

# Requisitos

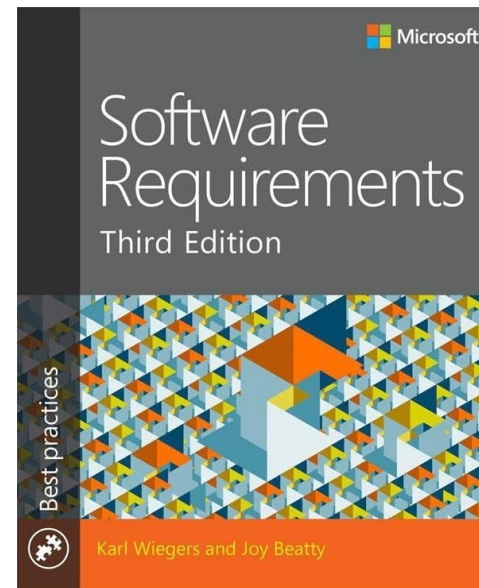
---

**Marcos de Oliveira**  
marcos.oliveira@ufc.br

*Disciplina Análise e Projeto de Sistemas*



<https://engsoftmoderna.info/cap3.html>



# 40% a 50%

dos **problemas** encontrados na entrega de um software são **originados** das atividades de **requisitos** de software.

---

# Importância dos Requisitos

- **Requisitos** definem o que um sistema deve fazer e sob quais restrições
- Os requisitos de um sistema devem atender às necessidades de seus usuários
  - Pois devem resolver os seus problemas
- Problemas na especificação de requisitos também têm um custo alto
  - Pois podem requerer retrabalho para os requisitos mal especificados
  - Ou trabalho extra para os requisitos que faltaram
- Corre-se o risco de entregar um sistema que vai ser rejeitado pelos seus usuários
  - Pois não se trata daquilo que eles precisavam

# Tipos de Requisitos

- Os dois tipos principais são

**Requisitos Funcionais**

*As funcionalidades que o sistema deve conter*

**X**

**Requisitos Não-Funcionais**

*As características que o sistema deve apresentar*

# Requisitos Funcionais

- São escritos utilizando **verbos de ação**, em linguagem natural.
- Alguns exemplos de Requisitos Funcionais para o sistema de um banco:
  - Informar o saldo de uma conta
  - Informar o extrato de uma conta
  - Realizar transferências entre contas
  - Pagar um boleto bancário
  - Cancelar um cartão de débito

# Requisitos Não Funcionais

- Esses requisitos estão relacionados com a qualidade do serviço prestado pelo sistema.
  - Definem restrições ao funcionamento do sistema.
- Escrever os Requisitos Não-Funcionais de um sistema não é uma tarefa fácil, pois:
  - Deve-se evitar especificações genéricas
    - Ex: "O sistema deve ser **rápido** e ter **alta** disponibilidade"
      - Quão rápido?
      - O que significa alta disponibilidade?
  - Deve-se usar especificações mais precisas
    - Ex: "O sistema deve ter **99,99%** de disponibilidade e **99%** de todas as transações realizadas em qualquer janela de **5 minutos** devem ter um tempo de resposta máximo de **1 segundo**"

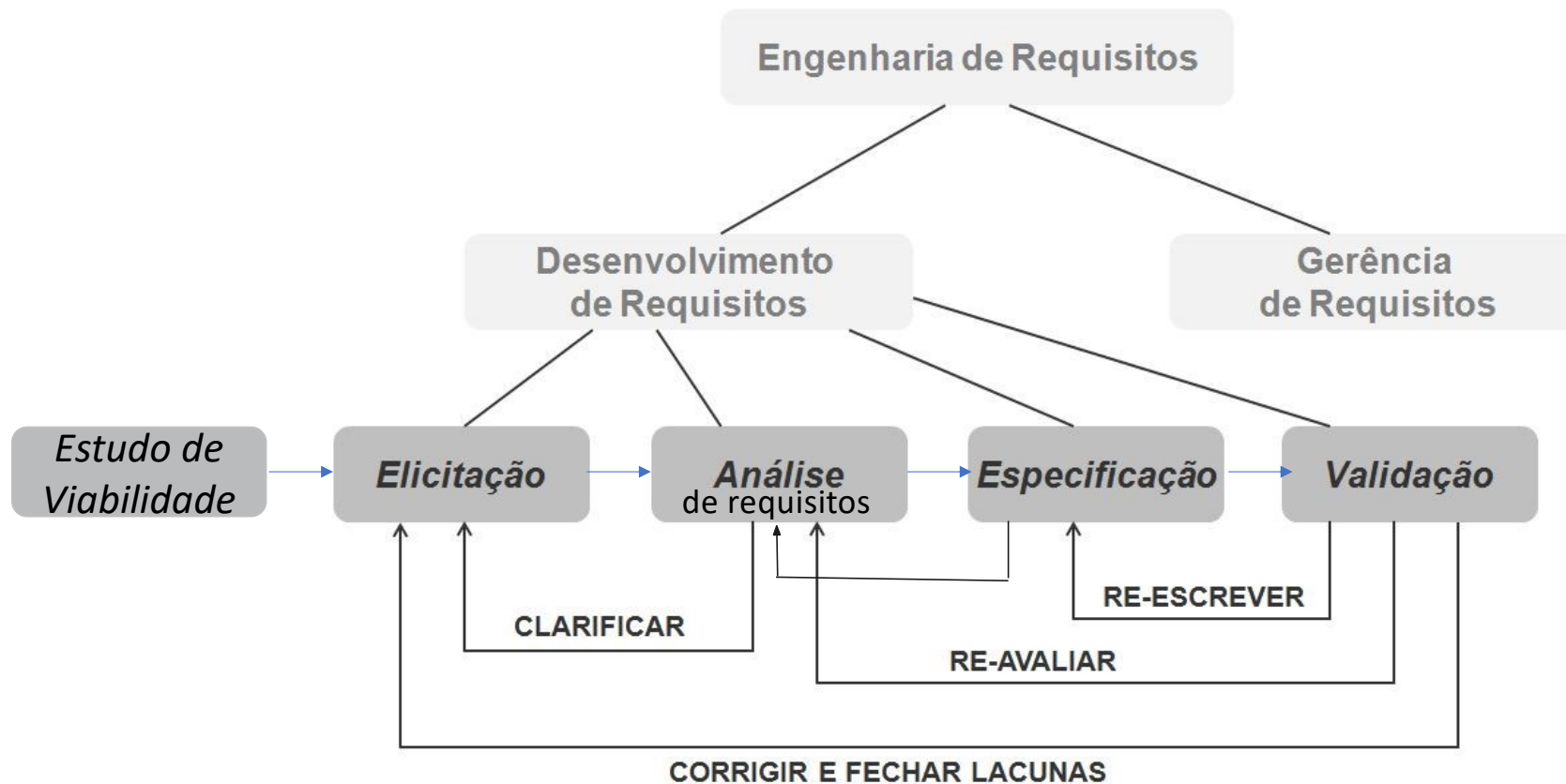
# Stakeholder

---

- Atividade de requisitos pode envolver profissionais com diferentes perfis na empresa
  - Em um sistema a ser desenvolvido para a área de telemarketing, os gerentes, atendentes, diretores, setor de recursos humanos, porteiros e profissionais do setor de Tecnologia da Informação da empresa podem estar envolvidos
- Dada esta diversidade, os envolvidos são denominados através do termo técnico Stakeholder



# Engenharia de Requisitos de Software



# Elicitação de Requisitos

- Primeiras atividades relacionadas com os requisitos de um sistema
- Objetivo: **descobrir** os principais requisitos do sistema que se pretende construir



# **Elicitação de Requisitos**

- Técnicas:
  - Entrevistas
  - Questionários
  - Análise de documentos e formulários da organização
  - Observação da execução de tarefas
  - Substituir o usuário
  - Brainstorming
  - Prototipação
  - Workshop ou oficina de requisitos

# Análise de Requisitos

- Atividades relacionadas ao estudo do resultado da elicitação
- Objetivo: **Entender** os principais requisitos do sistema que se pretende construir



Marcos de Oliveira (Análise e Projeto de Sistemas)

# Especificação de Requisitos

- Atividades relacionadas com a documentação dos requisitos do sistema
- Objetivo: documentar, especificar e priorizar os requisitos do sistema que se pretende construir
- Instrumentos:
  - Documento de Requisitos
  - Histórias do Usuário
  - Especificação de Casos de Uso
  - Diagramas de iStar
  - Diagramas de Casos de Uso



# Especificação de Requisitos

- \* Requisitos Relacionados com Interfaces Externas
  - \* Interfaces com o Usuário
  - \* Interfaces com Hardware
  - \* Interfaces com Outros Sistemas de Software
  - \* Interfaces de Comunicação
- \* Requisitos Funcionais
  - \* Requisito Funcional #1
  - \* Requisito Funcional #2
  - \* ....
- \* Requisitos de Desempenho
- \* Requisitos de Projeto
- \* Outros Requisitos

**As principais seções de um documento de requisitos IEEE 830**

# **Especificação de casos de uso**

- **Especificações de Casos de Uso são** documentos bastante detalhados para especificação de requisitos
- Recomendados para sistemas com requisitos mais estáveis, e podem ser críticos

# Especificação de Casos de Uso

- ✓ São documentos textuais de especificação de requisitos
- ✓ Documentam um acordo entre clientes e time de desenvolvimento
- ✓ Incluem descrições mais detalhadas do que histórias de usuário
- ✓ Escritos na fase de Especificação de Requisitos
- ✓ Podem ser lidos, entendidos e validados pelos usuários
- ✓ São escritos na perspectiva de um ator que deseja usar o sistema com um objetivo
- ✓ Enumera os passos que um ator realiza em um sistema com um determinado objetivo
- ✓ Inclui duas listas de passos: normal e extensões
  - **Fluxo normal:** um cenário onde tudo dá certo
  - **Fluxo de extensões:** representam alternativas ou situações de erro



# **Especificação de Casos de Uso - Formato**

- Deve ter um nome
  - Cujas primeiras palavras devem ser um verbo no infinitivo
- Deve informar atores envolvidos no caso de uso
- Pode também incluir um outro caso de uso
- Deve conter um fluxo normal de passos (happy day)
- Pode conter uma lista de extensões dos passos normais
  - Tratar erros, exceções, cancelamentos, etc
- Descrições de casos de uso podem incluir seções adicionais:
  - Propósito do caso de uso
  - Pré-condições: o que deve ser verdadeiro antes da execução do caso de uso
  - Pós-condições: o que deve ser verdadeiro após a execução do caso de uso
  - Uma lista de casos de uso relacionados

# Especificação de Casos de Uso - Exemplo

**Sistema:** Bancário

**Nome:** Transferir Valores entre Contas

**Ator:** Cliente do Banco

-> **Passo 1:** inclui o caso de uso "Autenticar Cliente"

-> **Passos 2a e 3a:** tratamento de erro

-> **Passo 4a:** detalhamento do passo 4

-> **Passos 5a e 5b:** detalhamento do passo 5



**Transferir Valores entre Contas**

**Ator:** Cliente do Banco

**Fluxo normal:**

1 -

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

# **Especificação de Casos de Uso - Boas práticas**

- ✓ As ações de um caso de uso devem ser escritas em uma linguagem simples e direta
- ✓ Sempre que possível, use o ator principal como sujeito das ações, seguido de um verbo
- ✓ Casos de uso devem ser pequenos, com poucos passos
- ✓ É necessário um nível de abstração maior do que em algoritmos
- ✓ Casos de uso não devem tratar de aspectos tecnológicos ou de design
- ✓ Casos de uso não precisam mencionar a interface que o ator usará para se comunicar com o sistema
- ✓ Não necessariamente o fluxo normal de um caso de uso precisa ser uma enumeração de ações
- ✓ Padronize o vocabulário adotado nos casos de uso

# ***Especificação de Caso de Uso x Histórias de usuário***

# Histórias de usuário

- Histórias de Usuários são documentos simplificados de especificação de requisitos
- Recomendada para sistemas cujos requisitos que estão muito sujeitos a mudanças, mas não críticos.

# Histórias de usuário

- ✓ É uma técnica pragmática
- ✓ Usada em processos de desenvolvimento ágeis
- ✓ Com elas todas as atividades de Engenharia de Requisitos ocorrem ao longo de todo o desenvolvimento
- ✓ Conta com um representante dos clientes, no time de desenvolvimento, para tirar dúvidas e explicar os requisitos para os desenvolvedores
- ✓ Favorecem comunicação verbal, em vez de comunicação escrita

*“Uma história é uma promessa: o representante dos clientes promete ter tempo para definir e explicar os detalhes em conversas com os desenvolvedores, durante o sprint no qual a história será implementada.”*

# Histórias de Usuário

Uma história de usuário é composta de três partes, todas começando com a letra C:

1. **Cartão:** usado pelos clientes para escrever uma funcionalidade que o sistema deve ter.
2. **Conversas:** entre clientes e desenvolvedores, em que os clientes detalham o que está em cada cartão.
3. **Confirmação:** É um teste de aceitação especificado pelo cliente.

“

*História de Usuário = Cartão + Conversas + Confirmação*

Como um candidato,  
eu posso buscar vagas de emprego

## Cartão

- Apresenta um claro valor de negócio e objetivo do usuário
- No entanto, **é difícil começar a implementar e testar baseado somente nessa descrição**. Por quê?



**Como um candidato,  
eu posso buscar vagas de emprego**

-----

***+ Opções de busca: por cidade, por estado,  
por título do cargo***

## **Conversas**

- Detalhes que são expostos durante a elicitação
- Ajudando a definir os limites (escopo)
- Sem exagero de detalhes

**Como um candidato,  
eu posso buscar vagas de emprego**



**VERSO**

-----  
**+ Opções de busca:  
por título do cargo**

**Confirmações:**

- + Busca sem campo algum preenchido não é realizada (falha)**
- + Apenas cidades com vagas são apresentadas (sucesso)**

## **Confirmações**

- Como o cliente avaliará o sistema?
- Como saber que já está pronto?
- Uso de exemplos como critérios de aceitação

# Exemplo Biblioteca

**Sistema:** Controle de Bibliotecas

**Usuário:** usuário típico



*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*

# Exemplo Biblioteca

**Sistema:** Controle de Bibliotecas

**Usuário:** professor



*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*

# Exemplo Biblioteca

**Sistema:** Controle de Bibliotecas

**Usuário:** funcionário



*Como funcionário da biblioteca, eu gostaria de cadastrar novos usuários*

*Como funcionário da biblioteca, eu gostaria de cadastrar novos livros*

*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*

# Validação de Requisitos

- As atividades relacionadas a verificação e validação dos requisitos documentados
- Objetivo: garantir que eles estejam:
  - **Corretos:** as informações de cada requisito devem estar corretas
  - **Precisos:** a interpretação de cada requisito não pode gerar ambiguidade
  - **Completo:** todos os requisitos devem estar especificados
  - **Consistentes:** não podem haver requisitos contraditórios
  - **Verificáveis:** cada requisito deve apresentar pelo menos uma maneira de ser testável
- Após finalizada essa atividade, os requisitos já podem ser priorizados.
  - Cada requisito apresentará a sua prioridade no sistema