

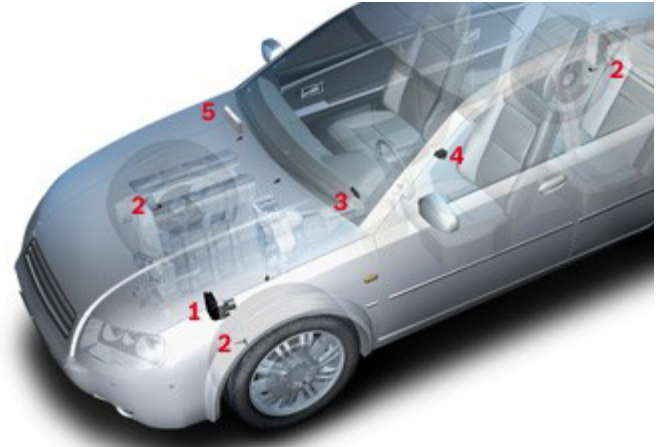
- Engenharia de Requisitos:
Introdução
- Prof. José Luiz de Freitas
Júnior

Para que as organizações desenvolvem e utilizam software?

- Reduzir custos, substituindo processos manuais e sujeitos a erros;
- Aumentar a produtividade das pessoas;
- Obter capacidades anteriormente não possíveis;
- Aumentar a capacidade de tomada de decisões.

Importância de Software na Sociedade Atual

- A economia das nações é altamente dependente de software;
- Vários sistemas são controlados por software;
- Os carros atuais tem milhões de linhas de código de software.



Air-bag system	Antilock brakes	Automatic transmission
Alarm system	Climate control	Collision-avoidance system
Cruise control	Communication system	Dashboard instrumentation
Electronic stability control	Engine ignition	Engine control
Electronic-seat control	Entertainment system	Navigation system
Power steering	Tire-pressure monitoring	Windshield-wiper control

Charette, R. N. (2009) This Car Runs on Code. Disponível em <http://www.spectrum.ieee.org/feb09/7649>

Importância de Software na Sociedade Atual

- Vários sistemas são controlados por software;
- Um celular típico contém milhões de linhas de código de software;
- Um Boeing 787 Dreamliner, possui cerca de 6,5 milhões de linhas de código de software;



Apostamos nossas vidas nos softwares...



E melhor que funcionem!

Ariane 5

- Ariane 5 e sua carga foram destruídos 37 segundos depois de levantar vôo;
- Erro devido a uma falha de software:
 - Conversão de número de ponto flutuante para número inteiro de 16 bits.

Conversão gerou uma exceção que não foi tratada;

Custo total do projeto: Us\$ 7 Bilhões;

Custo da carga: Us\$ 500 Milhões.



Engenharia de Software

- É a ciência da construção de Sistemas de Informação (SI) que propõe uma disciplina para o desenvolvimento dos Sistemas de Informação através do uso de técnicas e metodologias que possibilitem a obtenção de um Software (SW) confiável que funcione eficientemente em máquinas reais.

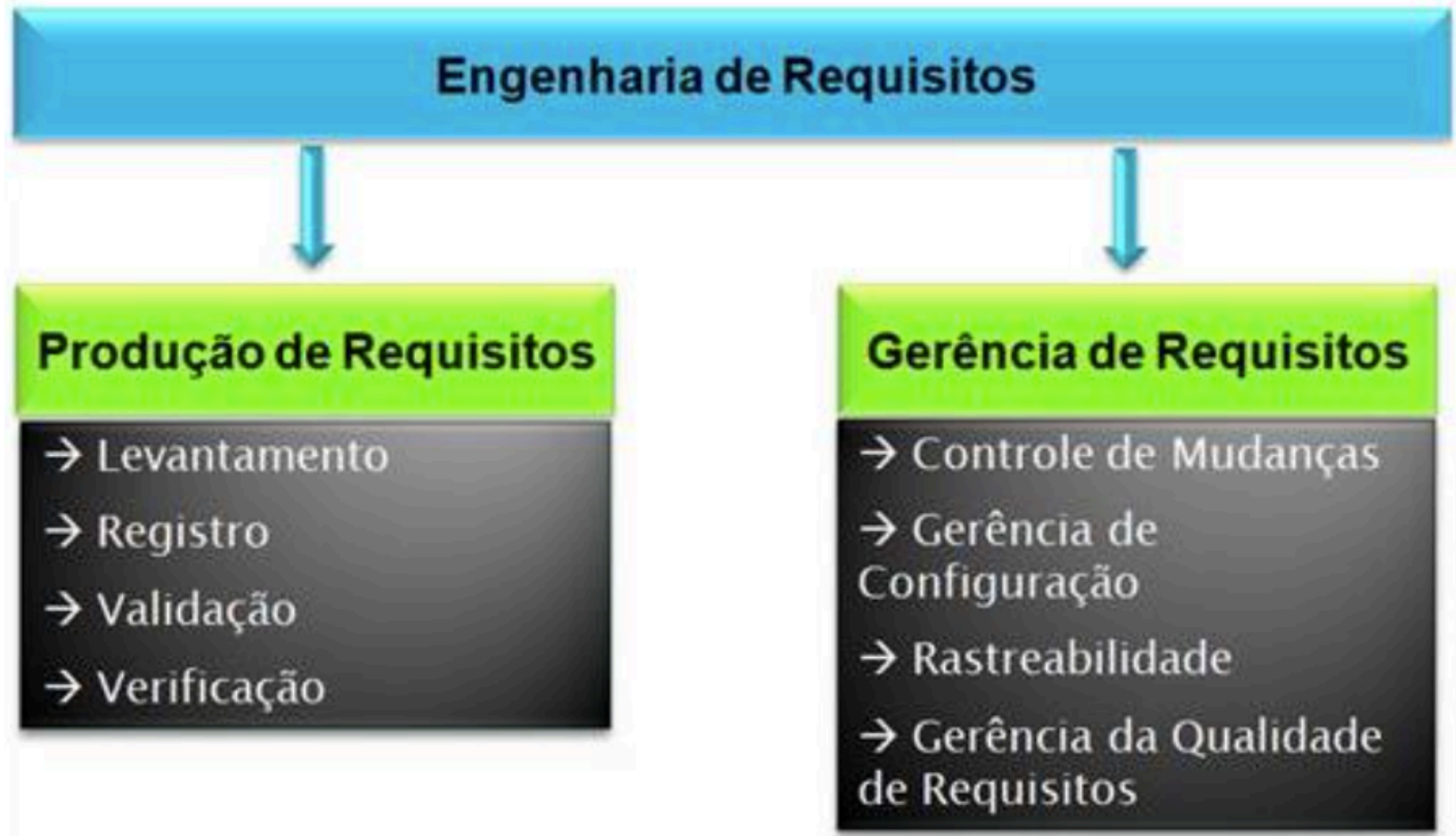
Engenharia de Requisitos

- Área da Engenharia de Software (EW) responsável por definir métodos e técnicas que devem ser seguidos para se produzir um **Documento de Especificação de Requisitos** que estabeleça o que o Sistema de Informação deve fazer e sob quais condições deve operar.

Engenharia de Requisitos – Definições

- Termo usado para descrever as atividades relacionadas à investigação e definição do escopo de um sistema de software;
- Processo sistemático de desenvolvimento de requisitos através de um processo cooperativo de análise onde os resultados das observações são codificados em uma variedade de formatos e a acurácia das observações são constantemente verificada;
- Processo de descobrir, analisar, documentar e verificar as funções e restrições do sistema.

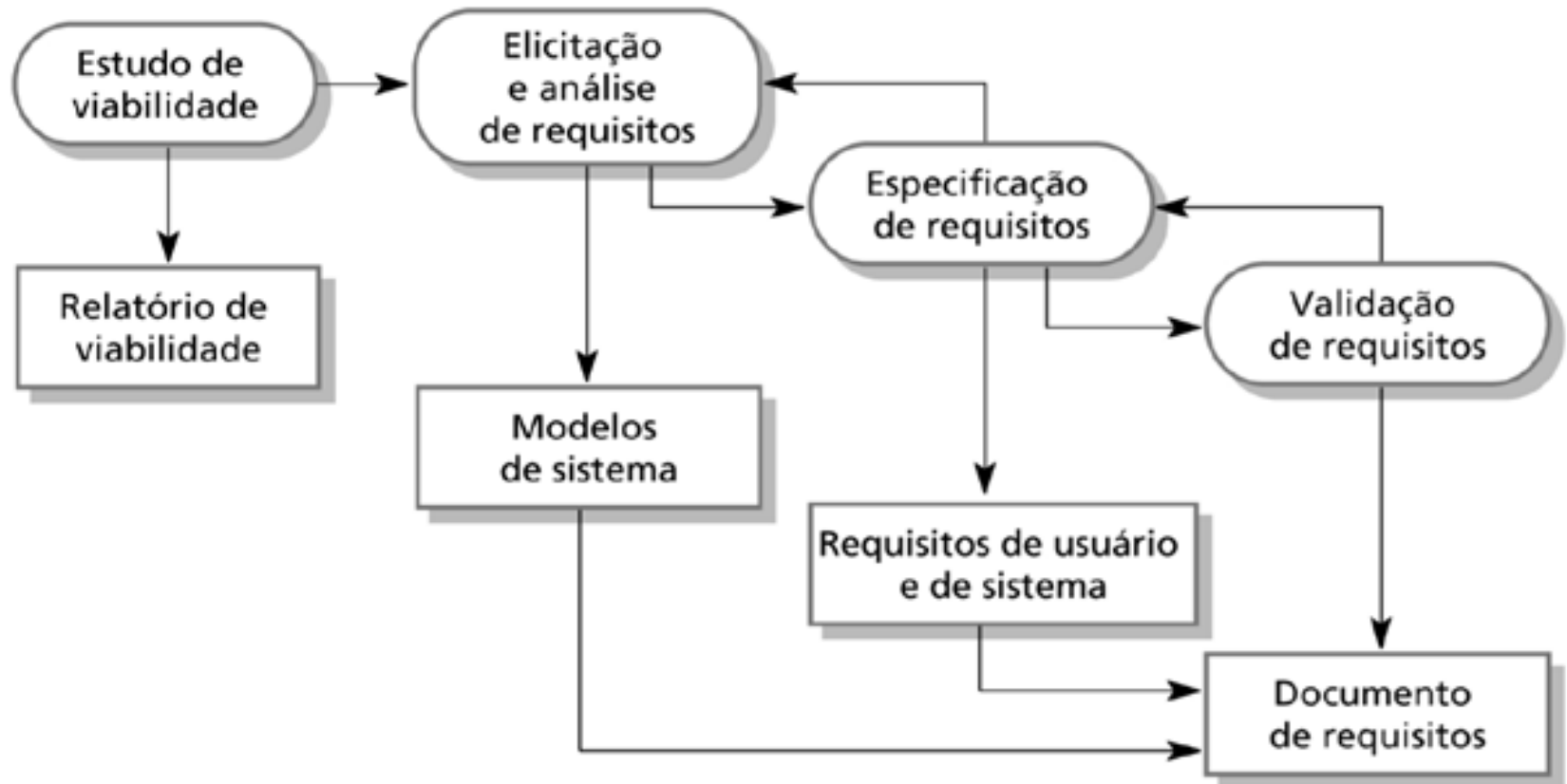
Engenharia de Requisitos – Organização



Engenharia de Requisitos – Objetivos

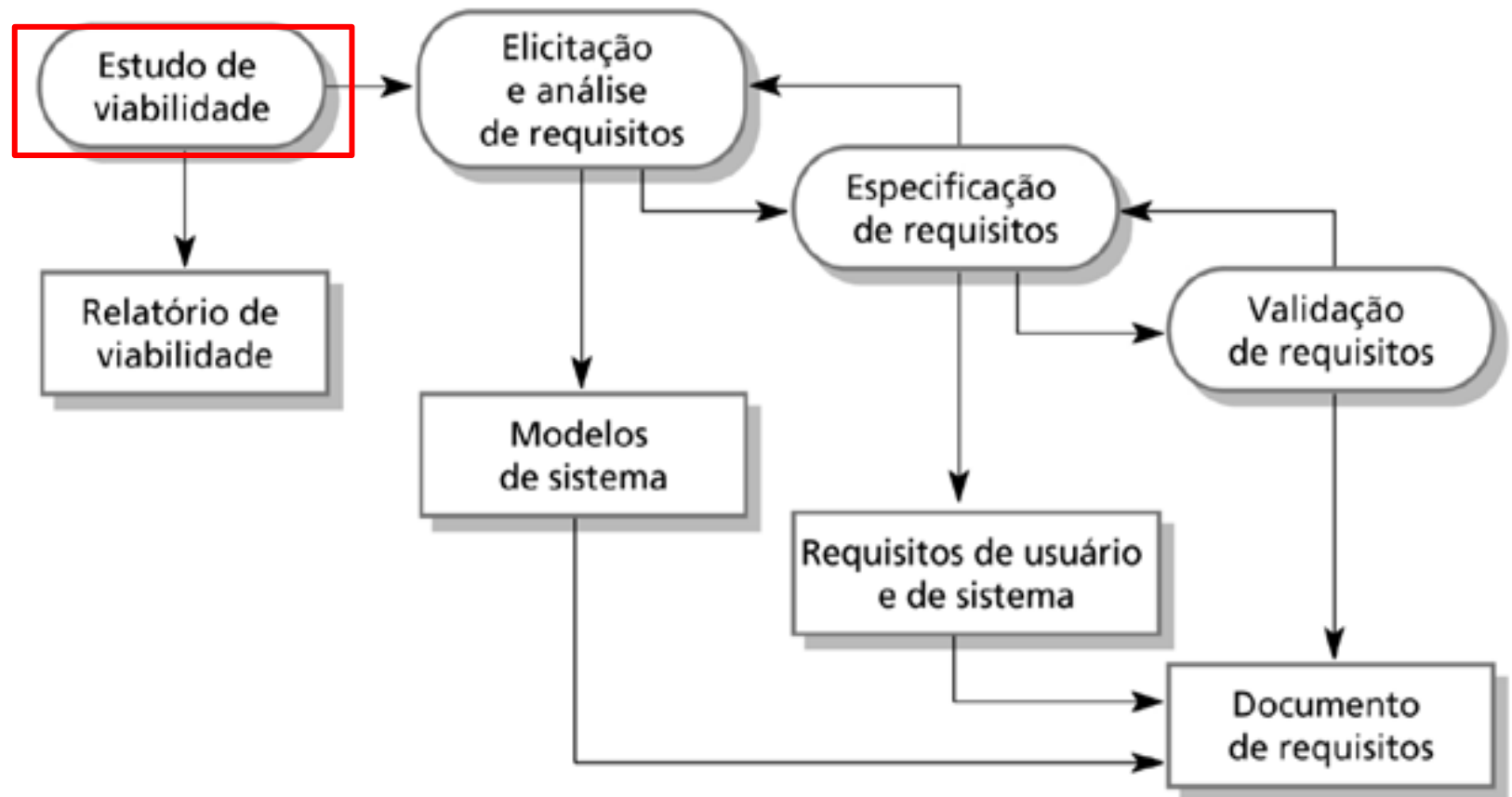
- Estabelecer uma visão comum entre o cliente e a equipe de projeto em relação aos requisitos que serão atendidos pelo projeto de software;
- Registrar e acompanhar requisitos ao longo de todo o processo de desenvolvimento;
- Documentar e controlar os requisitos alocados para estabelecer uma "baseline" para uso gerencial e da engenharia de software;
- Manter planos, artefatos e atividades de software consistentes com os requisitos alocados.

PROCESSO DE ENGENHARIA DE REQUISITOS



Deve lidar com diferentes pontos de vista, métodos, ferramentas e pessoas

PROCESSO DE ENGENHARIA DE REQUISITOS



ESTUDO DE VIABILIDADE

- Necessidade do **mercado** ou serviço potencialmente **novo**
 - Conjunto **preliminar** de requisitos de negócios
 - **Esboço** da descrição do sistema
 - **Esboço** de como o sistema pretende apoiar os processos de negócios do cliente
- **Pessoal** envolvido
 - Gerentes de departamentos onde o sistema será usado
 - Pessoal de marketing
 - Engenheiros de software familiarizados com o tipo de sistema proposto
 - Especialistas em tecnologia
 - Usuários finais do sistema

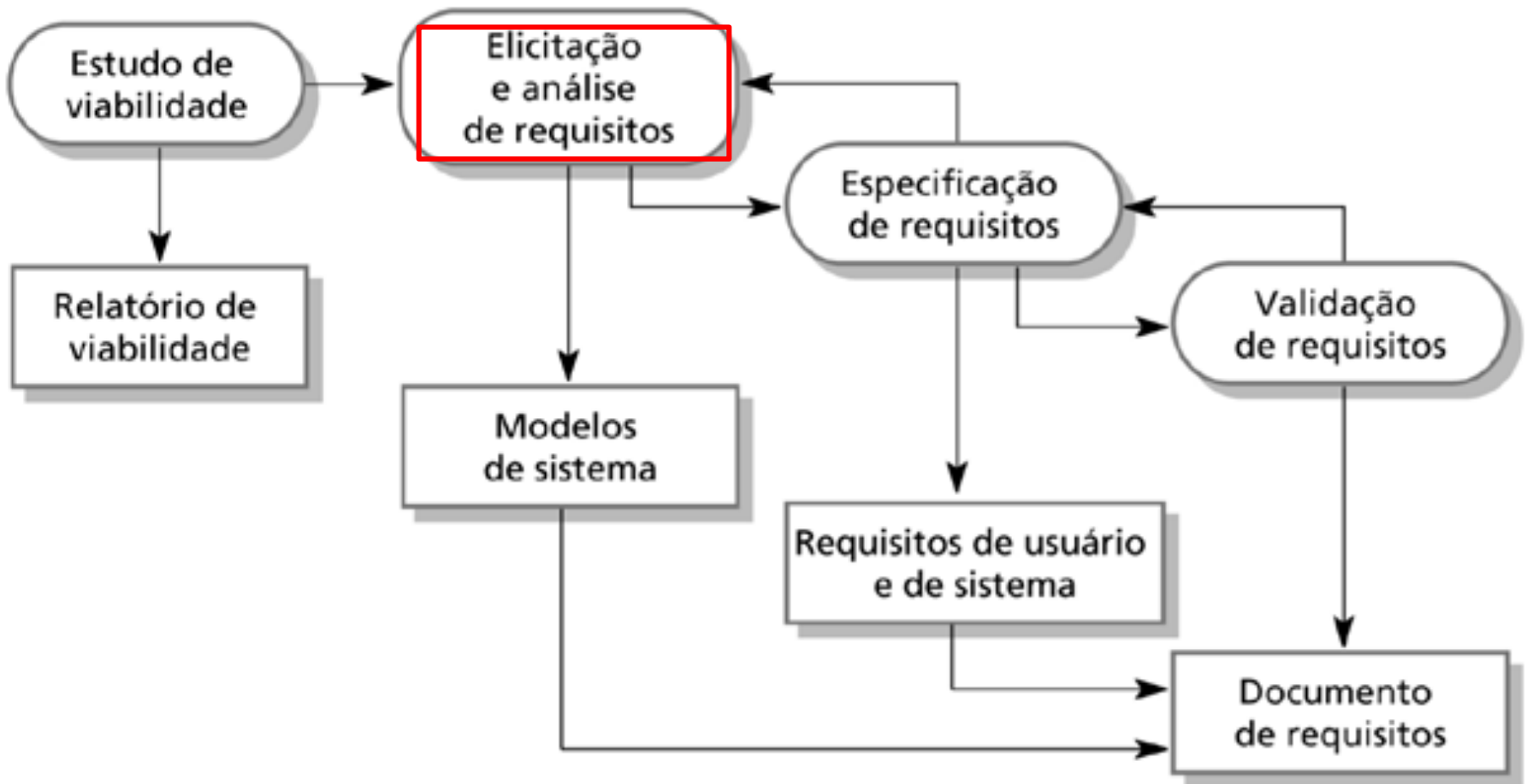
QUESTÕES USUAIS (ALGUMAS)

- Quem está por trás da solicitação deste trabalho? (cliente)
- Quem vai usar a solução? (usuário final)
- Qual será o benefício econômico com essa solução?
- A solução contribui para os objetivos gerais da organização?
- Quais os problemas com os processos atuais e como a nova solução ajudaria a resolver esses problemas?
- Qual será a contribuição direta da solução para os objetivos e requisitos da organização?
- Como a organização se comportaria se o solução não fosse implementada?
- A solução deve ser implementada com tecnologia atual e dentro das restrições definidas de custo e prazo?
- A solução pode ser integrada a sistemas já implementados?
- Você é a pessoa ideal para responder a estas questões? Suas respostas são “oficiais”?
- Alguém mais pode fornecer informações adicionais?

RESULTADO DO ESTUDO DE VIABILIDADE

- Relatório (de viabilidade) que recomenda (ou não) prosseguir com os processos de engenharia de requisitos e de desenvolvimento de sistema
 - Viabilidade econômica
 - Viabilidade tecnológica
 - Viabilidade técnica

PROCESSO DE ENGENHARIA DE REQUISITOS

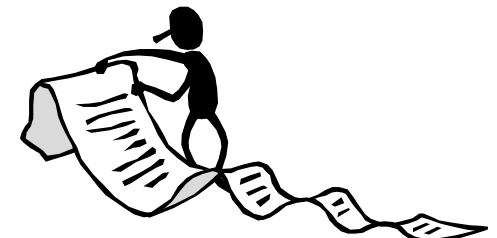


ELICITAR

- ELICITAR = Clarear + Extrair + Descobrir
 - Tornar **explícito**, obter o máximo de **informação** para o conhecimento do objeto em questão
- Há **três atividades** principais
 1. Identificação de fontes de informação
 2. Coleta de Fatos
 3. Comunicação

1. IDENTIFICAR FONTES DE INFORMAÇÃO

- Documentação do macro-sistema
 - Políticas, manuais
 - Memorandos, atas, contratos...
 - Livros sobre tema relacionado
 - Outros sistemas da empresa
 - Sistemas externos
- Atores
 - Usuários finais
 - Engenheiros de sistemas
 - Gerentes de negócios
 - Especialistas do domínio, etc.



1. IDENTIFICAR FONTES DE INFORMAÇÃO

- Importante priorizar as fontes de informação
 - Atores mais importantes
 - Documentos mais mencionados
 - Rede de comunicação entre os componentes do macro-sistema
- Problemas
 - Pessoas envolvidas não sabem o que querem do sistema
 - Conhecimento implícito das pessoas envolvidas
 - Diferentes pessoas, diferentes requisitos, diferentes expressões
 - Fatores políticos e/ou econômicos podem influenciar

2. COLETAR FATOS

- Geralmente,
 - são feitas **entrevistas** com os interessados no sistema
 - são consultados os **materiais** existentes que descrevem os objetivos e desejos da organização
 - são realizadas **pesquisas** sobre a existência de sistemas similares para posterior análise (**análise de mercado ou concorrência**)

MÉTODOS CONVENCIONAIS DE COLETA

- **Entrevistas**

- Método mais comum de coleta de dados

- **Desestruturada**

- Deseja-se explorar um problema (estágios iniciais de um tópico considerado)

- Entrevista mais informal

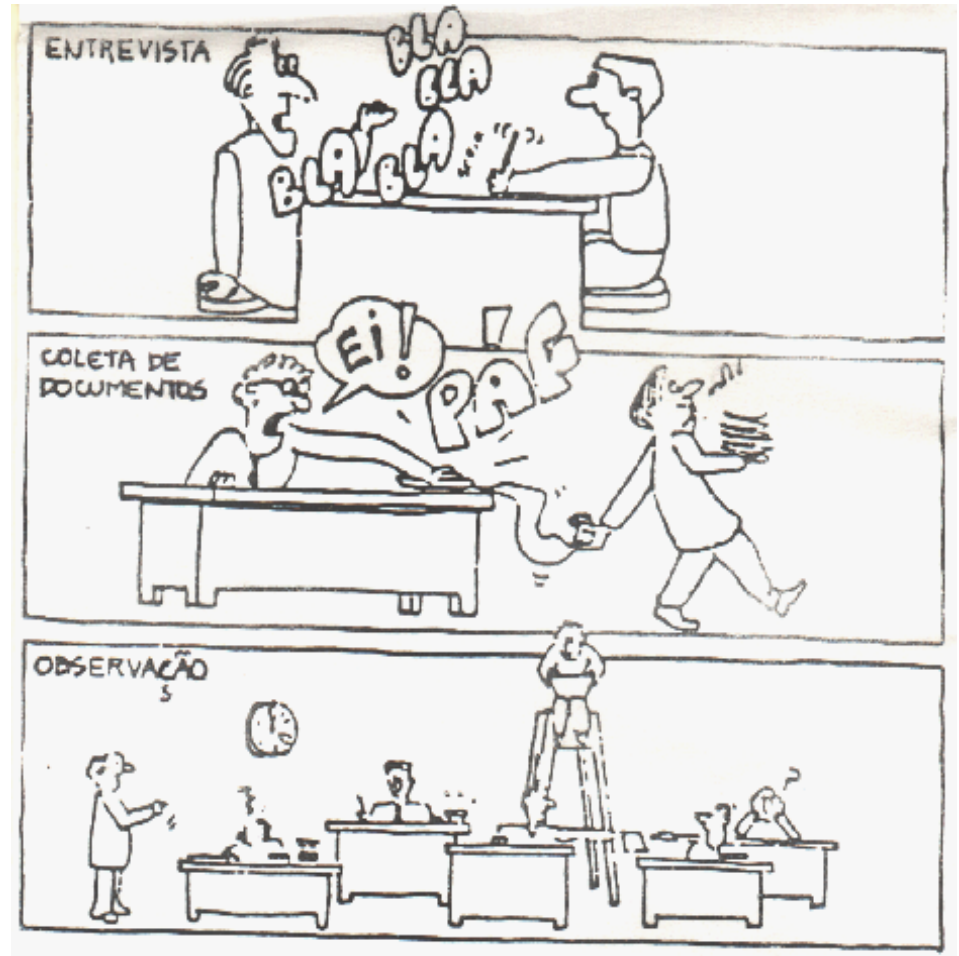
- **Estruturada**

- Deseja-se informações específicas do conteúdo e do problema

- Considerar o nível de experiência e conhecimento do entrevistado

MÉTODOS ALTERNATIVOS DE COLETA

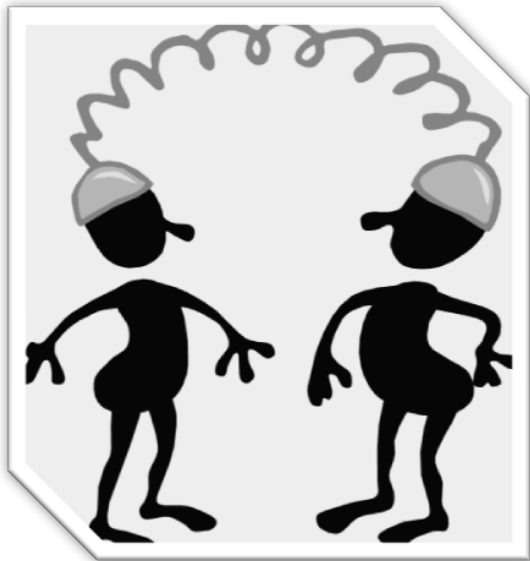
- ❑ Questionários
- ❑ Leitura de documentos
- ❑ Engenharia reversa do projeto do software
- ❑ Observação direta / estudo etnográfico
- ❑ Protótipos
- ❑ *Brainstorming*



3. COMUNICAÇÃO

- Para o sucesso da elicitação de requisitos, é fundamental que engenheiros de software se comuniquem de maneira eficaz com os clientes “**que entendem o problema**”
- Preocupar-se com
 - a forma como a informação é apresentada
 - o estabelecimento de contextos comuns
 - linguagem (por exemplo, tecnicista)
 - nível de abstração (geral, específico)
 - Feedback

PROBLEMAS DA ELICITAÇÃO



Cliente

Analista



ANALISTA X USUÁRIO

ELICITAÇÃO E ANÁLISE DE REQUISITOS

- **Categorizar** e **organizar** os requisitos coletados em subconjuntos
- **Relacionar** os requisitos coletados
- **Priorizar** os requisitos coletados segundo a necessidade dos clientes/usuários
- Construir **modelos** que explicitem o entendimento **colaborativo** do sistema sob os diferentes pontos de vista envolvidos (clientes, usuários finais, arquitetos de sistema, etc.)

Requisitos Funcionais e Não Funcionais

- Requisitos funcionais
 - O sistema deve fornecer declarações de serviços, como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações. ✓ Pode explicitar o que o sistema não deve fazer.
- Requisitos não-funcionais
 - Restrições aos serviços ou funções oferecidas pelo sistema, tais como restrições de tempo, restrições no processo de desenvolvimento, padrões. ✓ Muitas vezes se aplica ao sistema como um todo ao invés de características individuais ou serviços.

ELICITAÇÃO E ANÁLISE DE REQUISITOS

Modelo de Domínio

- O Modelo de Domínio é a **representação visual** (em UML) das **classes conceituais do mundo real em um domínio de problema**.
- É um artefato que **representa o domínio do problema**, portanto, não é utilizado para modelar a arquitetura de software (diagrama de classes de projeto), pois esta, embora inicialmente derivada do modelo conceitual pertence ao domínio da solução.
- O modelo pode **evoluir durante as iterações**. (Melhor entendimento do negócio e/ou requisitos).

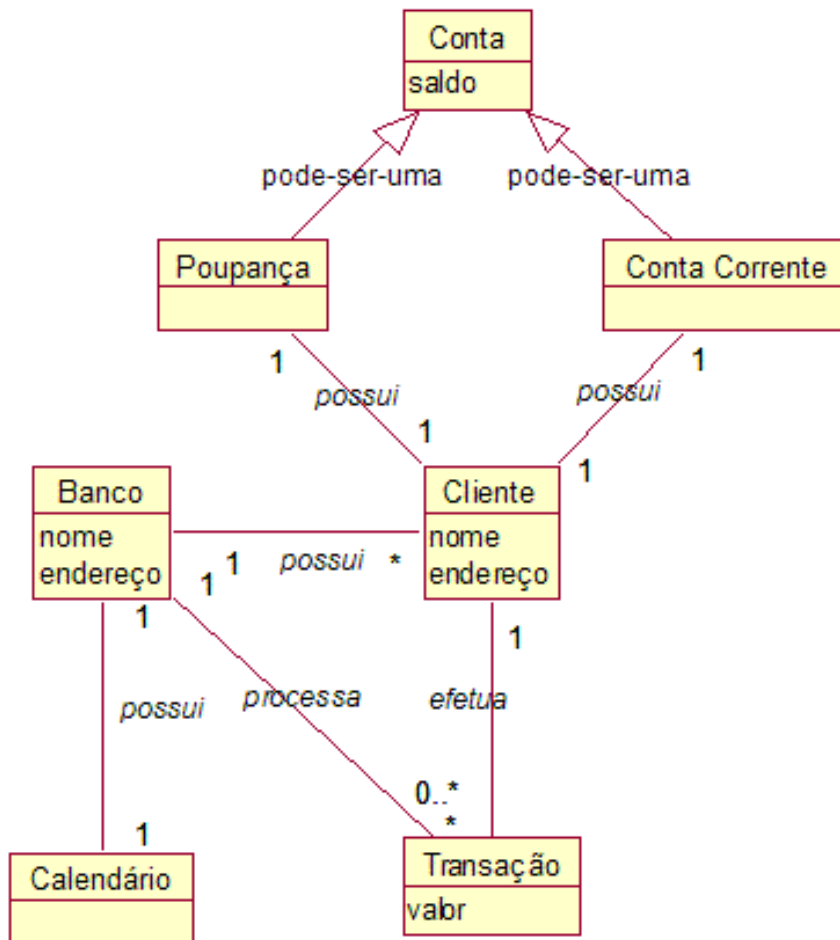
ELICITAÇÃO E ANÁLISE DE REQUISITOS

Modelo de Domínio - Diretrizes

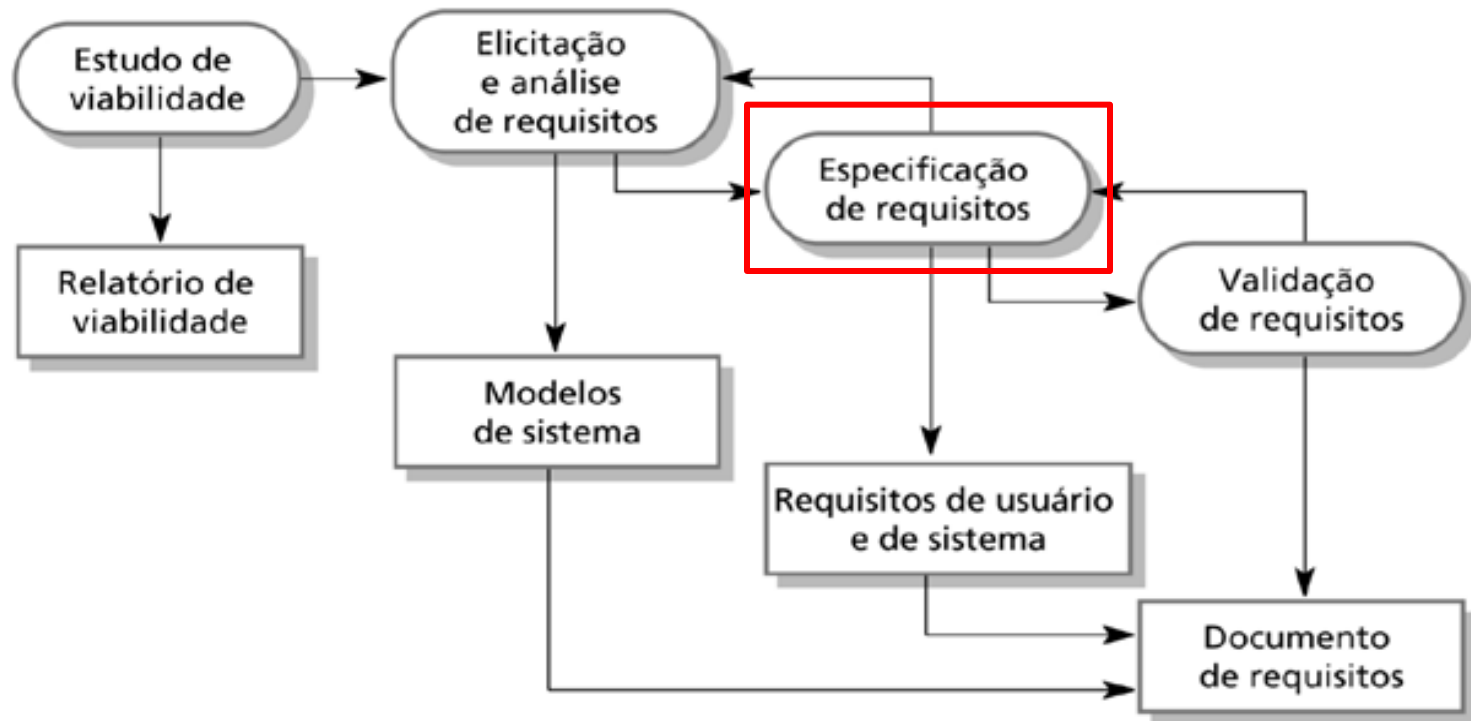
1. Liste classes conceituais candidatas: Através de conversas com usuários, estudos, levantamento de requisitos, etc.
2. Desenhe-as em um modelo de domínio
3. Acrescente as associações necessárias entre as classes conceituais
4. Acrescente os atributos necessários para entendimento dos conceitos

ELICITAÇÃO E ANÁLISE DE REQUISITOS

Modelo de Domínio - Exemplo



PROCESSO DE ENGENHARIA DE REQUISITOS



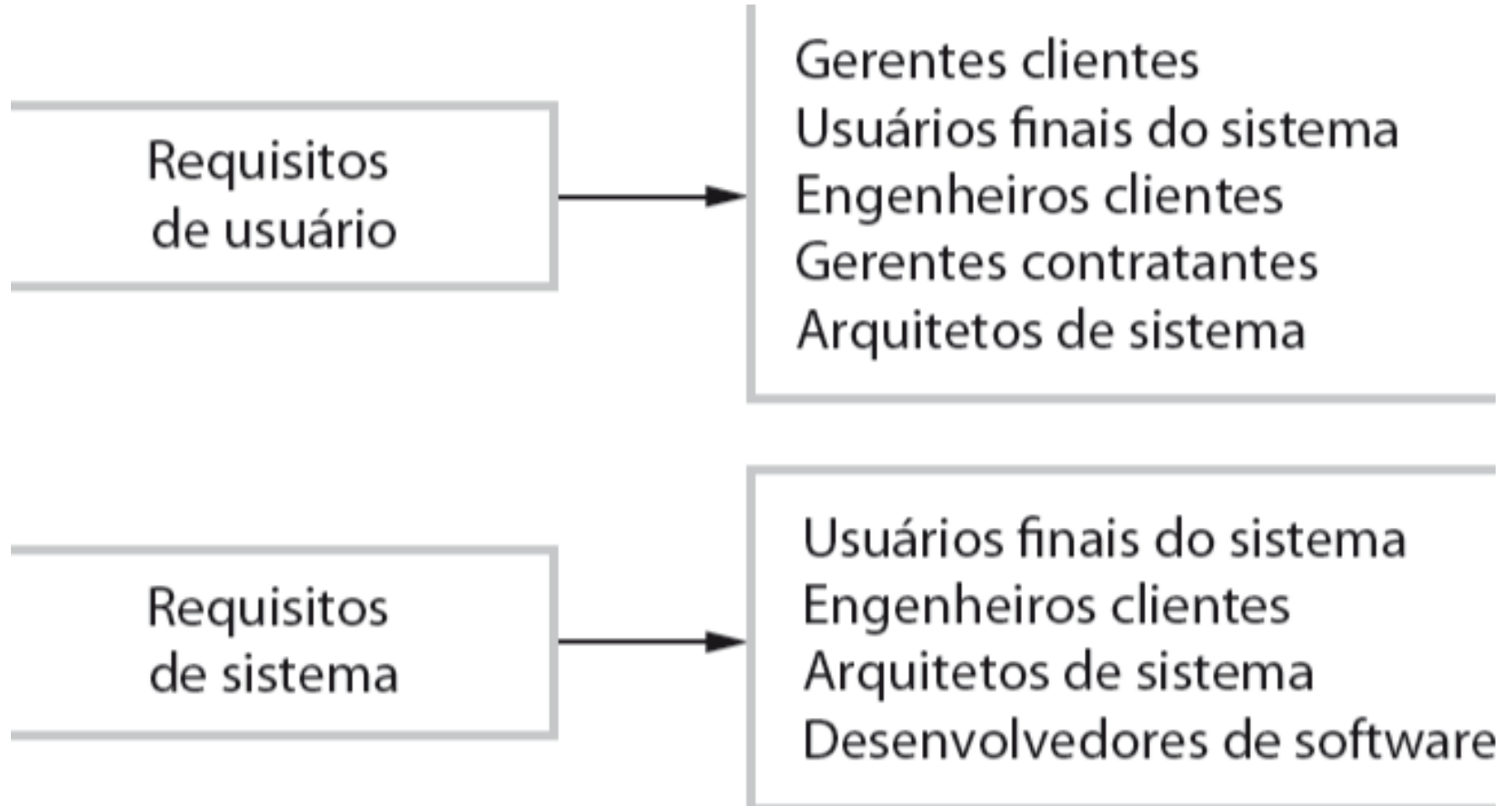
DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS

- É a declaração oficial (contrato) do que é requisitado pelos desenvolvedores do sistema
- Deve incluir ambos, uma definição dos requisitos de usuário e uma especificação dos requisitos de sistema
- NÃO é um documento de projeto! Apenas descreve **O QUE** o sistema deve fazer ao invés de **COMO** deve ser feito

Tipos de Requisitos

- Requisitos de usuário
 - Declarações em linguagem natural com diagramas dos serviços que o sistema deverá fornecer e suas restrições operacionais. Escrito para os clientes.
- Requisitos de sistema
 - Um documento estruturado estabelecendo descrições detalhadas das funções do sistema, serviços e restrições operacionais. Define o que deve ser implementado assim, pode ser parte de um contrato entre o cliente e o empreiteiro.

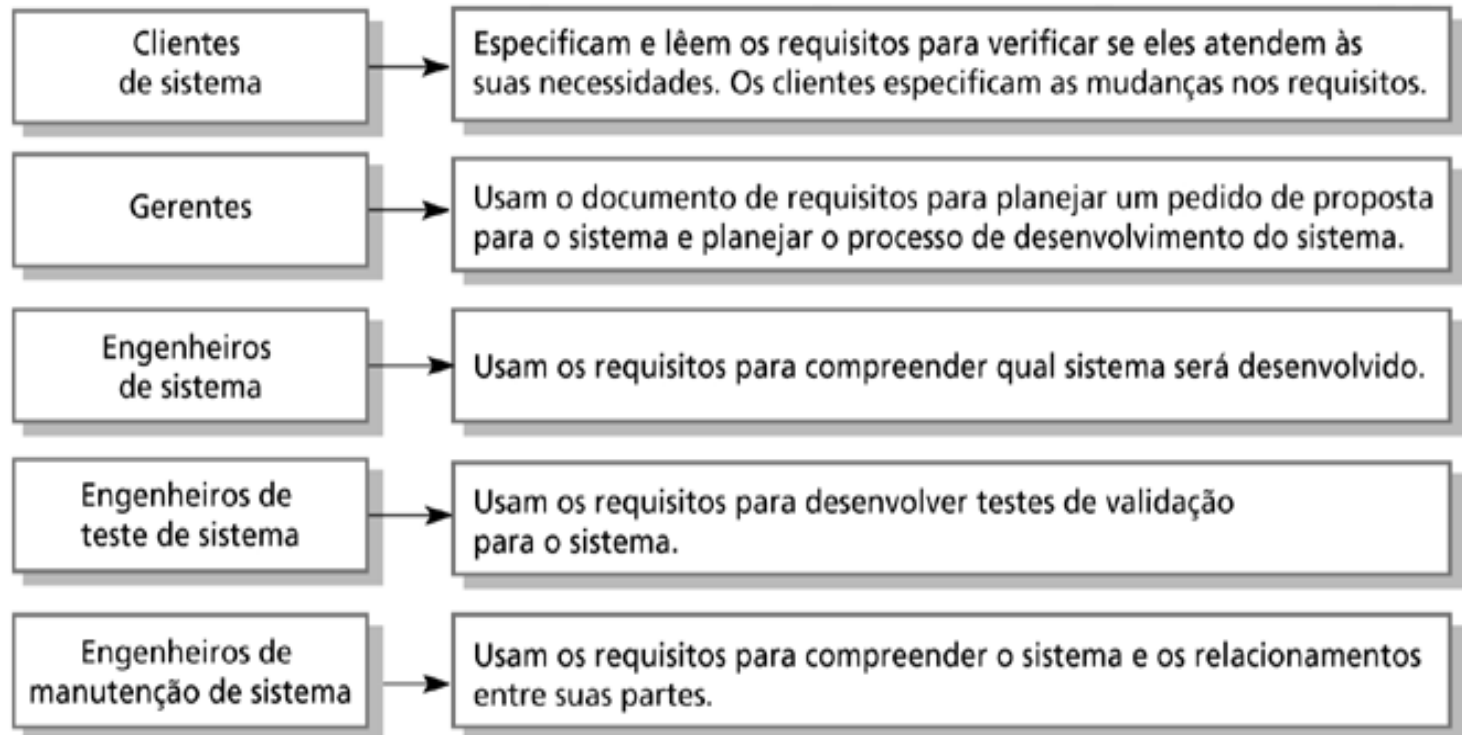
Leitores de diferentes tipos de requisitos



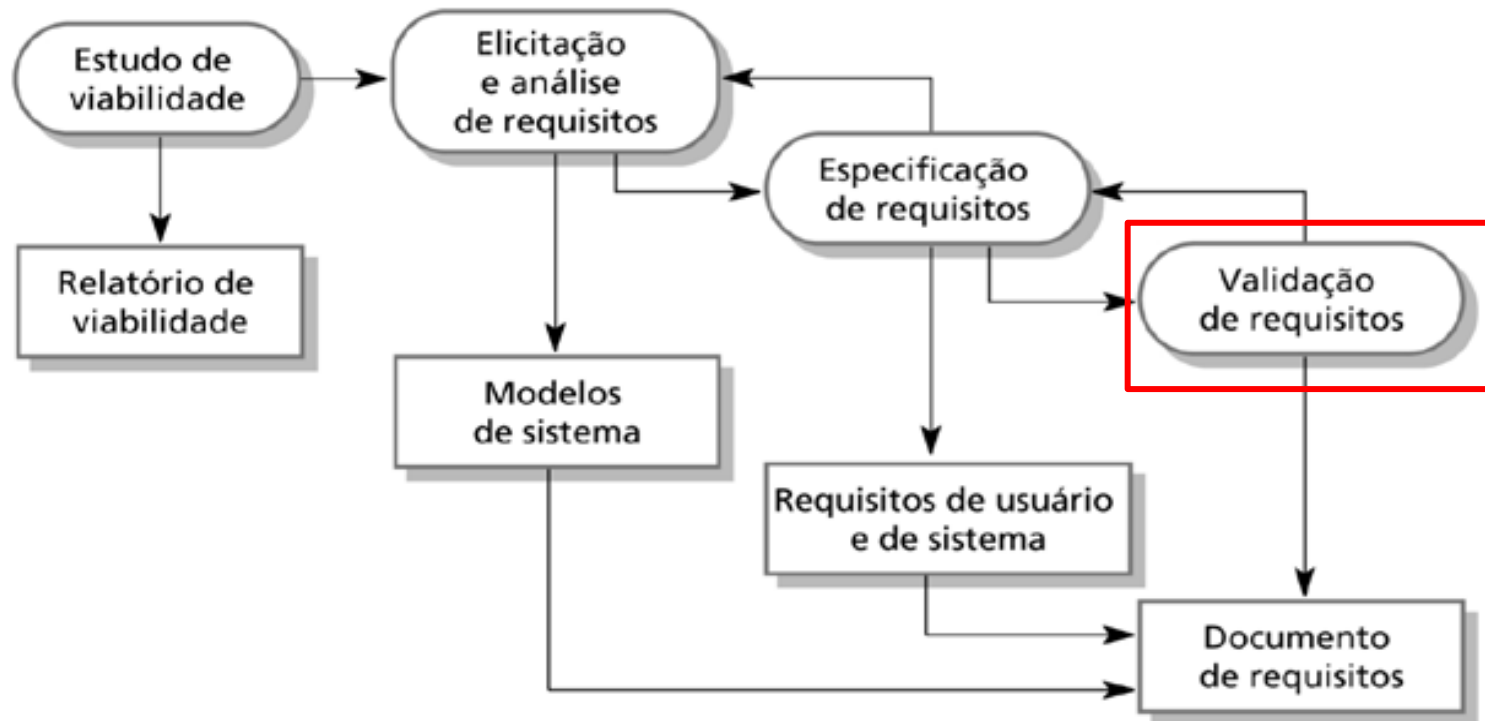
USUÁRIOS DO DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS

Figura 6.5

Usuários de um documento de requisitos.



PROCESSO DE ENGENHARIA DE REQUISITOS



VALIDAÇÃO DE REQUISITOS

☐ Verificação de **validade** de requisitos

- O sistema fornece as funções que melhor apoiam as necessidades do cliente?
- Cada requisito tem uma fonte de informação atribuída?
- Padrões de requisitos foram usados para simplificar o modelo de requisitos?

☐ Verificação de **consistência**

- Existe algum tipo de conflito de requisitos?
- Cada requisito é não ambíguo?

• Verificação de **completeza**

- Todas as funções requisitadas pelo cliente foram incluídas?

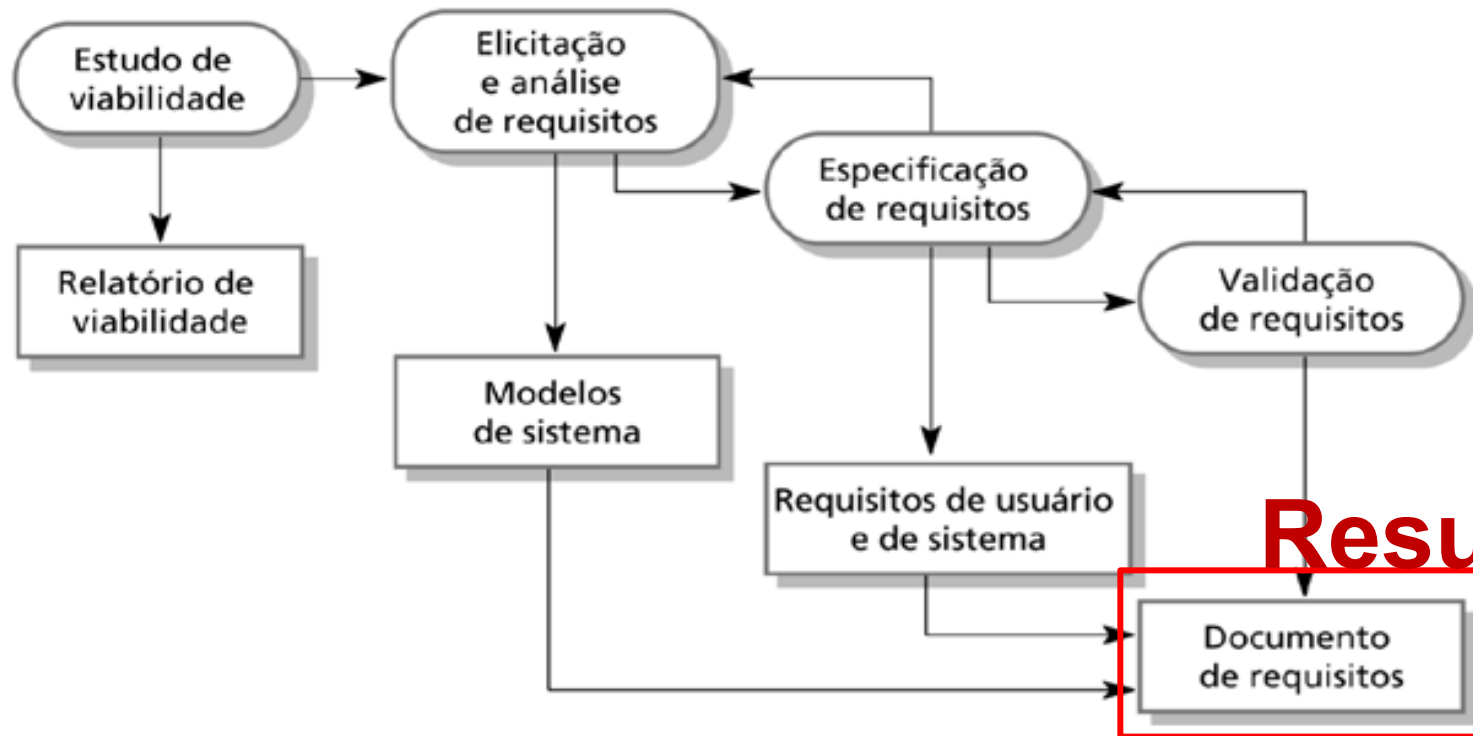
• Verificação de **realismo**

- Os requisitos podem ser implementados com o orçamento e a tecnologia disponíveis?

• Facilidade de **verificação**

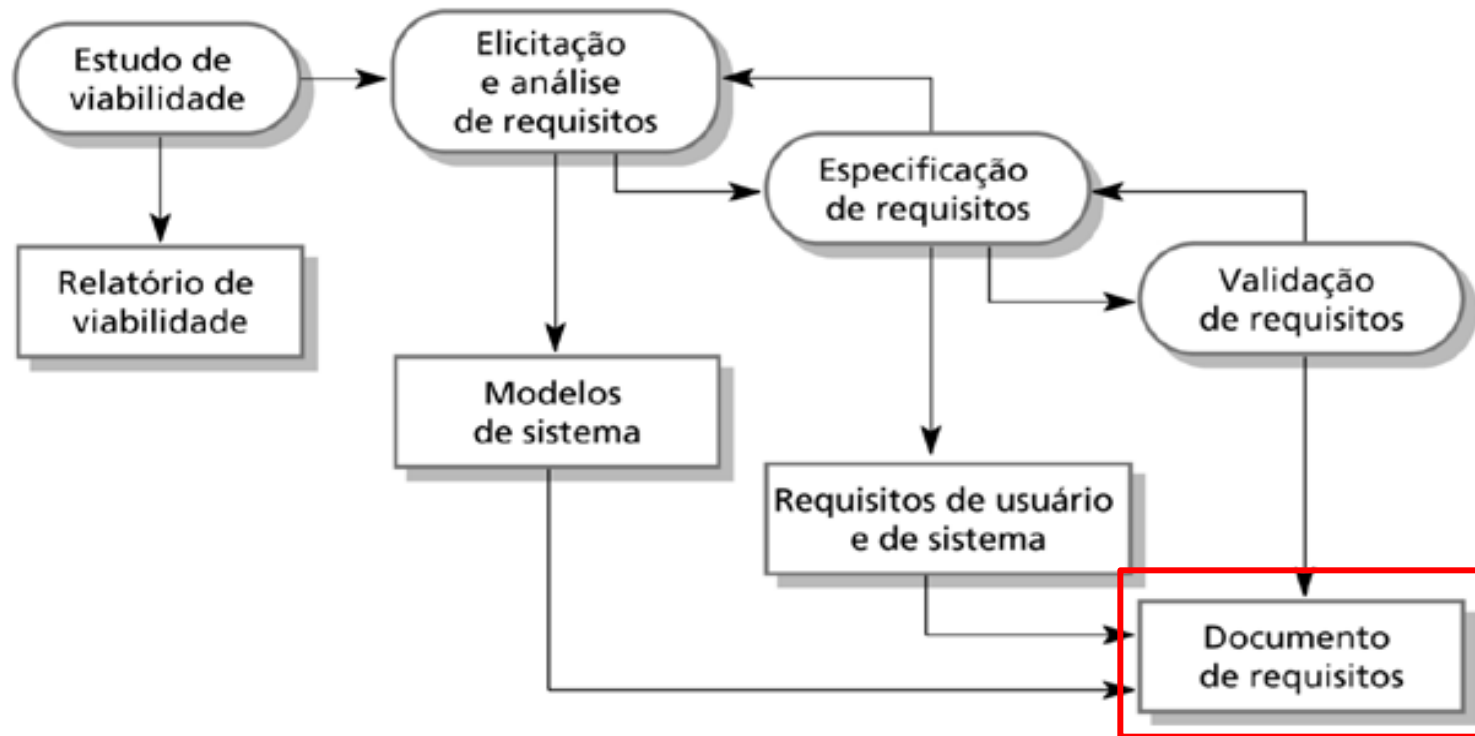
- Os requisitos podem ser testados quando estiver implementado?

PROCESSO DE ENGENHARIA DE REQUISITOS



Resultado

PROCESSO DE ENGENHARIA DE REQUISITOS



~~GERENCIAMENTO DE REQUISITOS~~

GERENCIAMENTO DE REQUISITOS

- Processo de **gerenciamento de mudanças de requisitos** que ocorre durante o processo de engenharia de requisitos e o desenvolvimento de sistema
- Requisitos são, inevitavelmente, incompletos e inconsistentes
 - **Novos requisitos** surgem durante o processo à medida que as necessidades de negócio mudam e/ou uma melhor compreensão do sistema é desenvolvida
 - Os **diferentes pontos de vista** têm requisitos diferentes, e estes são, frequentemente, contraditórios
 - Mudanças em requisitos podem advir também de mudanças **técnicas** e **tecnológicas**, situações **financeiras**

ETAPAS DO GERENCIAMENTO DE REQUISITOS

1. Identificação de requisitos

- Definir identificador **único** para cada requisito

2. Controle e rastreamento de requisitos

- Processo seguido quando da análise de uma **mudança** de requisitos
- Importância de **políticas de rastreabilidade** para definir a quantidade de informações que é mantida sobre os relacionamentos entre requisitos (**tabelas de rastreamento**)

ETAPAS DO GERENCIAMENTO DE REQUISITOS

Tabelas de Rastreamento

- Tabela de rastreamento de características
 - Entre requisitos e características do sistema
- Tabela de rastreamento de fontes
 - Entre requisitos e fontes de informação
- Tabela de rastreamento de dependência
 - Dependência entre requisitos
- Tabela de rastreamento de projeto
 - Entre requisitos e módulos de projeto

ETAPAS DO GERENCIAMENTO DE REQUISITOS

Tabelas de Rastreamento

ID de requisito	1.1	1.2	1.3	2.1	2.2	2.3	3.1	3.2
1.1		D	R					
1.2			D			R		D
1.3	R			R				
2.1			R		D			D
2.2								D
2.3		R		D				
3.1								R
3.2							R	

D= requisito da linha depende do requisito da coluna

R= existe algum relacionamento entre os requisitos

PROPOSTA DE UMA ESTRUTURA GENÉRICA: PADRÃO DE DOCUMENTAÇÃO IEEE 830-1998

1. INTRODUÇÃO

1.1 Propósito do documento de requisitos

1.2 Escopo do produto

1.3 Definições, acrônimos e abreviações

1.4 Referências

1.5 Visão geral do restante do documento

2. DESCRIÇÃO GERAL

2.1 Perspectiva do produto

2.2 Funções do produto

2.3 Características do usuário

2.4 Restrições gerais

2.5 Suposições e dependências

3. REQUISITOS ESPECÍFICOS

- Funcionais

- Não-funcionais

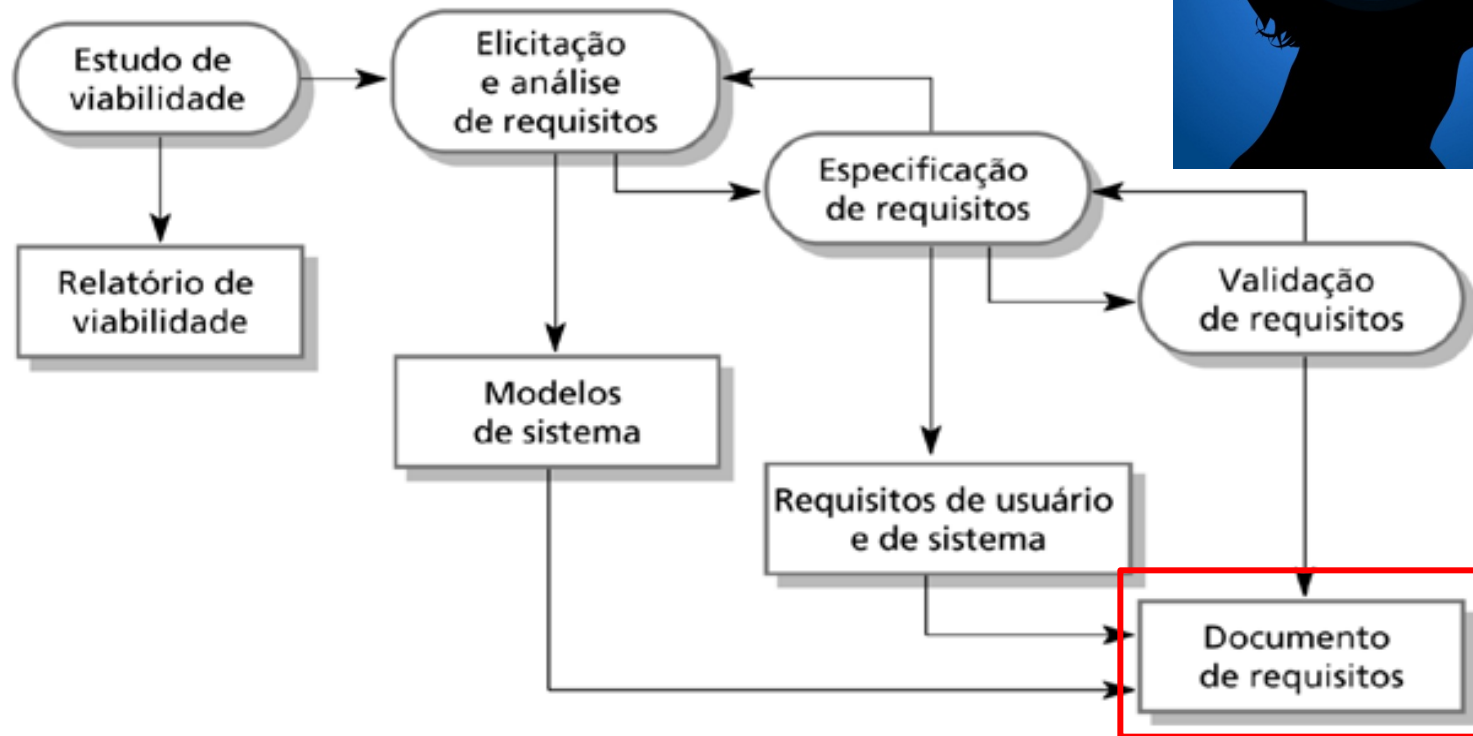
- De interface

OBS: requisitos podem documentar interfaces externas, descrever funcionalidade e desempenho do sistema, especificar requisitos lógicos de banco de dados, restrições de projeto, propriedades emergentes do sistema e características de qualidade

4. APÊNDICES

5. ÍNDICE

FICOU CLARO?



~~GERENCIAMENTO DE REQUISITOS~~