



APRESENTAÇÃO

Olá, seja bem-vindo à Kogui.

A Kogui é uma empresa nascida a partir do Grupo Kogui que está no mercado há 15 anos atuando em projetos nacionais e internacionais. Parceira oficial da SAP e SAS, desenvolve soluções para diferentes mercados, destacando-se em projetos inovadores. Suas verticais de negócios estão divididas em:



Fábrica de Software

Adota uma abordagem ágil, garantindo interação constante com cliente durante 0 oprojeto. é principalpremissa desenvolver aplicações garantindo arquitetadas sua evolução manutenção.



Projetos Cognitivos KOGUI

Área de desenvolvimento de projetos voltados para tecnologias e plataformas cognitivas com aplicação intensa de Inteligência Artificial e BigData.



Produtos (SaaS)

Sistemas para as áreas de Recursos Humanos, Área Financeira e Controladoria e Área Jurídica.







Prazo de entrega: 20/08/2025

Objetivo

Avaliar se o(a) candidato(a) consegue subir um ambiente Docker com Django (back/front) + Postgres + n8n, implementar um formulário simples em Django, persistir no banco e disparar um webhook no n8n.

Entregáveis

- Repositório Git com o código.
- README.md explicando como rodar (passo a passo) e as variáveis de ambiente (fornecer .env.example).
- docker-compose.yml com todos os serviços prefixados challenge_.
- Dockerfile da aplicação Django.
- Workflow do n8n funcionando (Exportar o JSON do fluxo e adicionar ao repositório do Git).

Requisitos Mínimos (OBRIGATÓRIOS)

1. Docker/Compose

- Serviços obrigatórios:
 - challenge_web (Django)
 - challenge_db (Postgres)
 - challenge_n8n (n8n)
- o docker compose up -d --build deve subir tudo sem passos manuais fora do Docker.

2. Django (back + front)

- Página / com formulário (campos mínimos: nome e email).
- Validação server-side + CSRF habilitado.
- Salvar no Postgres (modelo mínimo Lead(nome, email, criado_em)).
- Servir estáticos (ao menos 1 CSS e 1 imagem).

3.Integração n8n

- Ao receber o POST do formulário, enviar um POST JSON para um webhook do n8n (URL via variável de ambiente).
- Workflow do n8n precisa validar payload e responder 200 em sucesso.

4. Config via .env

Usar .env/.env.example (sem segredos no repositório).





Prazo de entrega: 20/08/2025

Requisitos Opcionais (DIFERENCIAIS)

- Página /leads/ listando registros com busca e/ou paginação.
- Mensagens de sucesso/erro (Django messages).
- · Admin Django habilitado.
- WhiteNoise para estáticos / build multi-stage no Dockerfile.
- Healthchecks no compose; evitar latest.
- Testes unitários (mínimos) e linters (flake8/ruff, black).
- Limitar logs do Docker (ex.: max-size: "50m").

Descrição do Desafio (o que deve ser feito)

Construa uma solução mínima, conteinerizada com Docker, composta por:

1. Aplicação Django (back + front):

- Exponha a rota pública / com um formulário de lead contendo nome e e-mail (mínimo).
- Valide os campos no servidor (e ative CSRF).
- Persista os dados em Postgres via ORM (modelo simples Lead(nome, email, criado_em)).
- Após salvar, dispare um POST JSON para um webhook do n8n (URL configurável por variável de ambiente).
- Sirva ao menos um CSS e uma imagem (estáticos) pela app.

2. Workflow no n8n:

- Crie um Webhook (POST) que receba o payload do Django.
- Faça uma validação básica do corpo (nome/e-mail).
- Responda HTTP 200 com um JSON de confirmação ({"ok": true}) quando válido; trate erro com 400/422.

3.Infra com Docker Compose:

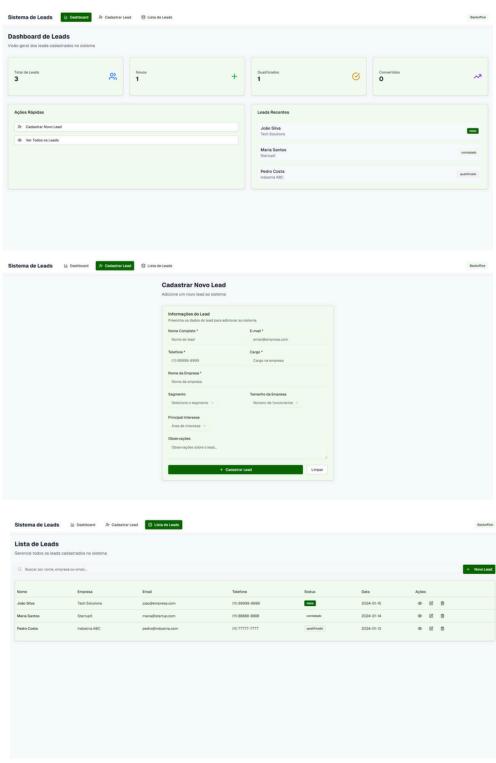
- Suba três serviços com prefixo challenge_: challenge_web (Django), challenge_db (Postgres) e challenge_n8n (n8n).
- O comando docker compose up -d --build deve levantar tudo sem passos manuais fora do Docker.
- Variáveis sensíveis via .env (forneça .env.example, sem segredos reais).





Prazo de entrega: 20/08/2025

Exemplos de Telas



Boa Sorte.

Atenciosamente,

Alexandre Cilani



Alameda Francisco Alves, 169 – 8° andar -Edifício Infinite Bairro Jardim– Santo André – SP www.kogui.com.br