



Requisitos de Software

Rosana T. Vaccare Braga
rtvb@icmc.usp.br

ICMC/USP

2017

Requisitos de Software

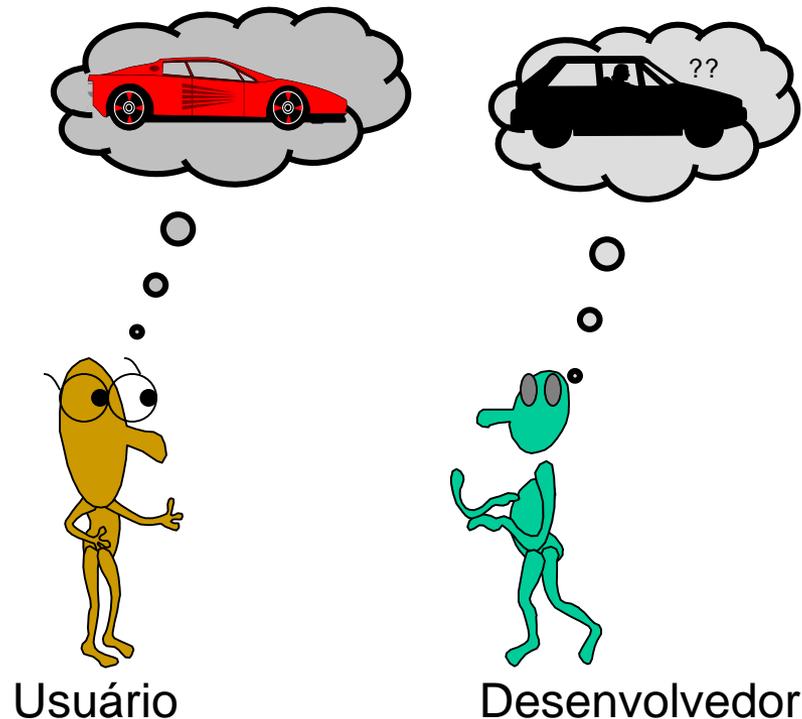
- 
- ⇒ Descrições do que **o sistema deve fazer**
 - ⇒ Inclui: os **serviços** fornecidos pelo sistema, suas **qualidades específicas** e suas **restrições operacionais**.
 - ⇒ Esses requisitos refletem as necessidades dos clientes de um sistema.

Por que especificar os requisitos?



Requisitos de Software

⇒ Por que é difícil entender os requisitos de um software?





How the customer explained it



How the Project Leader understood it



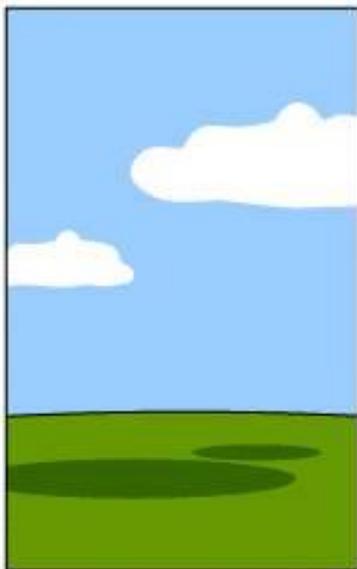
How the Analyst designed it



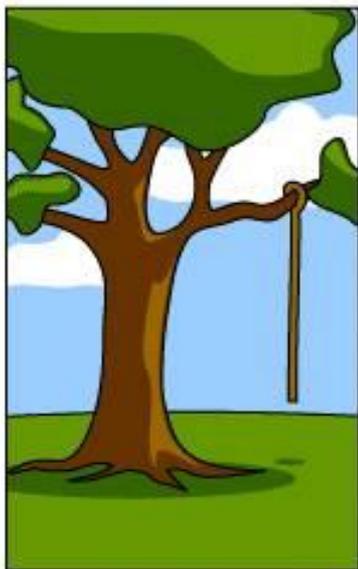
How the Programmer wrote it



How the Business Consultant described it



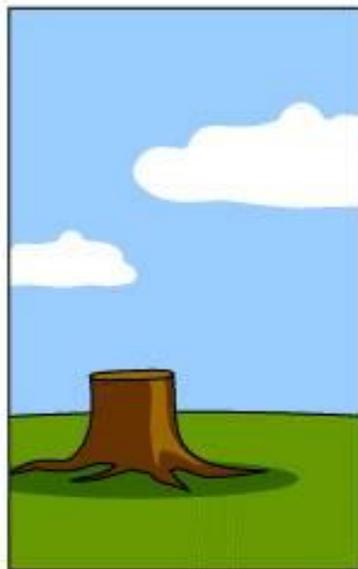
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

Por que é difícil entender os requisitos?

Diferentes níveis de descrição

Exemplo:

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

Tipos de Requisitos



- ⇒ Requisitos Funcionais
- ⇒ Requisitos Não-Funcionais

Requisitos Funcionais



⇒ Requisitos diretamente ligados a...

- ☞ **Funções** que o sistema deve fornecer.
- ☞ Como o sistema deve **reagir a entradas específicas**.
- ☞ Como o sistema deve se **comportar em determinadas situações**.
- ☞ Podem também declarar o que o sistema **não deve fazer**.

Requisitos Funcionais - Exemplos



- ⇒ “O usuário **deve** conseguir fazer buscas em todo o acervo de materiais bibliográficos.”
- ⇒ “O sistema **deve** fornecer **telas apropriadas** para o usuário ler documentos disponíveis no repositório de documentos.”
- ⇒ “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”

Requisitos Funcionais - Qualidade e Precisão

- ⇒ Surgem vários problemas quando os requisitos não são declarados de forma precisa.
- ⇒ Requisitos ambíguos podem ser interpretados de diferentes maneiras pelos desenvolvedores e usuários.
- ⇒ Considere o termo '**telas apropriadas**'.
 - ☞ Intenção do Usuário: telas especiais para cada tipo diferente de documento.
 - ☞ Interpretação do Desenvolvedor: fornecer uma tela texto que mostra o conteúdo do documento.

Requisitos Funcionais

Qualidade - Completeza e Consistência



⇒ Os requisitos devem ser completos e consistentes.

👉 **Completo**

- Eles devem incluir descrição de todas as facilidades que estão sendo requeridas.

👉 **Consistente**

- Eles não devem apresentar conflitos ou contradições entre as descrições das facilidades fornecidas pelo sistema.

⇒ Na prática, é impossível produzir um Documento de Requisitos completo e consistente.

👉 **Importante a validação do Documento de Requisitos!!**



Requisitos Não-Funcionais

Requisitos Não-Funcionais



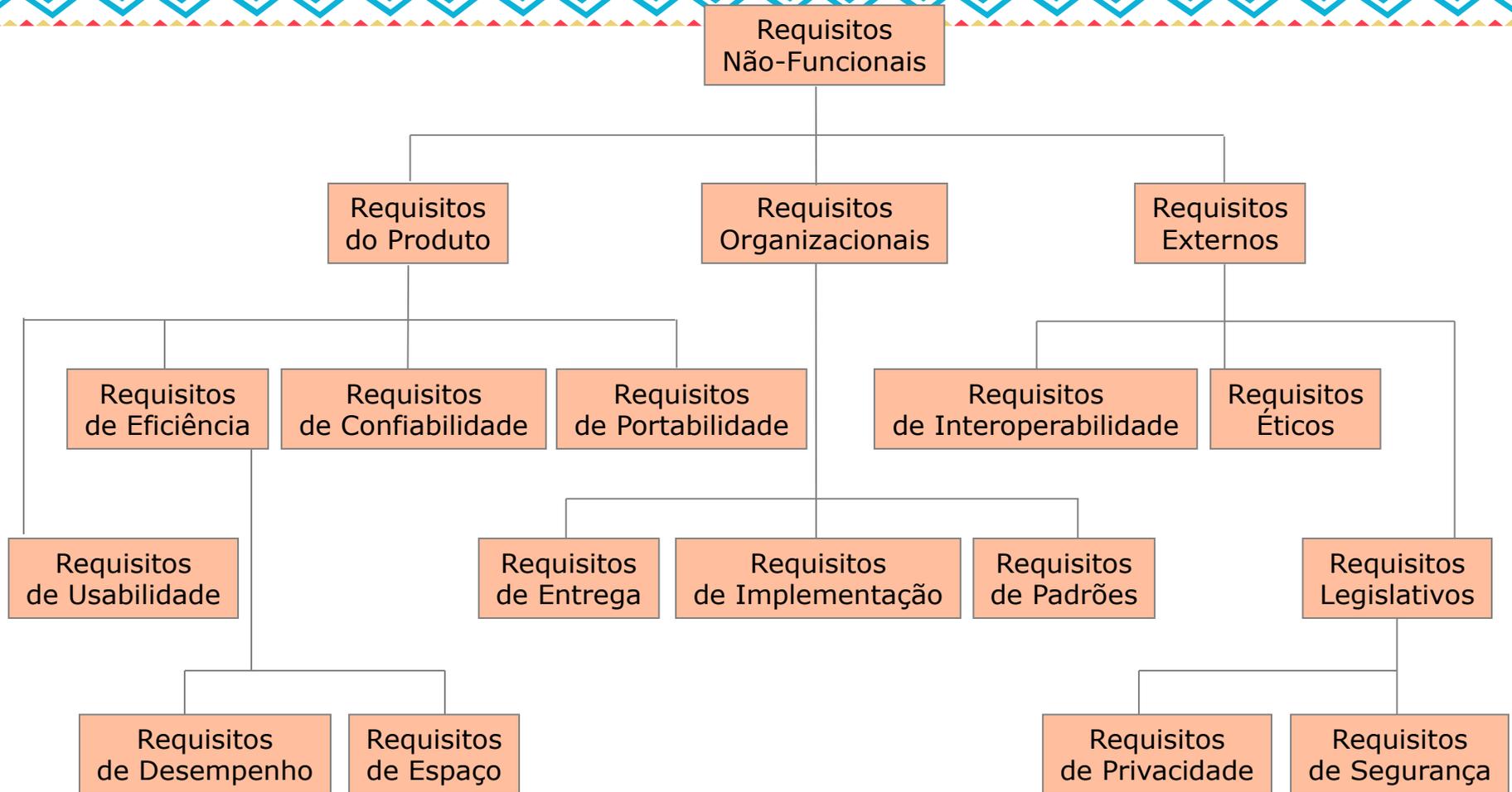
⇒ São requisitos que expressam:

☞ **Restrições** que o software deve atender.

☞ **Qualidades específicas** que o software deve ter.

Requisitos Não-Funcionais

Tipos



Requisitos Não-Funcionais

Exemplos



⇒ Requisitos do Produto

- ☞ O sistema deve ser robusto e tolerante a falhas, de forma a continuar sua operação ou abortar de forma segura o modo autônomo caso haja falha de um ou mais sistemas essenciais

⇒ Requisitos Organizacionais

- ☞ O processo de desenvolvimento do sistema e os produtos liberáveis devem estar em conformidade com o padrão empresarial XYZ.

⇒ Requisitos Externos

- ☞ Os operadores do sistema não devem ter acesso a qualquer dado que não necessitem.

Requisitos Não-Funcionais

Metas

- ⇒ Requisitos Não-Funcionais podem ser muito difíceis de serem declarados precisamente.
- ⇒ Podem ser utilizadas “**Metas**”.
 - ☞ Transmitem as intenções dos usuários do sistema.
 - ☞ **Exemplo:** O sistema de controle de aeronave deve ser fácil de ser usado por controladores experientes e deve estar organizado de tal maneira que os erros dos usuários sejam minimizados.

Requisitos Não-Funcionais

Interação entre Requisitos



- ⇒ Em sistemas complexos são comuns conflitos entre diferentes Requisitos Não-Funcionais.
- ⇒ Exemplo: Sistema para aeronaves.
 - ☞ Para minimizar o peso, o número de chips do sistema deve ser minimizado.
 - ☞ Para minimizar o consumo de energia, chips de menor potência devem ser usados.
 - ☞ *Entretanto, usar chips de menor potência pode significar que mais chips devem ser usados. Qual é o requisito mais crítico?*

Exercícios de Fixação



Identifique os **requisitos funcionais** e **não funcionais**. 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”.



Exercícios de Fixação



Requisitos funcionais (RF), não funcionais (RNF):

RF1: 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.

RF2: Uma vez selecionado um destino, pede-se que os usuários insiram seu cartão de crédito.

RF3: O sistema deve informar se existem vagas no destino escolhido

RF3: A validade do cartão é checada e o usuário então deve fornecer um número de identificação pessoal.

RF4: Quando a transação de crédito é validada, a passagem é emitida e o custo dessa passagem é incluído em sua conta do cartão de crédito.

RNF1: As telas devem facilitar a escolha do destino

RNF2: O tempo de resposta sobre vaga no trem deve ser adequado

RNF3: O formato do bilhete de passagem deve seguir ao padrão definido pelo Sistema Nacional de Tráfego Ferroviário”.

Exercícios de Fixação



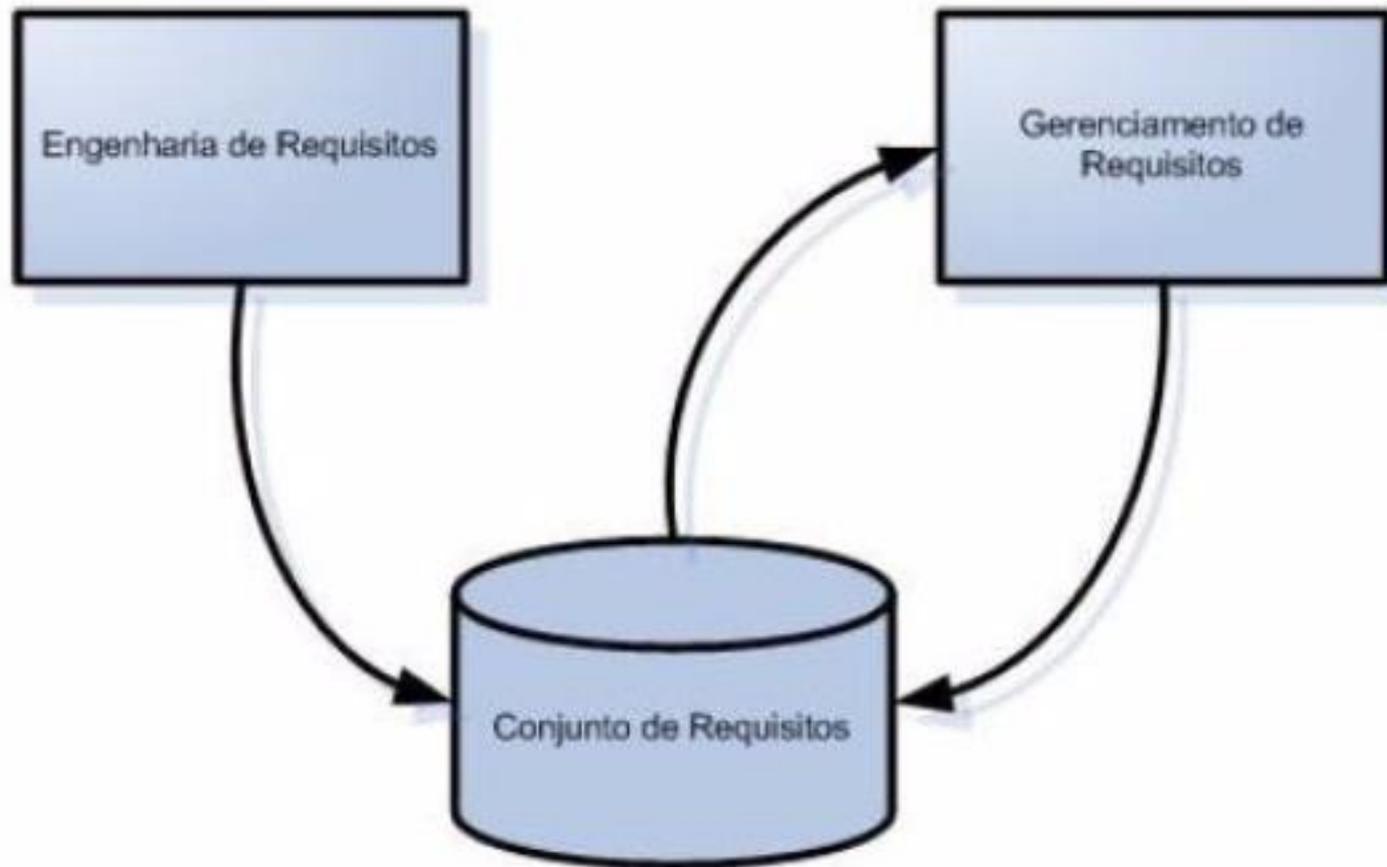
Aponte possíveis incertezas nessa descrição.

“Um sistema automático de emissão de passagens vende passagens de trem. Os usuários escolhem seu destino e apresentam um cartão de crédito e um número de identificação pessoal. 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”.

Engenharia de requisitos

- 
- Os requisitos e as formas de obtê-los e documentá-los **variam** drasticamente de um projeto para o outro
 - Contudo, existe uma série de atividades genéricas comuns a todos os processos:
 - Extração (elicitação) de requisitos;
 - Análise de requisitos;
 - Validação de requisitos;
 - Gerenciamento de requisitos.

Engenharia e Gerenciamento de Requisitos



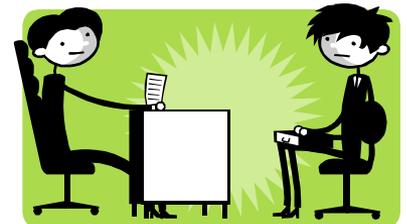
Extração de Requisitos de Software



Dilbert by Scott Adams, 2006

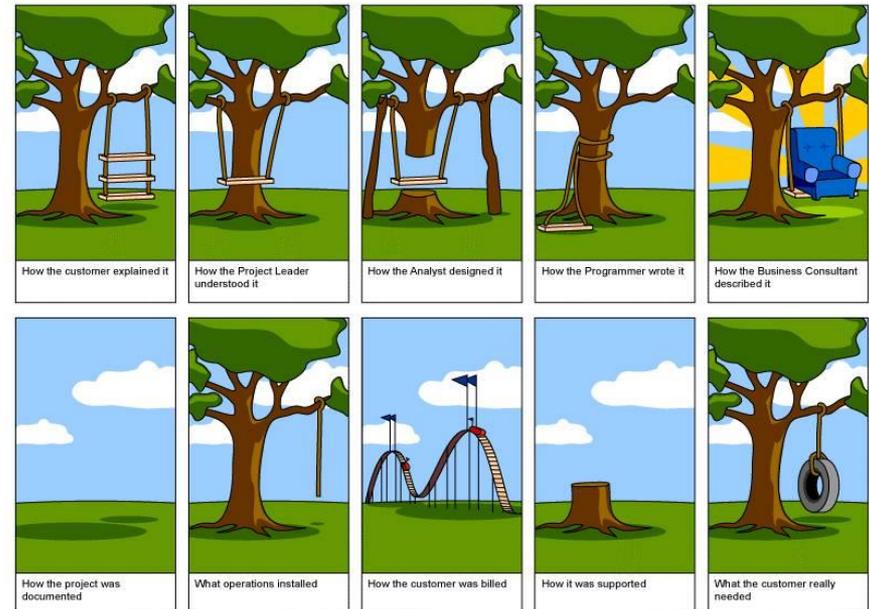
Extração de Requisitos de Software

- Envolve pessoal técnico trabalhando com os clientes para descobrir sobre o **domínio da aplicação**, os **serviços que o sistema deve fornecer** e sobre as **restrições operacionais**.
- Pode envolver:
 - Usuários finais
 - Gerentes
 - Engenheiros envolvidos na manutenção
 - especialistas de domínio
 - representantes de sindicato, etc.
 - Estes são chamados **stakeholders** (partes interessadas)

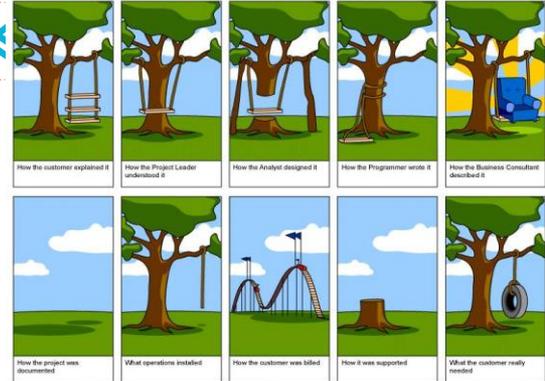


Extração de Requisitos de Software

Problemas?



Extração de Requisitos de Software



Problemas?

Stakeholders não sabem o querem do software

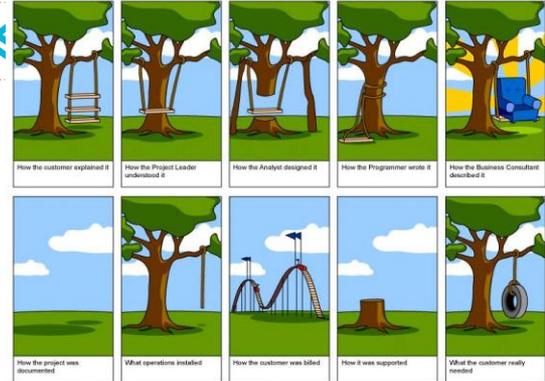
Stakeholder não sabe

explicar o quer do software

Stakeholders usam sua própria

linguagem

Extração de Requisitos de Software



Problemas?

Stakeholders podem ter requisitos conflitantes

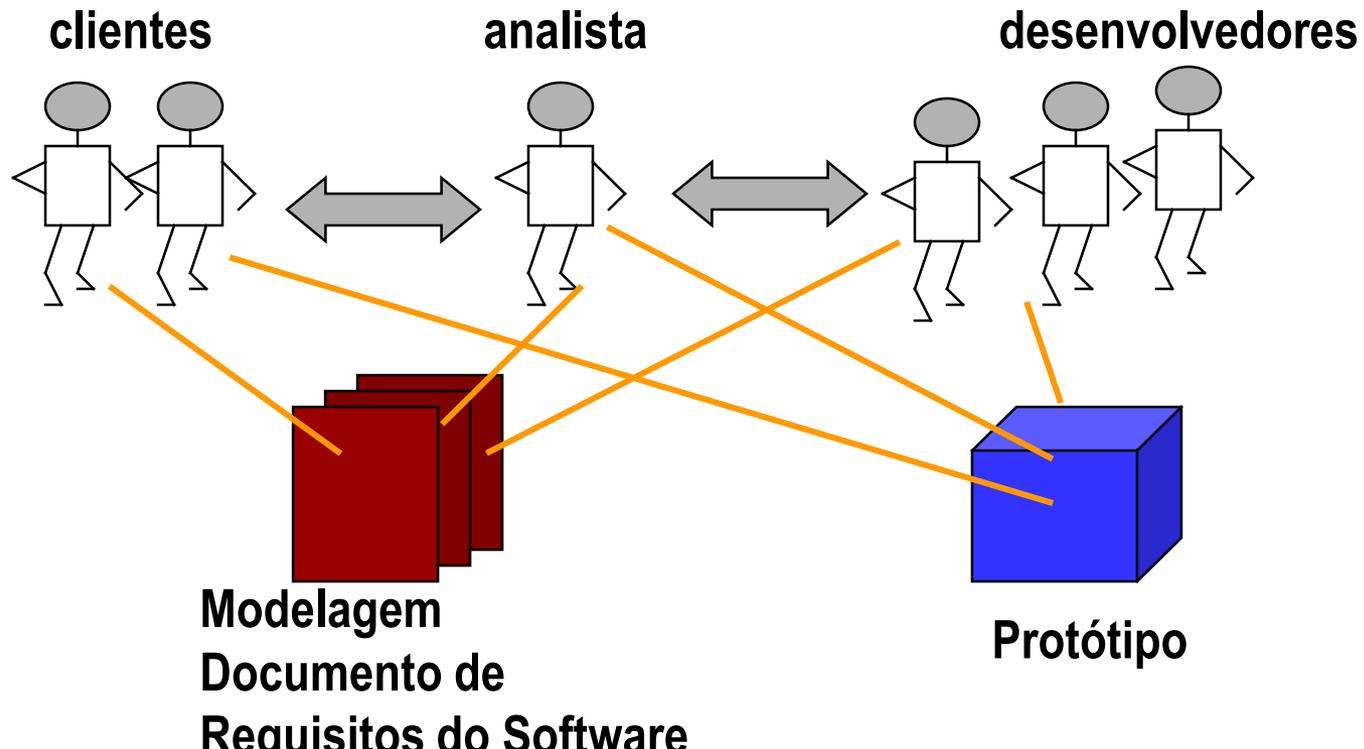
Fatores organizacionais podem influenciar

Requisitos mudam durante a
engenharia de requisitos

Extração dos Requisitos

Extração de requisitos é o processo de transformação das idéias que estão na mente dos usuários (a entrada) em um documento formal (saída).

A meta é o reconhecimento dos elementos básicos do problema, conforme percebidos pelo cliente.



Extração dos Requisitos

- 
- ⇒ **Processo crítico** em um projeto de software
 - ⇒ Requisitos incompletos, incorretos ou mal entendidos são as causas mais frequentes **da baixa qualidade, excesso de custo e atrasos nas liberações do software**
 - ⇒ Pesquisas têm mostrado que a maioria dos softwares vendidos **não satisfaz as necessidades do usuário**

Extração de Requisitos

⇒ Algumas técnicas são propostas visando auxiliar a comunicação e a extração dos requisitos

☞ *Entrevistas*

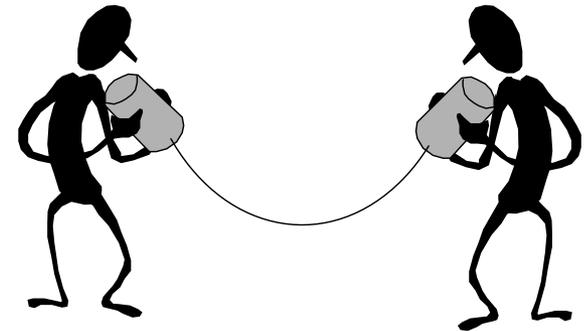
☞ *Cenários*

☞ *Estórias do usuário*

☞ *Etnografia*

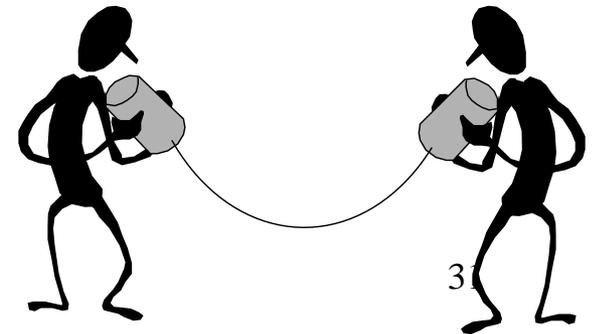
☞ *Brainstorm*

☞ *Prototipação*



Extração de Requisitos: Entrevista

- ⇒ Em entrevista formal ou informal, a equipe de ER formula questões para os *stakeholders* sobre os sistemas que eles usam e o sistema a ser desenvolvido.
- ⇒ Dois tipos de entrevistas:
 - ☞ **Entrevistas fechadas:** um conjunto de questões predefinidas são respondidas.
 - ☞ **Entrevistas abertas:** não há um roteiro predefinido, uma variedade de assuntos são explorados com os stakeholders.



Extração de Requisitos: Entrevista



⇒ Planejamento da entrevista

⇒ **Início:** Questões livres de contexto (Quebrar o gelo!)

☞ *Quem está por trás da solicitação deste trabalho?*

☞ *Quem vai usar a solução?*

☞ *Qual será o benefício econômico para uma solução bem-sucedida?*

Extração de Requisitos: Entrevista



⇒ Questões que ajudam a entender o problema:

- ☞ *Você pode me mostrar ou descrever o ambiente no qual a solução será usada?*
- ☞ *Que tipo de saídas você considera importante?*
- ☞ *Que problemas existem para a solução de software?*
- ☞ *Existem questões de desempenho ou restrições que podem afetar o software?*

Extração de Requisitos: Entrevista



⇒ **Final: focalizam a efetividade da reunião**

- ☞ *Você é a pessoa certa para responder a essas questões? Suas respostas são “oficiais”?*
- ☞ *Minhas questões são relevantes para o problema que você tem?*
- ☞ *Estou formulando muitas questões?*
- ☞ *Alguém mais pode fornecer informação adicional?*
- ☞ *Tem alguma questão que não fiz que você julga pertinente?*

Extração de Requisitos: Entrevista



- ⇒ Entrevistas são boas para obtenção de um entendimento geral do que os **stakeholders** fazem e como eles podem interagir com o sistema.
- ⇒ Entrevistas não são boas para a compreensão de **requisitos de domínio**
 - ☞ Os engenheiros de requisitos podem não entender a terminologia específica de domínio;
 - ☞ Alguns conhecimentos de domínio são tão específicos que as pessoas acham difícil explicar ou pensam que não valem a pena mencioná-los

Extração de Requisitos: Entrevista

⇒ ENTREVISTAS EFETIVAS

- ⇒ Os entrevistadores devem ter mente aberta, desejarem **ouvir** os stakeholders e **não ter idéias preconcebidas** sobre os requisitos.
- ⇒ Eles devem induzir os entrevistados com uma questão ou uma proposta, e não simplesmente esperar que eles respondam a uma questão tal como 'o que você quer?'



Extração de Requisitos: Cenários



⇒ CENÁRIOS

⇒ Cenários são exemplos reais de como um sistema pode ser usado.

⇒ Eles devem incluir

- ☞ Uma descrição da situação inicial;
- ☞ Uma descrição do fluxo normal de eventos;
- ☞ Uma descrição do que pode dar errado;
- ☞ Informação sobre outras atividades concorrentes;
- ☞ Uma descrição do estado quando o cenário termina.

Extração de Requisitos: Cenários

⇒ Exemplos de cenário?

- ☞ Saque em caixa eletrônico
- ☞ Empréstimo de livro em biblioteca
- ☞ Compra de livro na internet
- ☞ Pilotar um avião

- ☞ Descrição da situação inicial;
- ☞ Descrição do fluxo normal de eventos;
- ☞ Descrição do que pode dar errado;
- ☞ Informação sobre outras atividades concorrentes;
- ☞ Descrição do estado quando o cenário termina.

Extração de Requisitos: Estórias



- ⇒ São frases escritas **pelo cliente** na sua linguagem, sobre algo que a aplicação deve fazer.
- ⇒ Detalhes de cada estória não aparecem:
 - ☞ uma estória é “uma promessa de uma conversa futura entre cliente e desenvolvedores”.

Extração de Requisitos: Estórias



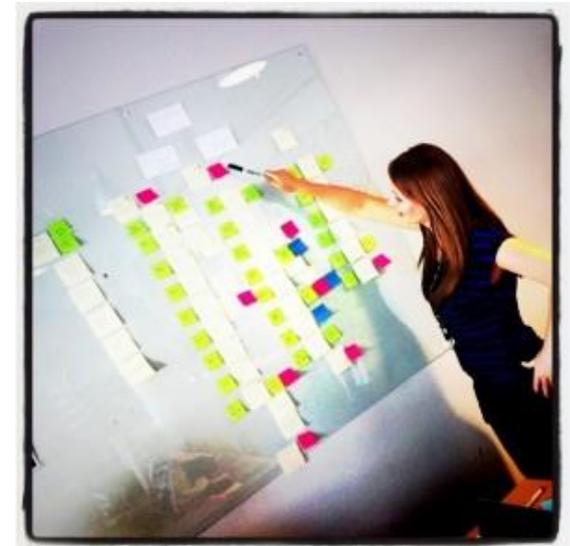
⇒ Exemplo de estórias

⇒ Para uma loja virtual:

- ☞ “Um usuário possui um carrinho de compras no qual ele adiciona produtos que quer comprar”
- ☞ “Um usuário faz o pagamento com cartão de crédito ou boleto bancário”
- ☞ “Um usuário lê comentários feitos por outros sobre os produtos da loja”
- ☞ “Um usuário recebe um e-mail de confirmação de compra quando efetua um pagamento”.

Extração de Requisitos: Estórias

- ⇒ As estórias conduzem novas reuniões com usuários que podem ocorrer durante a fase de desenvolvimento.
- ⇒ Feitas em cartões (manuscritas) que serão fixados em painéis
 - ☞ Ajudam a acompanhar o desenvolvimento (estória concluída, em desenvolvimento, não iniciado)
- ⇒ Auxiliam durante os **testes de aceitação**



Extração de Requisitos: Etnografia



- ⇒ Um analista observa e analisa como as pessoas realmente trabalham.
 - ☞ *As pessoas não explicam seu trabalho.*
- ⇒ Fatores sociais e organizacionais de importância podem ser observados.
- ⇒ Estudos de etnografia têm mostrado que o trabalho é, geralmente, mais rico e mais complexo do que o sugerido pelos modelos simples de sistema.

Extração de Requisitos: Etnografia



- ⇒ Requisitos do sistema se originam do modo como as pessoas realmente trabalham
 - ☞ Independem de como definições de processo sugerem que elas devam trabalhar.

- ⇒ Ideal complementar com prototipação

Extração de Requisitos

⇒ Brainstorming

- ☞ Tempestade de idéias, uma técnica realizada em grupo para gerar idéias relativas a um determinado assunto.



Extração de Requisitos



⇒ Brainstorming

- ☞ Útil na geração de uma ampla variedade de pontos de vista sobre o problema e na formulação do mesmo de diferentes maneiras
- ☞ Útil no início do processo de extração de requisitos
- ☞ Técnica básica para geração de idéias
- ☞ Permite que pessoas sugiram e explorem idéias sem que sejam criticadas e julgadas
- ☞ Funciona melhor com um número mínimo de quatro e máximo de dez pessoas

Extração de Requisitos



⇒ Brainstorming

☞ Se aplicada corretamente, pode ajudar em algumas dificuldades implícitas da extração de requisitos:

- Estimula o pensamento imaginativo para ajudar os usuários a se tornarem cientes das suas necessidades;
- Ajuda a construir um quadro mais completo das necessidades dos usuários
- Evita a tendência em limitar o problema muito cedo
- Para algumas pessoas, fornece uma interação social mais confortável (menos formal)

Extração de Requisitos

⇒ Brainstorming - Sessão

- ☞ Proibido criticar as idéias (não limitar a criatividade);
- ☞ Idéias não convencionais são encorajadas – podem levar a boas soluções;
- ☞ Geração de um número bem grande de idéias (aumentam as chances de boas idéias);
- ☞ Encorajar a combinação e o enriquecimento de idéias.



Extração de Requisitos

⇒ Brainstorming – Fases:

1. **Fase de Geração:** participantes são encorajados a fornecer idéias, sem discussão quanto ao mérito das mesmas
2. **Fase de Consolidação:** as idéias são discutidas, revisadas e organizadas



Extração de Requisitos

⇒ Brainstorming

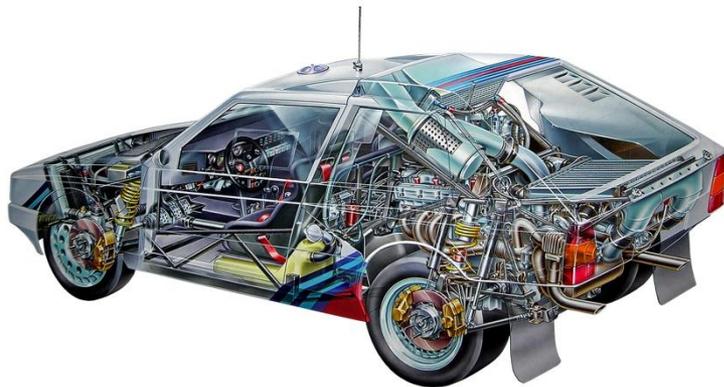
- ☞ Útil para sanar limitações cognitivas dos participantes
- ☞ Ausência de críticas ajuda a eliminar barreiras de extração de requisitos
- ☞ Fácil de aprender e requer pouco investimento
- ☞ Desvantagem: técnica pouco estruturada que pode não produzir a mesma qualidade ou nível de detalhes de outras técnicas



Extração de Requisitos

⇒ Prototipação

- ☞ Construção de modelos que representam o software
- ☞ Protótipos executáveis (papel ou software)
- ☞ <http://www.youtube.com/watch?v=k9mTvt0LXgk>



Problemas com Requisitos

⇒ Erros mais comuns cometidos no desenvolvimento:

- ☞ Ignorar um grupo de clientes.
- ☞ Ignorar um único cliente.
- ☞ Omitir um grupo de requisitos.
- ☞ Permitir inconsistências entre grupos de requisitos.
- ☞ Aceitar requisito inadequado.
- ☞ Aceitar requisito incorreto, indefinido, ou impreciso.
- ☞ Aceitar um requisito ambíguo e inconsistente.

Conclusão



⇒ Análise de Requisitos

⇒ primeiro passo do processo de engenharia de software.

☞ Especificação do software serve de base para todas as atividades de engenharia de software

⇒ Concentra-se nos domínios funcionais, comportamentais e de informação de um problema

⇒ O Documento de Requisitos de Software serve como um "**contrato**" de desenvolvimento entre o cliente e o desenvolvedor

☞ Entretanto: mesmo com o melhor dos métodos, “o problema” é que o “problema” continua mudando (volatilidade)