

# CENTRO UNIVERSITÁRIO FEI

Autor: Luciano Ventura Monegatto

## FASTDELIVERYINFORMATIC

São Paulo 2025

### **Relatório projeto FASTDELIVERYINFORMATIC**

## **1. Introdução**

O presente relatório tem como objetivo apresentar o desenvolvimento do sistema FastDeliveryInformatic, um programa em linguagem C destinado à simulação de um sistema de gestão de pedidos, produtos e usuários voltado para uma empresa fictícia de tecnologia. O projeto foi desenvolvido com foco em modularização, organização de código e aplicação prática de conceitos fundamentais de programação estruturada.

## **2. Desenvolvimento**

O sistema foi dividido em diversos módulos, cada um responsável por uma funcionalidade específica, como o gerenciamento de produtos, pedidos e usuários. A modularização foi implementada através de arquivos separados (.c e .h), permitindo melhor organização e manutenção do código.

O menu principal oferece opções para navegação entre as seções do sistema: usuários, produtos e pedidos. Dentro de cada módulo, o usuário pode realizar operações de cadastro, edição, exclusão e listagem, além de associar produtos aos pedidos realizados.

## **3. Estrutura do Sistema**

O projeto é composto pelos seguintes arquivos:

- **main.c**: contém o menu principal e o controle geral do sistema.
- **usuarios.c / usuarios.h**: responsável pelo cadastro, login e gerenciamento de usuários.
- **produtos.c / produtos.h**: responsável por armazenar, listar e buscar produtos.
- **pedidos.c / pedidos.h**: controla a criação, edição, exclusão e listagem de pedidos.

### **3.1. Funcionamento das Funções**

#### **3.1.1 cadastrarUsuario():**

A função cadastrarUsuario é responsável por permitir que novos usuários sejam registrados.

Durante sua execução, o programa solicita que o usuário informe um nome de login e uma senha. Esses dados são então armazenados de forma persistente no arquivo usuarios.txt, garantindo que as informações permaneçam disponíveis mesmo após o encerramento do programa.

O objetivo dessa função é criar perfis individuais de acesso, assegurando que apenas usuários cadastrados possam realizar pedidos e acessar funcionalidades internas do sistema.

#### **3.1.2 loginUsuario():**

A função loginUsuario() é responsável pelo processo de autenticação de usuários. Ela lê os dados do arquivo usuarios.txt e compara as credenciais fornecidas pelo usuário (nome e senha) com os registros armazenados.

Caso haja correspondência, o acesso é liberado e o usuário pode navegar pelo menu principal do sistema.

Essa função é essencial para garantir a segurança e o controle de acesso, permitindo que cada cliente tenha um ambiente próprio dentro da aplicação.

#### **3.1.3 buscarProduto():**

A função buscarProduto() permite que o usuário procure por um produto específico no catálogo da loja virtual.

A busca é realizada com base no nome ou parte do nome do produto, verificando as informações presentes no arquivo produtos.txt.

Ao localizar o item desejado, o sistema exibe os detalhes do produto, como o nome, o tipo e o preço.

Essa funcionalidade proporciona agilidade na navegação, permitindo que o usuário encontre facilmente os produtos de seu interesse.

#### **3.1.4 listarProdutos:**

A função listarProdutos() tem como objetivo exibir todos os produtos disponíveis no sistema.

Os dados são lidos diretamente do arquivo produtos.txt e apresentados de forma organizada, contendo informações como ID, nome, tipo e preço de cada item.

Essa função é fundamental para que o usuário tenha uma visão geral do catálogo

antes de realizar um pedido, simulando a experiência de navegação em um site de comércio eletrônico.

### 3.1.5 Gerenciar pedidos

O módulo Gerenciar Pedidos abrange as funções de criar, editar, excluir, adicionar e remover produtos de um pedido.

- **criarPedido()**: cria um novo pedido associado a um usuário logado.
- **editarPedido()**: permite alterar os dados de um pedido existente.
- **excluirPedido()**: remove um pedido do sistema.
- **adicionarProdutoPedido() e removerProdutoPedido()**: permitem adicionar ou retirar produtos de um pedido.
- **listarPedidos()**: listar os pedidos que o usuário já fez.

Essas operações permitem ao usuário personalizar suas compras conforme sua necessidade.

Cada pedido criado é associado ao ID do usuário que o realizou, garantindo um histórico individual.

As informações são armazenadas no arquivo pedidos.txt, de modo que todas as alterações são salvas permanentemente.

Esse módulo representa o núcleo do sistema, simulando de forma simplificada o processo de compra em uma loja online.

### 3.1.6 Avaliar pedido

A função **Avaliar Pedido** possibilita ao usuário atribuir uma nota de 0 a 5 estrelas a um pedido já finalizado.

Essa avaliação é registrada no arquivo pedidos.txt, associada ao respectivo pedido e ao usuário que a realizou.

O objetivo é simular o sistema de feedback encontrado em plataformas de e-commerce, permitindo que o cliente expresse sua satisfação com a compra.

Além de complementar o realismo da aplicação, essa função demonstra a integração entre diferentes módulos do programa.

## 4. Armazenamento de Dados

Os dados de usuários, produtos e pedidos são armazenados em arquivos de texto (.txt), simulando um banco de dados simples. Essa abordagem permite que as informações persistam mesmo após o encerramento do programa, possibilitando que as informações continuem permanecendo no sistema após seu fechamento.