

ANÁLISE DE SISTEMAS ASIS

Prof. Wallace Rodrigues



Descrição de Requisitos de Software

O que são Requisitos e por que são Essenciais

Definição Fundamental de Requisitos

- Um requisito de software é uma declaração das condições e capacidades que um sistema deve possuir. Simplificando, ele responde a três perguntas cruciais:
 - O que o sistema deve fazer? (Funções)
 - Como o sistema deve ser? (Qualidade, performance)
 - O que o limita? (Restrições técnicas ou regulatórias)
- O requisito é, essencialmente, a ponte de comunicação entre o desejo do cliente (ou a necessidade de negócio) e a implementação técnica pelo desenvolvedor.

O que são Requisitos e por que são Essenciais

A Importância Crítica: O Requisito como Contrato

- A principal função da descrição de requisitos é estabelecer um contrato formal entre todas as partes interessadas (stakeholders): O que o sistema deve fazer? (Funções)
 - Para o Cliente: É a confirmação de que o sistema construído atenderá às suas necessidades de negócio. É o que ele usará para aceitar ou rejeitar o produto final.
 - Para o Desenvolvedor: É o guia e a especificação técnica do que precisa ser codificado. Ele define o escopo do projeto.
 - Para o Testador: É a base para criar todos os planos e casos de teste. Um sistema só estará "pronto" quando todos os requisitos forem verificados e aprovados.)

O que são Requisitos e por que são Essenciais

A Importância Crítica: O Requisito como Contrato

- O custo de corrigir um erro ou um requisito mal interpretado aumenta exponencialmente quanto mais tarde no ciclo de vida do software ele é descoberto.
 - Corrigir um erro na fase de requisitos (no papel) é barato.
 - Corrigir o mesmo erro na fase de codificação é muito mais caro, pois exige retrabalho.
 - Corrigir o erro após a entrega do sistema (em produção) é o mais caro e pode causar prejuízos financeiros, danos à reputação e perda de clientes.

Tipos de Requisitos

A Importância Crítica: O Requisito como Contrato

- Os requisitos são tradicionalmente divididos em duas categorias principais, que definem escopos de trabalho diferentes para a equipe de desenvolvimento:
 - ✓ Requisitos Funcionais (RFs)
 - ✓ Requisitos Não-Funcionais (RNFs)

Requisitos Funcionais (RFs)

- **O que são:** Os Requisitos Funcionais definem as funções e os serviços que o sistema deve fornecer. Eles descrevem as ações que o sistema precisa executar para satisfazer as necessidades do usuário. Eles respondem à pergunta: "O que o sistema faz?"
- **Exemplos:**
 - ✓ "O sistema deve permitir que o usuário cadastre um novo produto."
 - ✓ "O sistema deve gerar um relatório de vendas mensais."
 - ✓ "O sistema deve permitir que o usuário se cadastre com nome, CPF e e-mail."
 - ✓ "O sistema deve enviar um e-mail de confirmação ao cliente após a compra."
 - ✓ "O sistema deve gerar um relatório de estoque em formato PDF."

Requisitos Não-Funcionais (RFNs)

- **O que são:** Os Requisitos Não-Funcionais definem as qualidades e restrições sob as quais o sistema deve operar. Eles especificam como o sistema deve fazer o seu trabalho. Eles respondem à pergunta: "Como o sistema deve ser?"
- **Categorias Chave:**
 - ✓ Desempenho: (Ex: "A página deve carregar em menos de 2 segundos.")
 - ✓ Segurança: (Ex: "O sistema deve usar criptografia SSL para todas as transações.")
 - ✓ Usabilidade: (Ex: "A interface deve ser acessível para daltônicos.")
 - ✓ Confiabilidade: (Ex: "O sistema deve ter 99.9% de disponibilidade.")

Características de um Bom Requisito

- Não basta ter requisitos; eles precisam ser de alta qualidade. Um bom requisito deve ser:
 - **Claro (Unambiguous)**: Deve ter apenas uma interpretação possível. (Evitar termos vagos como "rápido" ou "fácil de usar").
 - **Completo (Complete)**: Deve descrever a funcionalidade inteira, sem deixar pontas soltas.
 - **Consistente (Consistent)**: Não pode contradizer outro requisito.
 - **Verificável (Verifiable)**: Deve ser possível testar e provar que o requisito foi implementado (o desenvolvedor deve saber o que precisa fazer e o testador deve saber como testar).
 - **Rastreável (Traceable)**: Deve ser possível rastreá-lo da sua origem (o cliente) até a implementação no código e nos testes.

Como Descrever e Documentar (O Caso de Uso)

- A melhor forma de descrever requisitos funcionais é através de Casos de Uso.
- Caso de Uso: Uma descrição detalhada e estruturada de como um ator usa o sistema para atingir um objetivo.
 - Componentes Chave: Nome do caso de uso , Ator principal, pré-condições e pós-condições, fluxos principal e alternativo.

Como Descrever e Documentar (O Caso de Uso)

- A melhor forma de descrever requisitos funcionais é através de Casos de Uso.
- Caso de Uso: Uma descrição detalhada e estruturada de como um ator usa o sistema para atingir um objetivo.
 - Componentes Chave: Nome do caso de uso , Ator principal, pré-condições e pós-condições, fluxos principal e alternativo.

MODELOS

- Existe um modelo pronto e amplamente utilizado para documentar requisitos funcionais, conhecido como Template de Caso de Uso.
- Ele é uma estrutura padronizada que garante que todos os aspectos da funcionalidade sejam cobertos, resultando em um requisito claro, completo e verificável.

MODELO RF

Campo	Descrição	Exemplo
Nome do Caso de Uso	Nome conciso do objetivo.	Fazer Login no Sistema
ID do Requisito (RF)	Um identificador único para rastreabilidade.	RF-003
Ator Principal	Quem inicia o processo (usuário ou outro sistema).	Usuário Registrado
Objetivo	O que o Ator quer alcançar.	Obter acesso à área restrita do sistema.
Prioridade	Qual a importância deste requisito (Alta, Média, Baixa).	Alta
Pré-Condições	O que deve ser verdade antes que o caso de uso possa começar.	O usuário deve estar na página de login. O usuário deve ter uma conta cadastrada.
Pós-Condições	O estado do sistema após o fluxo principal ser concluído com sucesso.	O sistema exibe o painel principal do usuário.
Fluxo Principal (Caminho de Sucesso)	O passo a passo sequencial para o sucesso (o caminho mais comum).	1. O ator insere usuário e senha; 2. O sistema envia credenciais ao serviço de autenticação; 3. O sistema valida as credenciais; 4. O sistema concede acesso.
Fluxos Alternativos	Variações do fluxo principal que ainda levam ao sucesso.	2a. Se o usuário esquecer a senha, ele clica em "Recuperar Senha".
Fluxos de Exceção	O que acontece quando algo dá errado e impede o objetivo.	3a. Se as credenciais forem inválidas: O sistema exibe a mensagem "Usuário ou senha inválidos." O fluxo retorna ao passo 1.

MODELOS RNF

Campos		Exemplos	
<i>Campo</i>		Exemplo de RNF - Desempenho	
<i>ID do Requisito (RNF)</i>		RNF-D01	
<i>Categoria</i>		Desempenho	
<i>Requisito</i>		O tempo de resposta do servidor para a página inicial não deve exceder 1.5 segundos.	
<i>Métrica de Teste</i>		O teste deve medir o tempo de resposta em 1000 acessos simultâneos.	
	-----	-----	
<i>Campo</i>		Exemplo de RNF - Segurança	
<i>ID do Requisito (RNF)</i>		RNF-S05	
<i>Categoria</i>		Segurança	
<i>Requisito</i>		Todas as senhas de usuários devem ser armazenadas utilizando o algoritmo de hash SHA-256.	
<i>Métrica de Teste</i>		Verificar se a coluna do banco de dados usa o padrão de criptografia SHA-256.	