

# Documentação Técnica - Integração com HubSpot

---

## Introdução

Este documento descreve as decisões técnicas, motivações e sugestões de melhorias relacionadas à integração entre uma aplicação Spring Boot e a API do HubSpot utilizando OAuth 2.0. A integração permite autenticação segura e operações CRUD sobre contatos.

## Estrutura do Projeto

- client: Interfaces Feign para integração com APIs externas.
- controller: Endpoints REST expostos para autenticação e manipulação de contatos.
- dto: Objetos de transferência de dados usados para comunicação entre camadas.
- exceptions: Manipulação centralizada de exceções customizadas.
- service: Regras de negócio da aplicação.
- token: Lógica de obtenção e renovação de tokens OAuth.
- webhook: Endpoint para recebimento de eventos do HubSpot.

## Tecnologias e Bibliotecas Utilizadas

- Java 17
- Spring Boot 3.4.4
- Spring Web: para criação de endpoints REST.
- Spring Validation: para validação dos dados de entrada.
- Spring Cloud OpenFeign: abstração para chamadas HTTP externas (HubSpot).
- Lombok: para reduzir boilerplate (getters/setters/constructors).

## Decisões Técnicas

- Projeto pronto para execução com docker-compose, reduzindo a complexidade do setup inicial.
- Uso do Feign Client para simplificar chamadas HTTP, possibilitando melhor legibilidade.
- Configuração com variáveis externas (.env ou application.properties) para manter segurança e flexibilidade.
- Uso do Ngrok para exposição local segura e testes com webhooks do HubSpot.
- Implementada uma classe simples com escopo de componente (@Component) para armazenar o token OAuth em memória durante a execução da aplicação, facilitando os testes via API sem a necessidade de fornecer manualmente o token a cada requisição.
- Foram criados dois novos endpoints para listagem e deleção de contatos, permitindo a verificação dos dados enviados para o HubSpot e facilitando o monitoramento e a depuração durante o processo de integração.

## Melhorias Futuras

- Adicionar cache para reduzir chamadas repetidas à API do HubSpot.
- Persistência dos tokens em banco de dados para evitar nova autenticação a cada execução.
- Interface frontend para facilitar a visualização e testes dos dados integrados.
- Adicionar testes unitários e de integração automatizados.
- Implementação de uma lógica para gerar um novo token a partir do (refresh\_token) disponibilizado pela api do HubSpot.
- Implementação de ferramentas para monitorar os logs da aplicação.
- Criar mais endpoints para gerir os contatos, tais como atualizar um contato por exemplo, busca por id ou e-mail dentre outros.