Capa

Sumario

A documentação de requisitos é basicamente **o que o sistema deve fazer e o que não pode esquecer**. Ela é o mapa inicial do projeto.

Referências:

<http://uptech.team/blog/srs-document>

<https://github.com/jam01/SRS-Template>

## **1. Introdução**

* **Objetivo do Documento:** uma explicação rápida sobre o que esse documento cobre.
* **Escopo do Projeto:** o que o sistema vai abranger (e também o que não vai).

Exemplo:  
 "Este documento descreve os requisitos do sistema de controle de estoque para pequenas lojas de varejo. Não contempla integrações com marketplaces."

## **2. Visão Geral do Produto**

* **Descrição resumida:** o que é o sistema.
* **Usuários principais:** quem vai usar (tipos de usuários).
* **Restrições iniciais:** limites técnicos, de plataforma, de tempo etc.

Exemplo:  
 "O sistema será uma aplicação web responsiva, focada para dispositivos móveis e desktop. Deverá operar sem necessidade de instalação local."

## **3. Requisitos Funcionais (RF)**

* **O que o sistema deve fazer**, dividido em itens numerados.

Exemplo:

* RF001: O sistema deve permitir o cadastro de produtos.
* RF002: O sistema deve gerar relatórios de vendas semanais.
* RF003: O sistema deve permitir login de usuários com autenticação de dois fatores.

Esses requisitos são **ações concretas** que o software realizará.

## **4. Requisitos Não Funcionais (RNF)**

* **Como o sistema deve se comportar**: Qualidade, segurança, usabilidade, performance.

Exemplo:

* RNF001: O sistema deve ser capaz de atender 500 usuários simultâneos.
* RNF002: O sistema deve apresentar 99,9% de disponibilidade mensal.
* RNF003: As respostas do servidor devem ocorrer em até 2 segundos.

## **5. Regras de Negócio**

* **Normas e regras específicas** que precisam ser obedecidas no funcionamento do sistema.

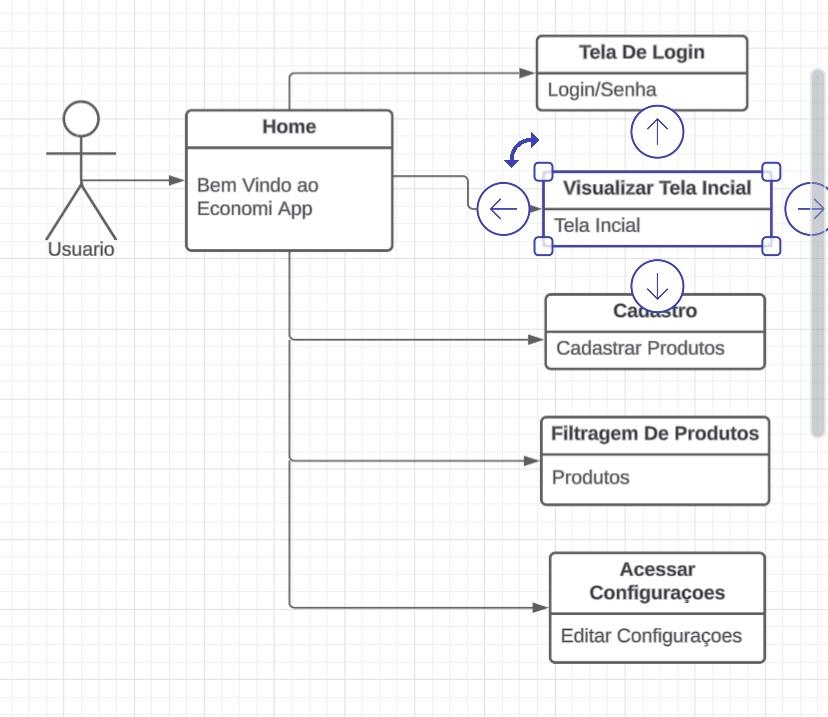
Exemplo:

* RN001: Um produto sem estoque não pode ser vendido.
* RN002: Apenas gerentes podem alterar os preços dos produtos.

## **6. Casos de Uso ou Histórias de Usuário**

Aqui você descreve **cenários** de como os usuários interagem com o sistema.

Formato básico (Casos de Uso):

* **Título:** Cadastro de Produto
* **Ator:** Usuário Administrador
* **Descrição:** O usuário acessa o menu, seleciona "Novo Produto" e preenche o formulário.  
  

## **7. Critérios de Aceitação**

* **Condições mínimas** para dizer que aquele requisito foi atendido.

Exemplo para Cadastro de Produto:

* O produto deve ter nome, preço e quantidade obrigatórios.
* Não pode haver produtos duplicados.
* **7.1 Cadastro de Produto**
* O produto deve conter obrigatoriamente: nome, categoria.
* O nome do produto deve ter no mínimo 3 caracteres.
* A imagem deve ser válida e carregável via URL ou upload.

### **7.2 Busca de Produto**

* A busca deve retornar produtos com nome semelhante ao termo pesquisado.
* A busca não pode demorar mais que 3 segundos com boa conexão.
* Deve funcionar com acentos e variações .

### **7.3 Comparação de Preços**

* Deve exibir pelo menos 2 ofertas diferentes para um mesmo produto (caso existam).
* Deve apresentar: nome da loja, preço,link para a loja.
* Ofertas com o mesmo preço devem ser ordenadas por melhor avaliação da loja.

### **7.4 Exibição do Histórico de Preço (opcional)**

* Caso não haja histórico, deve exibir uma mensagem clara: "Sem dados de histórico."

### **7.5Cadastro/Login de Usuário**

* O cadastro deve validar e-mail único.
* O login deve funcionar com e-mail e senha válidos.
* O app deve manter o login do usuário ativo.

### **7.6 Interface Mobile**

* A interface deve ser responsiva e adaptada para telas de 5” a 6.5”.
* Ícones e botões devem ser clicáveis com precisão em telas sensíveis ao toque.
* Texto deve ter legibilidade (mínimo 14px).

## **8. Restrições**

* Limitações que precisam ser consideradas (legais, técnicas, etc).

Exemplo:

* A aplicação deve seguir a LGPD (Lei Geral de Proteção de Dados).
* Deve funcionar em servidores Linux.
* **Seja claro e objetivo:** evite textos ambíguos ("o sistema deve ser rápido" não é bom. "responder em 2s" é melhor).
* **Numere tudo:** facilita rastrear mudanças.
* **Mantenha atualizado:** requisitos mudam durante o projeto.

### **8.1 Legais e Comerciais**

* **Limitações de Scraping**: Algumas lojas não permitem coleta de dados via scraping; o sistema deve respeitar os Termos de Uso de terceiros.
* **Uso de APIs de terceiros**: A integração com APIs deve respeitar as políticas de uso e limites de requisições.
* **Propriedade intelectual**: Nomes, imagens e descrições de produtos devem ser utilizados apenas para fins informativos e não podem ser revendidos.

### **8.2 Técnicas**

* **Dependência de Internet**: A busca e a comparação de preços dependem de conexão ativa com a internet.
* **Tempo de resposta**: As requisições devem ser otimizadas para não ultrapassar 3 segundos, exceto em condições de rede ruim.
* **Atualização de dados**: Os preços e ofertas não são em tempo real; atualizações periódicas podem ocorrer a cada 1h ou conforme política de coleta.

### **8.3 Operacionais**

* **Privacidade**: Dados do usuário devem ser armazenados e processados de acordo com a LGPD (Lei Geral de Proteção de Dados).

### **8.4 Financeiras**

* **Sem transações diretas**: O app é apenas informativo e não realiza vendas ou intermedia pagamentos.
* **Preços sujeitos a mudança**: Os preços exibidos podem variar no site da loja; o app não garante 100% de precisão em tempo real.