



## Unidade 4 – Princípios de Engenharia de Requisitos Parte 2

### CERTIFICAÇÃO CPRE-FL



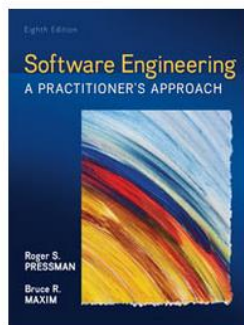
Prof. Aparecido V. de Freitas  
Doutor em Engenharia  
da Computação pela EPUVSP

[aparecidovfreitas@gmail.com](mailto:aparecidovfreitas@gmail.com)

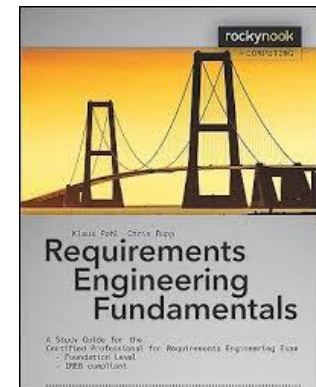
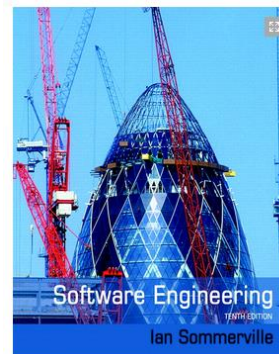


# Bibliografia

- **Fundamentos de Engenharia de Requisitos – POHL K., RUPP C. – IREB 2012**
- **Software Engineering – A Practitioner's Approach – Roger S. Pressman – Eight Edition – 2014**
- **Software Engineering – Ian Sommerville – 10<sup>th</sup> edition - 2015**
- Engenharia de Software – Uma abordagem profissional – Roger Pressman - McGraw Hill, Sétima Edição - 2011
- Engenharia de Software – Ian Sommerville – Nona Edição – Addison Wesley, 2007
- Requirements Engineering Fundamentals – IREB Compliant – Klaus Pohl e Chris Rupp – **CPRE-FL**, 1996.



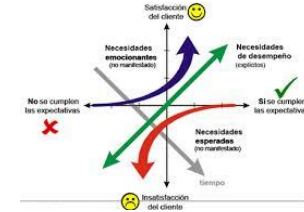
Software Engineering: A Practitioner's Approach, 8/e





# Categorização de Requisitos

## Modelo de Kano

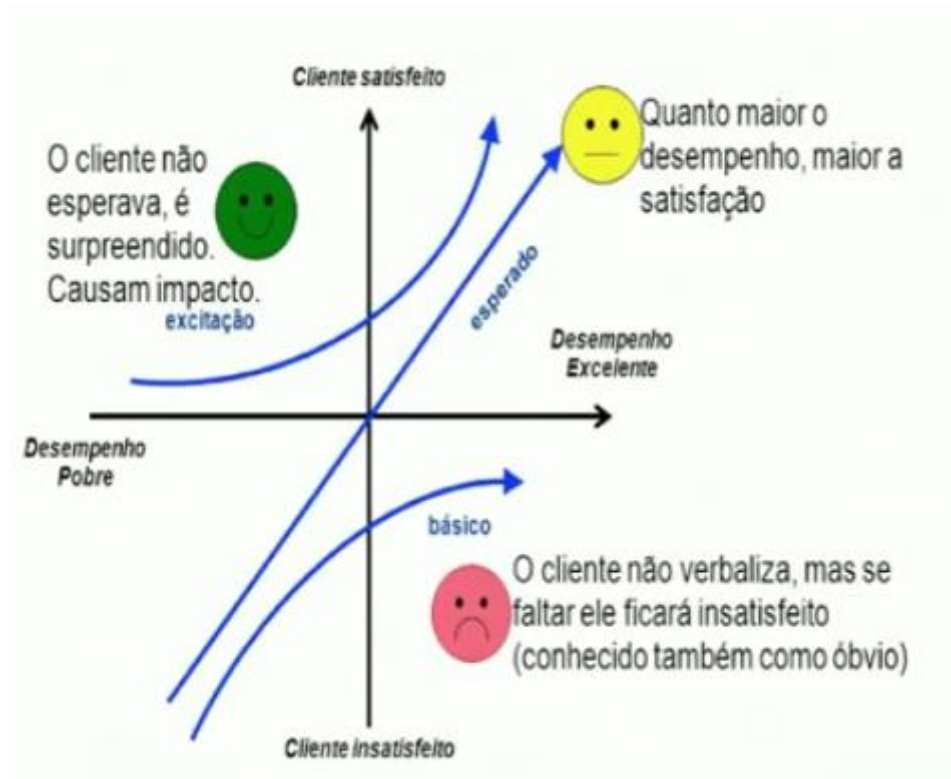


Os fatores que determinam a satisfação do cliente são classificados nas seguintes categorias:

- ✓ **Fatores básicos de satisfação:** Propriedades evidentes e pressupostas; (auto evidentes e tidas como certas) (conhecimento subconsciente)
- ✓ **Fatores esperados de satisfação:** Propriedades explicitamente exigidas do sistema; (conhecimento consciente)
- ✓ **Fatores inesperados de satisfação:** Propriedades do sistema que o stakeholder não conhece ou espera, e que ele descobre apenas ao utilizar o sistema - encantamentos. (conhecimento inconsciente)



# Modelo de Kano





# Fatores Básicos de Satisfação

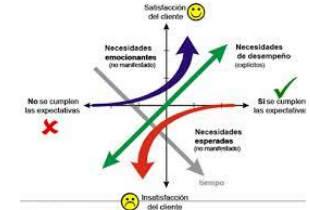


- Devem ser atendidos pelo sistema de qualquer maneira;
- Do contrário, os stakeholders ficarão decepcionados (**dissatisfiers**);
- Mesmo se atendidos, não geram uma atitude positiva por parte dos stakeholders;
- São predominantemente influenciados pelos sistemas existentes;
- Assim, técnicas de observação e técnicas centradas em análise de documentos são adequadas para se elicitar esses fatores.





## Fatores Esperados de Satisfação

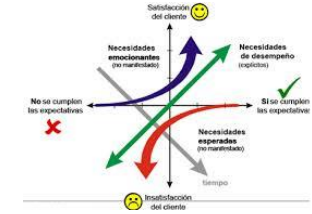


- Correspondem aos requisitos **conscientes** (explícitos);
- Propriedades conscientemente conhecidas e explicitamente exigidas pelos stakeholders;
- Quando atendidas, os stakeholders ficam contentes e satisfeitos (daí o termo **satisfiers**), o que é desejável;
- Se algumas dessas propriedades estiverem faltando, provavelmente o stakeholder **não** aceitará o sistema;
- São comumente elicitados por meio de questionários e entrevistas.





# Fatores **Inesperados** de Satisfação



- Propriedades do sistema cujo valor somente é reconhecido quando o stakeholder pode testar o sistema na prática, ou quando o ER as propõe (**delighters**);
- Correspondem aos **fatores inconscientes** de conhecimento (ocultos);
- Geram encantamentos dos stakeholders;
- Técnicas de criatividade são as mais indicadas para elicitar esses tipos de fatores.





## Técnicas de Elicitação

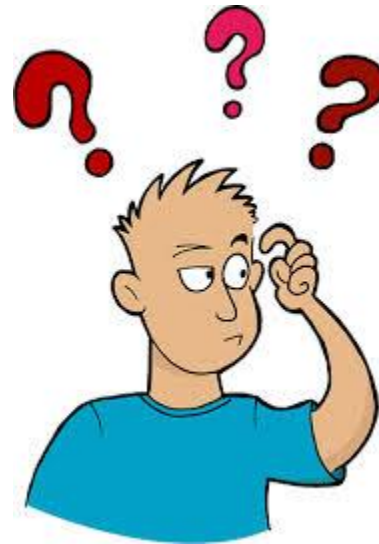
- Têm a finalidade de identificar os requisitos conscientes, inconscientes e subconscientes dos stakeholders.







Quais os fatores que influenciam na  
escolha das Técnicas de Elicitação ?





## Fatores para a escolha da Técnica de Elicitação

- Distinção entre requisitos conscientes, inconscientes e subconscientes;
- Restrições em termos de Tempo e Orçamento, bem como disponibilidade dos stakeholders;
- Experiência do ER;
- Oportunidades e Riscos do Projeto.





## Técnicas de Pesquisa

- Partem do pressuposto de que o respondente é capaz de expressar explicitamente seu conhecimento;
- Respondente está comprometido a investir tempo e esforço para a elicitacão;
- São geralmente dirigidas pelo ER;
- ER que faz as perguntas;
- Stakeholder pode se esquecer ou ignorar preocupações;
- Baseiam-se em entrevistas ou questionários.





Quais as vantagens e desvantagens da Técnica de Pesquisa baseada em Entrevistas ?





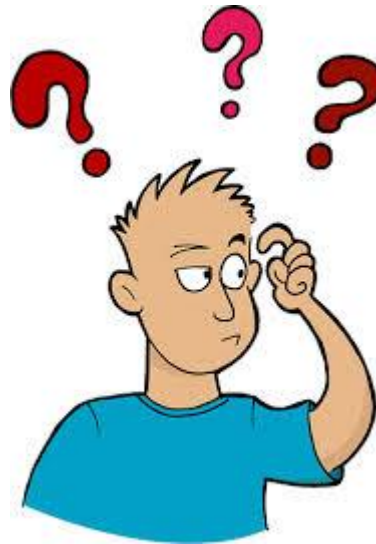
## Entrevistas

- ER faz as perguntas para um ou mais Stakeholders;
- ER documenta as respostas;
- Perguntas bem formuladas podem revelar requisitos subconscientes;
- Principal desvantagem: tempo consumido para a elicitação.





Quais as vantagens e desvantagens da Técnica de Pesquisa baseada em Questionários ?





## Questionários

- ER pode fazer uso de questões abertas ou fechadas (ex. Múltipla escolha) ;
- Para grande número de Stakeholders, questionário online é uma opção viável;
- Podem elicitar grande número de informações em curto espaço de tempo;
- Baixo custo;
- **Desvantagem**: coleta de requisitos que o ER já conhece ou conjectura;
- Não fornecem retroalimentação imediata entre stakeholder e ER.





## Técnicas de Criatividade

- Tem a finalidade de desenvolver requisitos inovadores;
- Podem esboçar uma visão inicial do sistema;
- Podem elicitar fatores inesperados de satisfação (delighters);
- Em geral, não são adequadas para estabelecer requisitos precisos sobre comportamento do sistema;
- Dentre as técnicas, cita-se: brainstorming e mudança de perspectiva.







# Brainstorming

- Ideias são coletadas durante certo período de tempo, geralmente com grupos de 5 a 10 pessoas;
- Ideias são documentadas por um moderador, sem serem discutidas, julgadas ou comentadas inicialmente;
- Participantes usam ideias de outros participantes para desenvolver ou modificar ideias existentes;
- Em outro tempo, as ideias coletadas são submetidas à uma análise rigorosa;
- Vantagem: grande número de ideias são coletadas em curto espaço de tempo;
- Desvantagem: Menos eficaz quando a dinâmica do grupo é confusa ou quando há membros muito dominantes no grupo.





## Brainwriting

### 6-3-5 Brainwriting

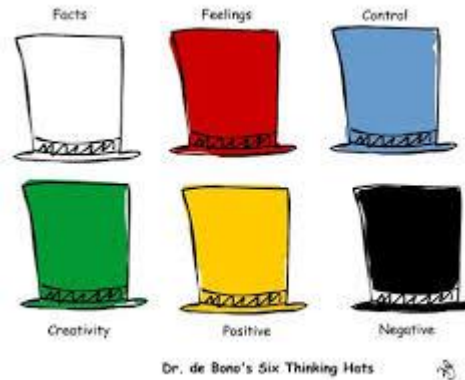
- Corresponde ao método 6-3-5;
- Seis participantes, cada um escreve três ideias numa folha de papel;
- Essa folha é repassada aos outros membros, que comentam e desenvolvem as ideias recebidas, processo que se repete 5 vezes.





## Mudança de Perspectiva

- Técnica conhecida por “Seis chapéus do pensamento” [DeBono, 2006]



- Cada chapéu representa uma perspectiva adotada, uma após a outra, por cada participante;
- As soluções daí resultantes abordam o problema a partir de perspectivas diferentes;
- Técnica benéfica quando stakeholders estão obstinadamente presos à suas opiniões;
- Difícil de ser aplicada se os requisitos exigem um nível de detalhamento muito grande, pois isso tornaria a técnica muito cansativa.



## Técnicas baseadas em Documentos

- Reutilizam soluções e experiências feitas com sistemas existentes;
- Ao substituir-se um sistema legado, essa técnica assegura que a funcionalidade completa do sistema possa ser identificada;
- Devem ser combinadas com outras técnicas de elicitação para que novos requisitos possam ser identificados;
- Fazem parte dessa técnica:
  - Arqueologia do Sistema: Extração de informações a partir da documentação ou código do sistema legado;
  - Leitura baseada em perspectiva: Foca-se em partes específicas da documentação do sistema, a partir de uma perspectiva específica, por exemplo: do implementador ou do testador.
  - Reutilização: Consulta-se documentação de requisitos que foram compilados anteriormente e reutiliza-os.



## Técnicas de Observação

- ER observa os stakeholders enquanto trabalham;
- ER documenta todos os passos e elicita os processos que o sistema deverá suportar, bem como potenciais erros, riscos e questões em aberto;
- Técnica apropriada para elicitar requisitos detalhados e **fatores básicos de satisfação** (**dissatisfiers**), pois o ER pode identificar fatores considerados óbvios, ou aqueles que os stakeholders apenas conhecem no subconsciente.
- Permite que o ER se familiarize com a linguagem do domínio, o que simplifica o processo de elicitação.





## Técnicas de Apoio

- Mapas Mentais
- Workshop
- CRC – Class Responsibility Collaboration
- Gravações de áudio e vídeo
- Modelagem de Casos de Uso
- Protótipos



## Documentação de Requisitos

- Na Engenharia de Requisitos, informações levantadas durante as diversas atividades devem, obrigatoriamente, ser documentadas.





## O que é Especificação de Requisitos ?







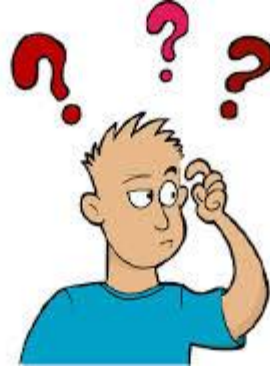
## Especificação de Requisitos

- É uma coleção de requisitos **representada de forma sistemática**, tipicamente para um sistema ou componente, atendendo a determinados critérios.





Quais as razões para se documentar requisitos ?





## Razões para se documentar Requisitos

- Requisitos formam a base para o desenvolvimento do sistema;
- Requisitos tem relevância legal ;
- Documentos de requisitos são complexos;
- Requisitos devem ser acessíveis para todas as partes envolvidas.





## Tipos de Documentação

- Requisitos para um sistema podem ser documentados a partir de três perspectivas diferentes. (Estrutural (dados), Funcional, Comportamental);
- Essas perspectivas são complementares (uma não substitui a outra...);
- Na prática, tanto linguagem natural (texto) quanto a modelagem conceitual são utilizadas para esse fim, ou muitas vezes, emprega-se uma combinação apropriada entre as duas.





## Documentação de Requisitos com Linguagem Natural

- É a forma de documentação de requisitos mais aplicada na prática;
- Vantagem: Stakeholders não precisam aprender nova notação;
- Entretanto, pode resultar em requisitos ambíguos.





## Documentação de Requisitos com Modelos Conceituais

- Estão associadas à cada perspectiva;
- Pode-se retratar os requisitos de forma mais compacta;
- Apresentam menor grau de ambiguidade;
- No entanto, exige conhecimentos específicos de modelagem de software.





## Diagramas – Modelos Conceituais

- Diagrama de Caso de Uso.
- Diagrama de Classes.
- Diagrama de Atividades.
- Diagrama de Estados





## Estruturas Padronizadas de Documentos

- RUP – Rational Unified Process [Kruchten,2001];
- Norma IEEE standard 830-1998;
- Modelo-V [V-Modell 2004] – Ministério do Interior da Alemanha.







## Estruturas Padronizadas de Documentos

- Padrão ISO **29148:2011** sugere que o documento de requisitos tenha 5 partes:
  - ✓ Informação introdutória e descrição geral do software;
  - ✓ Lista de documentos referenciados;
  - ✓ Lista de requisitos especificados;
  - ✓ Medidas planejadas para verificação;
  - ✓ Apêndices.





## Estrutura do Documento de Requisitos

As seguintes questões deveriam ser abordadas por qualquer estrutura de documento de Especificação de Requisitos:

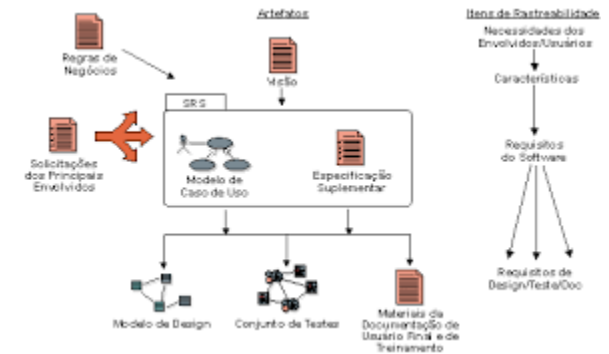


- ⊕ Introdução
- ⊕ Finalidade
- ⊕ Cobertura do Sistema
- ⊕ Stakeholder
- ⊕ Definições, acrônimo e abreviações
- ⊕ Referências
- ⊕ Visão Geral
- ⊕ Ambiente do Sistema
- ⊕ Descrição da Arquitetura
- ⊕ Funcionalidades do Sistema
- ⊕ Usuários e público-alvo
- ⊕ Restrições
- ⊕ Pressupostos
- ⊕ Requisitos
- ⊕ Apêndices
- ⊕ Índice



## Usos do Documento de Requisitos

- ⊕ Planejamento
- ⊕ Análise e Projeto
- ⊕ Implementação
- ⊕ Teste
- ⊕ Gerenciamento de Mudanças
- ⊕ Uso do sistema e manutenção
- ⊕ Gerenciamento do Contrato





## Critérios de Qualidade para o Documento de Requisitos

### IEEE 830–1998

- ✦ Não-ambiguidade e consistência;
- ✦ Estrutura clara;
- ✦ Modificabilidade e extensibilidade;
- ✦ Completude [IEEE-1998];
- ✦ Rastreabilidade [IEEE-1998].





# Critérios de Qualidade para o Documento de Requisitos

## IEEE 830-1998

Os critérios de qualidade definidos na norma IEEE 830-1998 podem ser aplicados tanto para requisitos individuais como para documentos completos de requisitos



- ⊕ Acordado
- ⊕ Priorizado
- ⊕ Não-ambíguo
- ⊕ Válido e atualizado
- ⊕ Correto
- ⊕ Consistente
- ⊕ Verificável
- ⊕ Realizável
- ⊕ Rastreável
- ⊕ Completo
- ⊕ Compreensível



## Regras fundamentais para a Legibilidade de Requisitos

- ⊕ Frases Curtas e parágrafos curtos;
- ⊕ Formular apenas um requisito por frase.





## Glossário

- ⊕ Uma causa frequente de **conflitos** na Engenharia de Requisitos ocorre quando pessoas envolvidas no processo de desenvolvimento de Software têm diferentes interpretações de termos.
- ⊕ Um **glossário** é uma coleção de definição de termos, apresentando os seguintes elementos:
  - ✓ Termos técnicos do contexto
  - ✓ Abreviações e acrônimos
  - ✓ Conceitos do dia-a-dia
  - ✓ Sinônimos
  - ✓ Homônimos





## Regras para utilizar o Glossário

- ⊕ Tem que ser gerenciado de forma centralizada;
- ⊕ Deve-se eleger quem fará a manutenção do glossário;
- ⊕ Deve ser atualizado de forma constante;
- ⊕ Deve ser acessível;
- ⊕ Seu uso deve ser obrigatório;
- ⊕ Deve conter a origem dos termos;
- ⊕ Precisa ser acordado entre as partes interessadas;
- ⊕ Deve ter uma estrutura consistente.





## Documentação de Requisitos com Linguagem Natural

- ⊕ Requisitos elicitados são frequentemente documentados usando linguagem natural;
- ⊕ É mais facilmente compreendida pelos stakeholders;
- ⊕ Não exige preparação dos stakeholders para compreender a especificação.





## Efeitos da Linguagem Natural

- ⊕ É inerentemente ambígua.
- ⊕ Requisitos podem se manifestar de forma diferente em cada pessoa (“Efeitos Transformacionais”)





## Efeitos Transformacionais

- ⊕ Nominalização;
- ⊕ Substantivos sem indicador de referência;
- ⊕ Quantificadores universais;
- ⊕ Condições especificadas de forma incompleta;
- ⊕ Verbos de processo especificados de forma incompleta.





# Nominalização

- ⊕ Um processo (às vezes de longa duração) é convertido em um evento (singular)
- ⊕ Dessa forma, todas as informações necessárias para descrever o processo acuradamente são perdidas.

## Exemplo: Nominalização

“Em caso de **crash** do sistema, será realizado um **restart** do sistema”



- ⊕ Os termos **crash** do sistema e **restart** descrevem, respectivamente, processos que deveriam ser analisados com maior precisão...
- ⊕ Termos nominalizados não devem deixar qualquer margem de interpretação dos processos e deve representar o processo de forma precisa, incluindo quaisquer exceções que possam ocorrer, bem como todos os parâmetros de entrada e saída.
- ⊕ Portanto, as nominalizações devem ser evitadas. Se presentes, deve-se examinar se foram suficientemente detalhadas em outra parte do Documento de Requisitos e, portanto, devem estar claras para todos os Stakeholders.



## Substantivos sem indicador de Referência

- ⊕ Substantivos especificados de forma **incompleta**;
- ⊕ Linguistas referem-se a isso como a falta de um indicador de referência.

Exemplo: Substantivos sem indicador de referência

“Os dados deverão ser exibidos para o usuário no terminal”



- ⊕ Quais dados ?
- ⊕ Qual usuário ?
- ⊕ Qual terminal, exatamente ?
- ⊕ Requisito deveria ser: “O sistema deve exibir os dados de faturamento para o usuário cadastrado no terminal em que ele estiver logado”



## Quantificadores Universais

- ⊕ Especificam quantidades e frequências. Agrupam um conjunto de objetos e fazem uma declaração sobre o comportamento desse grupo; Utilizam os seguintes termos: **Nunca, Sempre, Não, Nenhum, Cada, Todos, Alguns, Nada;**
- ⊕ Existe o risco do comportamento ou da propriedade especificada, **não** se aplicar à todos os objetos do conjunto especificado.

### Exemplo: Quantificadores universais

“ O sistema deverá mostrar **todos** os dados em **cada** submenu ”



- ⊕ Nesse casos, é preciso fazer as seguintes perguntas:
- ⊕ Realmente em cada submenu ?
- ⊕ Realmente todos os conjuntos de dados ?
- ⊕ Quantificadores universais podem facilmente ser identificados por meio de “palavras-gatilho” (trigger words) tais como: nunca, sempre, nenhum, cada, todos, alguns ou nada.



## Condições especificadas de forma incompleta

- ⊕ Requisitos contendo condições especificam o comportamento que deve ocorrer quando a condição for atendida. Empregam os termos Se.. Então, No caso de .... , Se ... , Em função de .... ;
- ⊕ Mas, também devem especificar o comportamento quando a condição não for atendida (essa parte costuma faltar nas especificações)

Exemplo: Condições especificadas de forma incompleta

“ O sistema do restaurante deverá oferecer todas as bebidas para um cliente registrado com mais de 20 anos”



- ⊕ Que bebidas o sistema vai oferecer para um cliente que tenha **20 anos ou menos** ?
- ⊕ Se essa questão for esclarecida, o requisito poderá ser formulado como segue:
- ⊕ “ O sistema do restaurante deverá oferecer:  
Todas as bebidas não alcoólicas para qualquer usuário registrado com menos de 21 anos. Todas as bebidas, inclusive as alcoólicas, para qualquer usuário registrado acima de 20 anos.”



## Verbos de processo especificados de forma incompleta

- ✦ Alguns verbos de processo requerem mais de um substantivo para serem especificados de forma completa. Por exemplo, o verbo “**transmitir**” exige pelo menos três suplementos para ser considerado completo: **O que é transmitido**, **de onde é transmitido** e **para onde é transmitido**;
- ✦ Isso pode ser evitado, na maioria das vezes, se os requisitos passarem a ser formulados na voz ativa e não na voz passiva.

### Exemplo: Verbos de processo especificados de forma incompleta

“ Para logar um usuário, os dados de login são inseridos”



- ✦ O requisito foi formulado na voz passiva. Não está claro quem vai inserir os dados de login.
- ✦ Também não está claro onde e como isso será feito.
- ✦ Se esse requisito fosse especificado na voz ativa, a formulação poderia ser:

“O sistema deve permitir ao usuário inserir seu username e senha usando o teclado do terminal”





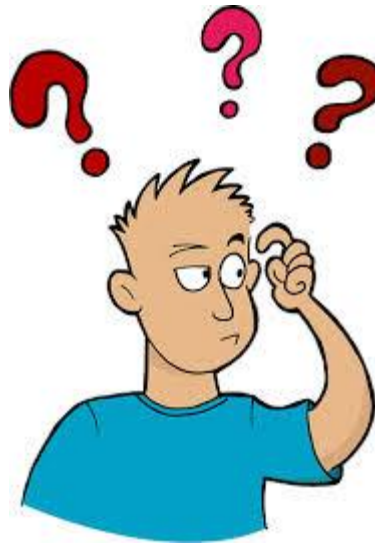
## Requisitos com o uso de Templates

- ✦ Fornecem uma abordagem simples para se reduzir os efeitos transformacionais de linguagem ao se documentar requisitos.
- ✦ Fornecem o apoio para que o ER possa obter alta qualidade e não-ambiguidade sintática, com baixo custo e tempo otimizado.





## O que é Template de Requisitos ?





## Template de Requisitos

⊕ É um padrão para a estrutura sintática de requisitos individuais.





# Modelagem, Validação e Gestão de Requisitos



Prof. Aparecido V. de Freitas  
Doutor em Engenharia  
da Computação pela EPUVSP



O que significa documentar Requisitos com o uso de Modelos ?





## Documentação de Requisitos com Modelos

- Na prática, os requisitos são frequentemente documentados por meio da linguagem natural;
- Entretanto, cada vez mais Requisitos tem sido documentados com o emprego de modelos;
- Modelos de requisitos são utilizados **adicionalmente** à documentação de requisitos em linguagem natural e substituem parcialmente requisitos que seriam documentados em linguagem natural.





O que é Modelo ?



## Modelo

- Um modelo é uma imagem que abstrai da realidade ou que funciona como uma representação abstrata da realidade a ser criada;
- A modelagem pode ser aplicada a objetos materiais ou imateriais de uma realidade existente ou de uma realidade a ser desenvolvida;
- Um modelo é uma representação abstrata de uma realidade existente ou de uma realidade a ser criada. [Stachowiak, 1973]

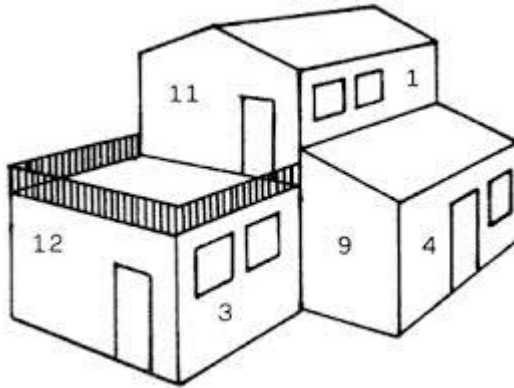






## Propriedades de Modelos

- Representação da realidade;
- Redução da realidade;
- Pragmatismo (construído para uma finalidade prática e específica)





## Linguagens de Modelagem

- Linguagens específicas para construção de modelos conceituais;
- São definidas por sua sintaxe e semântica;
- **UML** (Unified Modeling Language) é frequentemente utilizada para construir modelos de requisitos;
- **UML** tornou-se o padrão para a construção de sistemas de software baseada em modelos;





## Vantagens dos Modelos de Requisitos

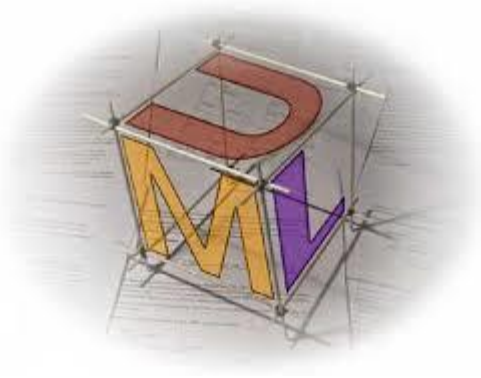
- Informações podem ser percebidas e memorizadas de forma mais rápida e melhor quando retratadas de forma gráfica ao invés de usar a linguagem natural.
- Linguagens de modelagem têm um enfoque estritamente definido. (Ex. Planta de Arquitetura, Planta Elétrica, Planta Hidráulica, Planta Estrutural...)





## Casos de Uso





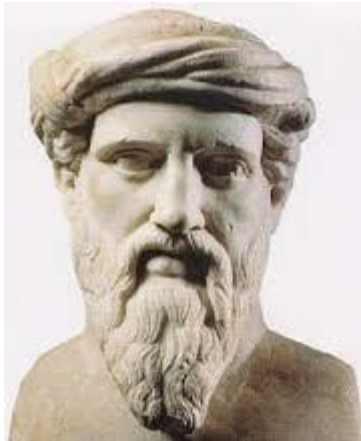
O que é Modelo de Caso de Uso?





## Modelo de Caso de Uso

- Modela os requisitos funcionais de um sistema;
- Na UML, para a modelagem de caso de uso empregam-se dois conceitos:
  - ✓ Diagramas de Casos de Uso;
  - ✓ Especificações de Casos de Uso



*Não diga pouco em muitas palavras,  
mas sim, muito em poucas.* **Pitágoras**



## Diagramas de Caso de Uso

- Descreve de forma **esquemática** um **cenário** que exhibe as funcionalidades do sistema sob ponto de vista do usuário.
- Apresentam o relacionamento das funções de um sistema.
- Apresentam também o relacionamento das funções de um sistema com seu ambiente.





## Qual a importância do Diagrama de Caso de Uso ?







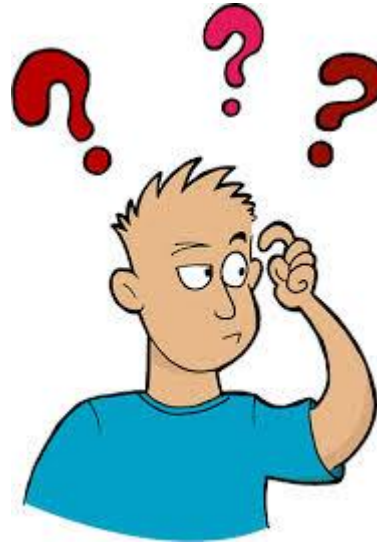
## Diagrama de Caso de Uso

- Direciona diversas tarefas posteriores do ciclo de vida do software;
- Força os desenvolvedores a moldar o sistema de acordo com o usuário.





Quais os elementos essenciais da Modelagem com Caso de Uso ?





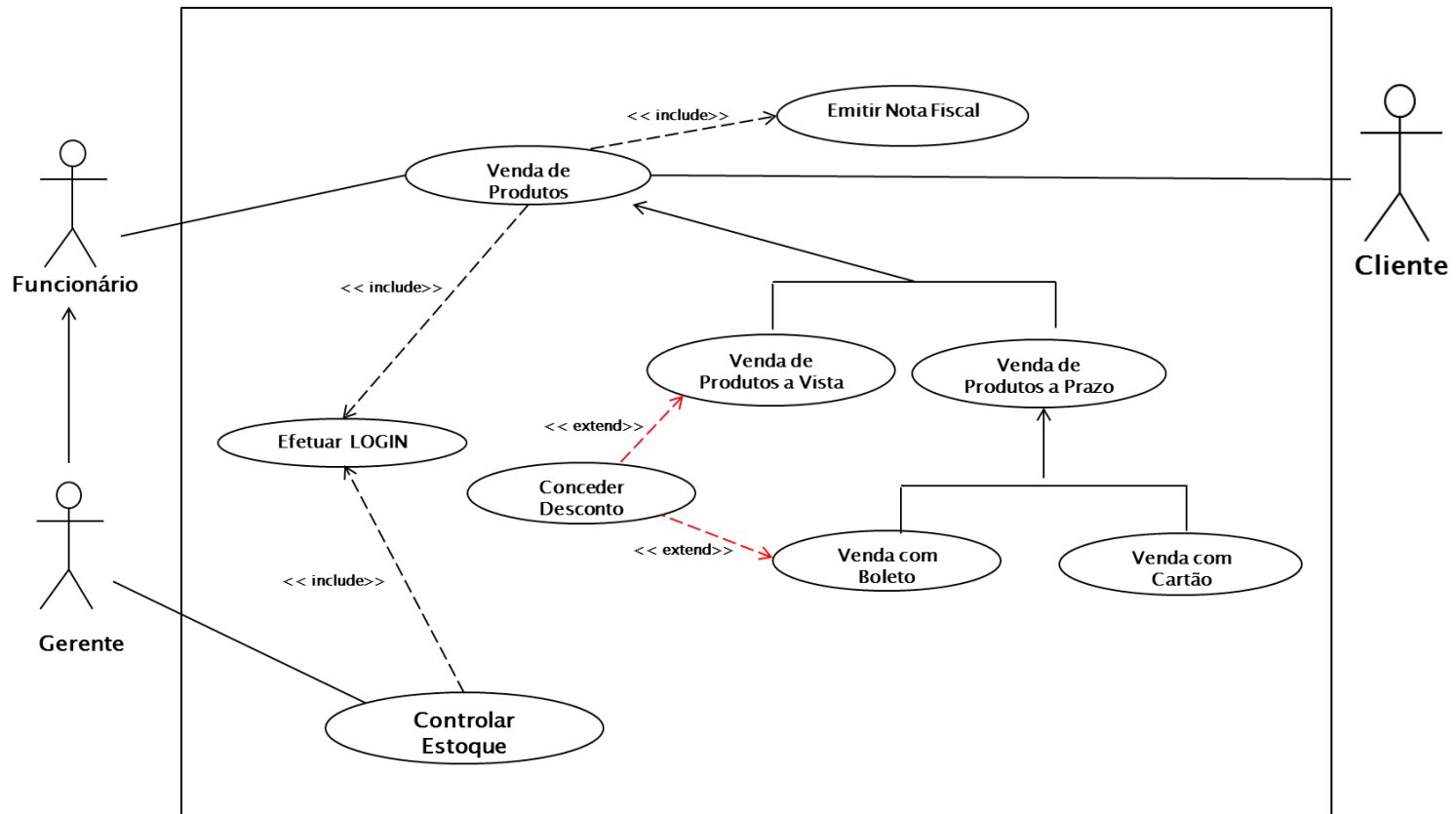
## Elementos essenciais da Modelagem com Caso de Uso

- Casos de Uso
- Atores
- Limites do Sistema
- Relação de Inclusão (Include)
- Relação de Extensão (Extend)
- Relacionamentos entre Atores e Casos de Uso



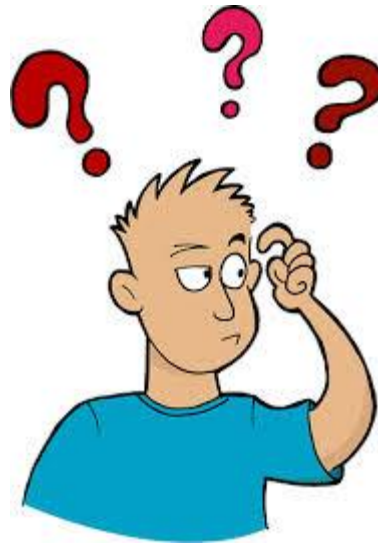


## Exemplo – Diagrama de Caso de Uso UML





Como documentar um caso de uso ?





## Documento de Caso de Uso



- ◆ Descreve, por meio de linguagem natural, a função em linhas gerais do caso de uso;
- ◆ Definem as etapas que devem ser executadas pelo ator e pelo sistema;
- ◆ Definem Restrições;
- ◆ Definem Validações.



## Documento de Caso de Uso



- ◆ UML não define um formato específico;
- ◆ Assim, o formato é bastante flexível;
- ◆ Pode-se usar pseudo-código, embora esse procedimento fuja bastante do objetivo do Diagrama de Casos de Uso, que é usar uma linguagem simples, de forma que até mesmo leigos possam entendê-la;
- ◆ Fornece a base para o plano de testes do sistema.



## Documento de Caso de Uso – Template



Nome do Caso de Uso	
Caso de Uso Geral	
Ator Principal	
Atores Secundários	
Descrição (Resumo)	
Restrições / Validações	
Pré-condições	
Pós-condições	
Fluxo Principal	
Fluxos Alternativos	
Fluxos de Exceção	
Restrições / Validações	





ENGENHARIA  
DE SOFTWARE



## Três Perspectivas sobre Requisitos

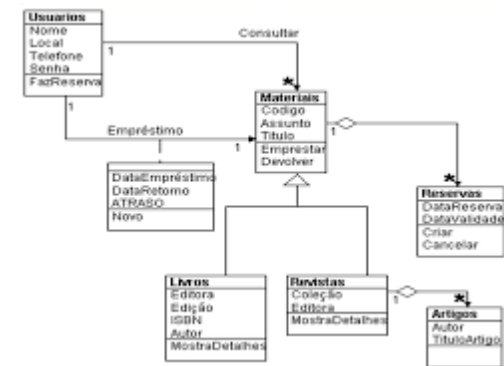
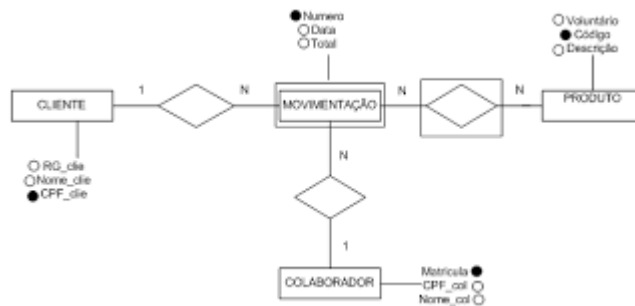
- ◆ **Perspectiva Estrutural:** Documentam-se as estruturas de dados de entrada e saída, bem como aspectos estático-estruturais das relações de uso e dependência no contexto do sistema;
- ◆ **Perspectiva Funcional:** Documentam-se as informações do contexto do sistema que estão sendo manipuladas pelo sistema e quais dados estão transmitidos para o contexto do sistema;
- ◆ **Perspectiva Comportamental:** Documentam-se a interação do sistema no contexto com base em estados.



## Modelagem de Requisitos na Perspectiva Estrutural

### ◆ Diagramas Entidade-Relacionamento

### ◆ Diagramas de Classe UML

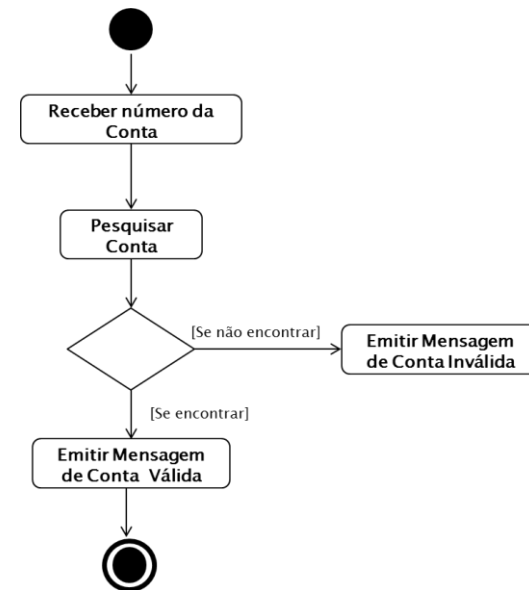
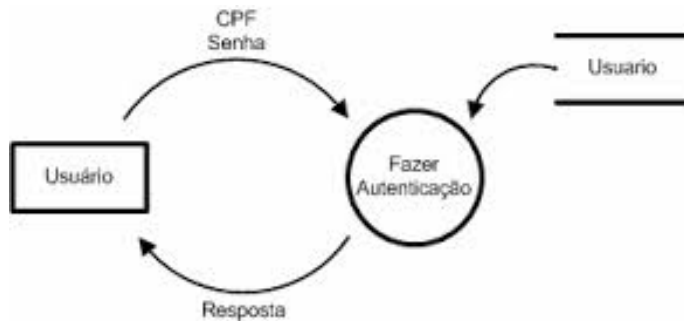




## Modelagem de Requisitos na Perspectiva Funcional

### ◆ Diagramas de Fluxo de Dados (DFD)

### ◆ Diagramas de Atividades UML

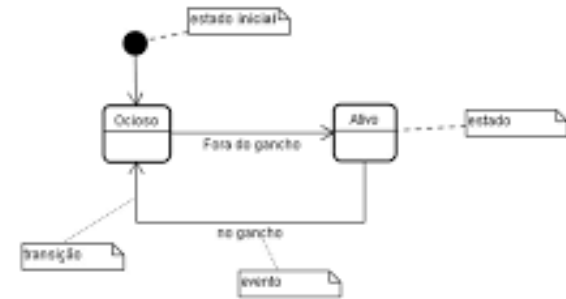
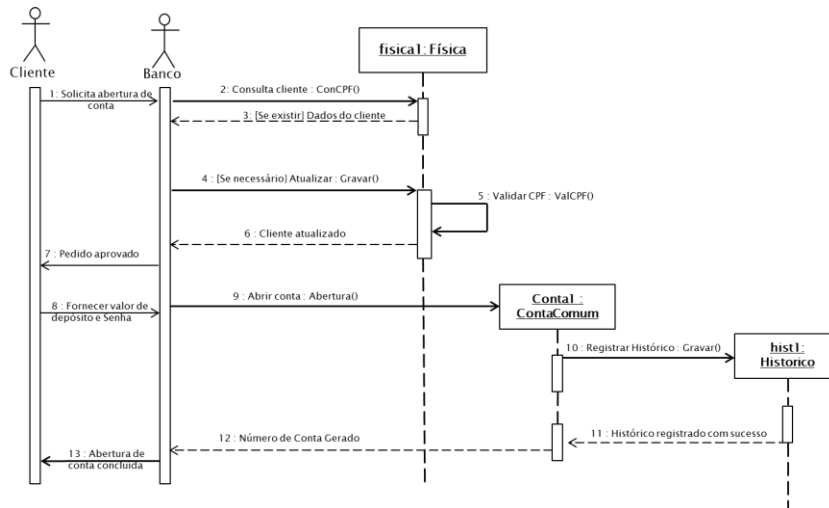




# Modelagem de Requisitos na Perspectiva Comportamental

## ◆ Diagrama de Estados UML

## ◆ Diagramas de Sequência UML





## Fundamentos da Validação de Requisitos

- ◆ Durante a Engenharia de Requisitos é necessário validar-se a **qualidade** dos requisitos desenvolvidos;
- ◆ O objetivo da Validação de Requisitos é descobrir **erros** nos requisitos documentados;
- ◆ A correção dos erros nos requisitos quando o sistema já está em operação implica em **custos significativamente altos**;
- ◆ **Erros críticos** em requisitos podem levar ao não cumprimento de acordos contratuais.





## Fundamentos da Negociação de Requisitos

- ◆ Se não houver consenso entre os stakeholders a respeito dos requisitos, cria-se um conflito;
- ◆ A aceitação do sistema é ameaçada por conflitos não resolvidos;
- ◆ No pior cenário, um conflito pode causar a retirada de apoio por parte do stakeholder, levando ao fracasso do projeto de desenvolvimento;
- ◆ O objetivo da negociação é chegar a uma compreensão comum e acordada dos requisitos do sistema a ser desenvolvido entre todos os stakeholders.





## Validação e Negociação de Requisitos

- ◆ São atividades realizadas ao **longo de todo o processo** de Engenharia de Requisitos;
- ◆ Portanto, **geram trabalhos adicionais** e, consequentemente, **custos adicionais**;
- ◆ Entretanto, trazem diversas **vantagens** tais como: redução do custo global do sistema, aumento da aceitação, estímulo para soluções criativas e inovações.







## Aspectos de Qualidade dos Requisitos

- ◆ **Conteúdo:** Todos os requisitos foram elicitados e documentados com o nível apropriado de detalhamento?
- ◆ **Documentação:** Todos os requisitos foram documentados em conformidade com as diretrizes de documentação e especificação previamente determinadas?
- ◆ **Acordo:** Todos os stakeholders concordam com os requisitos documentados e todos os conflitos conhecidos foram resolvidos?





## Aspectos de Qualidade “Conteúdo”

- ◆ **Compleitude global**: Todos os requisitos relevantes para o sistema a ser desenvolvido (ou para o próximo release do sistema) foram documentados?
- ◆ **Compleitude individual**: Cada requisito contém todas as informações necessárias?
- ◆ **Rastreabilidade**: Todos os relacionamentos relevantes de rastreabilidade foram definidos?
- ◆ **Exatidão/adequação**: Os requisitos refletem acuradamente os desejos e necessidades dos stakeholders?
- ◆ **Consistência**: É possível implementar todos os requisitos definidos para o sistema conjuntamente? Não há contradições?
- ◆ **Verificabilidade**: É possível definir critérios de aceitação e teste com base nos requisitos? Os critérios foram definidos?
- ◆ **Necessidade**: Cada requisito contribui para o cumprimento dos objetivos propostos?



## Riscos associados à Qualidade “Documentação”

- ◆ **Comprometimento das atividades de Desenvolvimento**: Formato empregado na documentação pode dificultar as atividades de desenvolvimento;
- ◆ **Compreensão**: Falha na documentação pode causar dificuldade de compreensão dos requisitos;
- ◆ **Incompletude**: Informações relevantes dos requisitos podem não estar documentadas;
- ◆ **Requisitos ignorados**: Se os requisitos não estiverem documentados no ponto onde deveriam, esses requisitos podem passar despercebidos em atividades posteriores.





## Aspectos de Qualidade “Documentação”

- ◆ Conformidade com o formato da documentação;
- ◆ Conformidade com a estrutura da documentação;
- ◆ Inteligibilidade;
- ◆ Não-ambiguidade;
- ◆ Conformidade com as regras de documentação.





## Aspectos de Qualidade “Acordo”



- ◆ Todos os stakeholders relevantes estão de acordo com cada requisito?
- ◆ Todos os stakeholders estão de acordo com cada requisito após o mesmo ter sido alterado?
- ◆ Todos os conflitos conhecidos com respeito a requisitos foram resolvidos?





## Gestão de Requisitos

- ◆ Designar **atributos** para requisitos (identificadores únicos ao longo de todo o processo de Engenharia de Requisitos);
- ◆ **Visualização** de Requisitos (para manter a complexidade dos requisitos dentro de limites práticos é necessário acessar os requisitos de forma seletiva, filtrando-se os requisitos de acordo com a tarefa atual);
- ◆ **Priorização** de Requisitos
- ◆ **Rastreabilidade** de Requisitos
- ◆ **Versionamento** de Requisitos
- ◆ Gerenciamento de **Mudanças** de Requisitos

