

Introdução a Requisitos De Software

Prof. André Takeshi Endo

Roteiro

- O que é um requisito
- Motivação
- Tipos de requisitos
 - Requisitos funcionais
 - Requisitos não funcionais
 - Requisitos de domínio
- Engenharia de requisitos
- Conclusão

O que vocês entendem por requisito?

- Definições

requirement

noun [C] UK  US  /rɪ'kwaɪə.mənt/ (US) /-'kwaɪr-/

B2 something that you must do, or something you need:

*A good degree is a minimum requirement **for** many jobs.*

[+ that] *It is a legal requirement **that** you have insurance for your car.*

*Students who fail to **meet** the requirements (**of** the course) will fail.*

Significado de Requisito

s.m. Condição que se deve satisfazer para alcançar certo fim.

Exigência de ordem legal para que determinado processo possa ter andamento.

Sinônimos de Requisito

Sinônimo de requisito: [condição](#), [exigência](#) e [quesito](#)

O que é um requisito? (ES)

- Definições:
 - “Um requisito é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar, para atingir os objetivos.” [Pfleeger]
 - “Requisitos são as descrições das funções e restrições sob o software.” [Sommerville]

O que é um requisito?

- Definições:
 - [IEEE]

requirement. (1) A condition or capability needed by a user to solve a problem or achieve an objective.

(2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.

(3) A documented representation of a condition or capability as in (1) or (2).

See also: **design requirement; functional requirement; implementation requirement; interface requirement; performance requirement; physical requirement.**

O que é um requisito?

- Definições [McLaughlin]:
 - Um requisito é uma “coisa” específica que o sistema deve possuir para satisfazer o cliente
 - Um requisito é algo que o sistema precisa fazer
 - Um requisito é uma **necessidade singular** que detalha o que um produto ou serviço particular deve **ser** ou **fazer**.

O que é um requisito?

- Requisitos podem ser especificados de diversas formas
- Notação matemática
- Apoiada pela UML
- Inicialmente, vamos considerar um descrição *textual em linguagem natural*

O que é um requisito?

- Exemplo
 - Considere o estudo de caso (Moodle)
- O que houve?
 - Cliente → Professor + Alunos
 - Elicitaram (identificaram)
 - Especificaram (escreveram)

Motivação

- O cliente sabe o que é necessário?
- Os usuários têm um bom entendimento das características e funções que trarão benefícios?
- Sim!
- Não!
- Como expressamos o desejo do cliente?

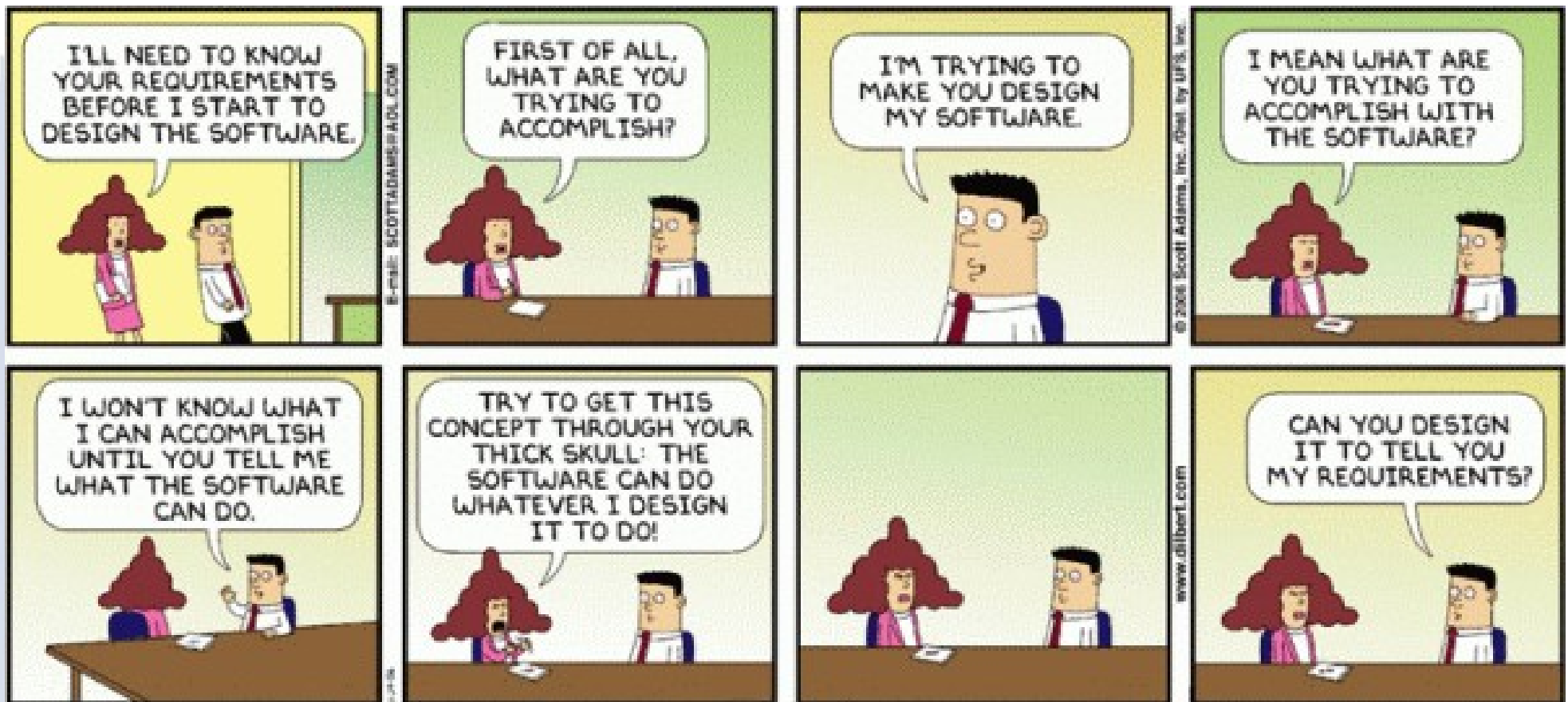
Motivação

...



Motivação

- O engenheiro e o cliente



Motivação



How the customer explained it



How the Project Leader understood it



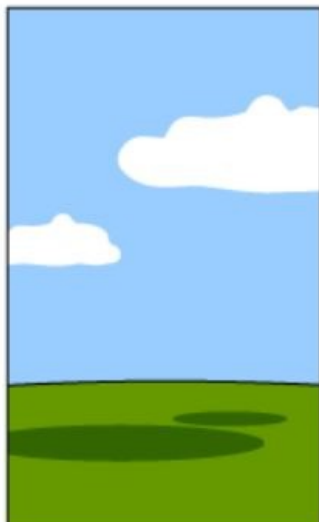
How the Analyst designed it



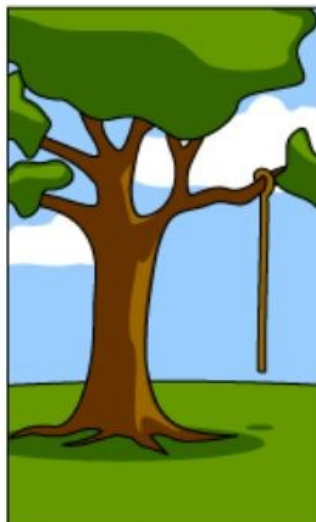
How the Programmer wrote it



How the Business Consultant described it



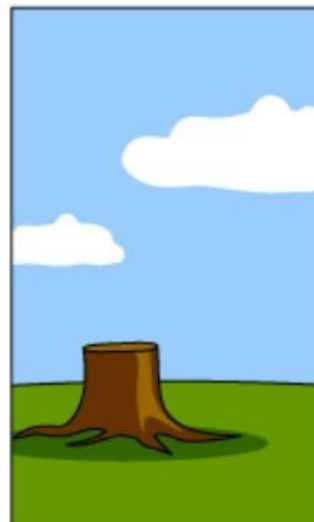
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

Motivação

- Em estágios iniciais de desenvolvimento...
- Como eu sei o que fazer?
- Como o cliente sabe o que eu vou fazer?
- Como eu defino um contrato?
- Como eu planejo?

Motivação

- Problemas comuns:
 - 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

Motivação

- Custo de correção
 - Reparo
 - Perda de oportunidades
 - Perda de clientes
- “Erros mais caros são aqueles cometidos na análise do requisitos e descobertos pelos usuários”
- Gráfico (custo x tempo)

Motivação

- Fred Brooks: “A parte mais difícil ao construir um sistema de software é decidir **o que** construir. Nenhuma parte do trabalho **afeta tanto** o sistema resultante se for feita a coisa errada. Nenhuma parte é **mais difícil** de consertar depois.”

Motivação

- A resposta para as perguntas anteriores
 - Requisitos
 - Engenharia de Requisitos

Tipos de Requisitos

- Existem diversas classificações
- A seguinte classificação [Sommerville]:
 - Requisitos funcionais
 - Requisitos não funcionais
 - Requisitos de domínio

Requisitos Funcionais

- São 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 procurar em todo o conjunto de dados armazenado no banco ou selecionar um subconjunto dele.
- O *sistema deve* fornecer telas apropriadas para o usuário ler documentos no repositório de documentos.
- A todo pedido deve ser alocado um identificador único (ORDER_ID) o qual poderá ser copiado para a área de armazenagem permanente da conta.

Requisitos Funcionais - Exemplos

- Piscina

- O sistema deve ligar o motor da limpeza da piscina quando a qualidade da água for inferior ao padrão pré-estabelecido pelo proprietário.
- O sistema deve enviar um sms para o proprietário quando for necessário adicionar produtos químicos ao reservatório ou fazer manutenção no motor.

- Acesso

- O sistema deve se comunicar com o celular do dono da casa para armar e desarmar o alarme, além da abertura do portão eletrônico, diminuindo o tempo de espera do cliente do lado de fora da casa enquanto o portão abre.

- Monitoramento

- O sistema deve permitir ao usuário monitoramento total da casa por meio de câmeras de segurança, enviando para o seu aparelho celular e notebook, imagens ao vivo das dependências da residência.

Requisitos - 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.

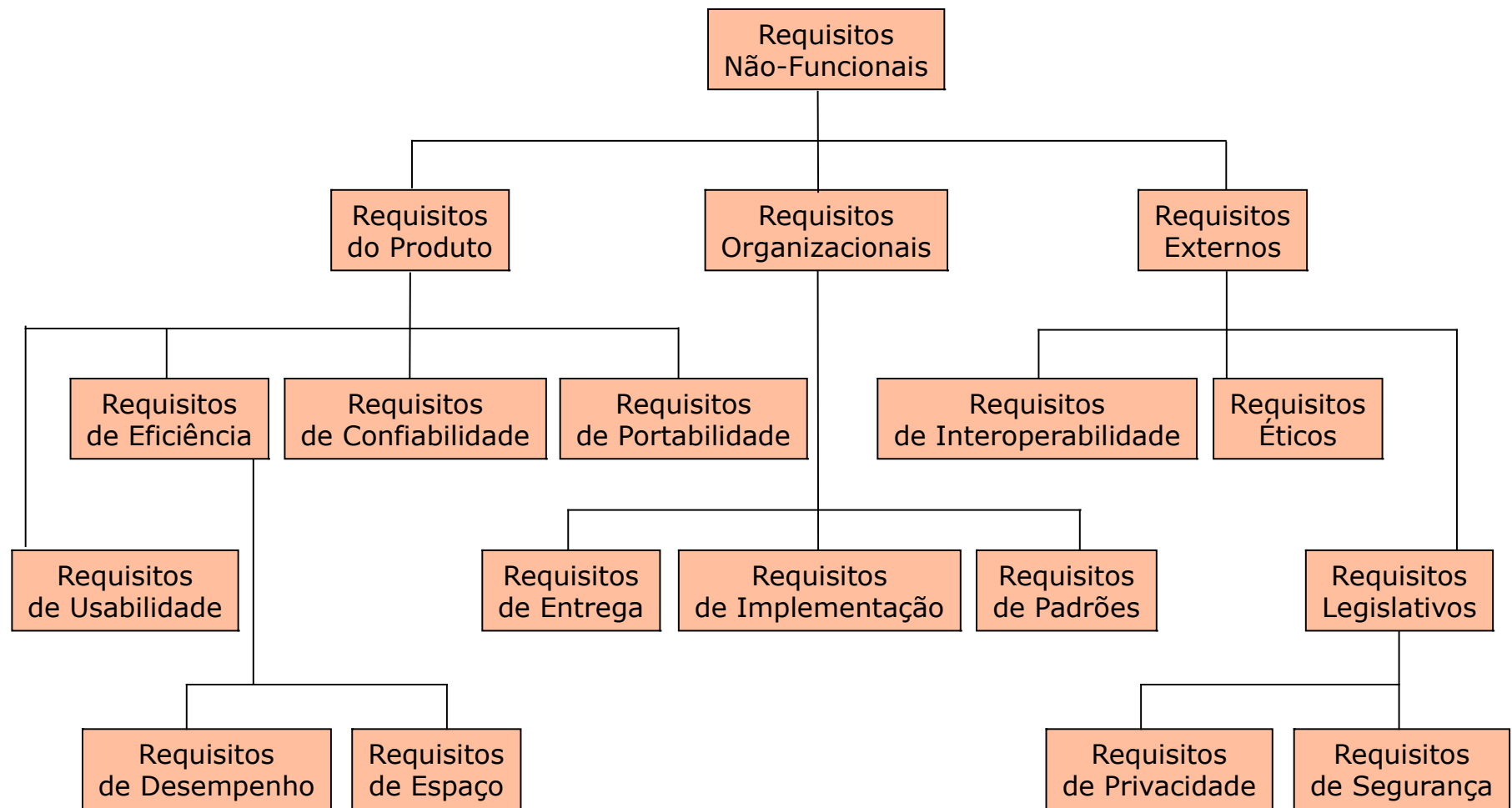
Requisitos – Completeza e Clareza

- 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.

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
 - Toda comunicação necessária entre o sistema e o usuário deve ser expressa usando o conjunto padrão de caracteres Unicode.
- Requisitos Organizacionais
 - O processo de desenvolvimento do sistema e os produtos liberáveis devem estar em conformidade segundo o padrão MY101.
- Requisitos Externos
 - Os operadores do sistema não devem ter acesso a qualquer dado que não necessitem.

Requisitos Não Funcionais

- 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 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 – Verificáveis?

- Requisitos Não-Funcionais imprecisos → difíceis de verificar
- Incluir mais informação
- Uma declaração que usa alguma **medida** 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 – Verificáveis?

- Algumas métricas

Requisito	Métrica
Velocidade	Transações processadas/segundo Tempo de resposta ao usuário/evento Tempo de refresh da tela
Tamanho	K Bytes Tamanho específico de memória
Facilidade de uso	Tempo de treinamento Número de frames de Help
Confiabilidade	Tempo médio de falha Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade
Robustez	Tempo de reinício depois de falha Porcentagem de eventos que causam falhas Probabilidade de que dados sejam corrompidos por falhas
Portabilidade	Porcentagem de declarações dependentes do sistema alvo Número de sistemas alvo

Requisitos Não Funcionais - Exemplos

- “O sistema da casa inteligente deve conter um gerador capaz de armazenar energia suficiente para suprir as necessidades elétricas da casa por pelo **menos 12 horas** quando a energia acabar. Uma opção é utilizar placas fotovoltaicas para armazenar energia solar durante o dia.”

Requisitos Não Funcionais – Verificáveis?

- 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?

Requisitos de Domínio

- São requisitos que são próprios do **domínio da aplicação** e que refletem características desse domínio
 - Podem ser **novos requisitos funcionais**, restrições sobre os requisitos existentes ou definições sobre computações específicas.
 - Se os requisitos do domínio não são satisfeitos, pode não ser possível operar o sistema de forma efetiva.

Requisitos de Domínio - Exemplos

- Deve haver uma interface-padrão com o usuário para todas as bases de dados, a qual deve estar baseada no padrão Z39.50
- Devido a restrições de direitos autorais, alguns documentos devem ser eliminados imediatamente à sua chegada. Dependendo dos requisitos do usuário, esses documentos serão impressos localmente no servidor do sistema para serem encaminhados manualmente ao usuário ou direcionados para uma impressora em rede

Requisitos de Domínio - Exemplos

- A desaceleração do trem deve ser calculada como:
- $D_{\text{trem}} = D_{\text{controle}} + D_{\text{gradiente}}$, onde:
- $D_{\text{gradiente}}$ é $9.81\text{ms}^2 * \text{gradiente compensado}/\alpha$
- e onde os valores de $9.81\text{ms}^2 / \alpha$ são conhecidos para diferentes tipos de trem

Requisitos de Domínio - Problemas

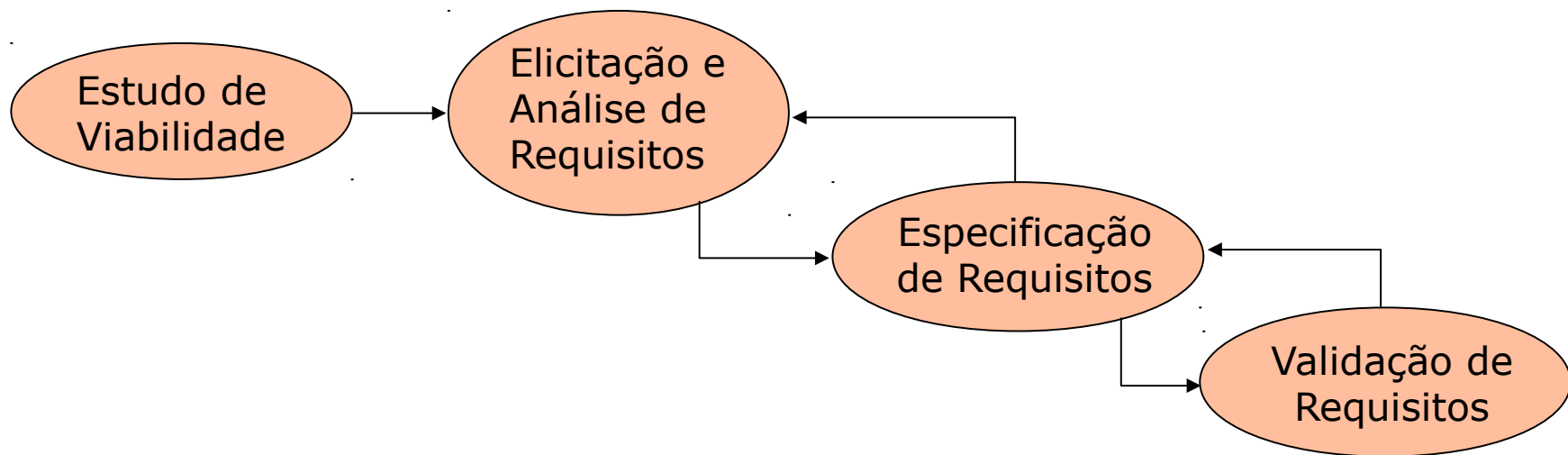
- Dificuldade de compreensão!!!
- Os requisitos são expressos em uma ***linguagem própria*** do domínio da aplicação
- Em geral, essa linguagem não é compreendida pelos engenheiros
- Conhecimento tácito
- Os especialistas no domínio entendem tão bem da área que não consideram necessário ***explicitar os requisitos do domínio***

Engenharia de Requisitos

- “É o **processo** de estabelecer os **serviços** que o cliente requer do sistema e as **restrições** sob as quais o sistema deve operar e ser desenvolvido.”
- Descobrir, especificar, analisar e validar
- Artefato principal:
 - ***Documento de requisitos***

Engenharia de Requisitos

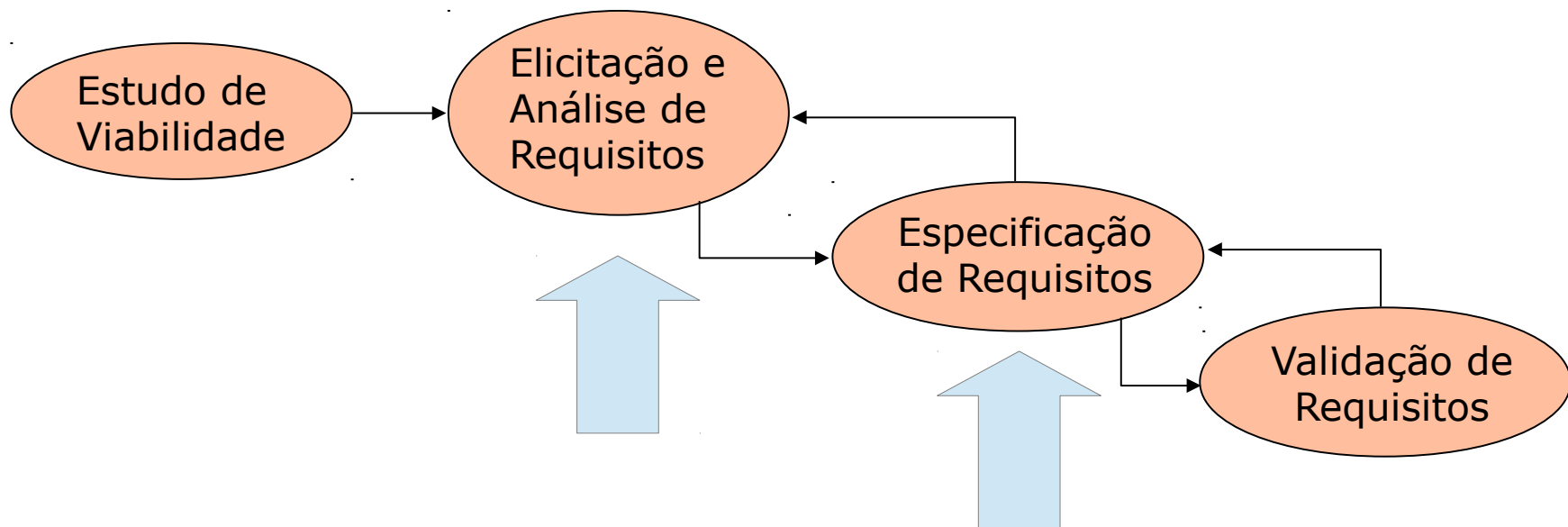
- Representação



Q: paralelo com o estudo de caso!

Engenharia de Requisitos

- Representação



Engenharia de Requisitos

- Como os requisitos são especificados?
- Como especificar os requisitos?
- Usar exemplos!!!

Bibliografia

- [Pfleeger07] S. L. Pfleeger, “Engenharia de Software: Teoria e Prática”, 2007.
- [Pressman11] R. S. Pressman, “Engenharia de Software: uma abordagem profissional”, 2011.
- [Sommerville03] I. Sommerville, “Engenharia de Software”, 2003.
- [Brooks87] “No Silver Bullet: Essence and Accidents of Software Engineering”, 1987.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1663532
- [IEEE90] “IEEE Standard Glossary of Software Engineering Terminology”, 1990.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=159342

Bibliografia

- [McLaughlin] Brett D. McLaughlin, Gary Pollice, Dave West. “Head First – Object Oriented Analysis & Design”. O'Reilly Media, 2007.
- [UUU] Materiais didáticos elaborados pelos grupos de engenharia de software do ICMC-USP, DC-UFSCAR e UTFPR-CP.
- Partes dessa apresentação foram adaptadas do material da profa. Ellen Francine.