

# StockPro — Documentação do Frontend e Banco de Dados

Versão 1.0 · Agosto/2025

Este documento descreve a arquitetura do frontend estático, pontos de integração com back-end e o modelo de dados com scripts SQL (PostgreSQL e MySQL).

## Sumário

- 1. Visão Geral
- 2. Estrutura do Repositório
- 3. Frontend (HTML/CSS/JS)
- 4. Páginas e Slots de Include
- 5. Acessibilidade e UX
- 6. Banco de Dados (modelo)
- 7. Scripts SQL e Seeds
- 8. Contratos de API (proposta)
- 9. Deploy e Ambiente
- 10. Contribuição e Roadmap

## 1. Visão Geral

Frontend estático com Bootstrap 5, Bootstrap Icons e jQuery via CDN. Dark mode com persistência em localStorage e sidebar sanfona/offcanvas no mobile.

## 2. Estrutura do Repositório

```
assets/  
css/app.css  
js/app.js  
img/logo.svg  
index.html  
produtos.html  
categorias.html  
fornecedores.html  
compras.html  
vendas.html  
clientes.html  
movimentacoes.html  
inventario.html  
relatorios.html  
usuarios.html  
configuracoes.html  
login.html  
recuperar.html  
404.html
```

## 3. Frontend (HTML/CSS/JS)

### 3.1 CDNs

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"  
rel="stylesheet">  
<link  
href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.css"  
rel="stylesheet">  
<link href="assets/css/app.css" rel="stylesheet">  
<script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.m  
in.js"></script>  
<script src="assets/js/app.js"></script>
```

### 3.2 app.css (destaques)

- Variáveis CSS para tema claro/escuro (cores, bg, texto).
- Topbar fixa com sombra; sidebar fixa (desktop) e offcanvas (<992px).
- Tabelas com thead sticky; cards com sombra/borda suave.

### 3.3 app.js (destaques)

- Persistência de tema (localStorage 'sp-theme').
- Toggle dark/light adicionando/removendo 'dark-theme' no .
- Inicialização de tooltips/toasts e helper showToast(msg, type).

## 4. Páginas e Slots de Include

Cada página possui um slot padrão e placeholders específicos para injeção do back-end.

```
<!-- INCLUDE: PAGE_CONTENT_START -->  
<!-- (back-end injeta conteúdo aqui) -->
```

```
<!-- INCLUDE: PAGE_CONTENT_END -->
```

- produtos.html →
- categorias.html →
- fornecedores.html →
- compras.html →
- vendas.html →
- clientes.html →
- movimentacoes.html →
- inventario.html →
- relatorios.html →
- usuarios.html →
- configuracoes.html →

## 5. Acessibilidade e UX

- Ícones com aria-label quando forem somente ícones.
- aria-expanded na sanfona, foco visível, contraste adequado.
- Feedback não só por cor (usar badges/ícones/texto).

## 6. Banco de Dados (Modelo)

- Categorias, Produtos, Fornecedores, Clientes, Usuários, Locais.
- Compras/Vendas e seus Itens.
- Movimentações (kardex) e Inventários.
- Views: v\_estoque\_atual, v\_estoque\_baixo, v\_entradas\_saidas\_por\_dia.

```
[produtos] 1---* [compras_itens] *---1 [compras] *---1 [fornecedores]
[produtos] 1---* [vendas_itens] *---1 [vendas] *---1 [clientes]
[produtos] 1---* [movimentacoes] *---1 [locais]
[inventarios] 1---* [inventarios_itens] *---1 [produtos]
```

## 7. Scripts SQL e Seeds

- PostgreSQL: stockpro\_schema.sql (ENUM types, views, índices opcionais TRGM).
- MySQL: stockpro\_mysql.sql (ENUM inline, FULLTEXT opcional em produtos.nome).
- Dados dummy coerentes com o frontend (SKU-0001, 0451, 0877 etc.).

## 8. Contratos de API (proposta)

- GET /api/produtos?busca=&categoria;=&status;=
- POST /api/produtos; PUT/PATCH /api/produtos/{id}; DELETE (inativar).
- CRUDs de categorias, fornecedores, clientes, usuários.
- GET /api/compras|vendas (com paginação) + POST para criar.

- GET /api/movimentacoes?periodo=&tipo;=&sku;=
- POST /api/inventarios + POST /api/inventarios/{id}/itens

## 9. Deploy e Ambiente

- Frontend em hosting estático (GitHub Pages/Netlify/S3/Nginx).
- Habilitar HTTPS e cache de assets.
- Backend expõe /api/\* com CORS para origem do frontend.
- Banco com os scripts fornecidos; Docker opcional p/ dev.

## 10. Contribuição e Roadmap

- Issues/PRs com descrição clara e prints de UI.
- Padrão: HTML semântico, classes Bootstrap, JS minimalista.
- Roadmap: autenticação real, RBAC, paginação server-side, exportações, gráficos (Chart.js), testes E2E.