

# Caiena,

**Olá, parabéns por chegar até aqui!**

Este é o desafio técnico da Caiena. Para realizá-lo, você deve desenvolver a solução do problema abaixo. Em um segundo momento, você compartilhará o resultado do desafio com nossas lideranças técnicas, que poderão fazer perguntas para se aprofundar no tema.

**Vamos lá?**

## Descrição do desafio

Faça uma aplicação integrada com o OpenWeatherMap e o Twitter para que seja possível enviar um tweet com a temperatura atual e a previsão dos próximos cinco dias (média diária).

A aplicação não precisa apresentar uma interface gráfica. Basta que disponibilize um endpoint HTTP, que receba dados por parâmetro e, aí, construa o texto e publique o tweet.

Não é necessário implementar controle de autenticação.

**Um exemplo de tweet:**

34°C e nublado em <cidade> em 12/12. Média para os próximos dias: 32°C em 13/12, 25°C em 14/12, 29°C em 15/12, 33°C em 16/12 e 28°C em 16/12.

## Entregáveis (o que o desafio deve contemplar):

- Uma gem de um SDK para utilização do OpenWeatherMap;
- Uma API que utiliza o SDK criado acima para realizar as requisições na API do OpenWeatherMap;
  - A API precisa expor um endpoint que deverá receber (POST) um ID para utilizar no SDK;
  - A API deverá se comunicar com o Twitter utilizando um SDK, [como esse](#). *É necessário habilitar a conta de [desenvolvedor do twitter](#). Portanto, é importante realizar esse passo para que o tempo de aprovação (geralmente de minutos) não impacte no desenvolvimento final.*

Os itens devem ser hospedados em alguma plataforma de gerenciamento/versionamento de código como GitHub, GitLab, etc.

Os repositórios devem contar com um arquivo README contendo instruções de configuração e inicialização do projeto em questão.

## Diferenciais:

- Uso de Docker
- Baixo número de dependências

## Processo de avaliação do desafio

### Primeira etapa:

- Demonstrar a aplicação em funcionamento para validação dos requisitos das funcionalidades
  - Não é necessário hospedar a aplicação em um servidor dedicado. A aplicação pode rodar em sua máquina de trabalho (notebook ou desktop)
- Explicar a construção e arquitetura da aplicação, passando pelo código-fonte do repositório.

### Segunda etapa:

- Avaliação e feedback do código e das funcionalidades entregues.

**Critérios de avaliação (a lista não está ordenada por importância):**

- Aplicar TDD (*Test-driven development*);
- Usar controle de versão (Git @ GitHub/GitLab/etc. - organização, commits etc.);
- Legibilidade e elegância do código;
- Arquitetura da solução;
- Funcionalidades entregues;
- Demonstração das funcionalidades;
- Documentação.