,
  - para facilitar a discussão, identificamos todos os requisitos unicamente.
- Tipo
  - Classificando-o como funcional, não funcional,...
- Evento que o atende
- Descrição
- Justificativa
- Fonte do requisito
  - A pessoa ou o grupo que o originou
- Critério de aceitação
  - Uma medida que possa ser usada para garantir que o requisito foi alcançado.
- Satisfação do usuário
- Insatisfação do usuário
- Dependências
  - Referências a outros requisitos que dependem de alguma forma desse requisito
- Conflitos
  - Referência aos requisitos que de alguma forma conflitam com esse
- Material de apoio
  - Listagem de material de apoio para atender esse requisito
- Histórico
  - Documentação da criação e das mudanças efetuadas

Volere

## Satisfeito x Insatisfeito

- ➔ Fazer as duas perguntas
- ➔ Quanto você ficaria satisfeito se essa função fosse implementada
- ➔ Quanto você ficaria insatisfeito se essa função não fosse implementada

		Implementado	
		Insatisfeito	Satisffeito
Não Implementado	Insatisfeito	ELIMINAR CONFLITO	REQUISITO CRÍTICO
	Satisffeito	NÃO FAZER	DECORAÇÃO

Professor:

*Geraldo Xexéo*

Conteúdo:

**Escrevendo  
Requisitos**

# Escrevendo Requisitos

- Use sentenças e parágrafos curtos.
- Use a voz ativa.
- Use verbos no futuro.
- Use os termos de forma consistente e mantenha um glossário.

# Escrevendo Requisitos

- Para cada requisito, avalie se a partir de sua definição é possível determinar se ele está pronto ou não.
- Garanta que todos os requisitos são verificáveis imaginando (e possivelmente documentando) uma forma de fazê-lo.
- Verifique requisitos agregados (termos como "e" e "ou" são uma boa indicação) e divida-os.
- Mantenha um nível de detalhe único em todos os requisitos.

# 5W2H

- Who?
- When?
- What?
- Where?
- Why?
- How?
- How Much?

Professor:

*Geraldo Xexéo*

Conteúdo:

**Verificação**

**e**

**Validação**

# Problemas com Requisitos

- Problemas de Escopo
- Problemas de Compreensão
- Problemas de Volatilidade

# Problemas de Escopo

- ➡ A fronteira do sistema fica mal definida
- ➡ É dada alguma informação de projeto desnecessária
  - Definindo previamente decisões de projeto

# Problemas de Compreensão

- Usuário não compreendem completamente suas necessidades
- Usuário tem um entendimento fraco das capacidades e limitações dos sistemas computacionais
- Analista tem um conhecimento fraco do domínio do problema
- Usuário e analista falam "linguagens" diferentes
- Omissão de informação que parece óbvia
- Visão conflitantes entre diferentes usuários
- Requisitos vagos ou não testáveis
  - "User friendly"
  - Robusto

# Problemas de Volatilidade

- Requisitos evoluem com o tempo

# Avaliando Requisitos

- ➔ Os requisitos estão completos?
  - ➡ Existe algum requisito ausente?
  - ➡ Existe alguma informação faltando em um requisito?
- ➔ Os requisitos são consistentes?
  - ➡ Existe alguma contradição na descrição entre requisitos diferentes?
- ➔ Os requisitos podem ser compreendidos?
  - ➡ Os leitores do documento entendem o que os requisitos significam?

# Avaliando Requisitos

- ➡ Os requisitos são ambíguos?
  - Existem diferentes interpretações de um mesmo requisito?
- ➡ O documento está estruturado?
  - As descrições de requisitos estão organizadas de forma que requisitos semelhantes estão agrupados?
  - Uma estrutura alternativa seria mais fácil de ler?
- ➡ Os requisitos são rastreáveis?
  - Cada requisito está identificado unicamente?
  - Os requisitos incluem referências a requisitos relacionados e justificativas para a sua inclusão
- ➡ O documento como um todo e cada requisito está de acordo com os padrões definidos?

## Checklist de elicitação

- ➔ O requisito foi identificado unicamente?
- ➔ Os requisitos descobertos tem uma fonte e justificativa válida?
- ➔ A fonte do requisito pode ser tracked?

## Checklist de elicitação

- O requisito tem um tipo?
- O requisito é ambíguo, não claro ou vago?
- O requisito está adequado aos objetivos de negócio do projeto?
- O requisito está em conflito com alguma restrição do domínio, política da empresa, regulamento ou lei?

# Checklist de elicitação

- O requisito está relacionado com algum assunto da organização (ou político), em oposição aos objetivos de negócio do sistema?
- O requisito leva em consideração as necessidades dos stakeholders?
- O requisito deve ser descrito em um cenário?

# Checklist Final de elicitação

- ➔ Todos os interessados foram consultados na fase de elicitação?
- ➔ Toda a informação recolhida na fase preliminar foi re-escrita como um requisito?

## Aula 9

Professor:

Geraldo Xexéo, D.Sc.  
DCC/IM/UFRJ  
PESC/COPPE/UFRJ

Conteúdo:

**Requisitos:**

**FIM**