



# Engenharia de requisitos

Herysson R. Figueiredo  
herysson.figueiredo@ufn.edu.br



Como o cliente explicou...



Como o líder de projeto entendeu...



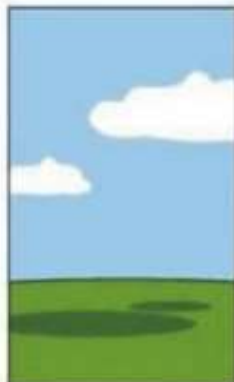
Como o analista projetou...



Como o programador construiu...



Como o consultor de negócios descreveu...



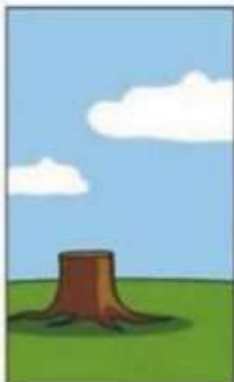
Como o projeto foi documentado...



Que funcionalidades foram instaladas...



Como o cliente foi cobrado...



Como foi mantido...



O que o cliente realmente queria...



## O que são requisitos?

“Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento.”

“Condição ou capacidade que um sistema (ou um componente) deve alcançar ou possuir para satisfazer um contrato, padrão especificação ou outro documento formalmente imposto”



## O que são requisitos?

Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações.



## O que é engenharia de requisito?

O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é chamado engenharia de requisitos (RE, do inglês *requirements engineering*).



## Como identificamos os requisitos?

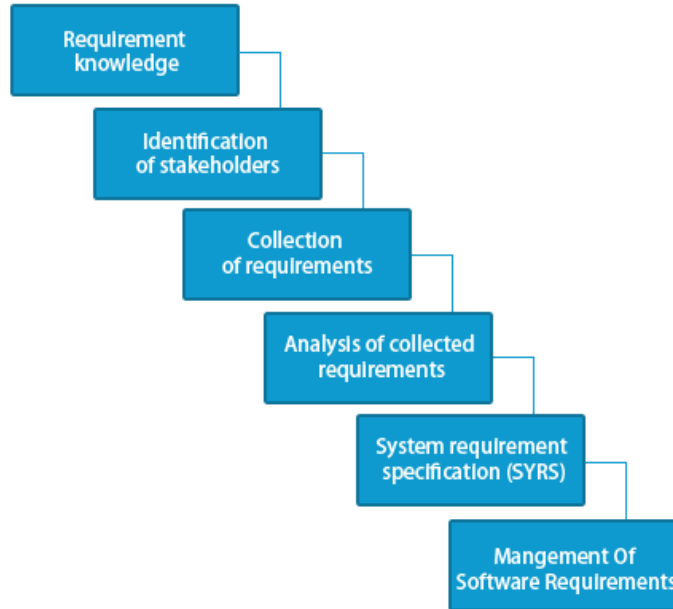
Identificamos os requisitos a partir dos **domínios de negócio**. Usando técnicas variadas para obtenção de informações sobre os requisitos que um sistema deve ter.

Os *stakeholders* (todos os interessados no sistema) possuem papel-chave na descoberta dos requisitos.

# Como identificamos os requisitos?

Identificamos os  
técnicas variadas par  
sistema deve ter.

Os *stakeholders*  
chave na descoberta



## Software Requirement Analysis

de negócio. Usando  
os requisitos que um

na) possuem papel-



## E o que é um Domínio de Negócio?

Também chamado de domínio do problema ou domínio da aplicação , trata-se da área específica na qual o *software* será desenvolvido.

O domínio é a área de conhecimento ou atividade específica caracterizada por um conjunto de conceitos e terminologias compreendidos por especialistas da área.

Corresponde à parte do mundo real que é relevante para o desenvolvimento do *software*.





## Análise de requisitos - Problemas

Problemas clássicos que precisam ser tratados com atenção no processo de levantamento de requisito são os seguintes:

- Comunicação do cliente com analista;
- Evolução dos requisitos;
- Gerenciamento de alterações;
- Falta de conhecimento sobre o Domínio.



## O que obtemos levantando Requisitos

Um documento contendo todos os requisitos que o sistema abará.

Muitas vezes esse documento é escrito em notação informal (linguagem natural) ou em forma de **Casos de Uso**.

Assim, pessoas não-técnicas podem compreendê-lo facilmente.



## Atividades do levantamento de requisitos

Há três tipos de atividades realizadas no processo de levantamento de requisitos:

- **Elicitação dos requisitos:**
- **Análise de requisitos:**
- **Registro dos requisitos:**



## Atividades do levantamento de requisitos

Há três tipos de atividades realizadas no processo de levantamento de requisitos:

- **Elicitação dos requisitos:** Comunicação com os *stakeholders* para determinar quais são os requisitos de sistemas.
- **Análise de requisitos:**
- **Registro dos requisitos:**



## Atividades do levantamento de requisitos

Há três tipos de atividades realizadas no processo de levantamento de requisitos:

- **Elicitação dos requisitos:**
- **Análise de requisitos:** Determina o estado do requisito (inacabado, incompleto, ambíguo, contraditório)
- **Registro dos requisitos:**



## Atividades do levantamento de requisitos

Há três tipos de atividades realizadas no processo de levantamento de requisitos:

- **Elicitação dos requisitos:**
- **Análise de requisitos:**
- **Registro dos requisitos:** Documentar os requisitos de várias formas, como usando linguagem natural, casos de uso, ou ainda processo de especificação.



# Tipos de Requisitos

Podemos classificar os requisitos em dois grandes grupos:

- Requisitos Funcionais (RF)
- Requisitos Não-Funcionais (RNF)



## Requisitos Funcionais (RF)

Definem as funcionalidades do sistema. Basicamente, especificam o que o sistema deve fazer. Sem a implementação deste requisito, o sistema simplesmente não funciona como deveria - se chegar a funcionar.





## Requisitos Funcionais (RF)

“Os requisitos funcionais de um sistema descrevem **o que** ele deve fazer. São declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações.” (SOMMERVILLE, 2007)



## Exemplo de Requisitos Funcionais

- [RF001] O sistema deve permitir que os professores realizem o lançamento de notas e faltas dos alunos;
- [RF002] O sistema deve permitir a emissão do histórico escolar.
- [RF003] Deve ser possível calcular as notas médias para cada disciplina
- [RF004] Deve ser possível emitir boletos para pagamento das mensalidades dos cursos.
- [RF005] O sistema deve gerar nota fiscal eletrônica (NF-e) dos serviços prestados.



## Requisitos Não-Funcionais (RNF)

Indicam características de qualidade que o sistema deve possuir que estão relacionadas às funcionalidades previstas.



## Requisitos Não-Funcionais (RNF)

“Os requisitos não funcionais, como o nome sugere, são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área.”(SOMMERVILLE, 2007)



## Requisitos Não-Funcionais (RNF)

Algumas características de qualidade são:

- Desempenho;
- Confiabilidade;
- Escalabilidade
- Portabilidade
- Usabilidade
- Segurança

Os critérios que qualificam os requisitos não-funcionais são geralmente mensuráveis.



## Classificação dos RNF

Os requisitos Não-Funcionais podem ser classificados em várias categorias de requisitos, como:

- **Requisitos de Produtos**
- **Requisitos Organizacionais**
- **Requisitos Externos**



## Classificação dos RNF

Os requisitos Não-Funcionais podem ser classificados em várias categorias de requisitos, como:

- **Requisitos de Produtos:** Esses requisitos especificam ou restringem o comportamento do software
  - Requisitos de usabilidade
  - Requisitos de Eficiência
  - Requisito de confiabilidade
  - Requisito de portabilidade
- **Requisitos Organizacionais**
- **Requisitos Externos**



## Classificação dos RNF

Os requisitos Não-Funcionais podem ser classificados em várias categorias de requisitos, como:

- **Requisitos de Produtos**
- **Requisitos Organizacionais:** Esses são os requisitos gerais de sistemas derivados das políticas e procedimentos da organização do cliente e do desenvolvedor
  - Requisito de entrega
  - Requisito de implementação
  - Requisito de padrões
- **Requisitos Externos**





## Classificação dos RNF

Os requisitos Não-Funcionais podem ser classificados em várias categorias de requisitos, como:

- **Requisitos de Produtos**
- **Requisitos Organizacionais**
- **Requisitos Externos:** Esse tipo abrange todos os requisitos que derivam de fatores externos ao sistema e seu processo de desenvolvimento
  - Requisitos de interoperabilidade
  - Requisitos éticos
  - Requisitos legais



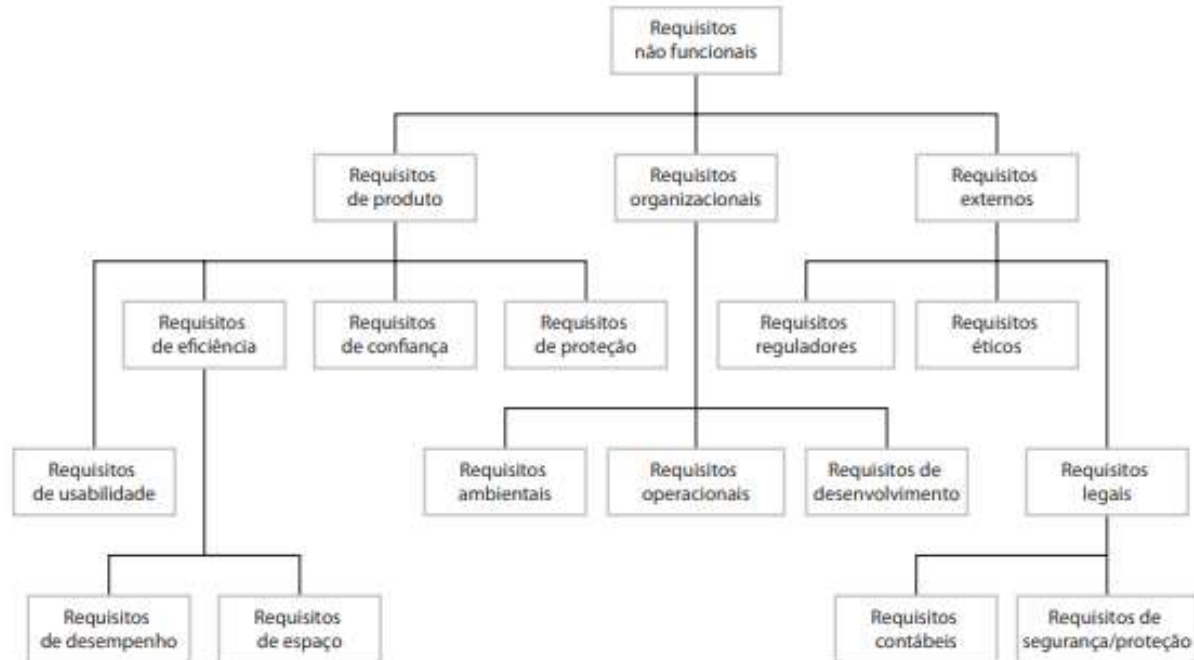
## Classificação dos RNF

Os requisitos Não-Funcionais podem ser classificados em várias categorias de requisitos, como:

- **Requisitos de Produtos**
- **Requisitos Organizacionais**
- **Requisitos Externos**

Na prática todos são listados no mesmo documento.

# Tipos de Requisitos Não-Funcionais





## Métricas para especificar requisitos não funcionais

Propriedade	Medida
Velocidade	Transações processadas/segundo Tempo de resposta de usuário/evento Tempo de atualização de tela
Tamanho	Megabytes Número de chips de memória ROM
Facilidade de uso	Tempo de treinamento Número de <i>frames</i> de ajuda
Confiabilidade	Tempo médio para falha Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade
Robustez	Tempo de reinício após falha Percentual de eventos que causam falhas Probabilidade de corrupção de dados em caso de falha
Portabilidade	Percentual de declarações dependentes do sistema-alvo Número de sistemas-alvo



## Exemplo de Requisitos Não-Funcionais

- [RNF001] O sistema poder ser acessado pela *web*
- [RNF002] O sistema deve ser implementado usando linguagem Python
- [RNF003] O acesso ao sistema não pode demorar mais do que 10 segundos
- [RNF004] Professores devem ser capazes de utilizar o sistema com um treinamento de 4 horas
- [RNF005] Um aluno não poderá ter acesso aos dados dos outros alunos



## Requisitos Funcionais x Não-Funcionais

Funcionais	Não-Funcionais
Mandatários	Não mandatários
Visualizados em Casos de Uso	Atribuídos (Características)
Funcionalidade do Produto	Propriedade do Produto
Relativamente fáceis de descobrir	Mais difíceis de descobrir em geral
Necessidades do negócio	Expectativas do negócio
Verbo	



# Técnicas para Levantamento de Requisitos

Existem diversas técnicas que podem ser empregadas para realizar levantamento de requisitos em um projeto de sistemas.

- *Workshop* de requisitos
- Entrevistas com *stakeholders*
- Questionários
- *Brainstorming*
- Prototipagem
- Etnografia
- JAD (*Join Application Design*)
- VORD (*Viewpoint-Oriented Requirements Definition*)



## Prática

1. Crie um documento contendo os requisitos funcionais e não funcionais para o desenvolvimento de um sistema informatizado de uma biblioteca. Quanto aos requisitos não funcionais, liste no mínimo um para cada uma das propriedades : velocidade, tamanho, facilidade de uso, confiabilidade, robustez, portabilidade e segurança.
2. Usando sua criatividade e seu imagine um software a ser desenvolvido e crie um documento contendo os seus requisitos.





## Bibliografia

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: guia do usuário. 2. ed. Rio de Janeiro: Campus, 2006.

PRESSMAN, Roger S. Engenharia de software. 5. Ed. Rio de Janeiro: McGraw-Hill, 2002.

SOMMERVILLE, I. Engenharia de software. 10. Ed. São Paulo, SP: Addison Wesley, 2018

BEZERRA, Eduardo. Princípios de Análise e Projeto de Sistemas com UML. Rio de Janeiro: Campus, 2006.