



Escola e Faculdade SENAI "Nadir Dias de Figueiredo"  
Isabella Gonçalves Lopreti

## **Documento de Requisitos do Sistema**

**Documentação Inicial**

**Cantina 10.0**

---

Osasco, São Paulo

2025



Escola e Faculdade SENAI "Nadir Dias de Figueiredo"  
Isabella Gonçalves Lopreti

# 1. Introdução

---

Este documento descreve os requisitos funcionais e não funcionais do sistema Cantina 10.0, desenvolvido como projeto final do semestre para o curso técnico Desenvolvimento de Sistemas. O objetivo é criar um sistema de vendas para cantina que atenda às necessidades de operação de pedidos, controle de estoque, preparação de alimentos, pagamentos e gerenciamento geral.

# 2. Requisitos Funcionais

---

**2.1 [RF001]** A tela de vendas deve permitir selecionar itens do cardápio, quantidade e forma de pagamento.

**2.2 [RF002]** O sistema deve calcular e exibir o troco quando a forma de pagamento for dinheiro.

**2.3 [RF003]** Os itens vendidos devem ser automaticamente deduzidos do estoque.

**2.4 [RF004]** A cozinha deve visualizar apenas itens que exigem preparo.

**2.5 [RF005]** A cozinha deve poder liberar ou cancelar pedidos.

**2.6 [RF006]** A tela de chamada deve mostrar o primeiro nome do cliente após liberação pelo balcão.

**2.7 [RF007]** O sistema deve permitir adicionar e retirar itens do estoque, sem permitir estoque negativo.

**2.8 [RF008]** O sistema deve possuir uma tela de login com três perfis: Cantineiro (admin), Balcão e Cozinha.

**2.9 [RF009]** Os acessos às telas devem ser restritos conforme o perfil de usuário.

**2.10 [RF010]** O sistema deve validar entradas do usuário para evitar erros e inconsistências.

**2.11 [RF011]** O nome do cliente deve ser exibido no telão apenas quando houver itens de chapa no pedido.

**2.12 [RF012]** O telão deve mostrar pedidos em preparação e pedidos prontos separadamente.

**2.13 [RF013]** O telão deve mostrar os pedidos por ordem de chegada (mais antigos primeiro).

**2.14 [RF014]** O pedido só deve aparecer para retirada quando todos os itens estiverem prontos.

**2.15 [RF015]** O sistema deve possuir um botão para entrega do pedido.

**2.16 [RF016]** O pedido só pode ser registrado após confirmação do pagamento.

**2.17 [RF017]** O sistema deve manter os seguintes status de pedido: Pago, Preparando, Pronto, Entregue, Extornado.

**2.18 [RF018]** Os pedidos devem ser organizados por ordem de prioridade (tempo de entrada).

**2.19 [RF019]** O sistema deve permitir marcar se o pedido é para viagem ou não.

**2.20 [RF020]** A data e hora devem ser registradas em todos os pedidos.

**2.21 [RF021]** O cantineiro pode cancelar apenas pedidos pagos.

**2.22 [RF022]** O caixa pode cancelar pedidos em qualquer fase.

**2.23 [RF023]** Sempre registrar os valores unitários e quantidade.

**2.24 [RF024]** O sistema deve permitir cadastrar novos produtos com nome e preço(perfil adm).

**2.25 [RF025]** Cada produto deve ter um código numérico único.

**2.26 [RF026]** O controle de estoque deve registrar apenas a quantidade.

**2.27 [RF027]** O sistema deve conter um menu para navegação entre telas.

**2.28 [RF028]** O sistema deve manter histórico de todos os pedidos.

## 3. Requisitos Não Funcionais

---

**3.1 [RNF001]** O sistema deve seguir a paleta de cores, fontes e logotipo do manual da marca Bolt.

**3.2 [RNF002]** Todas as telas devem exibir o logotipo da Bolt na parte superior.

**3.3 [RNF003]** Os botões devem manter padronização visual, sendo o botão de cancelar discreto.

**3.4 [RNF004]** O sistema deve ser desenvolvido em C# com Windows Forms.

**3.5 [RNF005]** A interface deve ser amigável e com mensagens claras para o usuário.

**3.6 [RNF006]** O sistema deve exibir mensagens de erro/aviso em casos de falha ou entrada inválida.

**3.7 [RNF007]** O sistema deve evitar ações que levem a inconsistências, como estoque negativo.