



Engenharia de Requisitos

Prof. Ingrid Nunes

INF01127 - Engenharia de Software N



Engenharia de Requisitos



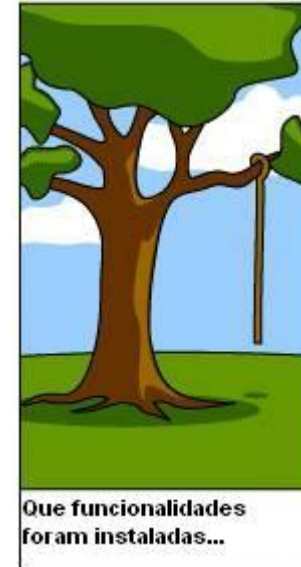
INTRODUÇÃO

Introdução



“A principal medida de sucesso de um sistema de software é o grau em que ele **atende o propósito** para o qual foi concebido”

(Nusebeich & Easterbrook)



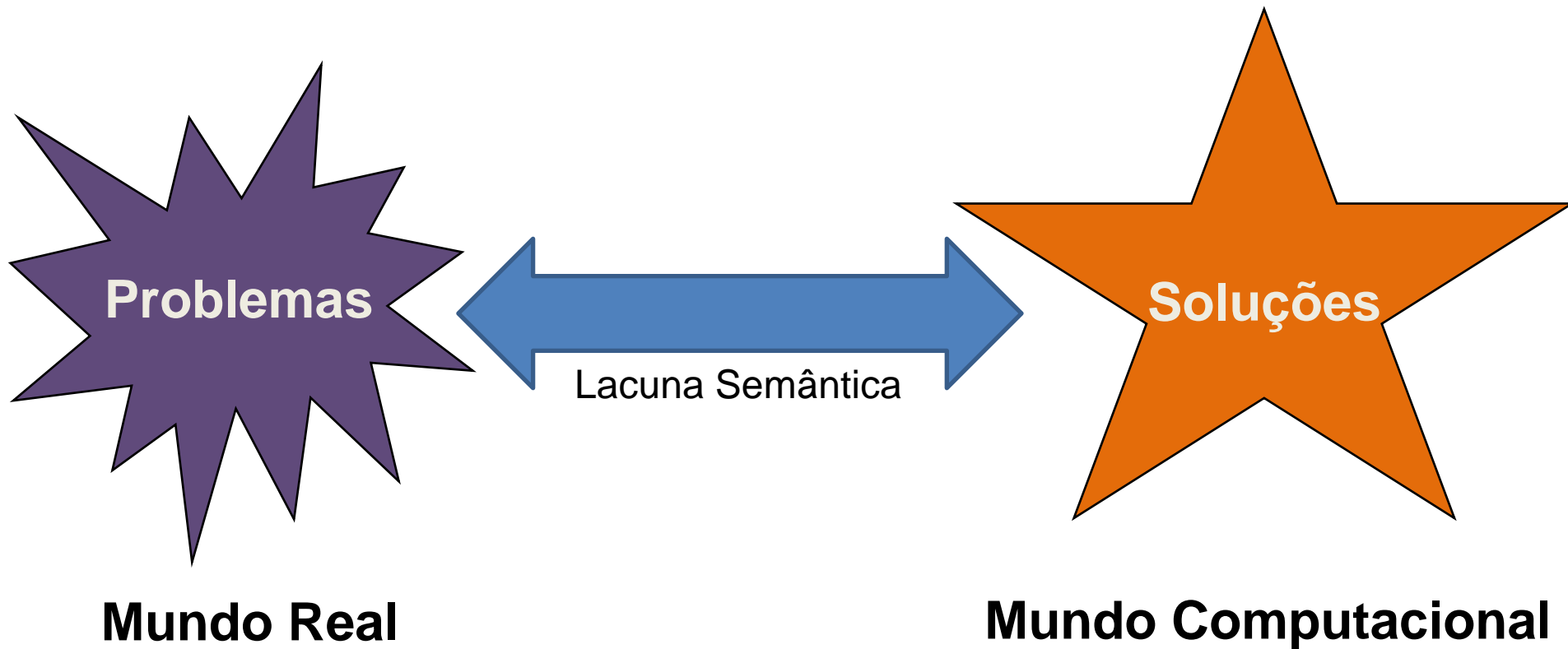
Introdução



“A parte mais difícil na construção de um sistema de software é decidir precisamente o que construir. Nenhuma outra parte do trabalho conceitual é tão difícil quanto estabelecer requisitos técnicos detalhados, incluindo todas as interfaces para pessoas, máquinas e outros sistemas de software. Nenhuma outra parte do trabalho prejudica o sistema resultante caso seja feita incorretamente. Nenhuma outra parte é tão difícil de corrigir mais tarde.”

F. Brooks

Engenharia de Requisitos



Engenharia de Requisitos





Customer requirement



1. Have one trunk
2. Have four legs
3. Should carry load both passenger & cargo
4. Black in color
5. Should be herbivorous

Engenharia de Requisitos



Customer requirement	Our Solution
	
<ol style="list-style-type: none">1. Have one trunk2. Have four legs3. Should carry load both passenger & cargo4. Black in color5. Should be herbivorous	<ol style="list-style-type: none">1. Have one trunk <input checked="" type="checkbox"/>2. Have four legs <input checked="" type="checkbox"/>3. Should carry load both passenger & cargo <input checked="" type="checkbox"/>4. Black in color <input checked="" type="checkbox"/>5. Should be herbivorous <input checked="" type="checkbox"/> <p>Our Value add: Also gives milk 😊</p>

Visão Histórica: Análise



- Análise
 - Estudo do Universo: entender o problema e seu contexto
 - Especificação do sistema: O QUÊ o sistema deve fazer
- Abordagens Principais
 - Orientadas a Atividades (Funções)
 - Análise Estruturada (SSA), Análise Estruturada Moderna, SADT, etc.
 - Orientadas a Dados
 - ER, JSD, Engenharia da Informação, etc.
 - Orientadas a Estados
 - Statecharts, Redes de Petri, etc.
 - Orientadas a Objetos
 - OOA (Coad/Yourdon), OMT, Objectory, Fusion, UML etc.

Requisitos: Análise Tradicional

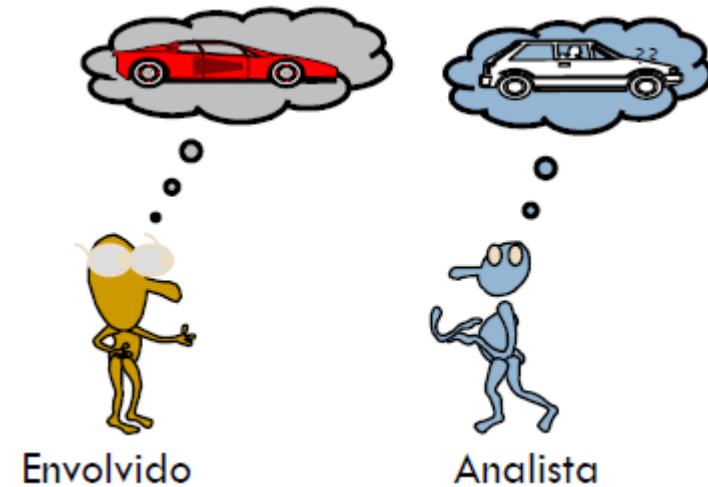


- Parte de um conjunto de premissas que dificilmente ocorrem
 - Há um problema bem definido e estável
 - Especificação de Requisitos = **contrato “congelado”** entre analista e cliente (\neq usuário)
 - **Cliente** conhece bem seu domínio e **sabe bem o que quer**
 - Basta saber lhe interrogar corretamente
 - Requisitos são “coletados”
 - Processo linear baseado em abstrações
 - É possível atingir um conjunto completo, coerente e não ambíguo de requisitos mesmo sem a participação do usuário

Requisitos: Conflitos



- Envolvidos (“*stakeholders*”)
 - **Não sabem** o que realmente querem
 - **Não conseguem** articular o que querem
 - **Pensam** que sabem o que querem
 - Mas não o reconhecem quando entregue
- Analistas
 - Pensam que entende os problemas dos usuários melhor que os próprios
- Todos
 - Acreditam que todos estão politicamente motivados



Requisitos: Realidade



- Problema a ser resolvido (e seu domínio) **não é claramente delimitado nem descrito** e está em constante mudança
 - Requisitos mudam mesmo durante o desenvolvimento
- Cliente sabe que tem um problema, mas **não necessariamente como resolvê-lo**
 - Em geral, suas exigências emergem ou se modificam a partir de interações com o analista e versões preliminares do sistema
 - Suas prioridades mudam, seu entendimento sobre o sistema também
- Nem sempre se chega a especificações **corretas e completas**
 - Deve-se lidar com diferentes graus de incompletude e inconsistência e atribuir prioridades diferentes aos requisitos
- Requisitos existem em um **contexto organizacional**
 - Existem interesses políticos diversos que influenciam o processo
 - Conflitos, negociações

Engenharia de Requisitos



- “A engenharia de requisitos estabelece o **processo de definição de requisitos** como um processo no qual o que deve ser feito deve ser **elicitado, modelado e analisado**. Este processo deve lidar com diferentes **pontos de vista**, e usar uma combinação de **métodos, ferramentas e pessoal**. O produto desse processo é um modelo, do qual um documento chamado requisitos é produzido.”

(Júlio Leite)

- Enfoque sistemático para identificação e manutenção dos requisitos
- Requisitos devem ser gerenciados



Engenharia de Requisitos



TIPOS DE REQUISITOS

Requisitos



- Objetivos, funções, propriedades, restrições que o sistema deve possuir/obedecer para satisfazer contratos, padrões ou especificações de acordo com o(s) usuário(s)
- Requisitos expressam comportamentos e propriedades que o sistema deve apresentar
 - Devem ser **compreendidos** pela equipe de desenvolvimento e validados pela parte interessada e comunidade de usuários
 - **Deve ser possível verificá-los**

Tipos de Requisitos



- **Requisitos Funcionais**
 - Requisitos diretamente ligados a funcionalidade do software
- **Requisitos Não Funcionais**
 - Requisitos que expressam restrições que o software deve atender ou qualidades específicas que o software deve ter
- **Requisitos⁻¹ (Requisitos Inversos)**
 - Estabelecem condições que nunca podem ocorrer.

Exemplos



- O sistema deve prover um formulário de entrada para a inserção dos resultados dos testes clínicos de um paciente.
 - RF
- O sistema deve emitir um recibo para o cliente, com o tempo máximo de 8 segundos após a transação.
 - RF, RNF de performance
- Os pagamentos devem ser realizados através do PagSeguro
 - RNF
- O sistema não pode apagar informação de um cliente
 - RIN

Requisitos Funcionais: Exercício



- Dê alguns exemplos de RFs para
 1. Sistema de **agendamento de reuniões**
 2. Sistema de **reserva de sala para reuniões**

Requisitos Não-Funcionais

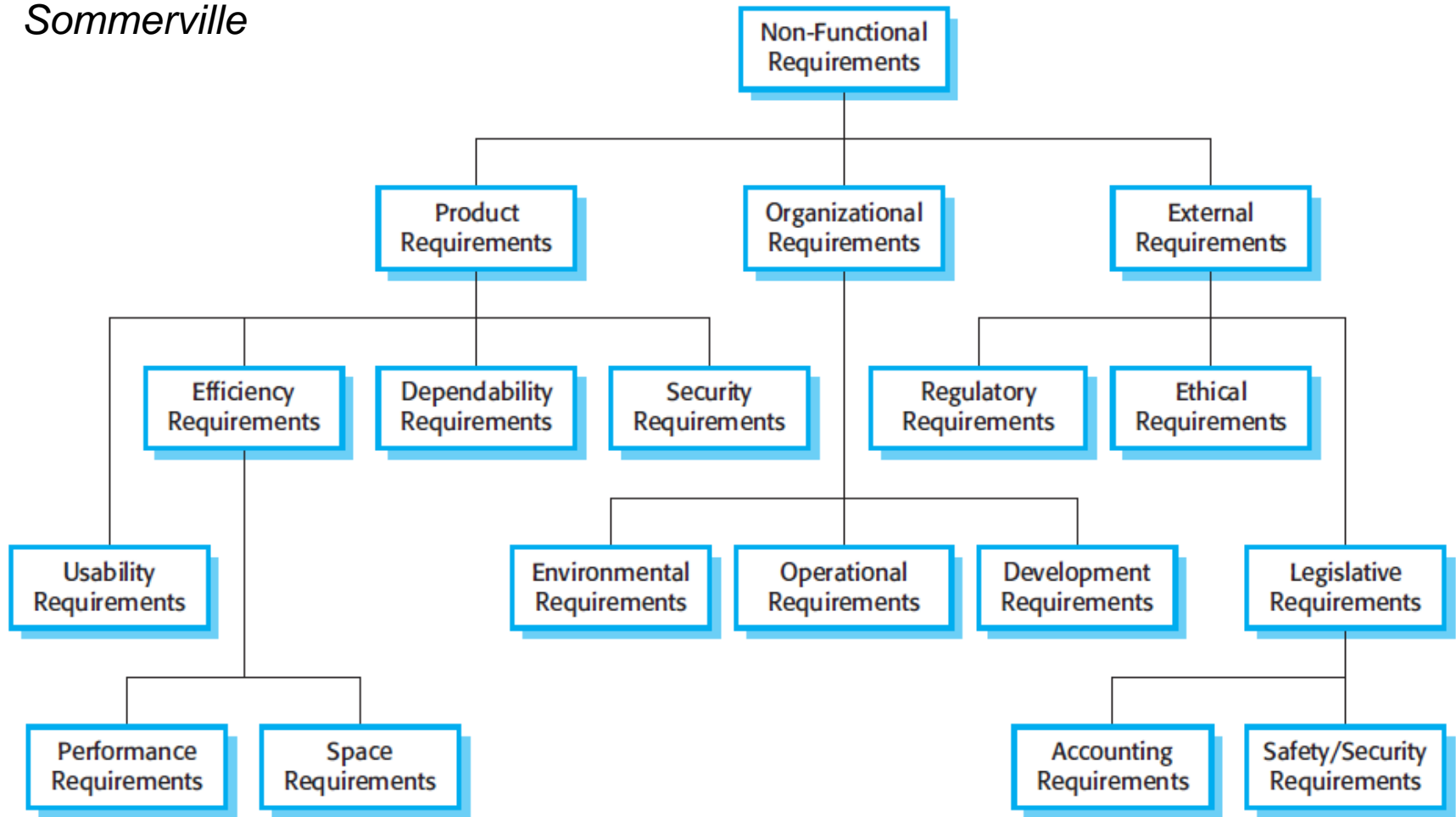


- Os requisitos não funcionais **são tão importantes** quanto os requisitos funcionais
- Estes requisitos são aqueles que geralmente restringem o comportamento do produto ou condicionam o projeto
- Como qualquer requisito, deve ser verificável
 - **Deve-se associar uma forma de medida/referência**

Requisitos Não-Funcionais (Tipos)



Sommerville



Requisitos Não-Funcionais



- Requisitos não funcionais devem ser elaborados até que se tornem **verificáveis**

Requisito Inverificável	Requisito Verificável
“O banco de dados ZZ deve ser flexível”	<ul style="list-style-type: none">• O banco de dados ZZ deve possuir oito campos por registro.
“MNOP deve ser seguro”	<ul style="list-style-type: none">• MNOP deve parar sua operação se uma pessoa se aproximar a mais de 2 metros de uma de suas partes móveis• MNOP deve desligar os aquecedores se a temperatura da água exceder 37°C• MNOP deve estar dentro dos padrões estabelecidos pela norma N567 seção 3.6 para temperaturas de superfícies externas.
“O sistema CE deve processar depósitos rapidamente”	<ul style="list-style-type: none">• O sistema CE deve escanear os dados do usuário e conta de cada folheto de depósito em 2 segundos ou menos”



Exemplos de Formas de Mensuração de RNF

Propriedade	Métrica
Velocidade	Transações processadas por segundo Tempo de resposta ao usuário/evento Tempo de <i>refresh</i> da tela
Tamanho	Kbytes Número de chips de RAM
Facilidade de uso	Tempo de treinamento Número de telas de ajuda
Confiabilidade	Tempo médio para falhar Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade
Robustez	Tempo de reinício após falha Porcentagem de eventos que causam falhas Probabilidade de que os dados sejam corrompidos por falhas
Portabilidade	Porcentagem de declarações dependentes de sistemas-alvo Número de sistemas-alvo

(Sommerville, 2003)



Engenharia de Requisitos



PROCESSO

Requisitos: Ciclo de Vida



- Requisitos **NÃO** são uma preocupação somente no início do projeto
- Requisitos têm papel central em todo o ciclo
 - Definição de critérios de aceitação e validação pelos *stakeholders*
 - Definição de O QUÊ o sistema deve fazer
 - especificação para a equipe
 - Verificação do sistema sendo desenvolvido
 - Alocação de tarefas para a equipe
 - Estimativa de custo/esforço/cronograma
 - Acompanhamento e controle do andamento do projeto
 - Informações para gerenciamento de mudanças
 - Rastreabilidade, análise de impacto
- Importantíssimo
 - PA Requirements Management (nível 2) e Requirements Development (nível 3) do CMMI

Atividades Genéricas



- ***Determinação***

- identificação das fontes de informação
- eliciação, refinamento e integração de informações

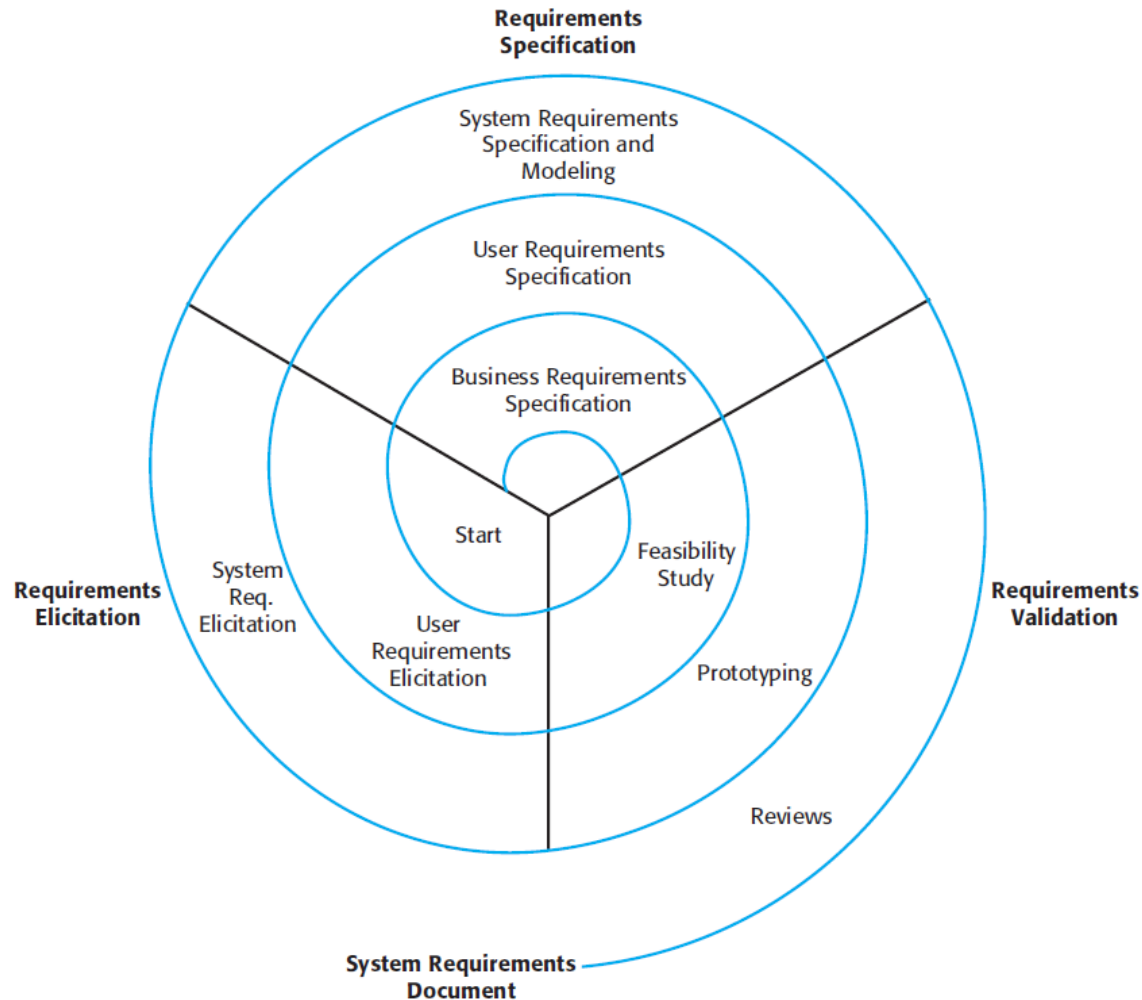
- ***Expressão***

- representação das informações obtidas
- representação das várias versões dos requisitos
- modelagem

- ***Avaliação***

- avaliação da informação recolhida e representada quanto à correção, completude, coerência para o(s) usuário(s)

Visão Espiral da Engenharia de Requisitos





Engenharia de Requisitos



PROCESSO: DETERMINAÇÃO

Engenharia de Requisitos



Determinação de
Requisitos

Problemas

Soluções

Lacuna Semântica

Mundo Real

Mundo Computacional

Determinação de Requisitos



- Coleta, Captura, Elicitação
- “Elicitação e captura (de requisitos) deveriam ser ilícitos” (Mike Cohn)
 - Requisitos estão prontos, basta fazermos as perguntas certas
- Metáfora do Arrastão (*trawling*)
 - Malhas de diferentes tamanhos
 - Tempo de vida
 - Múltiplas passadas
 - Pescadores experientes



Determinação



- Identificar as fontes de informação
- Busca de fatos

Identificação de Fontes de Informação



- Atores do Universo de Informações (UdI)
 - Clientes
 - Usuários
 - Desenvolvedores
- Documentos
- Livros
- Sistemas de Software

Heurísticas



- Quem é o cliente?
- Quem é o dono do sistema?
- Existe alguma solução (pacote) disponível?
- Existem sistemas similares?
- Quais são os livros relacionados à aplicação em discussão?
- Existe a possibilidade de reutilizar os artefatos (software)?

Técnicas para identificar requisitos



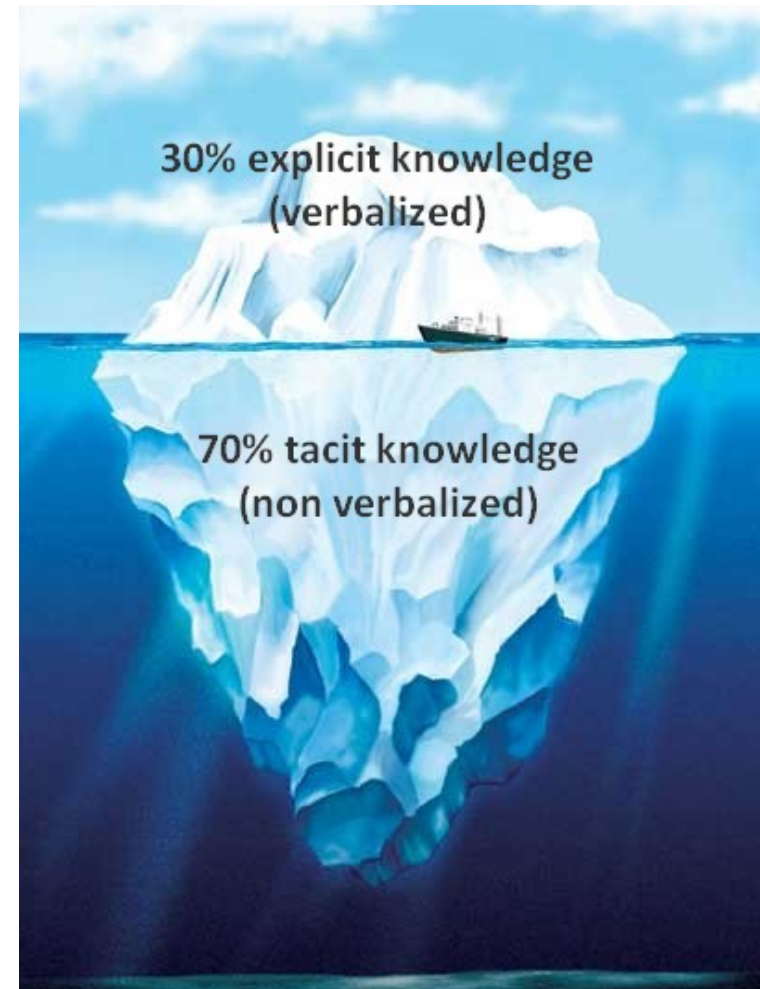
- Técnicas Baseadas em **Comunicação**
 - Comunicação Direta: Entrevistas, Reuniões, Workshops
 - Comunicação Indireta: Questionários, Textos de requisitos
- Técnicas Baseadas em **Estudo**
 - Estudo de documentos e formulários
 - Estudo de Sistemas Existentes (Engenharia reversa)
 - Estudo Bibliográfico
- Técnicas Baseadas em **Observação**
 - Observação Direta (pessoal ou vídeo)
 - Observação Verbalizada
 - Observação seguida de diálogo

Vilão: Conhecimento Tácito



- Trivial
- Internalizado
- Nunca é lembrado!

- Problema de comunicação
 - Não de requisitos!



Determinação



- Usuários misturam a solução com os requisitos
 - Separar NECESSIDADE da solução proposta ou atual
- Perguntar por quê?
 - Qual a razão disto?
 - Pode me explicar o porquê?
 - Qual o pensamento por trás disto?

Determinação



"The coffeepot must be made of stainless steel."

Why this particular material?

"Because a glass pot might break."

Why should it break?

"Because the pot is often left on the heat even when empty."

How does that happen? Can you explain why it matters?

"Because the pot is shared by many people who just pass through, and the heater does not switch off when the pot is empty."

Why doesn't it switch off?

"Because ...a requirement was missed."

Requirements

The coffeepot heater should switch off when the pot is empty.

I want hot fresh coffee whenever I walk into the kitchen.

I want to be able to make coffee without risking injury or damage.

Exercício



- “A cafeteira tem uma luz vermelha que pisca quando está ligada, quando a água chega na temperatura certa ela fica ligada (sem piscar)”
 - Quais seriam os “reais” requisitos?

Entrevistas



- Popular
- Escolha bem quem entrevistar
- Perguntas abertas e livres de contexto
- Tomar cuidado com a diferença entre desejo e necessidade
- (+) Contato direto com atores
- (+) Possibilidade de validação imediata
- (-) Conhecimento tácito
- (-) Diferenças culturais

Reuniões



- Extensão da entrevista
 - Brainstorm
 - Workshop de Requisitos
- (+) Dispor de múltiplas opiniões
- (+) Criação coletiva
- (+) Prototipação, storyboard → stakeholders
- (-) Dispersão
- (-) Custo

Observação



- (+) Baixo custo
- (+) Pouca complexidade da tarefa
- (+) Palavras tomam outro sentido
- (-) Dependência do ator (observador)
- (-) Superficialidade decorrente da pouca exposição ao universo de informações

Leitura de Documentos



- (+) Facilidade de acesso às fontes de informação
- (+) Volume de informação
- (-) Dispersão das informações
- (-) Volume de trabalho requerido para identificação dos fatos
- (-) Representação estática
- (-) Dependente de expressão e de interpretação



Engenharia de Requisitos

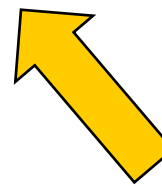


PROCESSO: EXPRESSÃO

Engenharia de Requisitos



Expressão



Problemas

Soluções

Lacuna Semântica

Mundo Real

Mundo Computacional

Documento de Requisitos



- *Software Requirements Specification* (SRS)
 - Afirmação oficial do que os desenvolvedores devem fazer
- Compromisso entre
 - Comunicação dos requisitos aos clientes
 - Detalhamento preciso para desenvolvedores e testadores
- Atributos de uma boa especificação
 - Clareza
 - Não ambígua
 - Completa
 - Simples
 - Bem escrita

Documento de Requisitos



- Sommerville (baseado no padrão IEEE)
 - Prefácio
 - Introdução
 - Glossário
 - Definição dos Requisitos do Usuário
 - Arquitetura do Sistema
 - Especificação dos Requisitos do Sistema
 - Modelos do Sistema
 - Evolução do Sistema
 - Apêndices
 - Índice

Documento de Requisitos



- Pressman (baseado em Karl Wiegers)
 - Introdução
 - Descrição Geral
 - Características do Sistema
 - Requisitos de Interfaces Externas
 - Outros Requisitos Não Funcionais
 - Outros Requisitos
 - Apêndice A: Glossário
 - Apêndice B: Modelos de Análise
 - Apêndice C: Lista de Problemas

Formas de Expressão



- **Sentenças em Linhagem Natural**
 - Sentenças numeradas
 - Cada sentença expressa um requisito
- **Linguagem Natural Estruturada**
 - Baseado em um *template* padrão
- **Linhagens de Descrição de Projeto**
 - Uso de uma linguagem como linhagem de programação
 - Características mais abstratas
- **Notações Gráficas**
 - E.g. Diagramas de Casos de Uso e Sequencia
- **Especificações Matemáticas**
 - Máquina de estados finitos

Expressão: Sentenças de Requisitos



O sistema deve + [verbo + objeto | frase verbal] +
[complemento de agente | nulo] + [condições | nulo]

- Três classes de sentenças
 - {Entrada, Saída, Mudança de Estado}
- Verbo/Frase verbal
 - Frase que expressa a funcionalidade do requisito
- Complemento de agente é a identificação de um agente relacionado com o requisito
 - Agente pode ser uma pessoa, uma instituição, um grupo ou um dispositivo físico externo ao software
- As sentenças podem ser de três tipos
 - Funcionais, não-funcionais e inversas

Sentenças de Requisitos



- O sistema deve emitir um recibo para o cliente.
- O sistema deve emitir o recibo para o cliente em no máximo 2 segundos.
- O sistema deve transformar uma fita disponível em fita emprestada, quando a fita for alugada pelo cliente.
- O sistema deve cadastrar bibliotecários.
- O sistema deve verificar a identidade do bibliotecário.
- O sistema não pode deixar que um livro fique na reserva mais de um mês.
- O sistema deve tornar um livro em livro emprestado, quando um usuário pegar este livro emprestado.

Expressão: Histórias

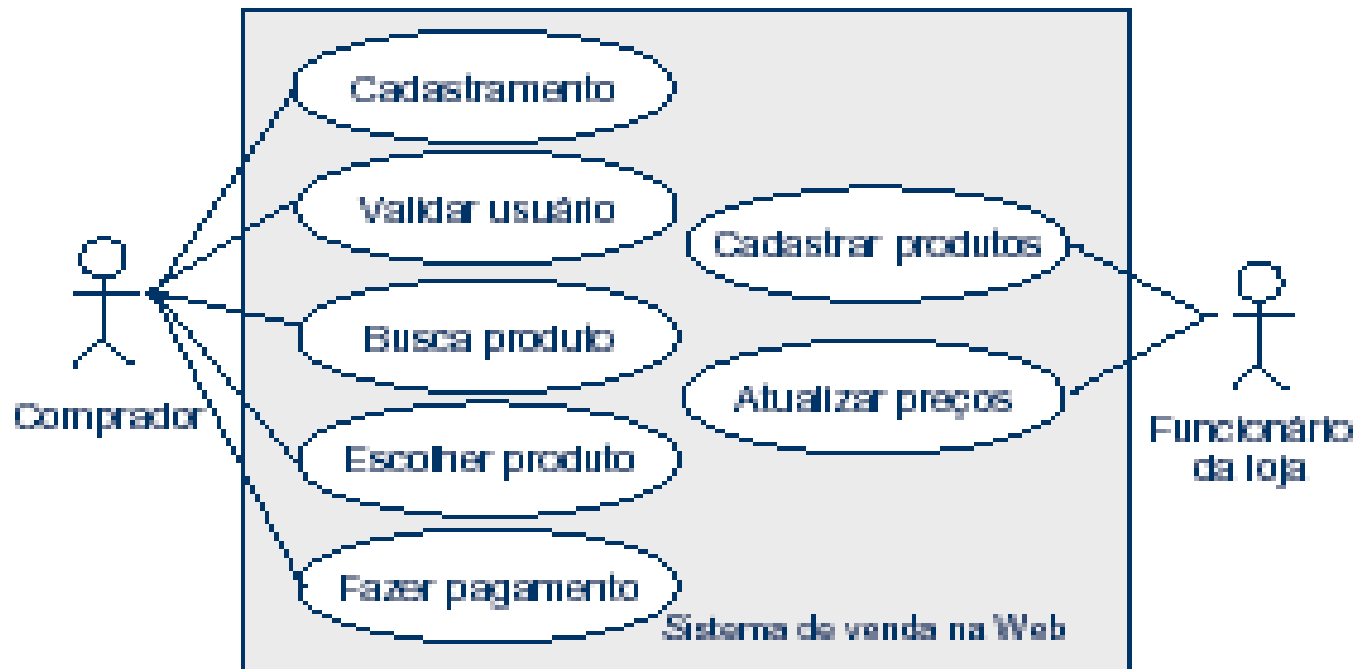


- Histórias
 - “Como cliente, gostaria de poder selecionar produtos por categorias para poder adquiri-los”
 - “Como aluno, gostaria de poder emitir meu histórico escolar para usar como comprovante oficial de rendimento escolar”
- Sem formato definido
- Adotados em métodos ágeis, como recurso de acomodação à mudança
 - “Uma expressão negociável de intenção” (Leffingwell)
- XP Essencial = CCC
 - Exige a **c**onversa
 - Exige a **c**onfirmação
 - **C**asos de testes de aceitação tornam-se a expressão do requisito
 - Documentados de forma mais ou menos formal

Expressão: Casos de Uso



- Representam a funcionalidade do sistema do ponto de vista de seus usuários



Casos de Uso



- Diagrama de casos de uso são semelhantes à histórias no seu poder semântico
 - “Como aluno, desejo emitir meu histórico escolar”
 - Precisa conversar, e documentar, possivelmente através de casos de teste
 - Caso de Uso Emitir Histórico Escolar
 - Diagrama tem de vir acompanhado de descrição textual descrevendo interação entre ator/sistema



powered by astah*

Descrição de Casos de Uso



- Descrição de casos de uso são informais
- Podem ser usados em vários níveis de abstração
 - Grau de detalhe reflete estágio do projeto de desenvolvimento
 - Conteúdo pode ser adequado às necessidades da aplicação
 - Formato é definido pela organização
 - Diferentes estilos de descrição

Descrição de Casos de Uso: Tipos



- **Preliminar (ou alto-nível)**
 - Conceitual, **abstrato** e pouco detalhado
 - Independente de implementação
 - Usados na delimitação de escopo em fases iniciais
 - ex: concepção
- **Essencial (ou expandido)**
 - Conceitual (**independente de implementação**)
 - Detalhado em termos de funcionalidade
 - Descreve cenários de interação entre ator e caso de uso
- **Real (ou concreto)**
 - Concreto (**dependente de tecnologia**)
 - Detalhado em termos de funcionalidade, comportamento (operação) e referências explícitas a elementos de interface

Exemplo: Caso de Uso Preliminar

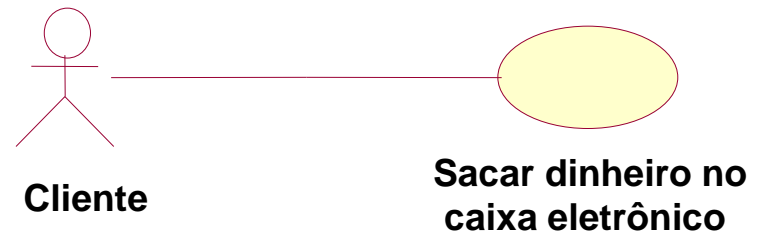


Identificação: UC1

Caso de uso: Sacar dinheiro no caixa eletrônico

Ator: Cliente

Descrição: Permite ao cliente sacar quantia especificada, uma vez validados cartão e senha. A transação deve ser registrada, o saldo atualizado, e dinheiro liberado.



Exemplo: Caso de Uso Essencial



Identificação: UC1

Caso de uso: Sacar dinheiro no caixa eletrônico

Ator: Cliente

Descrição: Permite ao cliente sacar quantia especificada, uma vez validados cartão e senha. A transação deve ser registrada, o saldo atualizado, e dinheiro liberado.

Pré-Condições: o Cliente possui cartão do banco e senha cadastrada.

Pós-Condições: lançada a transação na conta do Cliente, atualizado o saldo da conta corrente e liberado o dinheiro.

Sequência Típica de Eventos (Fluxo Básico):

Ação do ator


1. Este caso de uso começa quando o Cliente insere o cartão do banco no caixa eletrônico
2. O Cliente informa a sua senha
4. O Cliente informa o valor do saque

Resposta do sistema

3. O sistema valida a conta corrente e senha do Cliente, autorizando a operação
5. O sistema autoriza o saque e lança o débito na conta corrente do Cliente
6. O sistema libera o dinheiro

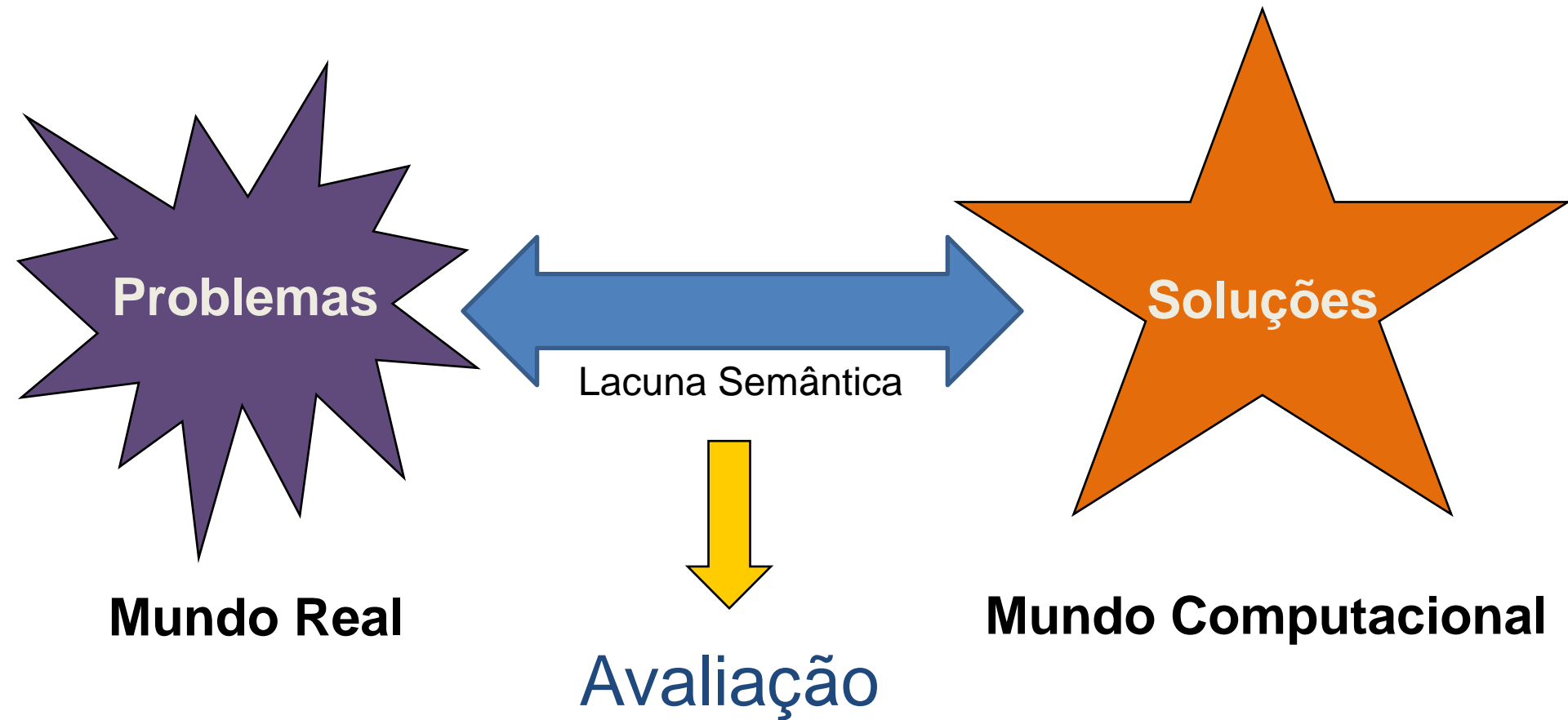


Engenharia de Requisitos



PROCESSO: AVALIAÇÃO

Engenharia de Requisitos



Verificação X Validação



Verificação

estamos construindo o
produto de maneira certa?
(em relação a outros
produtos)



entre modelos

Validação

estamos construindo o
produto certo?
(em relação a intenção dos
fregueses)



entre o Udl e um modelo

Avaliação



- Diferentes tipos de checagem devem ser realizados
 - Validade
 - Consistência
 - Completude
 - Realismo
 - Verificabilidade

Avaliação



- Definição de critérios de aceitação
 - Casos de teste de aceitação
 - Evolução de casos de teste para guiar as demais atividades
- Discussão de Cenários
- Protótipos
- Inspeção
- Revisão



Engenharia de Requisitos



GERENCIAMENTO DE REQUISITOS

Gerenciamento de Requisitos



- Requisitos sempre mudam
 - Ambientes de negócio e técnico do sistema sempre mudam após a instalação
 - Clientes e usuários não são as mesmas pessoas
 - Conflito nos requisitos dos usuários finais
 - Grandes sistemas tem uma grande comunidade

Gerenciamento de Requisitos

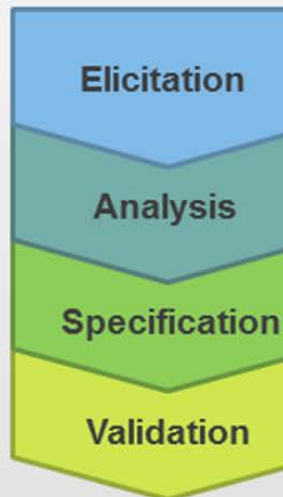


- Planejamento
 - **Identificação dos requisitos**
 - **Processo de gerenciamento de mudanças**
 - Avaliação do impacto e custo de mudanças
 - Análise do problema e especificação da mudança
 - Análise e custo da mudança
 - Implementação da mudanças
 - **Políticas de rastreamento**
 - Localização dos requisitos no projeto/implementação
 - **Suporte de Ferramentas**
 - Armazenamento dos requisitos
 - Gerenciamento de mudanças
 - Gerenciamento do rastreamento



Requirements Engineering

REQUIREMENTS DEVELOPMENT



Stakeholder consultation, review and context normalization



Understanding of desired system



Structured documentation of desired system



Identification of omitted, redundant & inconsistent req'mts

REQUIREMENTS MANAGEMENT



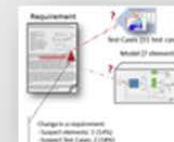
Traceability

Tracking where requirements are met



Change Mgmt

Req'mt maintenance & propagation



Fulfillment

Results & confirmation of successful fulfillment

Referências



- **Leitura Obrigatória**
 - **Sommerville, Ian. Engenharia de software. 9ª edição. Pearson Education, 2011.**
 - **Capítulo 6: discute o conceito de requisitos, os tipos de requisitos, e o conteúdo de um documento de especificação de requisitos**
 - **Capítulo 7: discute o processo de engenharia de requisitos**
- **Leitura complementar**
 - **Pressman, Roger. Engenharia de Software. McGrawHill, 2011.**
 - **Capítulo 5: as atividades da engenharia de requisito**
 - **Capitulo 6: o processo de coleta e descrição de requisitos**
 - **Alexander, I. & Stevens, R. Writting better requirements. Chapter 3: Gathering requirements from stakeholders.**
- **Inclui material público ou cedido com permissão de autoria dos professores Júlio Leite e Karin Breitmann**

Perguntas?



- Este material tem contribuições de
 - Ingrid Nunes
 - Karin Becker
 - Lucinéia Thom
 - Marcelo Pimenta

Prosoft

