**Manual de Utilização – Pizzaria App**

**1. Introdução**

O **Pizzaria App** é uma aplicação de linha de comando (CLI) desenvolvida em **TypeScript/Node.js** para gerenciar pedidos de uma pizzaria. Ele permite registrar pedidos, listar pedidos ativos, finalizar pedidos, gerar comprovantes e emitir relatórios de vendas.

Todos os dados são armazenados localmente em **arquivos CSV e JSON**, garantindo histórico completo de pedidos e controle financeiro.

**2. Requisitos**

* Node.js instalado (versão >= 14)
* Permissão de leitura e escrita na pasta da aplicação
* Terminal ou console para execução

**3. Estrutura de arquivos**

* **CSV**:
  + entradas.csv – Registro de todos os pedidos
  + ativos.csv – Pedidos ainda não finalizados
  + historico.csv – Histórico de pedidos finalizados
* **JSON**:
  + config.json – Configuração de preços
  + last\_id.json – Último ID de pedido gerado
  + sabores.json – Lista de sabores de pizza
  + sobremesas.json – Lista de sobremesas disponíveis
* **Comprovantes**:
  + Arquivos .txt gerados em csv/comprovante\_ID\_DATA.txt com detalhes do pedido

**4. Inicialização da aplicação**

1. No terminal, execute:

ts-node pizzaria\_app.ts

ou se compilado para JavaScript:

node js/pizzaria\_app.js

1. O aplicativo cria automaticamente as pastas e arquivos necessários caso não existam.

**5. Menu principal**

Ao iniciar, o usuário visualiza o seguinte menu:

=== MENU ===

1 - Registrar pedido

2 - Listar pedidos ativos com total por cliente

3 - Finalizar pedido

4 - Limpar pedidos ativos e reiniciar ID

5 - Relatório de vendas (diário/mensal)

0 - Sair

**6. Funcionalidades**

**6.1 Registrar pedido**

1. Escolha a opção **1**.
2. Será exibido o **cardápio de pizzas, preços e sobremesas**.
3. Informe os dados do cliente:
   * Nome
   * Telefone
   * Pizza escolhida
   * Bebida
   * Tamanho da pizza (Pequena/Média/Grande)
   * Quantidade de pizzas
   * Borda recheada (sim/não)
   * Quantidade de bebidas
   * Sobremesa
   * Quantidade de sobremesas
   * Endereço de entrega
   * Forma de pagamento (Dinheiro, Cartão ou PIX)
4. O sistema gera automaticamente um **ID de pedido** e o **horaPedido**.
5. O pedido é salvo em ativos.csv e um **comprovante em TXT** é gerado.

**6.2 Listar pedidos ativos**

1. Escolha a opção **2**.
2. O sistema exibe todos os pedidos em aberto, agrupados por cliente, com **total por cliente**.
3. Exemplo de saída:

Cliente: João | Telefone: 99999-9999

Pedido ID 1: Pizza: Calabresa (Grande) x2, Bebida: Coca-Cola x2, Sobremesa: Pudim x1, Forma: Dinheiro | Total: R$ 111.00

Total acumulado do cliente: R$ 111.00

**6.3 Finalizar pedido**

1. Escolha a opção **3**.
2. Informe o **ID do pedido** a ser finalizado.
3. O pedido será removido de ativos.csv e adicionado em historico.csv, com registro da **horaSaida** e **preço total**.
4. Confirmação no console:

Pedido 1 finalizado e registrado no histórico.

**6.4 Limpar pedidos ativos**

1. Escolha a opção **4**.
2. Confirme a exclusão de **todos os pedidos ativos**.
3. O arquivo ativos.csv será reiniciado com apenas o cabeçalho e last\_id.json será resetado para 0.

**6.5 Relatório de vendas**

1. Escolha a opção **5**.
2. O sistema gera **relatórios diários e mensais**, agrupando por **forma de pagamento**:
   * Quantidade de pizzas vendidas
   * Total arrecadado
3. Exemplo de saída:

=== RELATÓRIO DIÁRIO DE PIZZAS VENDIDAS ===

Dinheiro: 5 pizzas | Total: R$ 250.00

Cartão: 3 pizzas | Total: R$ 120.00

PIX: 2 pizzas | Total: R$ 90.00

=== RELATÓRIO MENSAL DE PIZZAS VENDIDAS ===

Dinheiro: 15 pizzas | Total: R$ 750.00

Cartão: 10 pizzas | Total: R$ 400.00

PIX: 8 pizzas | Total: R$ 360.00

**7. Emissão de comprovante**

* Todo pedido registrado gera automaticamente um **arquivo TXT** com detalhes.
* Local: csv/comprovante\_ID\_DATA.txt
* Conteúdo inclui:
  + Dados do cliente
  + Pedido completo
  + Forma de pagamento
  + Valor total

**8. Observações**

* Arquivos CSV e JSON não devem ser alterados manualmente para evitar inconsistências.
* Para encerrar a aplicação, escolha **0** no menu ou pressione CTRL+C.
* O sistema trata automaticamente a criação de pastas e arquivos ao iniciar.