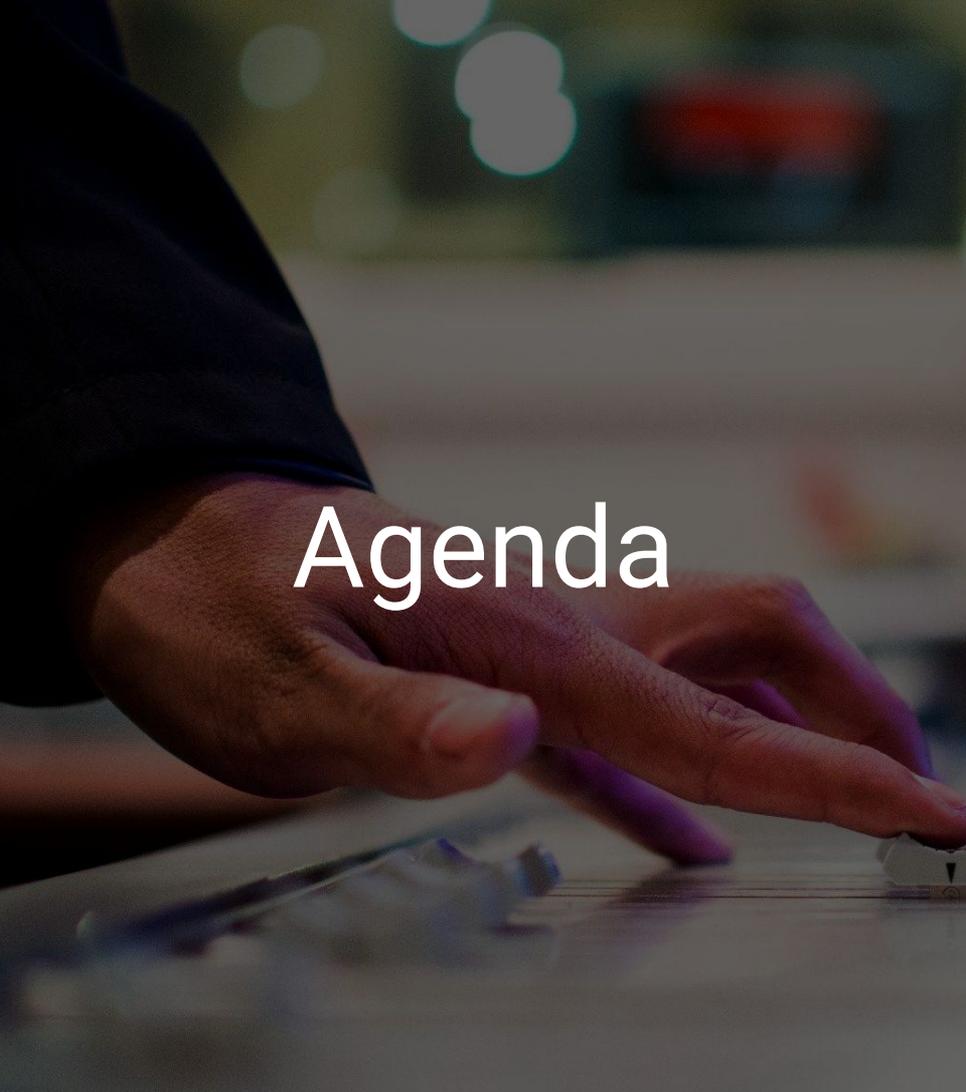


# Requisitos de Software

Prof: Nilson Júnior  
[nilson.junior@jaboatao.ifpe.edu.br](mailto:nilson.junior@jaboatao.ifpe.edu.br)

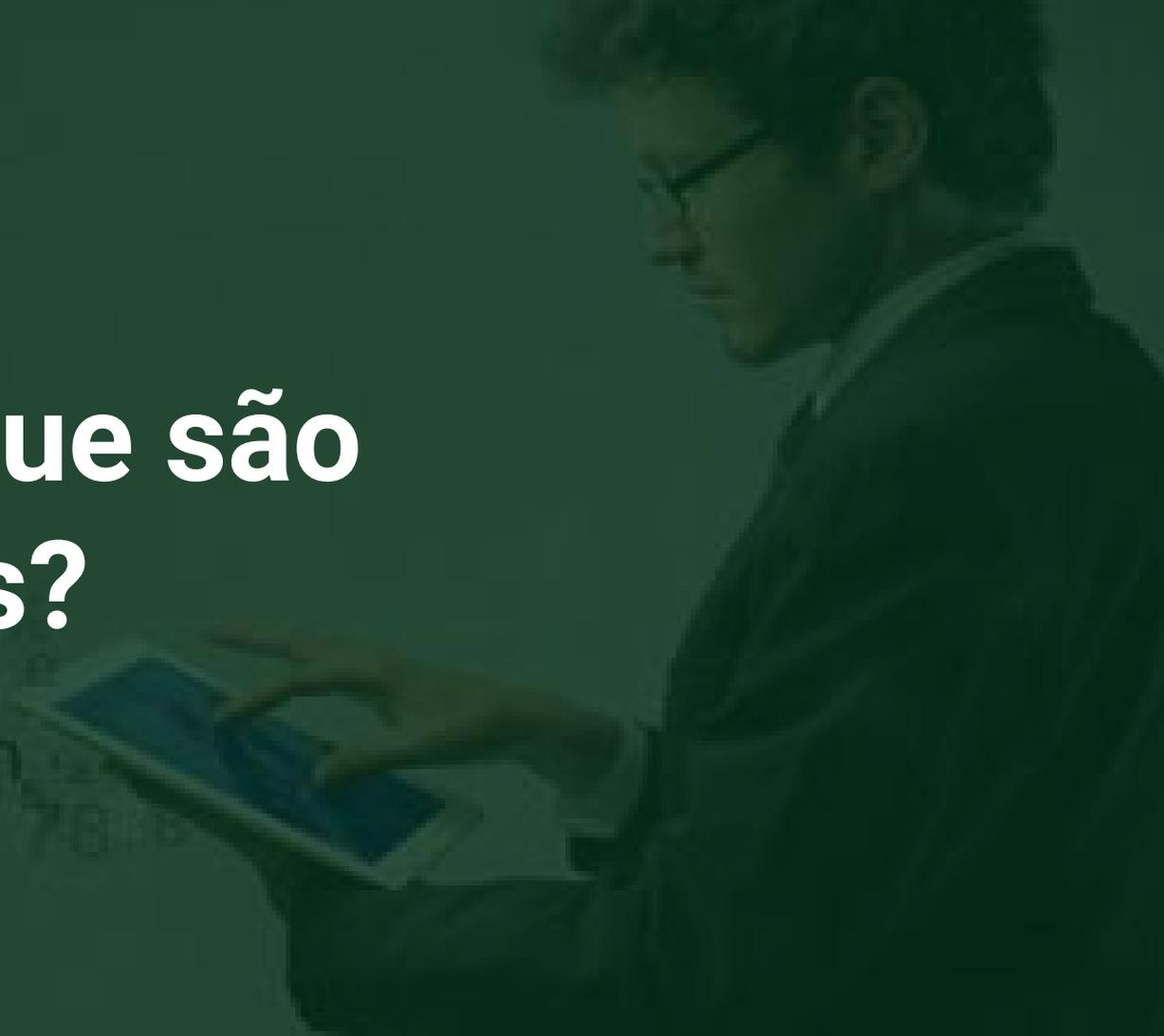
[www.professornilson.wix.com/ifpe](http://www.professornilson.wix.com/ifpe)

A close-up photograph of a person's hands writing on a whiteboard with a purple marker. The background is blurred, showing some bokeh lights. The word 'Agenda' is overlaid in white text on the left side of the image.

# Agenda

- Introdução a Requisitos de Software
- Conhecer um Processo de Gerenciamento de Requisitos
- Técnicas de Levantamento de Requisitos

**Afinal, o que são  
Requisitos?**



# Requisitos

É o que **estabelece** os serviços que o **cliente requer** de um sistema e as **restrições** sob as quais tal sistema operará e será **desenvolvido**.

Tais serviços e restrições são chamados de **requisitos**

# O que são Requisitos?

Pode ser uma descrição abstrata de alto nível de um serviço, uma restrição de sistema, entre outras coisas.

O problema cujo desenvolvimento do sistema deve resolver

O sistema tem que ser construído de modo a satisfazer todos os seus requisitos



# Tipos de Requisitos

# Características

## Requisitos de usuário

- Declarações de alto nível escritas em linguagem natural Escritos para os clientes.

## Requisitos de sistema

- Um documento estruturado estabelecendo descrições detalhadas das funções, serviços e restrições operacionais do sistema.
- Define o que deve ser implementado e pode até ser parte de um contrato entre o cliente e o desenvolvedor.

# Requisitos de Usuário

## Requisitos de usuário

- Requisitos funcionais e não-funcionais descritos de modo a ser compreensíveis por usuários que não têm conhecimento técnico detalhado.
- São definidos usando uma linguagem simples, tabelas e diagramas quando estes podem ser compreendidos por todos os usuários.
- Histórias de usuários são similares a requisitos de usuários.

# Diretrizes para escrever Requisitos

- Usar um formato padrão para todos os requisitos.
- Usar a linguagem de uma forma consistente. 'deve' para requisitos obrigatórios, e 'deveria' para requisitos desejáveis.
- Realçar o texto para identificar as partes principais do requisito. Evitar o uso de jargões de computação.

# Requisitos de Sistemas

- Especificações mais detalhadas das funções do sistema, dos serviços e das restrições
- Visam fornecer ser uma base para o desenvolvimento do sistema Em XP: histórias de usuário + tarefas de desenv.
- Eles podem ser incorporados no contrato de sistema.
- Requisitos de sistema podem ser definidos ou ilustrados usando notações gráficas

## Definição de requisitos de usuário

1. LIBSYS deve manter o acompanhamento de todos os dados exigidos pelas agências de licenciamento de direitos autorais no Reino Unido e em outros lugares.

## Especificação dos requisitos de sistema

1.1 Ao solicitar um documento ao LIBSYS, deve ser apresentado ao solicitante um formulário que registra os detalhes do usuário e da solicitação feita.

1.2 Os formulários de solicitação do LIBSYS devem ser armazenados no sistema durante cinco anos, a partir da data da solicitação.

1.3 Todos os formulários do LIBSYS devem ser indexados por usuário, nome do material solicitado e fornecedor da solicitação.

1.4 O LIBSYS deve manter um registro de todas as solicitações feitas ao sistema.

1.5 Para materiais aos quais se aplicam os direitos de empréstimo dos autores, os detalhes do empréstimo devem ser enviados mensalmente às agências de licenciamento de direitos autorais que se registraram no LIBSYS.

The background of the slide is a solid dark green color. Overlaid on this background are several semi-transparent, light green gear icons of various sizes and orientations, creating a mechanical or industrial aesthetic. The largest gear is positioned in the upper left quadrant, while other smaller gears are scattered across the upper and right portions of the slide.

# Requisitos Funcionais e não Funcionais

## Requisito Funcional

- São **declarações** de serviços que o sistema deve **prover, descrevendo** o que o sistema deve **fazer**. Um **requisito funcional** descreve uma interação entre o sistema e o seu ambiente, podendo descrever, ainda, como o sistema deve reagir a entradas específicas, como o sistema deve se comportar em situações específicas e o que o sistema não deve fazer.

## Requisito não Funcional

- Descrevem **restrições** sobre os serviços ou funções oferecidas pelo sistema, as quais **limitam** as opções para criar uma solução para o problema. Neste sentido, os **requisitos não funcionais** são muito **importantes para a fase de projeto** (design), servindo como base para a tomada de decisões nessa fase.

## Requisitos funcionais

- Serviços que o sistema deve fornecer
- Como o sistema deve reagir a entradas específicas
- Como o sistema deve se comportar em determinadas situações.

## Requisitos não-funcionais ou de qualidade

- Restrições sobre serviços ou funções oferecidos pelo sistema tais como restrições de timing, restrições sobre o processo de desenvolvimento, padrões, etc.

**Exemplo**

**EXAMPLE**

# Sistema LIBSYS

- Um sistema de biblioteca que fornece uma interface única para uma série de banco de dados de artigos em bibliotecas diferentes. Os usuários podem pesquisar, baixar e imprimir estes artigos para estudo pessoal.

## Requisito Funcional

- O usuário deve ser capaz de **pesquisar** em todo o conjunto inicial de banco de dados ou selecionar um subconjunto a partir dele.
- Para todo pedido deve ser alocado um identificador único que o usuário possa copiar para a área de armazenamento permanente da sua conta.
- O sistema deve fornecer **telas apropriadas** para o usuário ler os documentos no repositório de documentos.

# Imprecisão de Requisitos

Problemas surgem quando os requisitos não são precisamente definidos. Requisitos ambíguos podem ser interpretados de maneiras diferentes pelos desenvolvedores e usuários.

## Considere o termo 'telas apropriadas

- **Intenção do usuário** – tela de propósito especial para cada tipo diferente de documento;
- **Interpretação do desenvolvedor** – fornece uma tela de texto que mostra o conteúdo do documento.

# Requisitos completos e consistentes

Em princípio, requisitos devem ser completos e consistentes.

## **Completude**

Eles devem incluir descrições de todos os recursos requeridos.

## **Consistência**

Não deve haver conflitos ou contradições nas descrições dos recursos de sistema.

Na prática, é impossível produzir um documento de requisitos completo e consistente.

# Requisitos não Funcionais

**Definem propriedades e restrições de sistema**

Exemplos incluem **confiabilidade, tempo de resposta e requisitos de armazenamento.**

Restrições são **capacidades** de dispositivos de E/S, representações de sistema, etc.

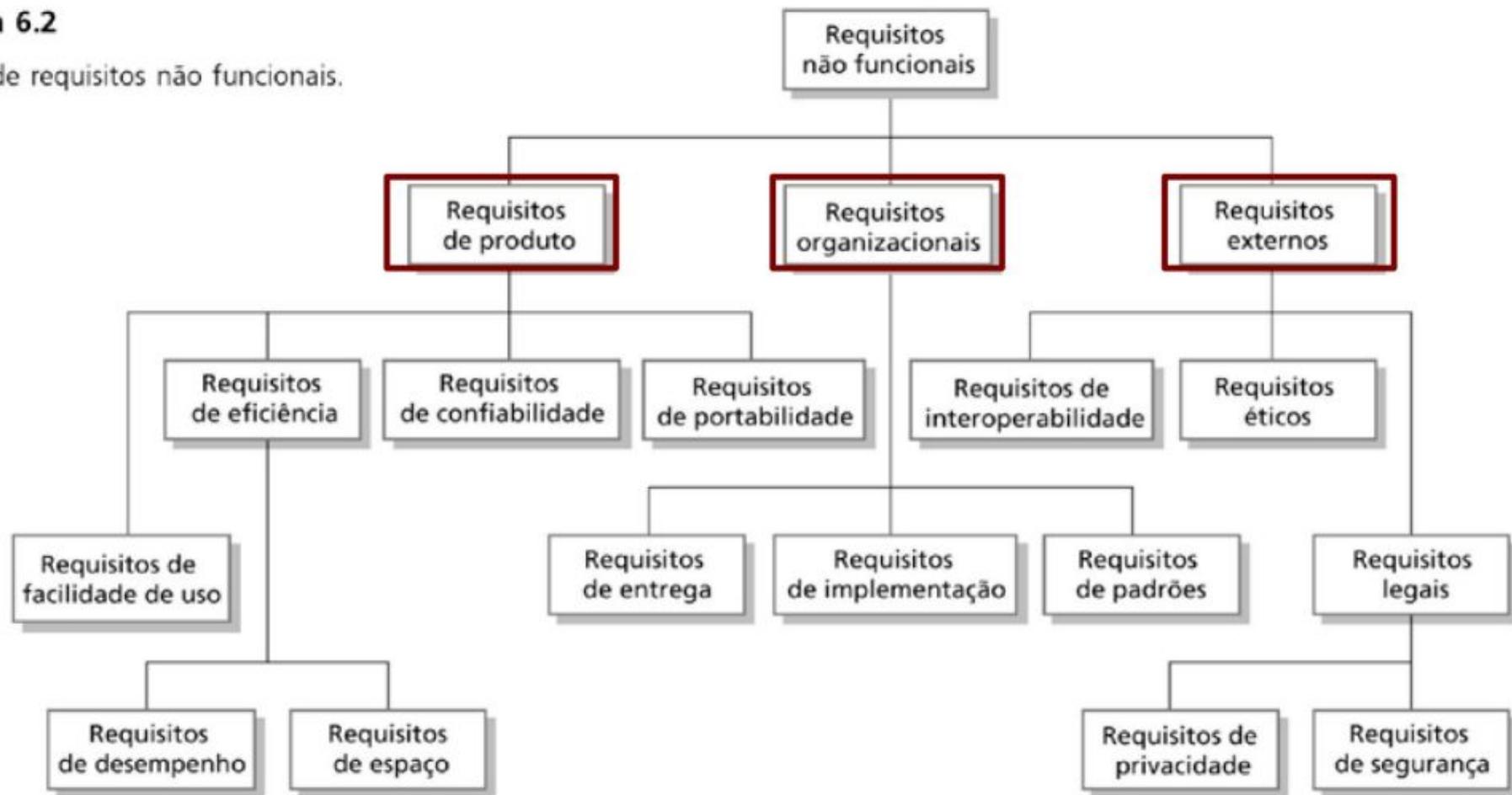
**Requisitos de processo** podem também ser **especificados**, impondo uma linguagem de programação, IDE ou método de desenvolvimento particular

**Requisitos não-funcionais** podem ser mais **críticos** do que os **requisitos funcionais.**

# Tipos de Requisitos não Funcionais

Figura 6.2

Tipos de requisitos não funcionais.



### **Requisito de produto**

8.1 A interface de usuário para o LIBSYS deve ser implementada como simples HTML, sem frames ou applets de Java.

### **Requisito organizacional**

9.3.2 O processo de desenvolvimento do sistema e os documentos a serem entregues devem estar em conformidade com o processo e produtos a serem entregues definidos em XYZCo-SP-STAN-95.

### **Requisito externo**

10.6 O sistema não deve revelar quaisquer informações pessoais sobre os usuários do sistema ao pessoal da biblioteca que usa o sistema, com exceção do nome e número de referência da biblioteca.

# Metas e Requisitos



# Metas e Requisitos

**Requisitos não-funcionais podem ser difíceis de definir precisamente**

Requisitos imprecisos podem ser difíceis de verificar.

## **Meta**

Uma intenção geral do usuário, tal como facilidade de uso.

## **Requisito não-funcional verificável**

Uma declaração usando alguma medida que pode ser objetivamente testada.

Metas são úteis para desenvolvedores quando exprimem as intenções dos usuários do sistema.

### **Meta do sistema**

O sistema deve ser fácil de ser usado pelos controladores experientes e ser organizado de modo que os erros dos usuários sejam minimizados.

### **Requisito não funcional verificável**

Os controladores experientes devem ser capazes de usar todas as funções do sistema depois de um treinamento no total de duas horas. Após esse treinamento, o número médio de erros cometidos pelos usuários experientes não deve exceder dois por dia.

# Métricas para especificar Requisitos não Funcionais

Propriedade	Medida
Velocidade	Transações processadas/segundo Tempo de resposta de usuário/evento Tempo de atualização da tela
Tamanho	Kbytes Número de chips de RAM
Facilidade de uso	Tempo de treinamento Número de frames de ajuda
Confiabilidade	Tempo médio de falha Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade
Robustez	Tempo para reiniciar após falha Porcentagem de eventos que causam falhas Probabilidade de corrupção de dados por falhas
Portabilidade	Porcentagem de declarações dependentes do sistema-alvo Número de sistemas-alvo

# Requisitos e Projeto

DESIGN =  
ETHICS +  
AESTHETICS

NEW  
DESIGN  
PRINCIPLES

The hope  
Lies in the  
individuals

HERITAGE

10/11  
WORLD  
BRAIN

EMPLOYMENT

DIFFERENCE =  
ETHICS

SMALL CAUSES  
↓  
BIG EFFECTS

GUIDING  
PRINCIPLES  
↓  
BUSINESS DESIGN

GLOBAL  
PLAY



CRITERIA  
CHECK LIST

- Requisitos devem definir o que o sistema deve fazer e o projeto deve descrever como ele faz isto.
  - Requisitos => problema
  - Projeto => solução

## **Na prática, requisitos e projeto são inseparáveis**

- Uma arquitetura de sistema pode ser projetada para estruturar os requisitos;
- O sistema pode ter que interoperar com outros sistemas que geram novos requisitos;
- O uso de uma solução de projeto específica pode ser um requisito de domínio.

## **Falta de clareza**

É difícil atingir uma precisão sem tornar o documento difícil de ler e ambíguo

## **Confusão de requisitos**

Requisitos funcionais e não-funcionais tendem a estar misturados.

## **Fusão de requisitos**

Vários requisitos diferentes podem ser expressos juntos Dificuldade de estruturar a especificação

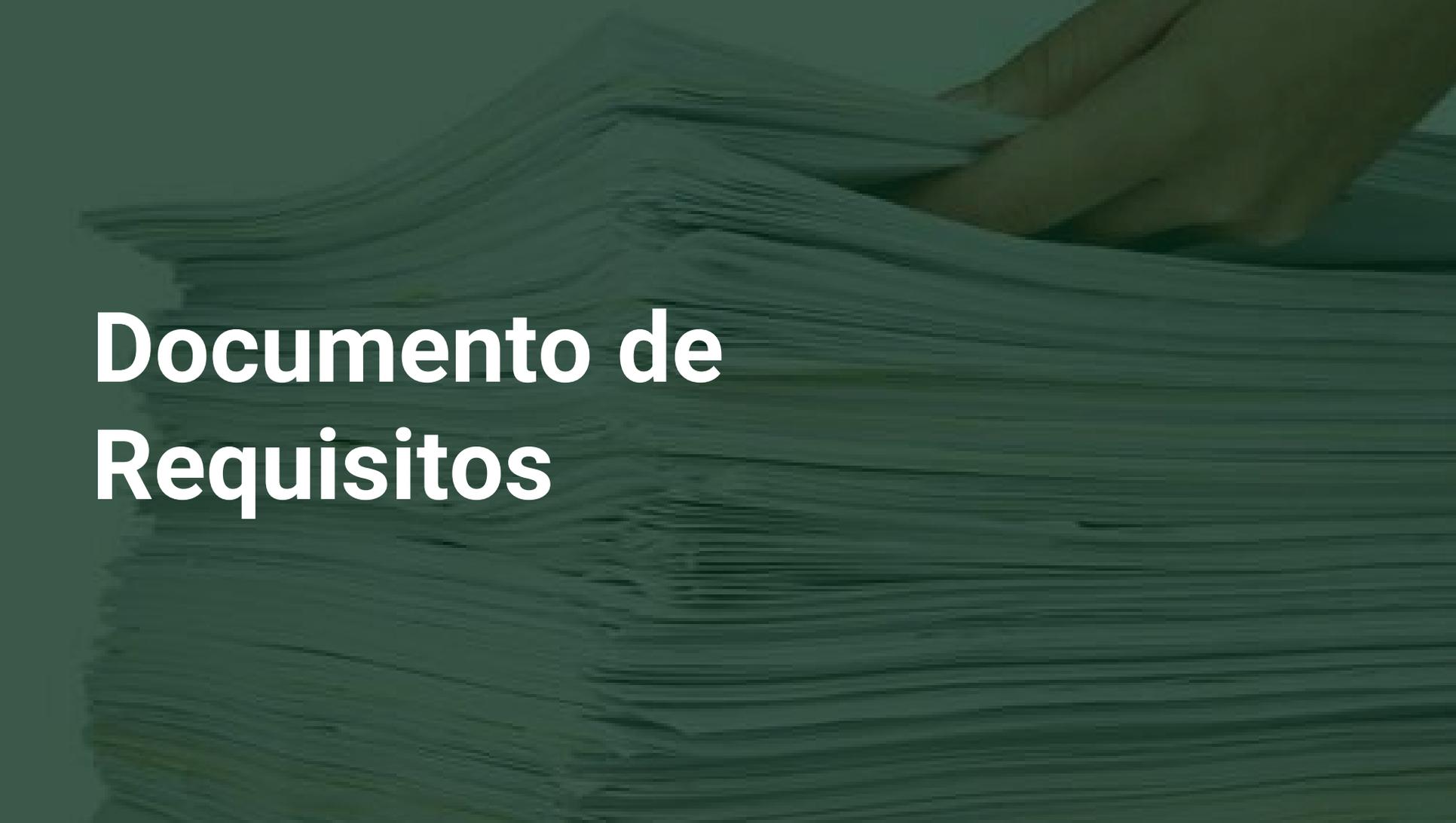
## Especificação Em linguagem Estruturada

A liberdade do elaborador de requisitos é **limitada** por um **template pré-definido** para requisitos.

**Todos os requisitos** são escritos de maneira **padronizada**.

A **terminologia** usada na descrição pode ser **limitada**.

A **vantagem** é que a maior parte da expressividade da linguagem natural é mantida Mas o **grau de uniformidade é imposto na especificação**.

A hand is shown holding a stack of papers, with the text 'Documento de Requisitos' overlaid in white. The background is a dark teal color.

# Documento de Requisitos

# Documento de Requisitos

O documento de requisitos é a declaração oficial do que é requisitado pelos desenvolvedores do sistema.

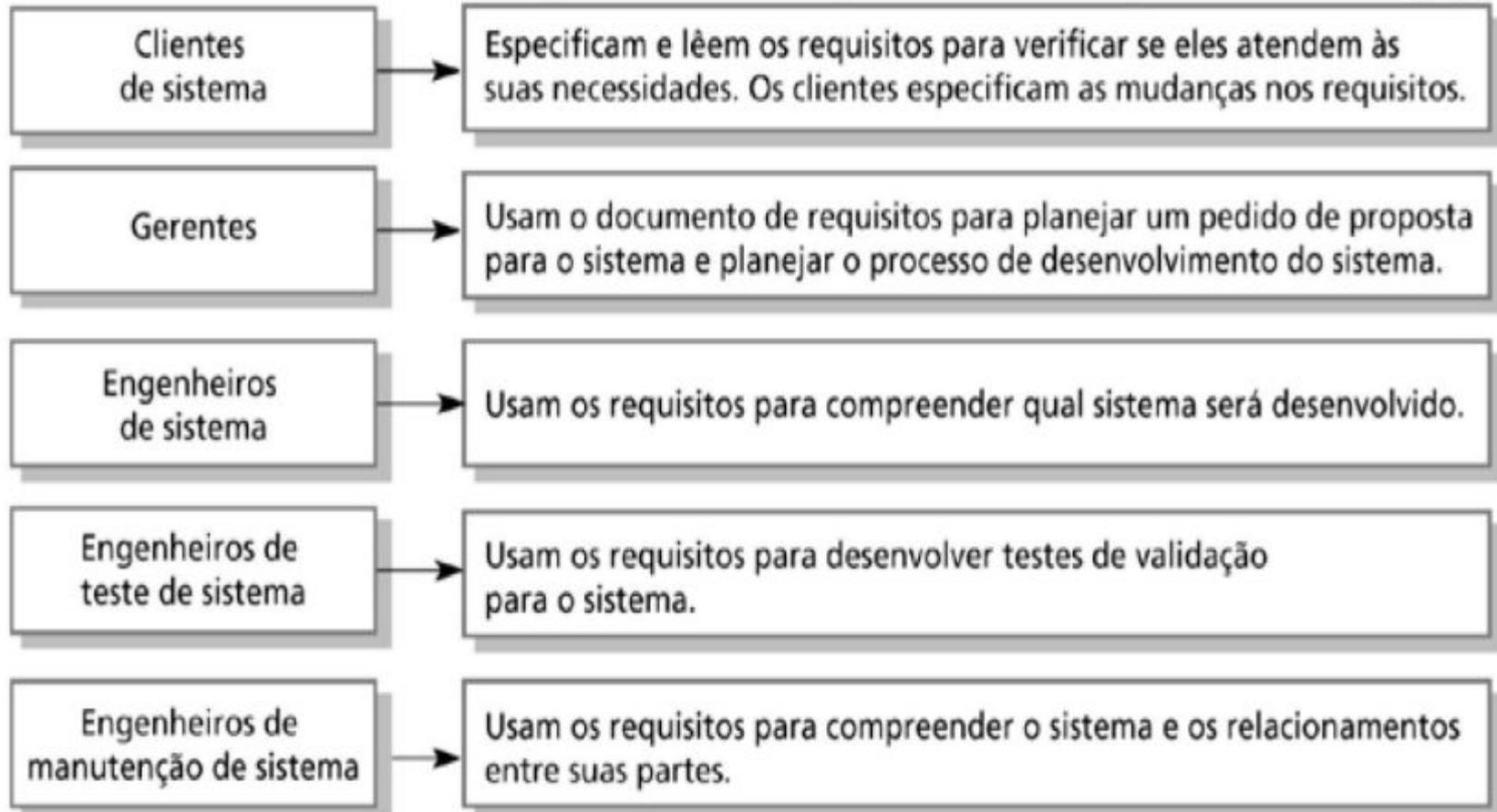
- Em XP é um pouco diferente

Deve incluir ambos, uma definição dos requisitos de usuário e uma especificação dos requisitos de sistema.

- NÃO É um documento de projeto.

Logo que possível, será preciso definir como o sistema deve fazer, ao invés de o que deve ser feito.

# Usuários de Documento de Software



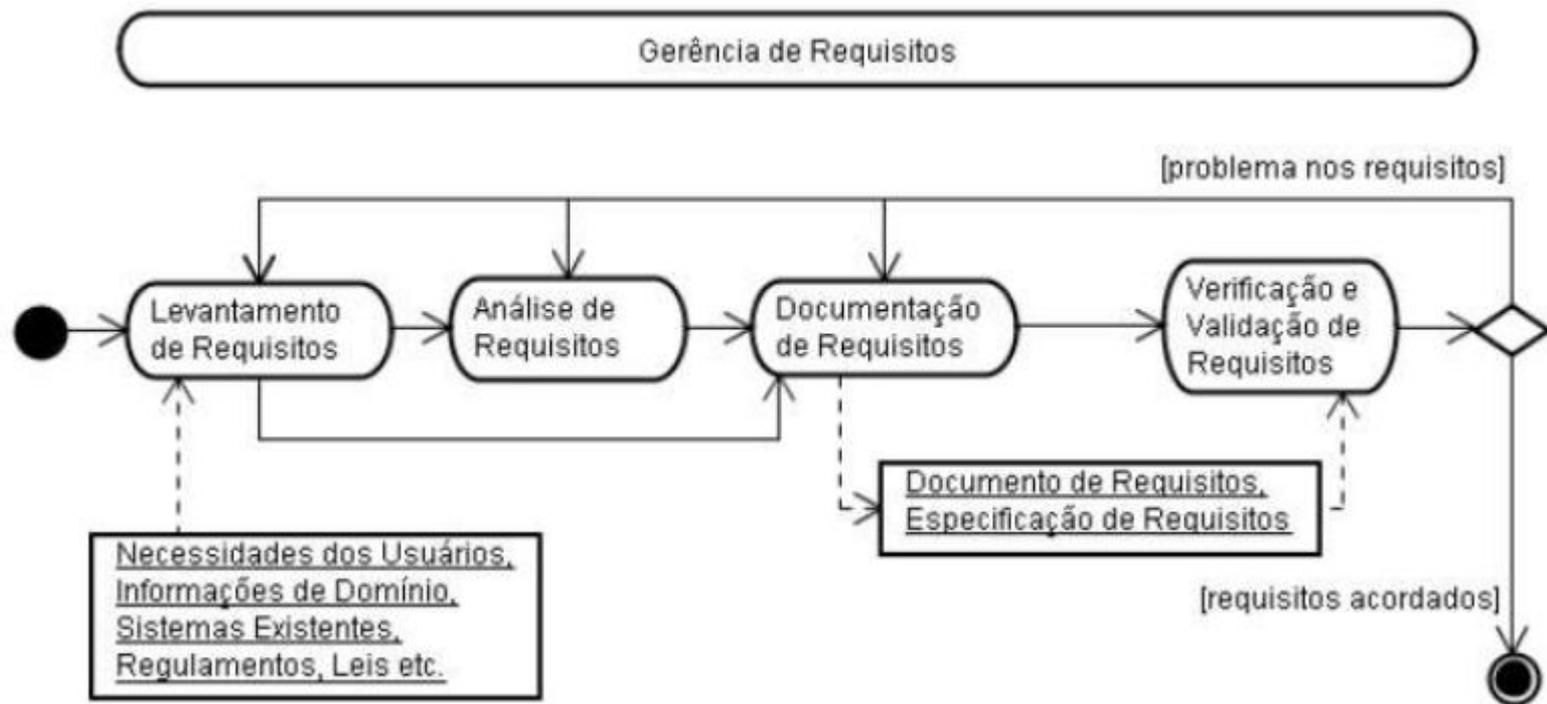
# Stackholders

É um termo usado em diversas áreas como **gestão de projetos**, comunicação social (Relações Públicas) **administração** e **arquitetura de software** referente às partes interessadas que devem estar de acordo com as práticas de governança corporativa executadas pela empresa.

The background is a dark green color with a pattern of white line-art icons. These icons include various business and technology symbols such as bar charts, pie charts, lightbulbs, a laptop, a smartphone, a cloud, and a person in a suit pointing at a screen. Some text elements are also visible in the background, including 'IDEA 45%', 'SALE!', 'TEAM', 'internet+', and 'CLOUD'.

# Gerenciamento de Requisitos

# Processo de Engenharia de Requisitos



# Levantamento de Requisitos

O Levantamento de Requisitos corresponde à fase inicial do processo de engenharia de requisitos e envolve as atividades de descoberta dos requisitos. Nessa fase, um esforço conjunto de clientes, usuários e especialistas de domínio é necessário, com o objetivo de entender a organização, seus processos, necessidades, deficiências dos sistemas de software atuais, possibilidades de melhorias, bem como restrições existentes.

# Análise de Requisitos

Uma vez identificados requisitos, é possível iniciar a atividade de análise, quando os requisitos levantados são usados como base para a modelagem do sistema. Requisitos de usuário são escritos tipicamente em linguagem natural, pois eles devem ser compreendidos por pessoas que não sejam especialistas

# Documentação de Requisitos

Os requisitos e modelos capturados nas etapas anteriores devem ser descritos e apresentados em documentos. A documentação é, portanto, uma atividade de registro e oficialização dos resultados da engenharia de requisitos. Como resultado, um ou mais documentos devem ser produzidos.

# Verificação e Validação de Requisitos

O objetivo da **verificação** é assegurar que o software esteja sendo construído de **forma correta**. Deve-se verificar se os artefatos produzidos atendem aos requisitos estabelecidos e se os padrões organizacionais (de produto e processo) foram consistentemente aplicados.

Já o objetivo da **validação** é assegurar que o software que está sendo desenvolvido é o **software correto**, ou seja, assegurar que os requisitos, e o software deles derivado, atendem ao uso proposto.



# Técnicas de Levantamento de Requisitos

# Técnicas de Levantamento de Requisitos

Hoje existem diversas técnicas que podem ser utilizadas no levantamento de requisitos, as quais podem possuir diferentes objetos de investigação ou podem ter foco em tipos diferentes de requisitos.

# Técnicas de Levantamento de Requisitos

**Entrevistas:** Esta técnica é amplamente utilizada, e consiste em conversas direcionadas com um objetivo específico e no formato “pergunta-resposta”. Sua finalidade é descobrir problemas a serem tratados, levantar procedimentos importantes e entender a opinião e as expectativas dos entrevistados sobre o sistema.

**Questionários:** o uso de questionários possibilita ao analista obter informações como postura, crenças, comportamentos e características de várias pessoas que serão afetadas diretamente pelo sistema.

## Técnicas de Levantamento de Requisitos

**Observação:** esta técnica consiste em realizar uma observação do comportamento e o ambiente dos indivíduos de vários níveis organizacionais. Utilizando-se essa técnica, é possível capturar o que realmente é feito e qual tipo de suporte computacional é realmente necessário. Ajuda a confirmar ou refutar informações obtidas com outras técnicas e ajuda a identificar tarefas que podem ser automatizadas e que não foram identificadas pelos interessados.

**Análise de documentos:** pela análise de documentos existentes na organização, analistas capturam informações e detalhes difíceis de conseguir por entrevista e observação. Documentos revelam um histórico da organização e sua direção.

## Técnicas de Levantamento de Requisitos

**Cenários:** com o uso desta técnica, um cenário de interação entre o usuário final e o sistema é montado e o usuário simula sua interação com o sistema nesse cenário, explicando ao analista o que ele está fazendo e de que informações ele precisa para realizar a tarefa descrita no cenário. O uso de cenários ajuda a entender requisitos, a expor o leque de possíveis interações e a revelar facilidades requeridas.

**Prototipagem:** um protótipo é uma versão preliminar do sistema, muitas vezes não operacional e descartável, que é apresentada ao usuário para capturar informações específicas sobre seus requisitos de informação, observar reações iniciais e obter sugestões, inovações e informações para estabelecer prioridades e redirecionar planos.



**Dúvidas ?**