

# **Projeto Atacado Eletrônico**

Autor: Fábio Adriano Corrêa de Oliveira Júnior

Blumenau - SC

2025

## **Sumário**

- 1. Introdução
- 2. Objetivo
- 3. Escopo do Projeto
- 4. Tecnologias Utilizadas
  - 4.1. Back-End
  - 4.2. Front-End
- 5. Requisitos do Sistema
  - 5.1. Requisitos Funcionais
  - 5.2. Requisitos Não Funcionais
  - 5.3 Métodos do Sistema
- 6. Casos de Uso
- 7. Arquitetura de Classes
- 8. Modelagem do Banco de Dados
  - 8.1 Ferramenta Utilizada
  - 8.2 Modelo
- 9. Views
  - 9.1 Tela Cadastro
  - 9.2 Tela Login
- 10. Tratamento de Erros

## 11. Melhorias Futuras

## 12. Referências

### **1. Introdução**

Este documento apresenta o projeto de um sistema web voltado para o atacado eletrônico. O sistema terá duas áreas principais: uma para clientes e outra para empresas (lojistas). O objetivo principal é facilitar o comércio em larga escala de produtos eletrônicos, com funcionalidades robustas e uma arquitetura moderna.

### **2. Objetivo**

O objetivo deste projeto é desenvolver um sistema de atacado eletrônico que permita:

- Cadastro e autenticação de usuários (clientes e empresas)
- Listagem e gerenciamento de produtos pelas empresas
- Navegação, compra e pagamento por parte dos clientes
- Emissão de notas fiscais e acompanhamento de pedidos

### **3. Escopo do Projeto**

O sistema contempla:

- Interface Web responsiva para clientes e empresas
- Área de administração para gerenciamento do conteúdo
- Integração com banco de dados relacional
- Geração automática de notas fiscais em formato PDF
- Controle de estoque
- Funcionalidade de busca e filtros de produtos

### **4. Tecnologias Utilizadas**

#### **4.1 Back-End**

- Java

- Spring Boot (framework principal)
- MySQL/Workbench (Sistema Gerenciador de Banco de Dados Relacional)
  - Version 8.0.42
  - Port 3306
  - X Protocol Port 33060
- JPA/Hibernate (persistência de dados)
- Express (para interações com serviços Node.js específicos)
- HeidiSQL (interface para o SGBD)

## **4.2 Front-End**

- React.js (framework principal de SPA)
- HTML5 / CSS3
- JavaScript (ES6+)
- Node.js (para build e servidor local)

## **5. Requisitos do Sistema**

### **5.1. Requisitos Funcionais**

- RF01: Cadastro de usuário (cliente e empresa)
- RF02: Login e autenticação
- RF03: Cadastro de produtos pelas empresas
- RF04: Visualização de produtos pelos clientes
- RF05: Adição de produtos ao carrinho
- RF06: Finalização de compra
- RF07: Emissão de nota fiscal
- RF08: Histórico de pedidos

## 5.2. Requisitos Não Funcionais

- RNF01: O sistema deve ser responsivo
- RNF02: O sistema deve armazenar dados com segurança
- RNF03: O tempo de resposta das requisições não deve exceder 2 segundos
- RNF04: O sistema deve suportar no mínimo 100 usuários simultâneos

## 6. Casos de Uso

COLOCAR DIAGRAMA

## 7. Arquitetura de Classes

### 7.1 Back-End(Spring Boot - Java)

COLOCAR CLASSE | DESCRIÇÃO | RESPONSÁVEL

name class	explicação	ex:entidade JPA, CONTROLLER SERVICE
------------	------------	-------------------------------------

## 7.2 Front-End(React.js)

src/

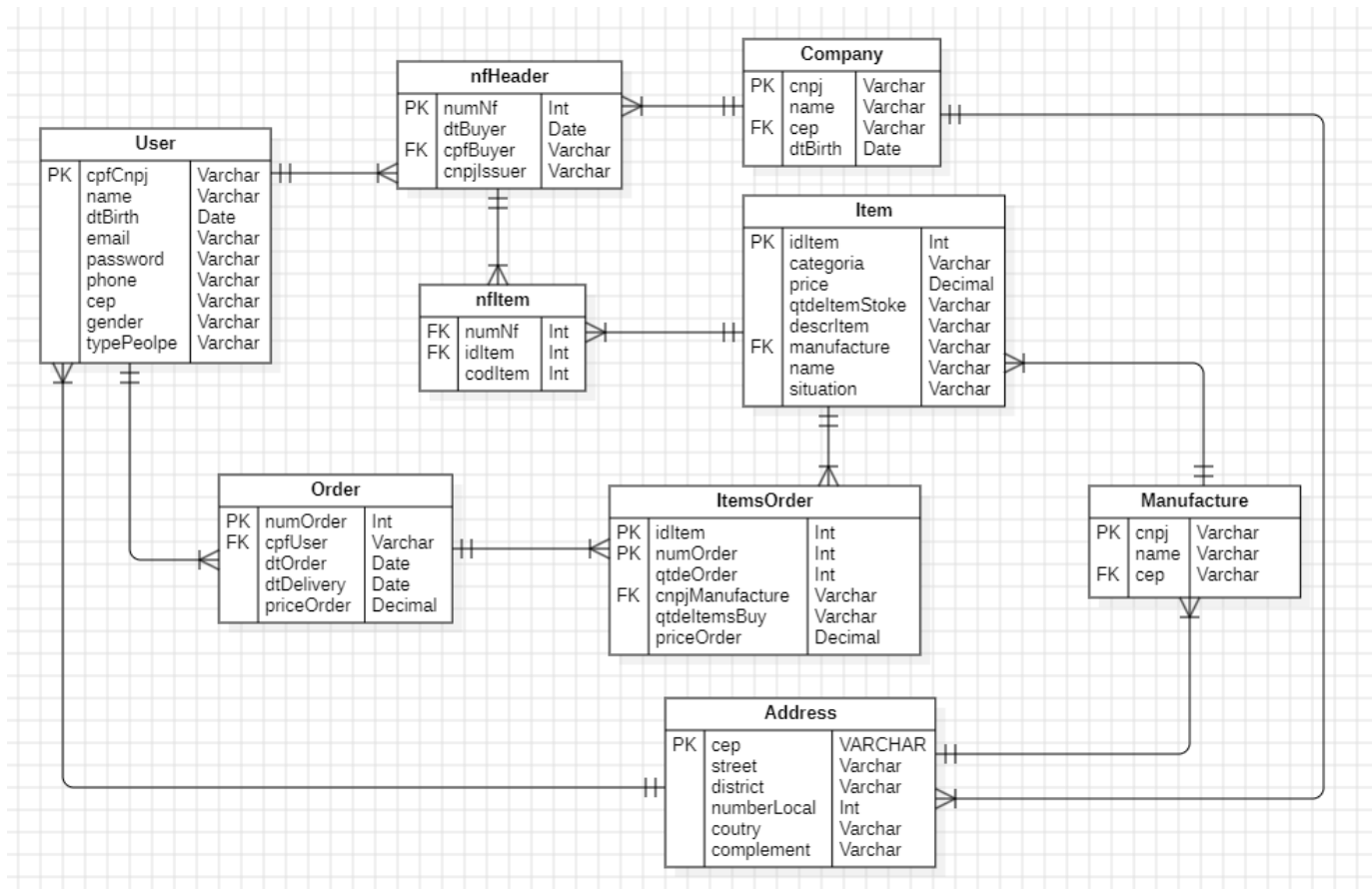
- |— App.jsx
- |— index.js
- |— routes/ ← rotas e navegação
  - | — PrivateRoute.jsx ← componente para proteger rotas logadas
- |— pages/ ← telas completas
  - | |— Login.jsx
  - | |— Cadastro.jsx
  - | |— ListaProdutos.jsx
  - | |— Carrinho.jsx
  - | |— PedidoFinalizado.jsx
- |— components/ ← componentes reutilizáveis (inputs, cards, navbar, etc.)
  - | |— ProdutoCard.jsx
  - | |— Header.jsx
  - | |— InputField.jsx
- |— services/ ← comunicação com o back (API)
  - | |— api.js ← configuração base do axios
  - | |— authService.js ← login, logout, verificar token
  - | |— produtoService.js ← busca, cadastro, edição
- |— context/ ← auth global ou carrinho (React Context API)
  - | |— AuthContext.jsx
- |— assets/ ← imagens, logos, etc.

## 8. Modelagem do Banco de Dados

### 8.1 Ferramenta Utilizada

- My SQL
- My SQL Workbench
- Draw.io ([BD.drawio - draw.io](https://bd.drawio.com/))

## 8.2 Modelo



## 9. Views

### 9.1 Tela Cadastro

Seu HARDWARE  
Sua História

**COMEÇE AQUI SUA HISTÓRIA**

COMPRE

Explore

Se DIVIRTA

FAÇA VOCÊ MESMO !!!

**CADASTRO**

CPF/CNPJ

CPF/CNPJ

CPF/CNPJ

CPF/CNPJ

CPF/CNPJ

CPF/CNPJ

CPF/CNPJ

CPF/CNPJ

SALVAR

## 9.2 Tela Login

## 10. Tratamento de Erros

## 11. Melhoria Futuras

- Integração com plataformas de pagamento (ex: Stripe, PagSeguro)
- Implementação de chat em tempo real entre cliente e loja
- Relatórios estatísticos com dashboards
- Alterar Senha usando e-mail

## 12. Referencias

EXTRAS MEU:

Ver incrementador unico para id \_order do banco mysql(next-value)