

DOCUMENTAÇÃO

Aplicação consumindo API do Gmail

1 - Proposta:

Utilizando as APIs do "Gmail" e do "Trello", desenvolver um código em Quarkus ou Spring capaz de executar as seguintes tarefas:

1. Monitorar a caixa de entrada do Gmail

- Criar log local com os intervalos de monitoramento

2. Sempre que tiver um novo email com a palavra "Trello"

- Criar um novo card no Trello, conteúdo:
 - Título: assunto do email
 - Descrição: corpo do email
- Persistir dados em database local
- Incluir no log

3. Expor dados da API para consultar os registros/históricos do database local

Level UP:

Mover o card para a lista "e-mails lidos" automaticamente após esse e-mail ter sido aberto no "Gmail"

- Incluir no log atividade

Adicionar uma label "importante" no card do Trello, se o e-mail correspondente for marcado com "estrela" no Gmail.

Ficha Técnica

Responsável pela Elaboração:

Edgar Costa Melo – edgar_cmello@hotmail.com / (98) 98208-5049

Visão Geral da Aplicação

Trata-se de uma aplicação back-end (API), construída utilizando a linguagem Java, responsável por gerenciar a caixa de e-mails da plataforma Gmail e realizar integração com a plataforma do Trello, por meio de sua API.

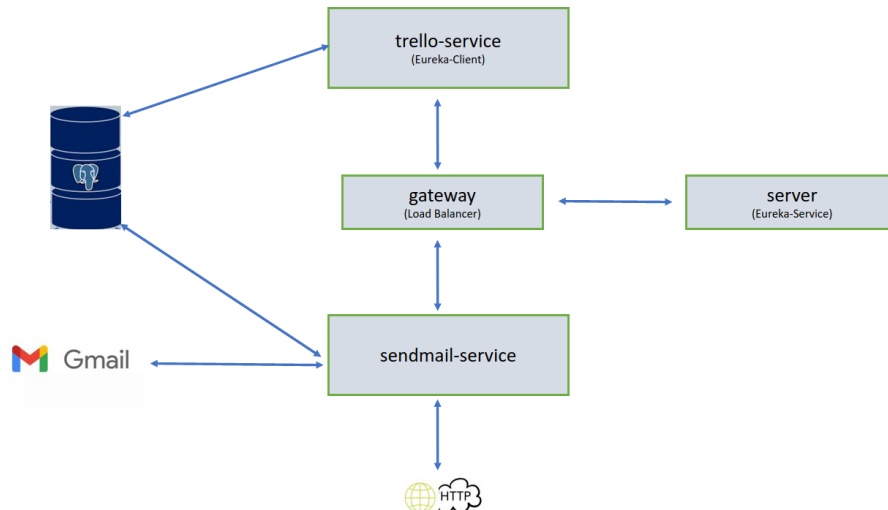
A aplicação foi desenvolvida com o ecossistema do Spring, utilizando a arquitetura de micro serviços sob forma de API REST.

Linguagens e Ferramentas:

- Java
- Spring (Spring-Boot, Spring-Cloud, Eureka, Gateway)

- PostgreSQL
- API Gmail
- Git / GitHub
- IntelliJ IDEA

Arquitetura dos Serviços da Aplicação



Requisitos Funcionais

1 – Envio de E-mail

A aplicação possibilita, através da API do Gmail integrada no modulo do serviço “*sendmail-service*” do pacote de nome “**Test**” (raiz do projeto), onde é possível enviar um e-mail utilizando Protocolo de Transferência de Correio Simples (SMTP) com as credencias do usuário especificado na classe **SendMailService**.

- A Requisição para envio de um e-mail deverá ser feita no diretório:
 - <http://localhost:8765/sendmail-service>
- Método utilizado:
 - POST
- Corpo da requisição no formato JSON:

```

{
  "remetente": "crvedgartest@gmail.com",
  "destinatario": " e-mail do destinatário ",
  "titulo": " (String) ",
  "corpo": "(String)"
}

```

*Os dados do corpo da requisição irão instanciar um objeto do tipo **MailInput**, o qual será validado utilizando o Bean Validation do Spring, os campos **remetente** e **destinatario** serão validados considerando o formato de e-mail, enquanto os campos **titulo** e **corpo** serão validados quanto a presença de valores do tipo String, não sendo aceito valores em branco ou *null*, sendo nesse caso lançada uma Exception.

Caso a requisição seja realizada com sucesso, será retornado o status 201 e o objeto do tipo **DtoMail** referente ao e-mail persistido no banco de dados.

Caso os dados do corpo ou da requisição sejam enviados com parâmetros diferente dos especificados acima, será lançada uma exception.

2 – Criação de Card Trello

Caso seja identificado que o e-mail enviado contenha a palavra “Trello”, o serviço “*sendmail-service*” irá automaticamente enviar uma requisição para o serviço “**trello-service**”, o qual encontra-se injetado no cliente “*sendmail-service*” sob forma de interface que expõe o método **criarAutomaticCard**, sendo este método nativo do pacote de nome “**demo**” (raiz do projeto);

Após receber a requisição, será criado um novo Card com o Título e a descrição informada, então os dados do Card serão persistidos no banco de dados local;

Após a criação, será mostrado no log do console as informações do novo Card.

A troca de informações entre os serviços ocorre através da FeignClient, Eureka e Gateway, os quais são responsáveis por identificar, balancear as cargas com LoadBalancer e redirecionar as uri e requisições para os serviços solicitados.

3 – Monitorando a Caixa de E-mail

A caixa de entrada do Gmail, representado pelos e-mails salvos no banco de dados, será monitorada em intervalos de 3 minutos.

O serviço responsável por realizar esta funcionalidade encontra-se disponível na classe **VerificadorCxEntrada** do pacote de nome “**Test**” (raiz do projeto).

Após realizar a busca no banco, será informado no log do console a hora da consulta e a quantidade de e-mails da caixa de entrada.

Orientações

1 – Baixar código disponível em <https://github.com/CRVEdgar/API-Gmail-Trello.git>

2 – Utilizar a IDE IntelliJ, STS ou similar;

3 – Iniciar o servidor do pacote de nome “server”

4 – Iniciar o redirecionador de rotas e balanceador de carga do pacote de nome “gateway”

5 – Iniciar os serviços do pacote de nome “demo” e “Test”

6 – Realizar uma requisição **POST** de teste, com a ferramenta Postman ou similar, no endpoint:

<http://localhost:8765/sendmail-service>

Dados(JSON):

```
{  
  "remetente": "crvedgartest@gmail.com",  
  "destinatario": "edgar\_cmello@hotmail.com",  
  "titulo": "Teste de envio do e-mail – API Gmail e Trello",  
  "corpo": "Ola, Edgar, tudo bem? estou testando seu código neste exato momento"  
}
```