

Documentação Técnica - Sistema CCO com API Transdata

1. Introdução

Este sistema foi desenvolvido para automatizar a visualização e análise dos dados de cumprimento de serviço (CCO), integrando dados da API Transdata e do ERP Globus. A aplicação oferece filtros por data, linha, empresa, prefixo e status, gráficos dinâmicos com Chart.js e exportação de relatórios em Excel e PDF. O projeto é estruturado em monorepo utilizando Turbo e PNPM, com backend em NestJS e frontend em React + Vite.

2. Tecnologias Utilizadas

Frontend:

- React + Vite com TypeScript
- Axios para chamadas HTTP
- Chart.js e Recharts para gráficos
- react-router-dom para roteamento
- Tailwind CSS com PostCSS
- Exportação com jsPDF, xlsx e file-saver

Backend:

- NestJS (Node.js + Express)
- Axios para integração com API Transdata
- TypeORM com SQLite
- Armazenamento simulado (mock)
- Exportações em Excel e PDF

Ferramentas de desenvolvimento:

Documentação Técnica - Sistema CCO com API Transdata

- Turbo (monorepo)
- PNPM (gerenciador de pacotes)
- ESLint, Prettier, TypeScript, Tailwind

3. Estrutura do Projeto

Monorepo com estrutura principal:

Dashboard/

```
├─ apps/  
|   ├─ backend/    → NestJS  
|   └─ frontend/   → React + Vite  
├─ package.json    → Gerencia os workspaces  
├─ turbo.json      → Orquestra os builds/dev  
└─ pnpm-workspace.yaml
```

Frontend:

src/

```
├─ components/  
|   ├─ charts/ → BarChartCumprimento.tsx  
|   └─ filters/ → FiltrosDashboard.tsx  
├─ pages/ → DashboardPage.tsx, HomePage.tsx  
├─ routes/ → AppRoutes.tsx  
├─ services/ → api.ts  
├─ types/ → cumprimento.ts  
└─ styles/ → global.css
```

Documentação Técnica - Sistema CCO com API Transdata

└─ App.tsx, main.tsx

4. Funcionalidades do Sistema

- Filtros por data, empresa, linha, prefixo e status
- Importação de dados da API Transdata
- Indicadores: furos, atrasos, colisões, quebras, defeitos
- Gráficos dinâmicos (barras, linhas, etc.)
- Exportação para Excel e PDF
- Armazenamento temporário em memória

5. Endpoints Disponíveis (Backend)

Importar dados:

Método: GET

Rota: /cumprimentos/importar

Parâmetros: dia (obrigatório), idempresa, idservico, numerolinha, prefixoprevisto, statusini, statusfim

Listar dados importados:

Método: GET

Rota: /cumprimentos

Retorno: JSON com lista de dados de cumprimento

6. Exemplo de Requisição

GET <http://localhost:3000/cumprimentos/importar?dia=2025-05-07&idempresa=1&idservico=2>

Documentação Técnica - Sistema CCO com API Transdata

Resposta esperada:

```
{  
  "mensagem": "Importação concluída",  
  "total": 42  
}
```

7. Considerações Finais

O sistema adota uma arquitetura moderna, com separação clara de responsabilidades, utilização de NestJS no backend, monorepo com Turbo, PNPM e frontend moderno com React + Vite. A análise por meio de gráficos e relatórios exportáveis facilita a tomada de decisões baseada em dados e performance operacional.