

# Sistema de Gestão de Estoque e Caixa

## Objetivo Geral

Desenvolver um sistema web para **controle de estoque, registro de vendas (caixa) e gestão de usuários**, com base em uma arquitetura cliente-servidor moderna, aplicando os conceitos de **formulários reativos, validações, rotas protegidas, serviços REST, e papéis de usuário (roles)**.

## 1. Escopo do Sistema

### Perfis e Permissões

Módulo	Administrador	Operador
Login	✓	✓
Manter Usuários (CRUD)	✓	✗
Cadastro e Edição de Produtos	✓	✗
Movimentação de Estoque (Entradas/Ajustes)	✓	✗
Caixa (Vendas)	✗	✓
Relatórios (Consulta de Vendas)	✓	✓

## 2. Módulos Principais

### 2.1 Login (Simples)

- Tela de autenticação com **e-mail e senha**.
- A senha deve ser **comparada** no back-end.
- Após login, o sistema retorna o **nome e perfil do usuário**.
- A sessão é **mantida no front-end** (ex.: `localStorage`).
- Exibe o nome do usuário no topo e permite **sair** (logout).

### 2.2 Manter Usuários (Administrador)

- CRUD completo: listar, cadastrar, editar e excluir.
- Campos:
  - Nome completo
  - E-mail (único)
  - Senha (armazenada criptografada)
  - Perfil: `ADMIN` ou `OPERADOR`
  - Status: ativo/inativo

#### Regras:

- Apenas Administradores podem acessar este módulo.
- Senha deve seguir política mínima (8 caracteres, 1 letra maiúscula e 1 número).
- E-mails duplicados não são permitidos.

### 2.3 Gestão de Estoque (Administrador)

- Cadastro e manutenção de produtos.

- Movimentações de estoque (entrada e ajuste).
- Exibição do **saldo atual** de cada produto.

**Campos:**

- Código do produto
- Nome
- Categoria
- Quantidade em estoque
- Preço unitário

 **Operações:**

- Criar, editar, excluir produtos.
- Registrar **entradas** (reposição) e **ajustes** (correção).
- Histórico de movimentações com data, tipo e motivo.

---

## 2.4 Caixa / Vendas (Operador)

- Registro de vendas com múltiplos itens.
- Cada venda gera **baixa de estoque automática**.
- Cálculo automático de **subtotal, total e troco**.

**Campos por item:**

- Produto (selecionado da lista de estoque)
- Quantidade (não pode exceder o estoque)
- Subtotal automático (quantidade × preço)

**Campos gerais:**

- Valor total
- Valor recebido
- Troco (calculado)
- Data e hora da venda
- Usuário responsável

---

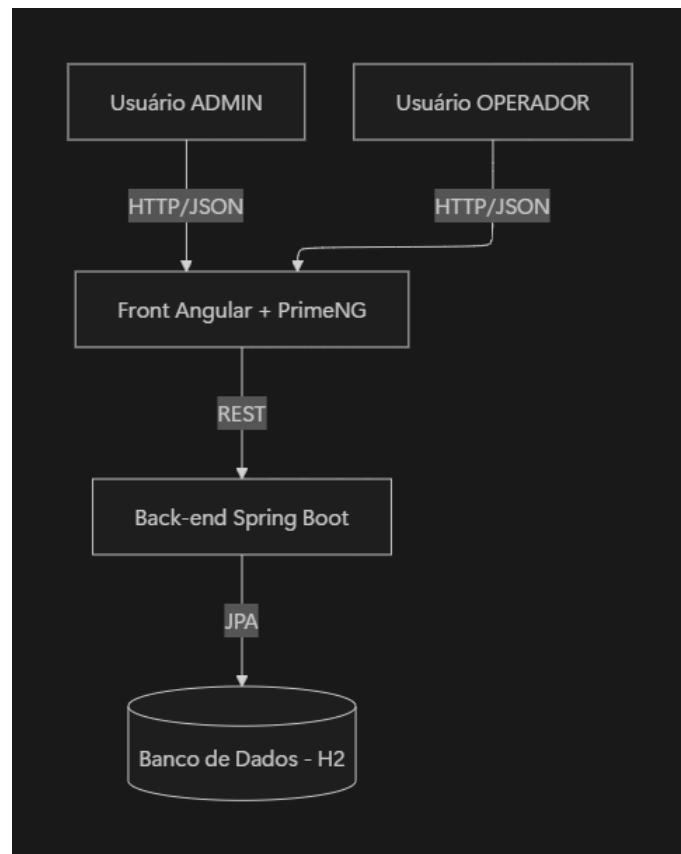
## 2.5 Relatórios (Administradores e Operadores)

- Listagem de todas as vendas registradas.
- Filtros por:
  - Período (data inicial/final)
  - Valor mínimo/máximo
  - Usuário
- Exibição de somatórios:
  - Total de vendas
  - Total de itens vendidos
- Acesso permitido a ambos os perfis.

---

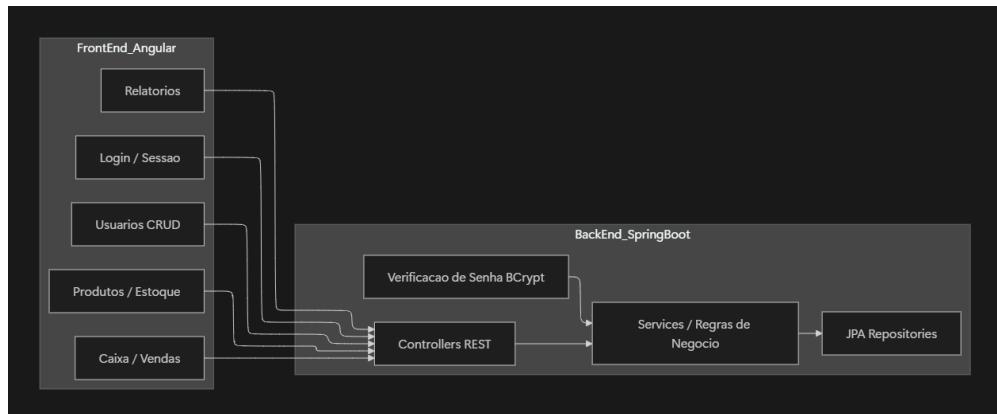
## 3. Arquitetura do Sistema

### 3.1 Visão Geral



---

### Componentes Lógicos

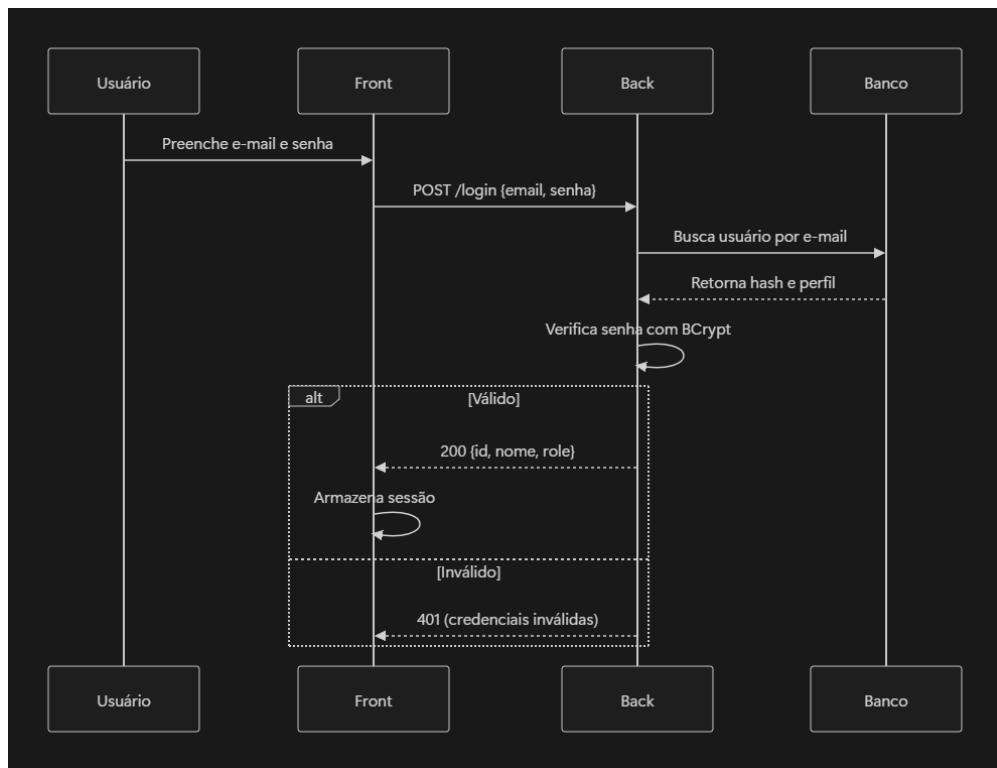


---

## 4. Modelo de Dados (ER Simplificado)

## 5. Autenticação Simplificada

### Fluxo



## 6. Validações Gerais

Campo	Regra
E-mail	Obrigatório, formato válido e único
Senha	mínimo 8 caracteres, 1 letra maiúscula, 1 número
Produto	Preço > 0, estoque $\geq$ 0
Quantidade	$\leq$ estoque atual
Venda	amountReceived $\geq$ total
Movimentação de Estoque	<b>ENTRADA</b> $\rightarrow$ +; <b>AJUSTE</b> $\rightarrow$ + ou -

## 7. Requisitos Não Funcionais

- Persistência em (H2 para desenvolvimento).
- Layout responsivo (PrimeNG).
- Tratamento de erros padronizado e mensagens amigáveis.
- Aplicação organizada em **camadas** (Controller → Service → Repository).
- Uso de **Reactive Forms** e **Guards** no Angular.

## 8. Planejamento (2 Semanas)

### Semana 1 — Administração

- Tela de Login
- CRUD de Usuários
- CRUD de Produtos e Movimentação de Estoque
- Guards e controle de acesso por perfil

## **Semana 2 — Caixa e Relatórios**

- Registro de Vendas (Operador)
  - Cálculo de total e troco
  - Atualização automática do estoque
  - Relatórios com filtros e somatórios
  - Revisão e testes funcionais
- 

## **10. Entregáveis**

- Código fonte (backend + frontend) em repositório público.
  - Script de banco ou migrations.
  - Documento de arquitetura (diagramas incluídos).
  - Vídeo curto ou apresentação mostrando os fluxos principais.
-