

ESPECIFICAÇÃO TÉCNICA DE SOFTWARE

PROJETO: MERCADINHO PONTO CERTO

Documento detalhado de arquitetura, requisitos,
modelo de dados e procedimentos de manutenção.

LOYOLA
DEVS

Data da Emissão: 13/01/2026

Autor: Equipe de Desenvolvimento

1. ESCOPO E VISÃO GERAL

O Sistema de Gestão Mercadinho (SGV) é uma solução Desktop desenvolvida para atender às necessidades críticas de pequenos comércios. O software foi projetado com foco em 'Alta Disponibilidade Local', garantindo operação contínua mesmo sem conexão com a internet.

1.1. Objetivo do Sistema

- Registrar e controlar vendas (PDV);
- Gerenciar produtos e estoque;
- Controlar abertura e fechamento de caixa;
- Gerar relatórios financeiros e operacionais;
- Auxiliar no cálculo de taxas de pagamento (débito, crédito e parcelado).

2. ARQUITETURA DE SOFTWARE

2.1. Padrão de Projeto (Design Pattern)

O sistema adota o padrão 'Repository Pattern'. Isso isola a camada de domínio (Regras de Negócio) da camada de persistência (Banco de Dados). No futuro, isso permite trocar o SQLite por Maria DB ou PostgreSQL apenas alterando os drivers de conexão no SQLAlchemy, sem reescrever a lógica de vendas.

2.2. Estrutura da Aplicação (.exe)

O binário final encapsula o interpretador Python e as 18 bibliotecas dependentes. A comunicação com o hardware (USB/Serial) é feita via chamadas diretas de API do Windows (pywin32), evitando drivers intermediários complexos.

3. STACK TECNOLÓGICA COMPLETA (BILL OF MATERIALS)

Listagem exata das bibliotecas homologadas para compilação do projeto.

Biblioteca	Versão	Função no Sistema
Flet	v0.28.3	Interface Gráfica (GUI) Reativa
SQLAlchemy	v2.0.44	ORM (Camada de Dados)
Alembic	v1.17.2	Controle de Migrações de Banco
SQLite	v3.x	Motor de Banco de Dados (Nativo)
Passlib	v1.7.4	Criptografia de Senhas
Bcrypt	v5.0.0	Algoritmo de Hash Seguro
Python-dotenv	v1.2.1	Configurações de Ambiente
FPDF	v1.7.2	Gerador de Relatórios PDF Básico
ReportLab	v4.4.5	Gerador de Gráficos e PDFs Complexos
PySerial	v3.5	Conexão com Balanças/Impressoras Seriais
PyUSB	v1.3.1	Conexão com Leitores USB

PyWin32	v311	Integração com Sistema Operacional
PyZbar	v0.1.9	Decodificação de Cód. Barras (Imagen)
QRCode	v7.4.2	Gerador de QR Code (Pix)
Tkcalendar	v1.6.1	Componente de Data (Datepicker)
Python-escpos	v3.1	Impressão Térmica (Cupom)
Pandas	v2.3.3	Análise de Dados (Dataframes)
OpenPyXL	v3.1.5	Exportação Excel (.xlsx)
OpenCV	v4.12	Processamento de Imagem (Opcional)

4. DICIONÁRIO DE DADOS (SCHEMA)

Estrutura das principais tabelas do banco de dados SQLite (mercadinho.db).

Tabela: PRODUTOS

- *ID (PK)*: Identificador único.
- *EAN (String)*: Código de Barras (Indexado para busca rápida).
- *DESCRICAO (String)*: Nome do produto.
- *PRECO_CUSTO (Decimal)*: Valor de compra.
- *PRECO_VENDA (Decimal)*: Valor final.
- *ESTOQUE_ATUAL (Int)*: Quantidade disponível.
- *VALIDADE (Date)*: Data de vencimento (Gatilho para alertas).
- *LOTE*: Código para controle e rastreabilidade.

Tabela: FORNECEDORES

- *ID (PK)*: Identificador.
- *RAZAO_SOCIAL (String)*: Nome da empresa.
- *CNPJ (String)*: Documento fiscal.
- *PRAZO_ENTREGA (Int)*: Dias médios para reposição.
- *FORMAS_PAGAMENTO (JSON)*: Lista de meios aceitos (Pix, Boleto, etc).

Tabela: VENDAS (Header)

- *ID (PK)*: Número da venda.
- *DATA_HORA (Datetime)*: Timestamp da operação.
- *TOTAL (Decimal)*: Valor final da transação.
- *FORMA_PAGTO (Enum)*: Dinheiro/Crédito/Débito/Pix.
- *STATUS (String)*: 'FINALIZADA' ou 'CANCELADA'.

5. ESPECIFICAÇÃO FUNCIONAL DETALHADA

5.1. Algoritmo de Venda (PDV)

1. Operador inicia venda (Tela em estado 'Aguardando').
2. Input de Produto: Via Leitor (EAN) ou Digitação Manual.
3. Sistema consulta Tabela PRODUTOS:

- Se não encontrado -> Exibe erro sonoro/visual.
- Se encontrado -> Adiciona à lista temporária e soma ao Subtotal.

4. Fechamento:

- Operador seleciona Pagamento (F1-F4).
- Sistema grava registro na Tabela VENDAS e Tabela ITENS_VENDA.
- Sistema decremente ESTOQUE_ATUAL na Tabela PRODUTOS.
- Impressora Térmica recebe comando ESC/POS.

5.2. Regras de Negócio (Validadores)

- Bloqueio de Caixa: Vendas são rejeitadas se não houver registro de 'Abertura de Caixa' no dia.
- Estoque Negativo: O sistema permite venda sem estoque (configurável), mas gera log de alerta.
- Validade: Itens vencidos são destacados em vermelho no cadastro.

6. PLANO DE MANUTENÇÃO E SUPORTE

6.1. Procedimentos de Backup

Como o banco é um arquivo local (.db), o backup consiste na cópia binária do arquivo. Recomenda-se automação via script BAT que copie 'mercadinho.db' para uma pasta segura ao desligar o PC.

6.2. Logs de Erro

O sistema gera um arquivo 'app.log' na raiz. Erros comuns de hardware (Impressora desconectada) são registrados lá com timestamp, facilitando o diagnóstico técnico.

7. Considerações Finais

O sistema Mercadinho Ponto Certo, desenvolvido como um Sistema de Gestão de Vendas (SGV), oferece uma solução integrada para controle de vendas, estoque e financeiro, atendendo às principais necessidades de um ponto de venda. A arquitetura adotada e as funcionalidades implementadas tornam o sistema consistente, funcional e preparado para futuras melhorias.