

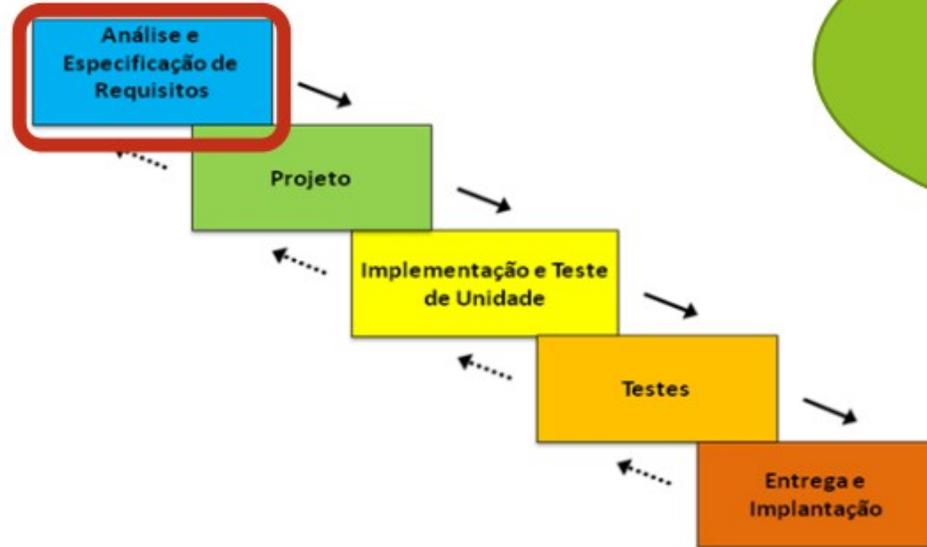
# Modelos Métodos e Técnicas de Engenharia de Software

Prof. Sônia A Santana



# Engenharia de Requisitos

# Introdução



Em um desenvolvimento de software, a primeira etapa a ser feita é capturar os requisitos que o sistema a ser desenvolvido tem de tratar.

- Um entendimento dos requisitos do software é essencial para o sucesso de um projeto de desenvolvimento de software.

# Importância da Etapa de Requisitos

- Identificar as necessidades dos usuários.
- Verificar a viabilidade de implementar estas necessidades.
- Distribuir as funções do sistema entre as pessoas, o hardware, o software e outros elementos do sistema.
- Criar um modelo do sistema que será utilizado nas fases de desenvolvimento seguintes.

# O que são Requisitos?

Requisitos são descrições dos serviços que devem ser providos pelo sistema e de suas restrições operacionais (SOMMERVILLE, 2007).

Um requisito é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir seus objetivos (PFLEEGER, 2004).

Um requisito é alguma coisa que o produto tem de fazer ou uma qualidade que ele precisa apresentar (ROBERTSON; ROBERTSON, 2006).

# Tipos de Requisitos

Descrições dos serviços fornecidos pelo sistema e as suas restrições operacionais



# Tipos de Requisitos: Funcionais

- **Funcionais**: são declarações de serviços que o sistema deve prover, descrevendo **o que o sistema deve fazer** (SOMMERVILLE, 2007). Um requisito funcional descreve uma interação entre o sistema e o seu ambiente (PFLEEGER, 2004), podendo descrever, ainda, como o sistema deve reagir a entradas específicas, como o sistema deve se comportar em situações específicas e o que o sistema não deve fazer (SOMMERVILLE, 2007).
  - **Ex.:** O sistema deve registrar locações, indicando o cliente, os itens locados, a data da locação, a data de devolução e o valor da locação.

# Exemplo Requisitos Funcionais

## Exemplo: Requisitos Funcionais do Portal do BB.

The screenshot displays the BB portal interface. At the top, there is a search bar labeled "Ache Fácil!" with the placeholder text "Digite transação desejada" and a search icon. To the right of the search bar are buttons for "Outra Conta" and "Sair". Below the search bar is a grid of service categories, each with a blue arrow icon: "Conta-corrente e Consultas", "Crédito", "Cartões", "Investimentos e Previdência", "Seguros", "Capitalização e Consórcios", "Serviços e Segurança", and "Ponto pra Você e BOMPRATODOS". To the right of this grid is a vertical list of user-related links: "Minha Página", "Minhas Mensagens", "Minhas Finanças", "Meus Limites", "Minha Segurança", and "Novidades". Below these elements is a section titled "Acesso Rápido" (Quick Access) with a blue header. This section contains several columns of links, each starting with a blue arrow icon:

- Extratos**
  - > Conta-corrente
  - > Poupança
  - > Cartão de Crédito
- Investimento**
  - > Fundo Premiado
- Cheque Especial**
  - > Adesão
- Pagamentos**
  - > Boletos bancários e títulos
  - > Água/Luz/Telefone/Gás
  - > Fatura Ourocard
- Seguros de Veículos**
  - > Simulação, Contratação e Renovação
- Recarga de Telefone**
  - > Recarga de celular pré-pago
- Transferências**
  - > Entre contas-correntes BB
  - > Conta-corrente para poupança
  - > Poupança para conta-corrente
  - > DOC - A débito de conta corrente
  - > TED - A débito de conta corrente
- Saque sem cartão**
  - > Solicitação de saque
- Soluções em Crédito**
  - > Crédito novo
  - > Renove seu crédito
- Comprovantes**
  - > Pagamentos
  - > Transferências
  - > Débitos automáticos
  - > Todos
- Débito Direto Autorizado**
  - > Consulta Boletos DDA

# Exercício de Fixação

Exemplo: Requisitos Funcionais do Facebook.

The screenshot displays the Facebook user interface. At the top, the Facebook logo is on the left, and a search bar with the text "Pesquise pessoas, locais e coisas" is on the right. Below the search bar, the user profile for "Vinicius Silva Andrade" is visible, including a profile picture and an "Editar perfil" link. A navigation menu on the left lists "FAVORITOS" (Feed de notícias, Mensagens, Eventos, Fotos) and "AMIGOS" (Melhores amigos, Família, Centro Universitár..., CEVA Logistics, IBM, IBM, Plan B (magazine), Takenet, Web Consult). The main content area shows a status update form with the placeholder text "No que você está pensando?". Below the form are icons for adding people, location, photos, and videos, along with a privacy dropdown set to "Amigos exceto conh..." and a "Publicar" button. A post by "Claudiney Barbosa Veloso" is shown below, featuring a photo of a person on a bicycle with a speedometer displaying "00 km/h". The post text reads: "Trilha dada, trilha cumprida... Esse foi o total da pedalada de hoje... Valeu galera!!!". A "CLASSIFICAR" dropdown menu is visible to the right of the post.

# Listas de Requisitos Funcionais

## Requisitos Funcionais

ID	Descrição	Priorid.	Reg. Negó
RF001	Cadastrar matérias-primas	Essenc.	
RF002	Cadastrar materiais de consumo	Essenc.	
RF003	Cadastrar fornecedores	Essenc.	
RF004	Emitir relatório de estoque de materia-prima	Impor.	RN01
RF005	Emitir relatório de estoque de material de consumo	Impor.	RN01
RF006	Cadastrar cotações	Essenc.	
RF007	Emitir solicitação de compra automaticamente	Essenc.	RN02
RF008	Ordenar as cotações de um mesmo produto	Desej.	RN03
RF009	Registrar justificativa de compra	Essenc.	RN04
	...		

# Tipos de Requisitos: Não Funcionais

- **Não Funcionais:** descrevem **restrições** sobre os serviços ou funções oferecidos pelo sistema (SOMMERVILLE, 2007), as quais limitam as opções para criar uma solução para o problema (PFLEEGER, 2004). Neste sentido, os requisitos não funcionais são muito importantes para a fase de projeto (design), servindo como base para a tomada de decisões nessa fase.
  - **Ex.:** A consulta ao acervo da locadora deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado. (requisito de portabilidade)

# Observação importante

- Os requisitos não funcionais têm origem nas necessidades dos usuários, em restrições de orçamento, em políticas organizacionais, em necessidades de interoperabilidade com outros sistemas de software ou hardware ou em fatores externos como regulamentos e legislações (SOMMERVILLE, 2007).
- Assim, os requisitos não funcionais podem ser classificados quanto à sua origem.

# Classificação dos Requisitos Não Funcionais

- Quanto à origem:
  - Requisitos de Produto;
  - Requisitos Organizacionais;
  - Requisitos Externos.

# Classificação do Requisitos Não Funcionais

- **Requisitos de produto:** especificam o comportamento do produto (sistema). Referem-se a atributos de qualidade que o sistema deve apresentar, tais como confiabilidade, usabilidade, eficiência, portabilidade, manutenibilidade e segurança.

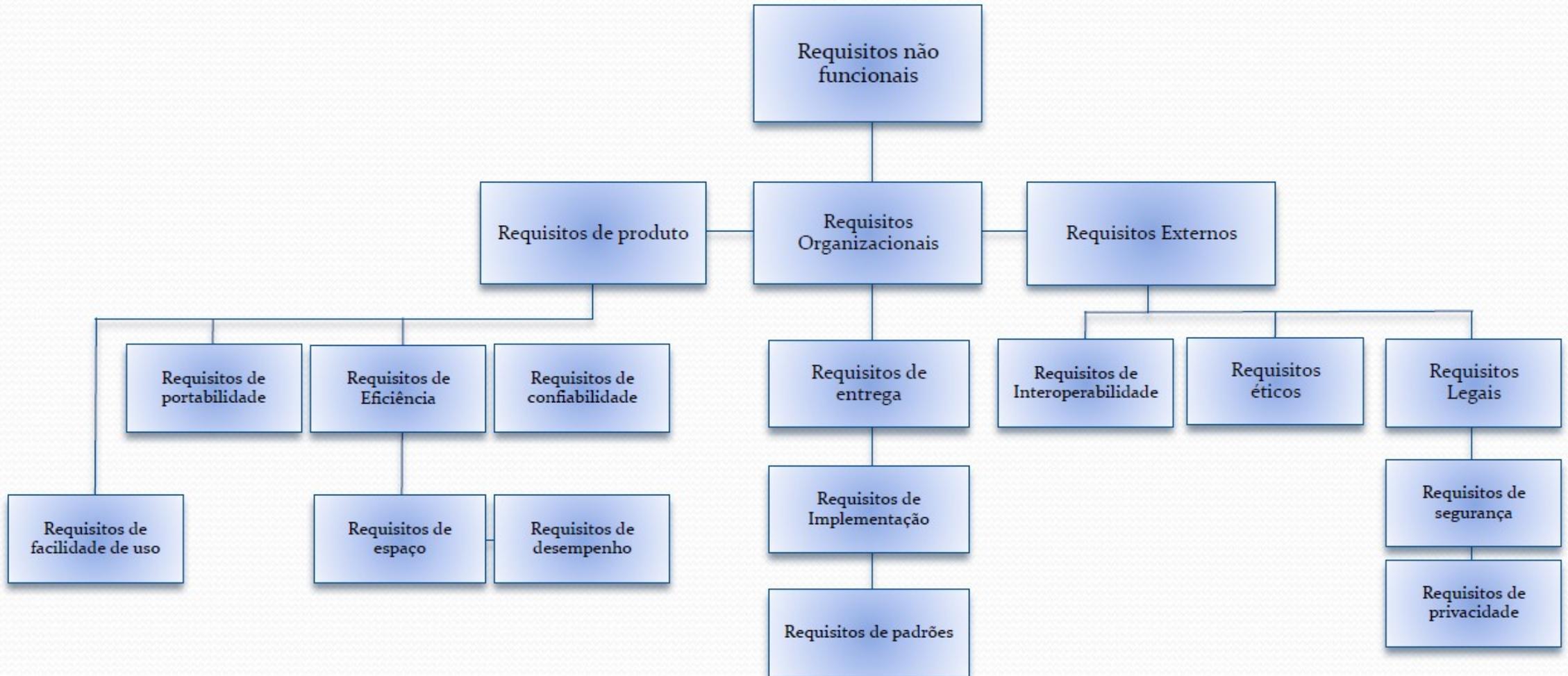
# Classificação do Requisitos Não Funcionais

- **Requisitos organizacionais:** são derivados de metas, políticas e procedimentos das organizações do cliente e do desenvolvedor.
- Incluem requisitos de processo (padrões de processo e modelos de documentos que devem ser usados), requisitos de implementação (tal como a linguagem de programação a ser adotada), restrições de entrega (tempo para chegar ao mercado - time to market, restrições de cronograma etc.), restrições orçamentárias (custo, custo-benefício) etc.

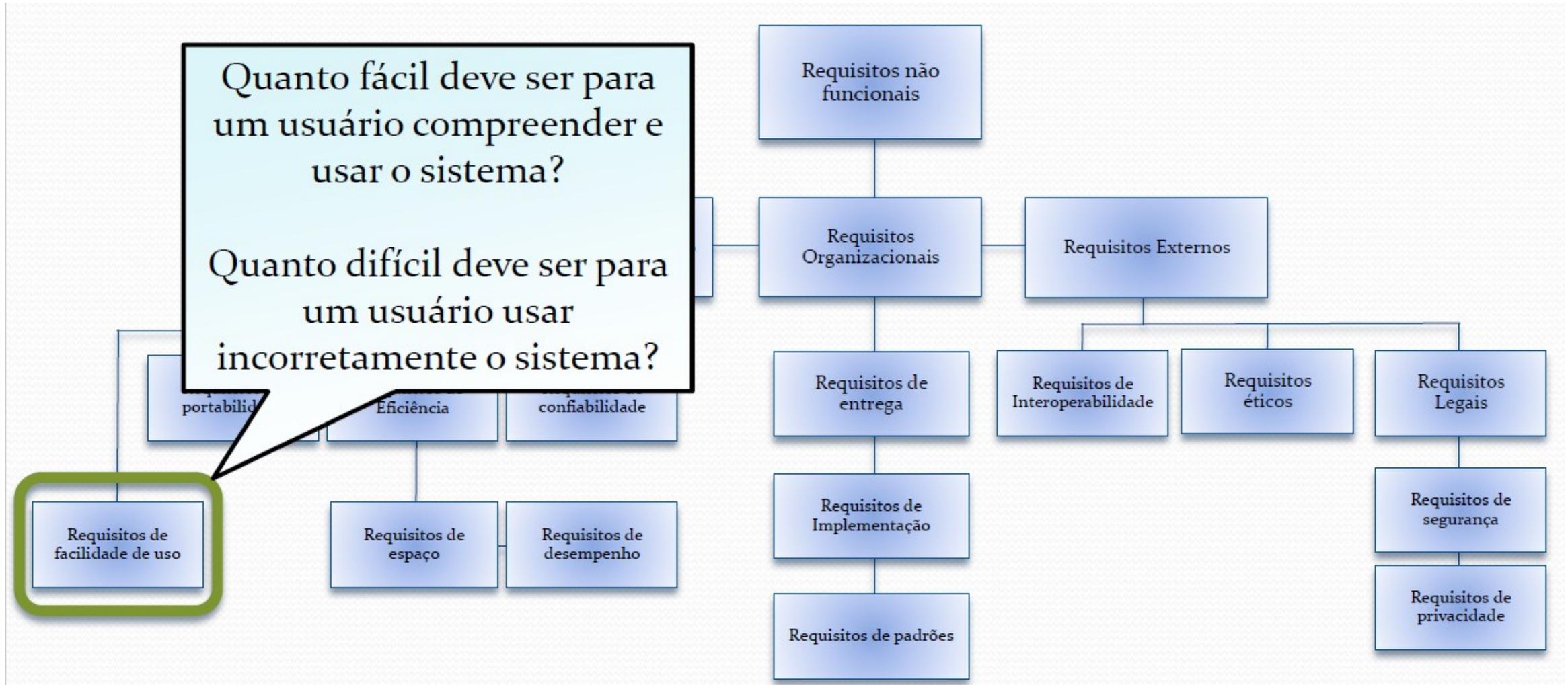
# Classificação do Requisitos Não Funcionais

- **Requisitos externos:** referem-se a todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento.
- Podem incluir requisitos de interoperabilidade com sistemas de outras organizações, requisitos legais (tais como requisitos de privacidade) e requisitos éticos.

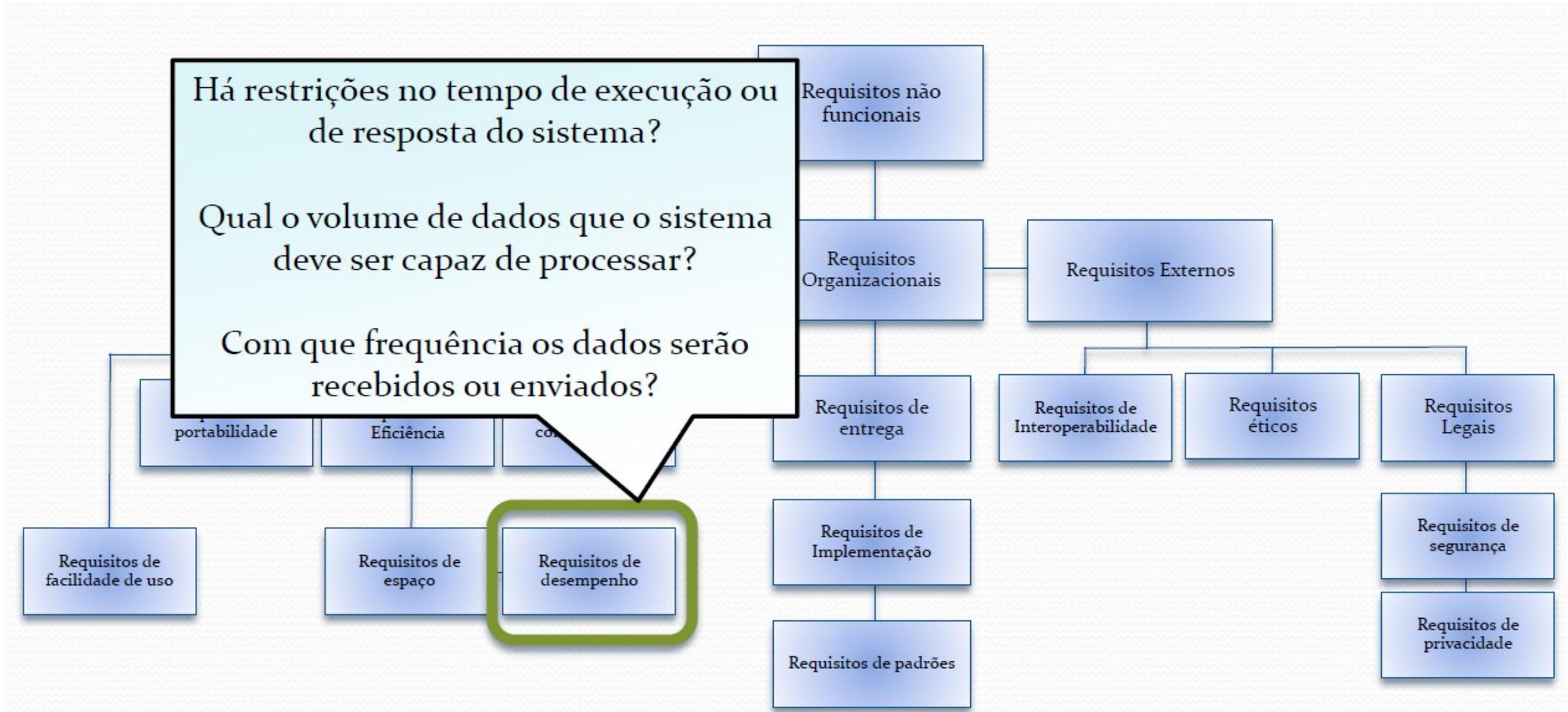
# Classificação do Requisitos Não Funcionais



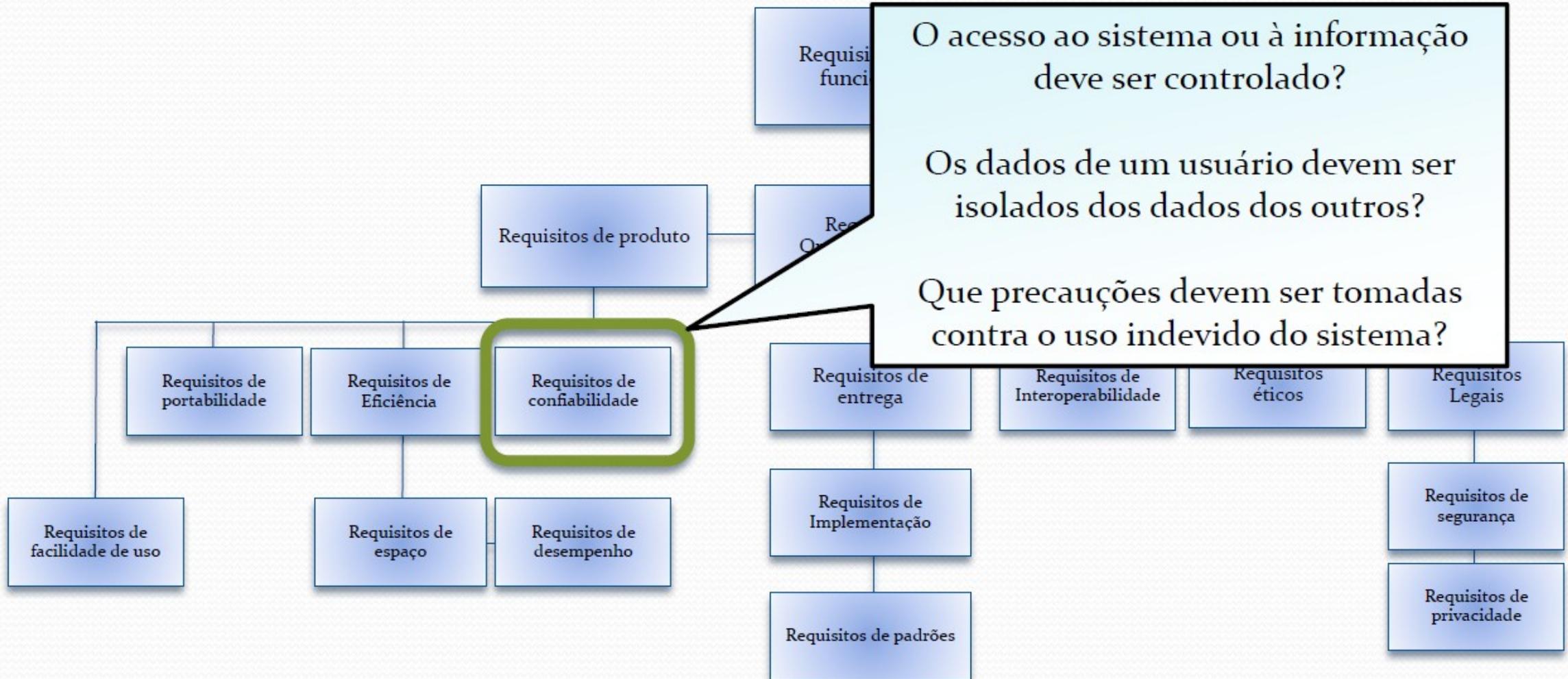
# Classificação do Requisitos Não Funcionais



# Classificação do Requisitos Não Funcionais



# Classificação do Requisitos Não Funcionais



# Exemplos de Requisitos Não Funcionais

- **Desempenho**: informar o saldo em menos de 5 segundos
- **Disponibilidade**: estar no ar 99.99% do tempo
- **Tolerância a falhas**: continuar operando se o centro de dados cair
- **Segurança**: criptografar dados trocados com as agências
- **Privacidade**: não disponibilizar dados de clientes para terceiros
- **Interoperabilidade**: integrar se com os sistemas do BACEN
- **Capacidade**: armazenar dados de 1 milhão de clientes
- **Usabilidade**: ter uma versão para deficientes visuais

# Requisitos Não Funcionais x Métricas

- Requisitos Não-Funcionais podem ser muito difíceis de serem declarados precisamente.
  - **Requisito Não-Funcional Verificável**
    - Declaração que usa alguma **métrica** que possa ser objetivamente testada.
    - **Exemplo:**
      - Controladores experientes devem ser capazes de usar todas as funções do sistema depois de duas horas de treinamento. Depois desse treinamento, o número médio de erros feito por um usuário experiente não deve exceder dois erros por dia.

# Requisitos Não Funcionais e Métricas

<p><b><i>Velocidade</i></b></p> <ul style="list-style-type: none"><li>- Transações processadas por segundo</li><li>- Tempo de resposta ao usuário</li></ul>	<p><b><i>Confiabilidade</i></b></p> <ul style="list-style-type: none"><li>- Tempo médio para falhar</li><li>- Taxa de ocorrência de falhas.</li><li>- Probabilidade de indisponibilidade.</li></ul>
<p><b><i>Tamanho</i></b></p> <ul style="list-style-type: none"><li>- Tamanho em bytes do software final</li><li>- Memória</li></ul>	<p><b><i>Robustez</i></b></p> <ul style="list-style-type: none"><li>- Tempo de reinício após falha</li><li>- Porcentual de eventos que causam falhas</li></ul>
<p><b><i>Facilidade de uso</i></b></p> <ul style="list-style-type: none"><li>- Tempo de treinamento necessário</li><li>- Número de telas de ajuda</li></ul>	<p><b><i>Portabilidade</i></b></p> <ul style="list-style-type: none"><li>- Número de ambientes operacionais nos quais o sistema pode rodar.</li></ul>

# Lista de Requisitos Não Funcionais

## Requisitos não Funcionais

ID	Descrição	Priorid.	Origem
RNF001	Integração com leitor de código de barras	Essencial	Externa (Interoperabilidade)
RNF002	Atualização do estoque na entrada de mercadoria deve ser < 1s	Importante	Produção (Desempenho)
RNF003	Usuários do sistema devem ser identificados individualmente	Essencial	Organização (Segurança)
RNF004	O sistema só permite uma única sessão de usuário por vez	Essencial	Organização (Segurança)
RNF005	O sistema deve ser compatível com Windows XP, Vista, 7, 8	Importante	Produção (Compatibilidade)
RNF006	O sistema deve funcionar 24/7	Importante	Prod. (Disponibilidade)

# Exemplo de Requisitos Não Funcionais

## ➤ Requisitos não funcionais do Google...

The image shows a screenshot of the Google search page. A dashed black box highlights the search bar and the advanced search options. The search bar is located in the center of the page, below the Google logo. The advanced search options are located on the right side of the page, below the search bar. The advanced search options include:

- Localizar páginas com...
- todas estas palavras:
- esta expressão ou frase exata:
- qualquer uma destas palavras:
- nenhuma destas palavras:
- números que variam de:
- Em seguida, limite seus resultados por...
- idioma:
- região:
- última atualização:

# Requisitos de Domínio (Regras de Negócio)

- São provenientes do **domínio de aplicação do sistema** e refletem características e restrições desse domínio. Eles são **derivados do negócio que o sistema se propõe a apoiar** e podem restringir requisitos funcionais existentes ou estabelecer como cálculos específicos devem ser realizados, refletindo fundamentos do domínio de aplicação (SOMMERVILLE, 2011).

*Ex.: Em um sistema de matrícula de uma universidade, uma importante regra de negócio diz que um aluno só pode se matricular em uma turma de uma disciplina se ele tiver cumprido seus pré-requisitos.*

# Regras de Negócio



## REGRAS DE NEGÓCIO

- 1.** O açai somente será entregue ao cliente após o recebimento do pagamento
- 2.** Clientes que comprarem 4 ou mais açais ganharão 1 açai de graça
- 3.** Não podem ser ofertados canudos de plástico

# Regras de Negócio



## REQUISITOS DE NEGÓCIO

1. Faturar a venda
2. Fechar o movimento do dia

# Outros

- Um aluno não pode se matricular em mais de 20 créditos por semestre
- A temperatura do paciente tem de ser medida a cada 2 horas

# Regras de Negócio

- Dicas: Prestar atenção em algumas palavras que os especialistas em negócios recomendam:
  - Sempre
  - Nunca
  - Obrigatoriamente
  - Somente se

# Lista de Regras de Negócio

## Regras de Negócio

ID	Descrição
RN01	O relatório de estoque DEVE ser atualizado a cada minuto
RN02	Uma solicitação de compra DEVE ser gerada QUANDO o estoque mínimo é atingido
RN03	Cotações com preços mais baixos que R\$500,00 DEVEM ter prioridade na compra
RN04	Compras geradas a partir de cotações com preços maiores R\$1.000,00 DEVEM ser justificadas
RN05	Empresas com idade < 1 ano OU com pendências no SERASA NÃO podem fornecer
	...

# Níveis de Descrição dos Requisitos

- **Requisitos de Cliente ou de Usuário:** são declarações em linguagem natural acompanhadas de diagramas intuitivos de quais serviços são esperados do sistema e das restrições sob as quais ele deve operar.
- Devem estar em um nível de abstração mais alto, de modo que sejam compreensíveis pelos clientes e usuários do sistema que não possuem conhecimento técnico.

# Níveis de Descrição dos Requisitos

- **Requisitos de Sistema**: definem detalhadamente as funções, serviços e restrições do sistema. São versões expandidas dos requisitos de cliente usados pelos desenvolvedores para projetar, implementar e testar o sistema.
- Como requisitos de sistema são mais detalhados, as especificações em linguagem natural são insuficientes e para especificá-los, notações mais especializadas devem ser utilizadas.

# Exemplo

Diferentes níveis de descrição, ex:

## Requisitos de usuário

1. O sistema **deve** gerar relatórios mensais que mostrem o custo dos medicamentos prescritos por clínica durante cada mês

## Requisitos de sistema

1. No último dia de cada mês **deve** ser gerado um resumo dos medicamentos prescritos por clínica durante aquele mês
2. Um relatório por clínica **deve** ser gerado, listando nome dos medicamentos, total de prescrições e o custo total
3. Se os medicamentos estão disponíveis em diferentes unidades de dosagem (10mg, 20mg) **devem** ser criados relatórios separados

# Níveis de Descrição dos Requisitos

- Requisitos de cliente são elaborados nos estágios iniciais do desenvolvimento (levantamento preliminar de requisitos) e servem de base para um entendimento entre clientes e desenvolvedores acerca do que o sistema deve contemplar. Esses requisitos são, normalmente, usados como base para a contratação e o planejamento do projeto ➡ Documento de Definição de Requisitos.
- Requisitos de sistema, por sua vez, são elaborados como parte dos esforços diretos para o desenvolvimento do sistema, capturando detalhes importantes para as fases técnicas posteriores do processo de desenvolvimento, a saber: projeto, implementação e testes.  
➡ Documento de Especificação de Requisitos.

# Exercícios: Classifique os requisitos

- O usuário deve conseguir fazer buscas em todo o acervo de materiais bibliográficos.
- O sistema deve ser protegido contra acesso não autorizado.
- O sistema deve cobrar multa para itens devolvidos com atraso, segundo a seguinte fórmula:  $M=n*VL$ , onde  $M$  é o valor da multa,  $n$  o número de dias de atraso e  $VL$  é o valor de locação do item.

# Exercícios: Classifique os requisitos

- O sistema deve ser capaz de lidar com o número necessário de usuários sem qualquer degradação no desempenho.
- O sistema deve ser capaz de aumentar ou diminuir conforme necessário.
- O sistema deve estar disponível quando necessário.
- O sistema deve permitir o cadastro dos fornecedores da loja
- O sistema deve utilizar os dados obtidos a partir dos sensores e interpretá-los para realizar a navegação

# Exercícios: Classifique os requisitos

- O sistema deve ser fácil de manter e atualizar.
- O sistema deve ser capaz de rodar em diferentes plataformas com alterações mínimas.
- O sistema deve ser confiável e atender aos requisitos do usuário.
- O sistema deve manter o histórico de locações e, portanto, clientes que tenham feito Locações não poderão ser excluídos.
- O sistema deve ser fácil de usar e entender.
- O sistema deve ser compatível com outros sistemas.

# Exercícios: Classifique os requisitos

- O sistema deve cumprir todas as leis e regulamentos aplicáveis.
- O sistema deve permitir o cadastro de novos usuários, realizando a validação através de um e-mail de confirmação.
- Reservas canceladas pelo sistema não deverão ser efetivamente excluídas pelo sistema, mas sim marcadas como expiradas.
- Deve ser possível ao usuário solicitar a redefinição de sua senha, informando o e-mail cadastrado

# Exercícios: Classifique os requisitos

- O sistema deve calcular o valor do frete com base no peso do produto e no endereço de entrega.
- O valor do frete deve ser calculado como R\$ 3,50 para cada 50 gramas de produto se o endereço de entrega for em outro país :
- O sistema deve permitir que sejam dados descontos nas locações, bem como que sejam ampliados os prazos de devolução de itens.

# Exercícios de Fixação

1- Identifique os **requisitos funcionais** e **não funcionais**.

2- Aponte possíveis incertezas nessa descrição.

*“Um sistema automático de emissão de passagens vende passagens de trem. A partir de uma lista de possíveis destinos, os usuários escolhem seu destino e apresentam um cartão de crédito e um número de identificação pessoal. Os destinos possíveis devem ser organizados de modo a facilitar a escolha. Após a escolha do destino, o sistema deve responder prontamente se há espaço disponível no trem. A passagem é emitida e o custo dessa passagem é incluído em sua conta do cartão de crédito. Quando o usuário pressiona o botão para iniciar, uma tela de menu com os possíveis destinos é ativada, juntamente com uma mensagem para que o usuário selecione um destino. Uma vez selecionado um destino, pede-se que os usuários insiram seu cartão de crédito. A validade do cartão é checada e o usuário então deve fornecer um número de identificação pessoal. Quando a transação de crédito é validada, a passagem é emitida. O formato do bilhete de passagem deve seguir ao padrão definido pelo Sistema Nacional de Tráfego Ferroviário”.*

# Como Identificar Requisitos?

- Etapa na qual o analista vai buscar as informações sobre o software a ser desenvolvido:
  - Buscar junto ao usuário, seus sistemas e documentos todas as informações possíveis sobre as funções que o sistema deve executar e as restrições sobre as quais o sistema deve operar.
  - Assume-se pouco conhecimento do analista sobre o sistema.

# Como Identificar Requisitos?

- Você percebe alguma relação?

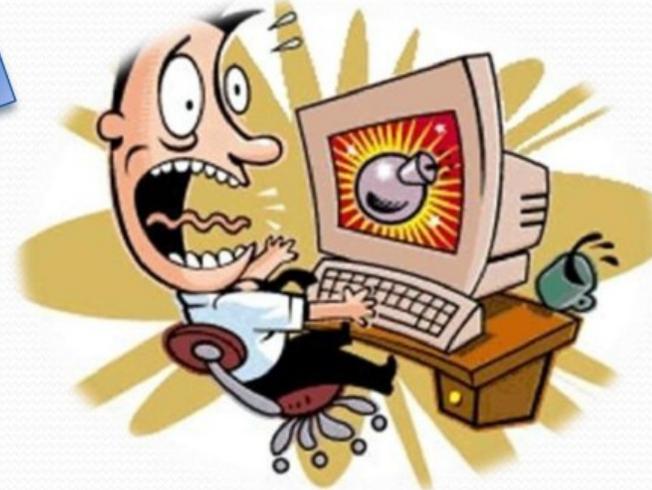
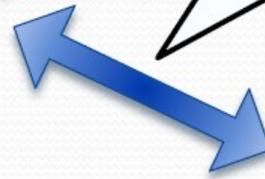


# Como Identificar Requisitos?



Um sistema a ser analisado é como  
uma floresta

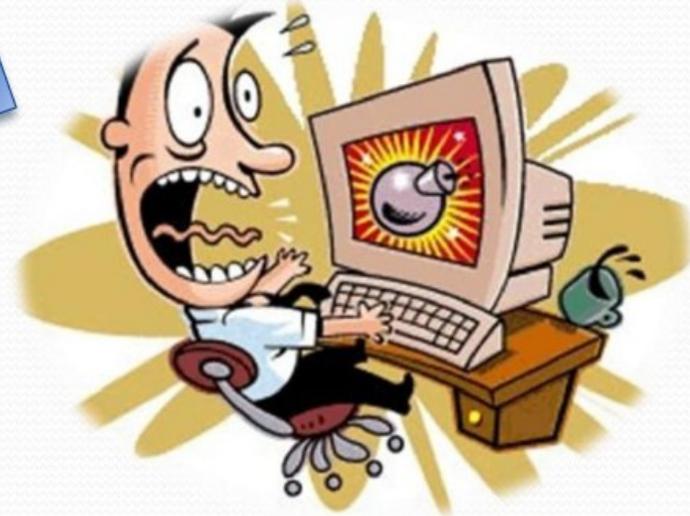
Não é possível, em um primeiro  
momento, conhecer cada planta e  
cada animal



# Como Identificar Requisitos?



No desenvolvimento de sistemas...  
é importante ter uma “**visão geral  
do todo**” e **depois dividir** em  
partes para analisar os **detalhes**



# Como Identificar Requisitos?

No desenvolvimento de sistemas...  
é importante ter uma “*visão geral do todo*” e **depois dividir** em partes para analisar os **detalhes**

## Exemplo 1:

**Visão do Todo:** O Serviço desejado consiste em um Portal Colaborativo, onde qualquer usuário Web deseja obter Informações sobre as Olimpíadas de Londres em 2012

# Como Identificar Requisitos?

No desenvolvimento de sistemas...  
é importante ter uma “*visão geral do todo*” e **depois dividir** em partes para analisar os **detalhes**

## Exemplo 1:

**Visão do Todo:** O Serviço desejado consiste em um Portal Colaborativo, onde qualquer usuário Web deseja obter Informações sobre as Olimpíadas de Londres em 2012

### ***Visão das Partes:***

- 1º) Que tipo de informação esse usuário deseja visualizar?
- 2º) Quem detém essa informação e vai disponibilizá-la no sistema....

# Como Identificar Requisitos?

No desenvolvimento de sistemas...  
é importante ter uma “*visão geral do todo*” e **depois dividir** em partes para analisar os **detalhes**

## Exemplo 2:

**Visão do Todo:** A Universidade precisa de um sistema que controle a reserva e utilização de recursos áudio visuais

# Como Identificar Requisitos?

No desenvolvimento de sistemas...  
é importante ter uma “**visão geral do todo**” e **depois dividir** em partes para analisar os **detalhes**

## Exemplo 3:

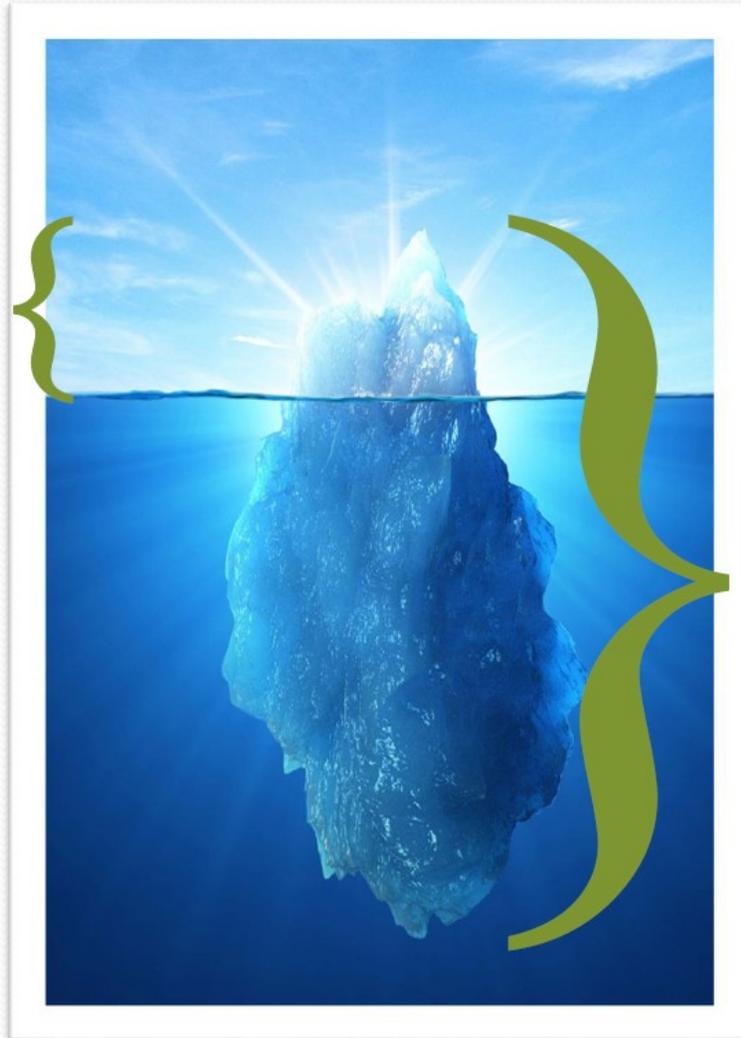
**Visão do Todo:** A empresa deseja controlar seu relacionamento com os clientes

Mas, e se o usuário não sabe explicar  
uma “*visão do todo*”?



# Iceberg de Requisitos

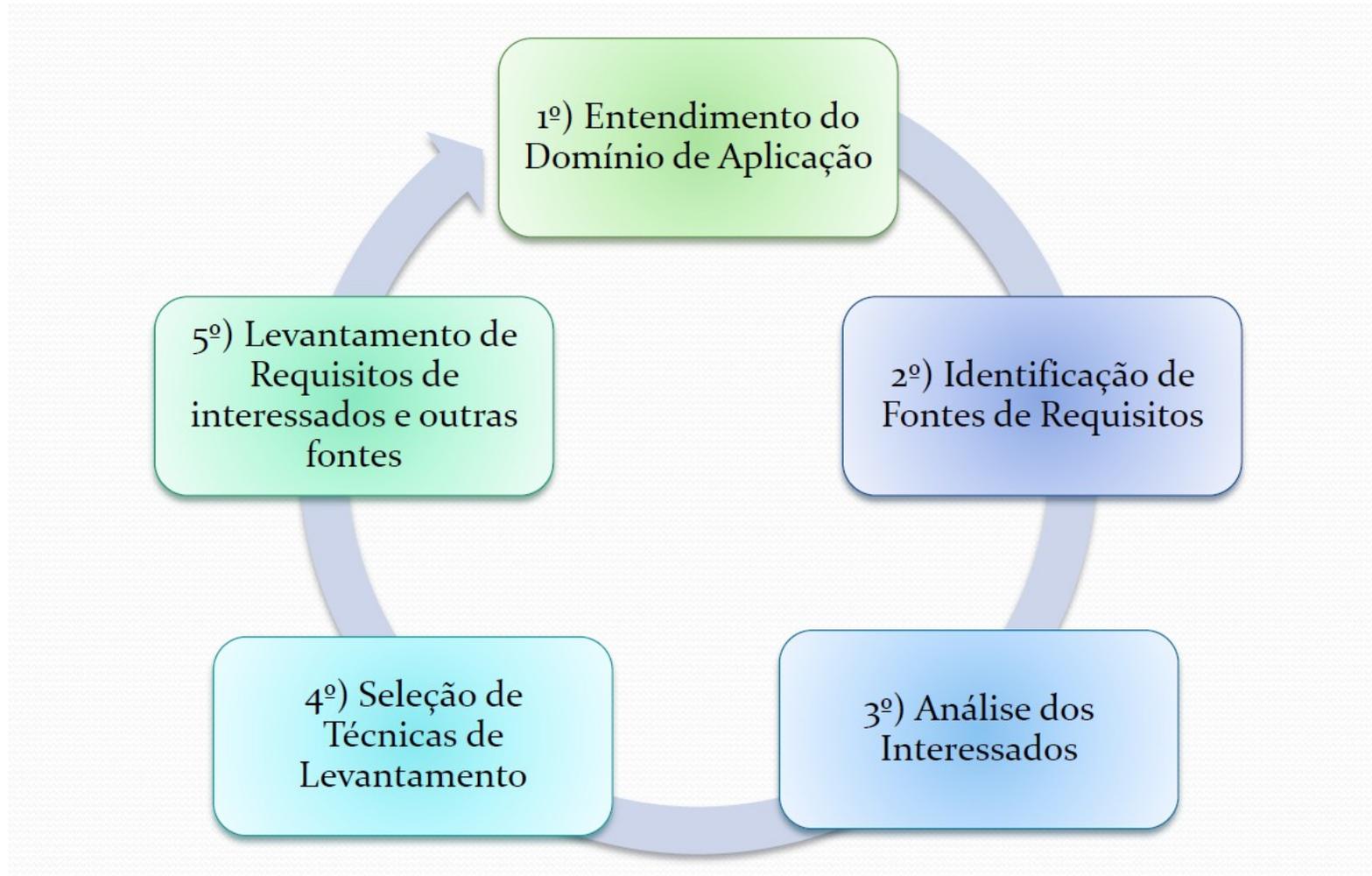
**Visão  
do  
Cliente**



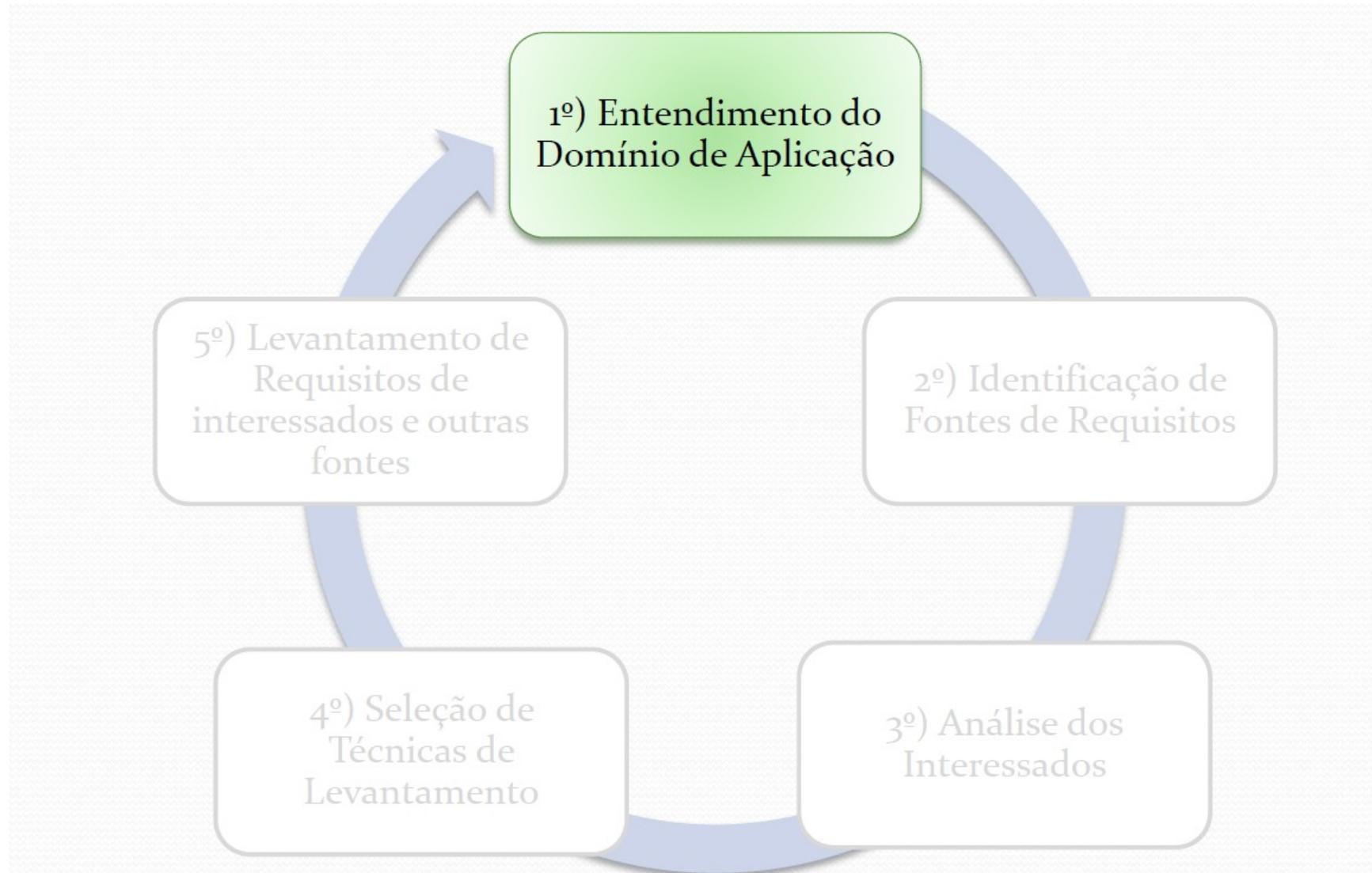
**Requisitos  
do  
Sistemas**

# Como Identificar Requisitos?

- Atividades básicas: Faça de forma incremental (etapas)



# Como Identificar Requisitos?



# Como Identificar Requisitos?

- Antes de iniciar a “*descoberta*” de requisitos propriamente dita, é importante...
  - Compreender os objetivos gerais do negócio a ser apoiado,
  - *Esboçar uma descrição* do problema a ser resolvido e identificar *por que o sistema é necessário* e quais são suas restrições
  - Levantar informações do contexto do desenvolvimento, informações sobre o domínio da aplicação e informações sobre *sistemas que estão em uso e serão substituídos* pelo sistema em desenvolvimento
  - Identificar as Regras de Negócio

# Como Identificar Requisitos?

## ➤ *Regras de Negócio*

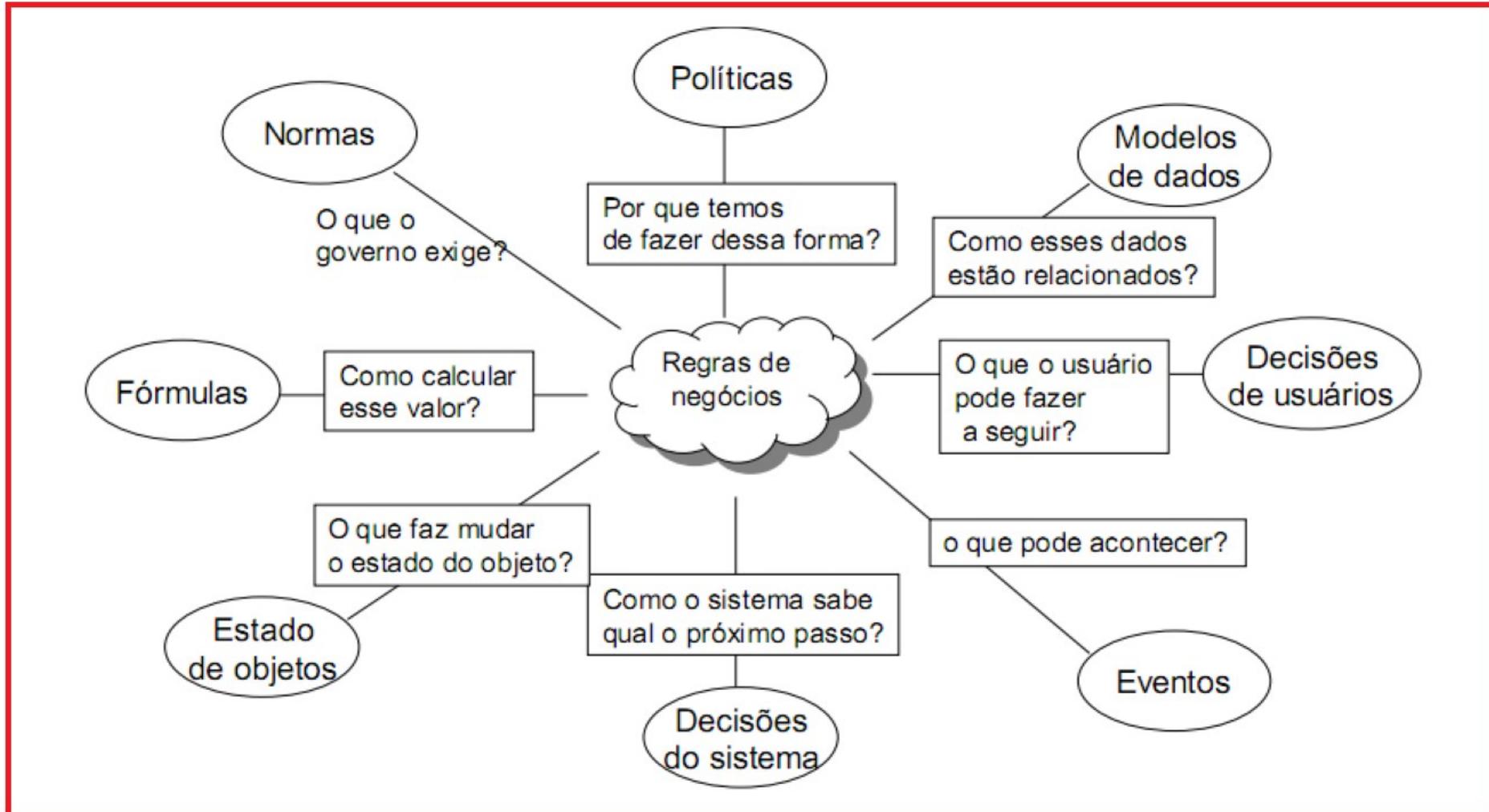
- Regras de negócios ou propriedades do domínio **não são requisitos**
- São fatos sobre o meio em que o sistema vai ser inserido.
- são verdadeiros mesmo se não houver um sistema
- São relevantes para o **desenvolvimento correto** do sistema

## ➤ *Exemplos de Regras de Negócio*

- Um aluno não pode se matricular em mais de 20 créditos por semestre
- Ativar o motor do videocassete faz os rolamentos girarem e puxarem a fita em uma direção
- Cheques vencem 30 dias após sua emissão
- A temperatura do paciente tem de ser medida a cada 2 horas

# Como Identificar Requisitos?

DOMÍNIO DE NEGÓCIO

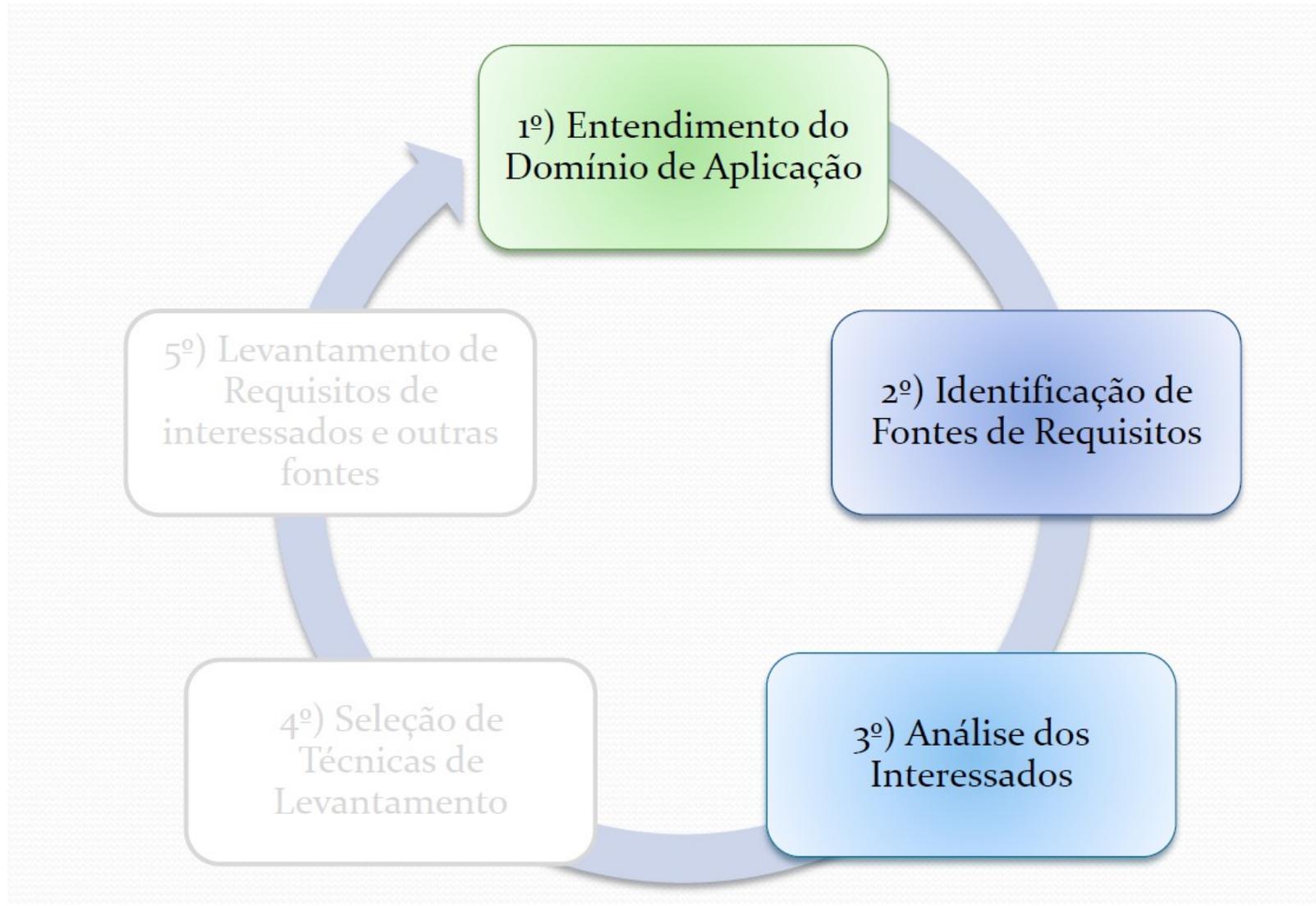


# Como Identificar Requisitos?

## ➤ Tipos de Regras de Negócio

- Fatos
- Restrições
- Gatilhos
- Inferências
- Cálculos
  - Especificam fórmulas ou algoritmos
    - *ex: toda compra acima de mil reais recebe um desconto de 10%*

# Como Identificar Requisitos?



# Como Identificar Requisitos?

- **Envolvimento de clientes e usuários é fator crítico para o sucesso do projeto.**
  - Importante engajar representantes deles desde o início do projeto
- **Para definir esses representantes, deve-se:**
  - Identificar diferentes classes de usuários
  - (i) frequência com que usam o sistema,
  - (ii) experiência no domínio de aplicação e perícia com sistemas computadorizados,
  - (iii) características do sistema que eles usam,
  - (iv) tarefas que eles realizam no apoio a seus processos de negócio e
  - (v) níveis de privilégio de acesso e segurança.

# Como Identificar Requisitos?

- Selecionar e trabalhar com indivíduos que representem cada grupo de usuários
- Estabelecer um acordo sobre quem serão as pessoas responsáveis por tomar decisões relativas a requisitos, sobretudo no que concerne a estabelecer prioridades e resolver conflitos
- Identificação e Análise *stakeholders* (i.e., toda pessoa que tem interesse no sistema)
  - Clientes, usuários, especialistas do domínio, marketing, desenvolvedores e investidores...

Quem são os *Stakeholders*?



Quais as fontes de Requisitos?

# Como Identificar Requisitos?

## ➤ *Stakeholders*

- Cliente
- Usuários
- Equipe de Desenvolvimento
- Equipe de Comunicação

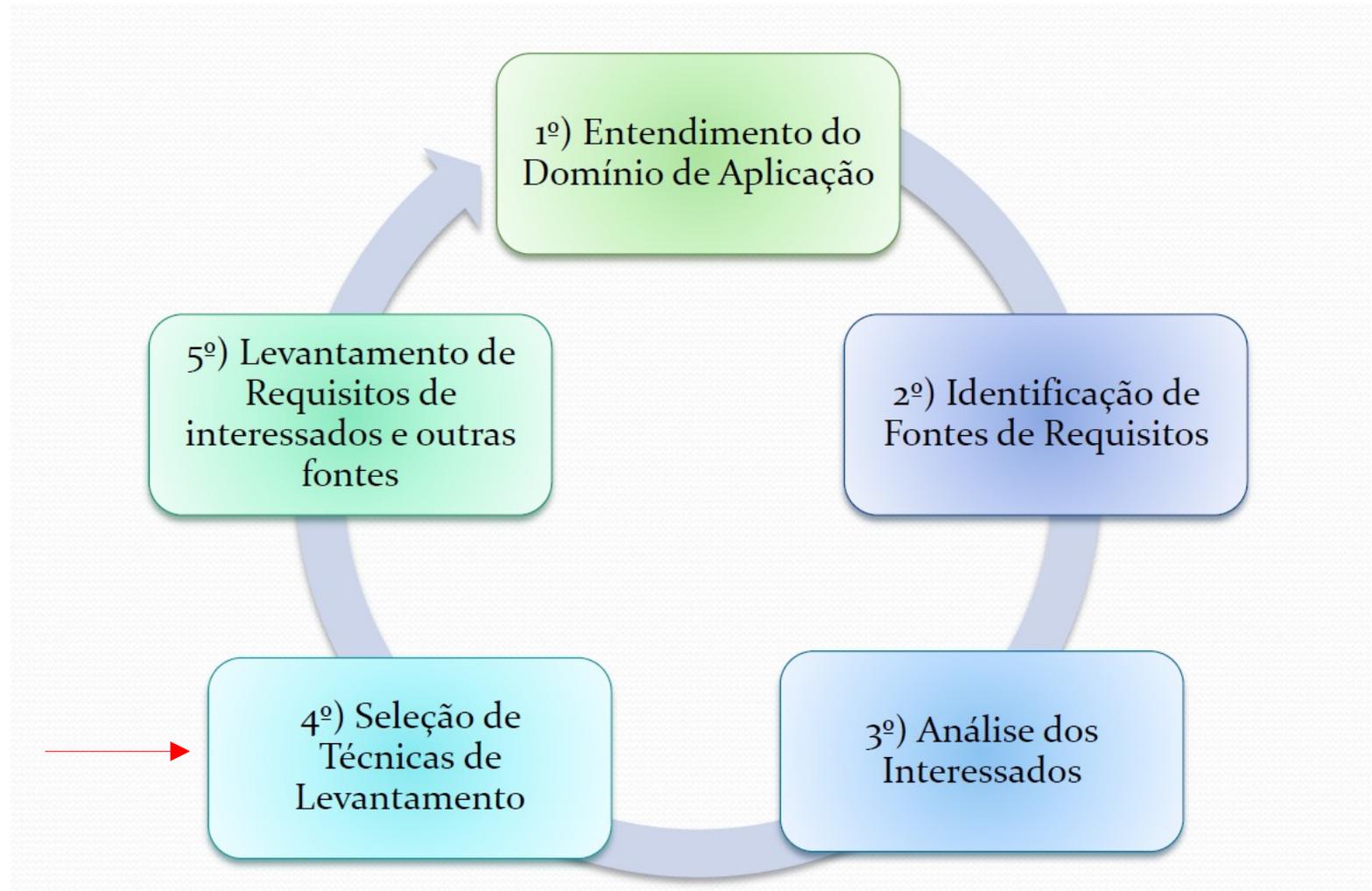
## ➤ *Fontes de Requisitos*

- Em relação aos Stakeholders, as principais fontes de requisitos são os usuários.

## ➤ *Análise dos Interessados Fontes de Requisitos*

- Levantar o perfil dos interessados.
  - Ex.: Aquele **usuário**, que trabalha em **determinado setor**, que deverá **interagir** com o sistema **diariamente**. Ele é o responsável por realizar os cadastros de outros usuários e de recursos ou produtos (informação/dados importantes), além de gerenciar todos os cadastros já realizados.

# Como Identificar Requisitos?



# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

Grupo de Usuários	Individual
Oficina de Requisitos Entrevista (em grupo) Protótipos Questionário <i>Brainstorming</i> Cenários Observação ...	Entrevista Protótipo Questionário ...

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

## - Entrevista

- Forma ***mais comum*** de levantamento (elicitación) de requisitos.
- **Objetivo:** ***entender*** os ***processos existentes*** e o domínio da aplicação
- Podem ser ***individuais*** ou em ***grupo***
- Deve-se ***preparar*** as perguntas ***antecipadamente***
  - Não se deve assumir qualquer conhecimento prévio sobre o problema
    - Chama-se essa postura de “***IGNORANTE INTELIGENTE***”

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

## - Questionários

- Muito utilizado quando os analistas identificam a necessidade de coletar informações de *multiusuários*
- FOCO:
  - Para obter *evidências* é necessário elaborar *questões fechadas*;
  - Para capturar *opiniões* e/ou *sugestões*, perguntas *abertas* podem ser incluídas
- Potencial *Ambiguidade*
  - Questionários podem apresentar problemas de interpretação (durante a aplicação e análise das respostas)
    - A *pessoa* que responde pode *não entender* as *perguntas*
    - O *analista* pode *não compreender* as *respostas*

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

- **Brainstorming (Chuva de Ideias)**
  - Um **grupo de pessoas** é reunido, um **cenário simulado** e um assunto é **discutido** para **elicitare os requisitos**
  - Importante que os **participantes** da reunião se sintam **confortáveis** o bastante para **discutir** sobre o assunto levantado **sem se sentirem ridículos**.
    - Na primeira sessão **não** se deve **julgar** as ideias
  - **Sessão posteriores** são usadas para **priorizar** o que foi **levantado**

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

## - Prototipação

- *Utilizada com frequência* na elicitação de requisitos
  - Tipos de protótipos: descartáveis (*papel/digital*) e evolutivos
- Protótipos *Descartáveis*
  - Função de ilustrar para os usuários e/ou clientes do sistema o que o analista entendeu sobre os requisitos – *Sem layout, preto e branco!*
- Protótipos *Evolutivos*
  - *Contém os requisitos* do sistema no âmbito de implementação incremental.
- Quais os *Benefícios* da utilização?
  - *Redução dos riscos* na construção
  - Baixo valor de correção, etc...

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

- **Seminários:**
  - Reuniões com usuários dos diversos departamentos envolvidos.
  - Planejar as reuniões.
  - Escolher criteriosamente as pessoas.
  - Conhecer previamente o assunto.
  - Atuar como um líder imparcial.
  - Esta técnica possibilita a quebra de feudos.

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

- **Observação**

- Vivenciar a rotina de trabalho dos usuários.
- Participar das atividades operacionais em conjunto com o usuário.
- Constatar problemas de relacionamentos entre os usuários e entre os departamentos.

# Seleção da Técnica de Levantamento de Requisitos (Elicitação)

- Pesquisas

- Solicitação de todos os documentos usados pelos departamentos.
- Documentos emitidos e recebidos pelo departamento.
- Confrontar e analisar as informações dos documentos com as informações obtidas nas outras técnicas.

# Perguntas para Identificar Requisitos

- Quais são alguns motivos pelos quais você e seus colegas usariam o novo sistema?
- Que objetivos você tem em mente que esse sistema o ajudaria a cumprir?
- Que problemas você espera que esse sistema resolva para você?
- Que eventos externos estão associados ao sistema?
- Que palavras você usaria para descrever o sistema?
- Algumas partes do produto são mais importantes que outras por motivos de desempenho, segurança, robustez, disponibilidade, ou alguma outra característica?
- Há alguma restrição ou regra à qual o sistema deve estar de acordo?
- Como você imagina que o sistema deve ser semelhante à forma como você trabalha hoje? Como ele deve ser diferente?
- Que aspectos dos seus processos de negócios você quer manter? quais você quer substituir?
- Que aspecto do sistema você acha mais interessante?
- Que aspecto terá mais valor pra você? e menos valor?
- Como você vai julgar se o sistema é um sucesso?
- Você pode descrever o ambiente onde o sistema vai ser usado?

# Atores do Processo de Elicitação

- **Analista de Sistemas/Negócios**
  - Entender as necessidades dos clientes/usuários
  - Definir os limites do sistema.
  - Identificar e documentar os requisitos do sistema.
  - Usar um vocabulário comum entre usuários e demais envolvidos.
  - Desenhar e descrever os modelos de Caso de Uso.
  - Desenvolver e manter o documento de requisitos.
  - Ter boas habilidades de comunicação (verbal e escrita)

# Artefatos Produzidos na Elicitação

- A Visão Geral do Sistema (texto corrido que contém as principais ideias do cliente sobre o sistema)
- Lista de requisitos funcionais e não funcionais
- Outros artefatos:
  - *Glossário (para alinhar termos utilizados entre analista, cliente e usuários – **Evitando assim o Pedantismo**)*
  - *Protótipos (ajudam a esboçar ideias sobre as funcionalidades)*

# Documento de Especificação de Requisitos

- Especifica os requisitos funcionais e não-funcionais do sistema.
- Fornece as principais informações relativas ao sistema possibilitando ao desenvolvedor os meios para a implementação, testes e homologação.

# Documento de Especificação de Requisitos

- Não deve conter informações sobre soluções técnicas.
- Procura responder a questão: Quais são as necessidades do usuário?
- Serve como um termo de consenso entre a equipe técnica (desenvolvedores) e o cliente.
- Estabelece o escopo do sistema, ou seja, define o que é sistema e o que não é sistema.
- Mostrar os requisitos pelo grau de prioridade (aconselhável).

# Componentes do Documento de Especificação de Requisitos

- Introdução
- Visão geral
- Convenções e nomenclaturas
- Partes interessadas (stakeholders)
- Requisitos funcionais
- Requisitos não funcionais
- Interface sugerida (IHM)

# Componentes do Documento de Especificação de Requisitos

- **Introdução**
  - Texto breve sobre o sistema, suas funcionalidades, objetivos e a motivação.
- **Visão geral**
  - Falar sobre o processo (ou sistema) atual, as funcionalidades e problemas. Especificar, em linhas gerais, o que será feito, ou seja, quais são as expectativas.
- **Convenções e nomenclaturas**
  - Tem como objetivo auxiliar o entendimento das abreviações usadas no documento.

# Componentes do Documento de Especificação de Requisitos

- Partes interessadas (stakeholders)
  - Stakeholder (em português, partes interessadas), são as pessoas que afetam ou são afetadas pelo sistema e que por sua vez terão influência sobre os requisitos desse sistema.
  - Trata-se de um termo usado inicialmente na administração de empresas
  - O sucesso de um projeto, sistema ou empresa depende diretamente da efetiva participação dos Stakeholders.

# Componentes do Documento de Especificação de Requisitos

- Requisitos funcionais
  - São os Casos de Uso.
  - Deve-se descrever os seguintes itens:
    - nome do caso de uso;
    - descrição resumida (objetivo);
    - ator principal;
    - ator secundário;
    - cenário principal;
    - cenário alternativo;
    - pré-condição;
    - pós-condição;
    - fluxo de exceção.
  - O diagrama de caso de uso deve ser representado no documento.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - São os requisitos relativos a: desempenho, segurança, usabilidade, confiabilidade e distribuição, entre outros.
  - Deve-se ainda mencionar os requisitos de hardware e software para o desenvolvimento e execução do projeto.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - Desempenho
    - Requisitos não funcionais relativos ao desempenho do sistema, como por exemplo tempo de resposta da abertura de telas do sistema, tempo necessário para emissão de um relatório de fatura.
    - Exemplos:
      - **REQNF001:** Todas as telas do sistema não deverão demorar mais que 5 (cinco) segundos para abertura.
      - **REQNF002:** O banco de dados relativo às informações gerenciais deve ser atualizado em tempo real.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - Segurança
    - Requisitos não funcionais relativos à integridade e privacidade dos dados armazenados pelo sistema.
    - Exemplos:
      - **REQNF001:** O sistema deverá restringir o acesso às informações financeiras relativas ao grupo.
      - **REQNF002:** Somente usuários autorizados com perfil “Gerência” poderão acessar o módulo de contas de usuário e definição dos parâmetros globais do sistema.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - Usabilidade
    - Requisitos não funcionais relativos à distribuição dos dados e dos arquivos de chamadas do sistema (módulo executável).
    - Exemplos:
      - **REQNF001:** O sistema deverá ser capaz de funcionar nos navegadores Internet Explorer e Firefox.
      - **REQNF002:** O sistema deverá possibilitar o uso de teclas de atalho para as principais opções.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - Confiabilidade
    - Requisitos não funcionais relativos à frequência de ocorrência de falhas do sistema e à capacidade de recuperação dessas falhas pelo próprio sistema.
    - Exemplos:
      - **REQNF001:** O tempo médio entre falhas (MTBF) para as operações referentes ao relatórios que necessitam de processamento intensivo não poderá exceder a NN meses (ou NN dias ou NN anos).
      - **REQNF002:** A exatidão das informações numéricas referentes aos resultados dos exames deverão possuir a precisão de 3 dígitos decimais.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - Distribuição
    - Requisitos não funcionais relativos à distribuição dos dados e dos arquivos de chamadas do sistema (módulo executável).
    - Exemplos:
      - **REQNF001:** O sistema deverá ser acessado no atendimento (marcação), consultório, administração e disponibilizado na WEB. Os dados deverão ser igualmente distribuídos em cada um desses setores e acessados mediante o perfil de cada usuário.

# Componentes do Documento de Especificação de Requisitos

- Requisitos não funcionais
  - Interface sugerida (IHM)
    - O desenho de uma interface parcialmente funcional para o sistema é muito importante à medida que auxilia na validação dos requisitos funcionais.
    - O protótipo pode ou não ser utilizado na continuação do desenvolvimento
    - Isso dependerá da linguagem de programação usada e de uma metodologia bem definida de modo a permitir uma migração natural do protótipo para versão final.

# Documentação

- Após **elicitados e documentados**, os requisitos devem ser:
  - (1) verificados e validados e
  - (2) priorizados.

# Documentação no Desenvolvimento Ágil

- No caso de **desenvolvimento ágil**, a documentação de requisitos é feita de forma simplificada, por meio de histórias do usuário.
- Por outro lado, em alguns projetos, ainda se exige um **Documento de Especificação de Requisitos**, no qual todos os requisitos do software que se pretende construir — incluindo requisitos funcionais e não-funcionais — são documentados em linguagem natural (português, inglês, etc.), conforme visto anteriormente.

# Verificação e Validação de Requisitos

- Após sua especificação, os requisitos devem ser verificados e validados.
- O objetivo é garantir que eles estejam corretos, precisos, completos, consistentes e verificáveis.

# Verificação e Validação de Requisitos

- **Requisitos devem estar corretos.** Um contra-exemplo é a especificação de forma incorreta da fórmula para remuneração das cadernetas de poupança em um sistema bancário. Evidentemente, uma imprecisão na descrição dessa fórmula irá resultar em prejuízos para o banco ou para seus clientes.

# Verificação e Validação de Requisitos

- **Requisitos devem ser precisos**, isto é, não devem ser ambíguos. No entanto, ambiguidade ocorre com mais frequência do que gostaríamos quando usamos linguagem natural.
  - **EXEMPLO**: considere essa condição: para ser aprovado um aluno precisa obter 60 pontos no semestre ou 60 pontos no Exame Especial e ser frequente. Veja que ela admite duas interpretações. A primeira é a seguinte: (60 pontos no semestre ou 60 pontos no Exame Especial) e ser frequente. Porém, pode-se interpretar também como: 60 pontos no semestre ou (60 pontos no Exame Especial e ser frequente). Conforme você observou, tivemos que usar parênteses para eliminar a ambiguidade na ordem das operações e e ou.

# Verificação e Validação de Requisitos

- **Requisitos devem ser completos.** Isto é, não podemos esquecer de especificar certos requisitos, principalmente se eles forem importantes no sistema que se pretende construir.
- **Requisitos devem ser consistentes.** Um contra-exemplo ocorre quando um stakeholder afirma que a disponibilidade do sistema deve ser 99,9% e outro considera que 90% já é suficiente.

# Verificação e Validação de Requisitos

- **Requisitos devem ser verificáveis**, isto é, deve ser possível testar se os requisitos estão sendo atendidos. Um contra-exemplo é um requisito que apenas requer que o sistema seja amigável. Como os desenvolvedores vão saber se estão atendendo a essa expectativa dos clientes?

# Priorização de Requisitos

- Por fim, os requisitos devem ser priorizados. Às vezes, o termo requisitos é interpretado de forma literal, isto é, como uma lista de funcionalidades e restrições obrigatórias em sistemas de software.
- No entanto, nem sempre aquilo que é especificado pelos clientes será implementado nas releases iniciais. Por exemplo, restrições de prazo e custos podem postergar a implementação de certos requisitos.

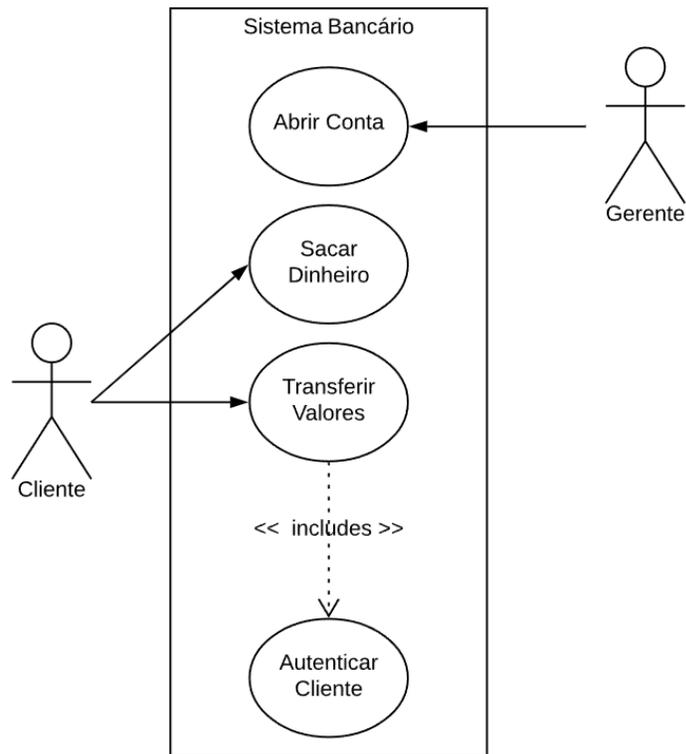
# Desafios da Engenharia de Requisitos

- Requisitos incompletos ou não-documentados (48%)
- Falhas de comunicação entre membros do time e os clientes (41%)
- Requisitos em constante mudança (33%)
- Requisitos especificados de forma abstrata (33%)
- Restrições de tempo (32%)
- Problemas de comunicação entre os próprios membros do time (27%)
- Stakeholders com dificuldades de separar requisitos e soluções (25%)
- Falta de apoio dos clientes (20%)
- Requisitos inconsistentes (19%)
- Falta de acesso às necessidades dos clientes ou do negócio (18%)

# Documentação de Requisitos

- “Documentações” de Requisitos
  - Documento de Especificação de Requisitos;
  - Casos de Uso;
- História de Usuários;

# Casos de Uso (UML)



“

## **Transferir Valores entre Contas**

**Ator:** Cliente do Banco

### **Fluxo normal:**

- 1 - Autenticar Cliente
- 2 - Cliente informa agência e conta de destino da transferência
- 3 - Cliente informa valor que deseja transferir
- 4 - Cliente informa a data em que pretende realizar a operação
- 5 - Sistema efetua transferência
- 6 - Sistema pergunta se o cliente deseja realizar uma nova transferência

### **Extensões:**

- 2a - Se conta e agência incorretas, solicitar nova conta e agência
- 3a - Se valor acima do saldo atual, solicitar novo valor
- 4a - Data informada deve ser a data atual ou no máximo um ano a frente
- 5a - Se data informada é a data atual, transferir imediatamente
- 5b - Se data informada é uma data futura, agendar transferência

# Histórias de Usuários

- No desenvolvimento de software e gerenciamento de produto, uma história de usuário é uma explicação informal e geral sobre um recurso de software escrito a partir da perspectiva do usuário final ou cliente
- É uma descrição simples e genérica de uma funcionalidade de software a partir da experiência do usuário. Ela define o que um usuário precisa da sua empresa, o que ajuda a priorizar o trabalho e a aumentar o valor para o cliente.

“

*História de Usuário = Cartão + Conversas + Confirmação*

# Histórias de Usuários

- **Cartão**, usado pelos clientes para escrever, na sua linguagem e em poucas sentenças, uma funcionalidade que esperam ver implementada no sistema.

# Histórias de Usuários

- **Conversas** entre clientes e desenvolvedores, por meio das quais os clientes explicam e detalham o que escreveram em cada cartão.
- Como dito antes, a visão de métodos ágeis sobre Engenharia de Requisitos é pragmática: como especificações textuais e completas de requisitos não funcionam, elas foram eliminadas e substituídas por **comunicação verbal entre desenvolvedores e clientes**. Por isso, métodos ágeis incluem nos times de desenvolvimento um representante dos clientes, que participa do time em tempo integral.

# Histórias de Usuários

- **Confirmação**, que é basicamente um teste de alto nível — de novo especificado pelo cliente — **para verificar se a história foi implementada conforme esperado**.
- Portanto, não se trata de um teste automatizado, como um teste de unidades, por exemplo. Mas a descrição dos cenários, exemplos e casos de teste que o cliente irá usar para confirmar a implementação da história. Por isso, são também **chamados de testes de aceitação de histórias**. Eles devem ser escritos o quanto antes, preferencialmente no início de uma iteração. Alguns autores recomendam escrevê-los no verso dos cartões da história.

# Características de Boas Histórias de Usuários

- **Histórias devem ser independentes:** dadas duas histórias X e Y, deve ser possível implementá-las em qualquer ordem. Para isso, idealmente, não devem existir dependências entre elas.

# Características de Boas Histórias de Usuários

- **Histórias devem ser abertas para negociação.**  
Frequentemente, costuma-se dizer que histórias (o cartão) são convites para conversas entre clientes e desenvolvedores durante um sprint.

# Características de Boas Histórias de Usuários

- **Histórias devem agregar valor para o negócio dos clientes.** Histórias são propostas, escritas e priorizadas pelos clientes e de acordo com o valor que elas agregam ao seu negócio.
- Por isso, **não existe a figura de uma história técnica**, como a seguinte: o sistema deve ser implementado em JavaScript, usando React no front-end e Node.js no backend.

# Características de Boas Histórias de Usuários

- **Deve ser viável estimar o tamanho de uma história.** Por exemplo, quantos dias serão necessários para implementá-la. Normalmente, isso requer que a história seja pequena, como veremos no próximo item, e que os desenvolvedores tenham experiência na área do sistema.

# Características de Boas Histórias de Usuários

- **Histórias devem ser sucintas e pequenas.** Na verdade, até se admite histórias complexas e grandes, as quais são chamadas de épicos.

# Características de Boas Histórias de Usuários

- **Histórias devem ser testáveis**, isto é, elas devem ter critérios de aceitação objetivos. Como exemplo, podemos citar: o cliente pode pagar com cartões de crédito. Uma vez definidas as bandeiras de cartões de crédito que serão aceitas, essa história é testável. Por outro lado, a seguinte história é um contra-exemplo: um cliente não deve esperar muito para ter sua compra confirmada. Essa é uma história vaga e, portanto, com um critério de aceitação também vago.

# Documentação da História de Usuários

- Escrita da história de Usuários

“

*Como um [papel de usuário], eu gostaria de [realizar algo com o sistema]*

# Exemplos de Histórias de Usuários

- Nesta seção, vamos mostrar exemplos de histórias para um sistema de controle de bibliotecas. Elas estão associadas a três tipos de usuários: usuário típico, professor e funcionário da biblioteca.

“

*Como usuário típico, eu gostaria de realizar empréstimos de livros*

*Como usuário típico, eu gostaria de devolver um livro que tomei emprestado*

*Como usuário típico, eu gostaria de renovar empréstimos de livros*

*Como usuário típico, eu gostaria de pesquisar por livros*

*Como usuário típico, eu gostaria de reservar livros que estão emprestados*

*Como usuário típico, eu gostaria de receber e-mails com novas aquisições*

# Exemplos de Histórias de Usuários

“

*Como professor, eu gostaria de realizar empréstimos de maior duração*

*Como professor, eu gostaria de sugerir a compra de livros*

*Como professor, eu gostaria de doar livros para a biblioteca*

*Como professor, eu gostaria de devolver livros em outras bibliotecas*

“

*Como funcionário da biblioteca, eu gostaria de dar baixa em livros estragados*

*Como funcionário da biblioteca, eu gostaria de obter estatísticas sobre o acervo*

*Como funcionário da biblioteca, eu gostaria que o sistema envie e-mails de cobrança para alunos com empréstimos atrasados*

*Como funcionário da biblioteca, eu gostaria que o sistema aplicasse multas quando da devolução de empréstimos atrasados*

# Exercício de Fixação: Solução

- Como especificar requisitos não-funcionais usando histórias de usuários?

# Requisitos x Histórias de Usuários

## 1 - Objetivo

**Requisito**



*O objetivo de um requisito é documentar as necessidades*

**História de Usuário**



*O objetivo da história de usuário é servir como ponto de apoio de conversas*

## 2 - Perspectiva

*Requisito*



*Requisitos são escritos sob a perspectiva da solução*

*História de Usuário*



*Histórias de usuários são descritas sob a perspectiva do usuário*

### 3 - Processo de trabalho

*Requisito*



*Requisitos são mais utilizados em abordagens preditivas*

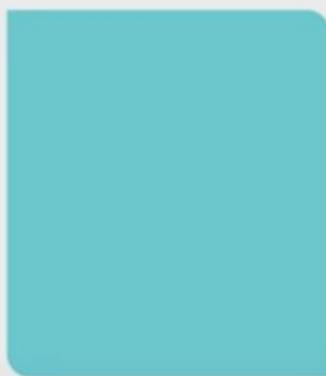
*História de Usuário*



*Requisitos são mais utilizados em abordagens adaptativas*

## 4 - Timming da elicitação

*Requisito*



*Requisitos são utilizados em  
um atividade de elicitação  
pontual*

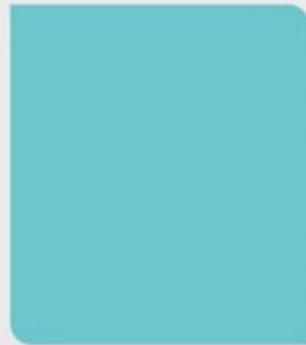
*História de Usuário*



*Histórias de usuário são  
utilizadas em um processo  
de descoberta contínuo*

## 5 - Propriedade

**Requisito**



*A propriedade do requisito é de quem o escreveu*

**História de Usuário**



*Histórias de usuário são de propriedade do grupo de pessoas que as descreveu*

## 6 - Tamanho

*Requisito*



*Requisitos são maiores e, portanto, difíceis de serem estimados e priorizados separadamente*

*História de Usuário*



*Histórias de usuário são menores e autocontidas. E possuem tudo que é necessário para serem estimadas e planejadas*

## 7 - Compreensão

*Requisito*



*Requisitos são escritos em linguagem técnica e, portanto, de difícil compreensão*

*História de Usuário*



*Histórias de usuário são escritas em linguagem em que todos podem compreender*

# Recomendações

- 1. Entenda bem o contexto no qual você utilizará as histórias de usuário.**
- 2. Nem tudo se resume às histórias de usuário.**
- 3. Faça das histórias um ponto de apoio de conversas produtivas.**

# Referências Bibliográficas

- PRESSMAN, Roger; MAXIM, Bruce. Engenharia de software: uma abordagem profissional. 8.ed. Bookman, 2016. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/books/9788580555349>
- SOMMERVILLE, Ian. Engenharia de software. 9. ed. São Paulo: Pearson Prentice Hall, 2011. E-book. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/2613/epub/0>
- Anotações de aula cedidas pelos professores: Jader Marques da Silva, Jean Paul Lopes, Osmar de Oliveira Braz Junior, Richard Henrique de Souza