

Documentação Técnica – Sistema de Ordem de Serviço (Projeto Integrador)

1. Descrição Geral do Sistema

O Sistema de Ordem de Serviço (Projeto0S) foi desenvolvido em Java utilizando o paradigma orientado a objetos. O objetivo principal é gerenciar clientes e ordens de serviço (OS), permitindo cadastrar, listar e alterar o status das OSs entre os estados ABERTA, ANDAMENTO e FINALIZADA. O sistema utiliza uma interface gráfica desenvolvida com Swing, persistência de dados em arquivos JSON e segue uma arquitetura modular baseada em camadas.

2. Arquitetura do Sistema

O sistema foi projetado seguindo uma arquitetura em três camadas principais:

- **Model:** Contém as classes que representam as entidades principais do domínio (Cliente e OS). Essas classes encapsulam dados e comportamentos relacionados às entidades.
- **DAO (Data Access Object):** Responsável pelo acesso e manipulação dos dados persistidos em arquivo JSON. Utiliza Collections (List e Map) para gerenciar os objetos em memória e a biblioteca Gson para serialização.
- **View:** Implementa a interface gráfica do usuário utilizando Swing. Os painéis principais permitem o cadastro e visualização de clientes e ordens de serviço.

3. Justificativas Técnicas

- **Uso de Map e Collections:** Foi utilizado HashMap para indexação rápida dos objetos por ID, melhorando o desempenho de busca e remoção. As Lists são usadas para manter a ordenação e integração com tabelas.
- **Modularização:** A divisão em pacotes (model, dao, view) reduz a complexidade e facilita a manutenção do código.
- **Polimorfismo:** O sistema adota uma interface genérica (GenericoDAO) implementada por ClienteDAO e OSDAO, permitindo operações comuns sobre diferentes tipos de entidades, aplicando o conceito de polimorfismo.

4. Diagrama de Classes (representação simplificada)

Classe	Atributos Principais	Relações
Cliente	id, nome, cpfCnpj, rua, numero, bairro	Usada por OS
OS	id, cliente, descricao, prioridade, status, valor	Associa Cliente
ClienteDAO	clientes: List, clienteMap: Map	Gerencia Cliente
OSDAO	osList: List, osMap: Map	Gerencia OS
MainFrame	painéis: ClientePanel, OSPanel, etc.	Integra interface

5. Execução e Interface

O sistema é iniciado pela classe **MainFrame**, que cria uma janela principal contendo várias abas para cada funcionalidade. O usuário pode cadastrar novos clientes, criar ordens de serviço e acompanhar o progresso de cada OS em tempo real. Os dados são automaticamente gravados e carregados a partir de arquivos JSON.

Exemplo da interface (representação):

[Janela Principal] → Abas: Clientes | Criar/Listar OS | OS Abertas | OS em Andamento | OS Finalizadas

6. Conclusão

O sistema atende aos requisitos do projeto integrador, apresentando arquitetura modular, uso de polimorfismo, persistência com JSON e uma interface gráfica completa. A estrutura adotada facilita futuras expansões, como autenticação de usuários, relatórios e integração com banco de dados.