

Documentação de Alternativas de Busca Processual

1. API Pública DataJud

- Autenticação via APIKey no cabeçalho Authorization.
- Endpoints: https://api-publica.datajud.cnj.jus.br/api_publica_<tribunal>/_search
- Exemplos de uso:
 - Busca por número de processo (match em numeroProcesso).
 - Filtro por classe e órgão julgador (bool + must).
 - Paginação com size + search_after.
- Campos retornados: id, tribunal, numeroProcesso, dataAjuizamento, grau, formato, sistema, classe, assuntos, orgaoJulgado
- Limitação: não suporta busca por nome de parte ou advogado.

2. Alternativas de busca por nome

2.1 Nested Query

- Consulta em dadosBasicos.polo.parte via nested:

```
{ nested: { path: 'dadosBasicos.polo.parte', query: match_phrase_prefix em pessoa.nome } }
```

2.2 Wildcard

- Uso de wildcard no campo .keyword:

```
{ wildcard: { 'dadosBasicos.polo.parte.pessoa.nome.keyword': '*NOME*' } }
```

2.3 Download MTD + filtro local

- Baixar JSON bruto (MTD) e filtrar em JavaScript:

```
filtrados = allDocs.filter(doc => ...includes(nomeInput))
```

3. SOAP MNI (Modelo Nacional de Interoperabilidade)

- WSDL:
<https://www.cnj.jus.br/.../servico-intercomunicacao-2.2.2.wsdl>
- Operação: consultarProcesso com filtro de parte:

```
<ser:nomeParte>JOÃO DA SILVA</ser:nomeParte>
```
- Retorno: XML com detalhes do processo.

4. API Codilo (terceiros)

- Endpoint exemplo:
GET <https://api.codilo.com.br/v1/processos?tribunal=TJSP&parte=JOÃO%20DA%20SILVA>
- Autenticação: Bearer token.
- Suporte a nome, CPF, número, classe, datas.

5. Considerações Finais

- Escolha entre solução oficial (SOAP), REST pública (DataJud) ou terceirizada.
- Balanceie performance, privacidade e complexidade.