

# FACULDADE ESTÁCIO - CAMPUS TERESINA

---

Alunos:

Thiago Samuel Rodrigues Rocha – 202309466678.

Erick John Mascarenha dos Santos – 202308623931.

Professor: Douglas Lopes.

Curso: Ciências da Computação.

## Documentação Técnica do Projeto Sistema de Cadastro e Vendas em Java

---

### 1. Introdução

Este projeto foi desenvolvido com o objetivo de aplicar de forma prática e aprofundar os conhecimentos adquiridos em Programação Orientada a Objetos (POO), por meio da construção de um sistema funcional e organizado (POO) por meio de um sistema completo de cadastro e controle de vendas, utilizando a linguagem Java. A aplicação simula o funcionamento básico de uma loja, permitindo o gerenciamento de clientes, produtos e vendas. Todas as interações com o usuário são realizadas por meio de uma interface gráfica construída com os pacotes Swing e AWT, para diálogos simples e outros componentes gráficos para interação avançada., o que garante praticidade no uso.

### 2. Objetivos

O principal objetivo é aplicar, de forma prática, os conceitos essenciais de POO, como encapsulamento, herança, polimorfismo e sobrescrita de métodos. Além disso, o projeto busca também alcançar metas complementares como:

- Desenvolver uma estrutura modular e coesa utilizando classes bem definidas;
- Implementar validações e controles nos cadastros de dados;
- Simular uma operação de venda com seleção de múltiplos produtos e cálculo do valor total;
- Apresentar ao usuário um sistema funcional, simples e intuitivo.

### 3. Funcionalidades Implementadas

- Cadastro de clientes com validação de CPF (formato numérico com 11 dígitos);
- Cadastro de produtos comuns e livros, com armazenamento de nome, preço e autor (no caso de livros);
- Realização de vendas, permitindo selecionar um cliente e adicionar vários produtos em uma única compra;
- Cálculo automático do valor total de cada venda realizada;
- Listagem completa e organizada de todos os clientes, produtos e vendas cadastradas no sistema;
- Interface baseada em componentes gráficos da biblioteca Swing, incluindo caixas de diálogo (JOptionPane), botões (JButton), rótulos (JLabel) e campos de texto (JTextField), o que torna a interação com o sistema mais intuitiva e moderna.

### 4. Estrutura do Projeto e das Classes

O sistema é composto por seis classes principais, descritas a seguir:

- Produto:

Classe que representa um produto genérico com atributos id (inteiro), nome (String) e preço (double). Inclui métodos construtores, getters e setters, além do método toString() que exibe as informações de forma organizada.

- Livro:

Classe que herda de Produto e acrescenta o atributo autor (String). Sobrescreve o método toString() para exibir também o nome do autor junto aos dados do produto.

- **Cliente:**

Classe que representa os clientes cadastrados no sistema, contendo os atributos nome e CPF. A validação do CPF é feita por meio de expressão regular, permitindo apenas números e obrigando 11 dígitos.

- **Venda:**

Classe que armazena os dados de uma venda, incluindo o cliente comprador, os produtos adquiridos e a quantidade. Possui métodos que calculam o total da venda e listam os produtos envolvidos.

- **Loja:**

Classe responsável por armazenar e gerenciar os dados do sistema. Contém vetores de até 100 posições para clientes, produtos e vendas. Implementa os métodos para adicionar registros e realizar listagens.

- **Main:**

Classe principal que contém o método `main()`, implementando menus interativos e funcionalidades visuais usando componentes da biblioteca Swing e AWT, como `JFrame`, `JButton`, `JLabel` e `JTextField`.

## **5. Aplicação dos Conceitos de POO**

Durante o desenvolvimento, os seguintes conceitos da Programação Orientada a Objetos foram aplicados:

- **Encapsulamento:**

Todos os atributos de classe foram encapsulados com o modificador de acesso privado, com acesso controlado por métodos públicos (`get/set`), garantindo maior controle e segurança no uso dos dados.

- **Herança:**

A classe Livro estende a classe Produto, aproveitando seus atributos e métodos, e adicionando novos comportamentos específicos (autor).

- Polimorfismo:

O vetor de produtos armazena tanto objetos do tipo Produto quanto do tipo Livro, permitindo o uso uniforme desses elementos.

- Sobrescrita de métodos:

O método toString() foi sobrescrito na classe Livro para adaptar a exibição dos dados com as novas informações.

- Construtores personalizados:

Todas as classes implementam construtores com parâmetros, facilitando a inicialização dos objetos com os dados necessários.

## 6. Interface com o Usuário

A interface do sistema foi desenvolvida utilizando a classe JOptionPane do pacote javax.swing. Essa abordagem permite criar caixas de diálogo para entrada e exibição de dados, dispensando o uso de console ou interfaces complexas. O menu principal oferece ao usuário as seguintes opções: cadastrar cliente, cadastrar produto/livro, realizar venda, listar clientes, listar produtos e listar vendas.

## 7. Repositório do Código Fonte

O código-fonte completo do projeto encontra-se disponível no GitHub no seguinte link:

[ [https://github.com/ThiagoSamuel98/Sistema\\_de\\_Cadastro\\_e\\_Vendas\\_Java](https://github.com/ThiagoSamuel98/Sistema_de_Cadastro_e_Vendas_Java) ]

Recomenda-se clonar o repositório ou baixar os arquivos diretamente para facilitar a importação e execução do projeto na IDE Eclipse.

## **8. Melhorias Futuras e Considerações Finais**

Durante o desenvolvimento foi identificado que a aplicação pode ser expandida para incluir recursos como:

- Salvamento e carregamento de dados usando arquivos ou banco de dados;
- Melhorias visuais na interface gráfica com organização em painéis e menus;
- Tratamento de exceções mais robusto nas entradas do usuário;
- Geração de relatórios em PDF ou exportação de dados em CSV.

Essas melhorias representam um caminho natural para a evolução do sistema de um protótipo acadêmico para uma aplicação mais próxima de ambientes reais de negócios.