

Desafio Técnico - Desenvolvedor Júnior

Contexto

Uma empresa de e-commerce precisa de um sistema interno para gerenciar produtos e suas avaliações. Os dados serão futuramente utilizados para relatórios e dashboards, então a performance de leitura é essencial.

Atualmente, a empresa já conta com milhares de produtos, e cada produto pode acumular dezenas de milhares de avaliações. A leitura dessas informações, especialmente o detalhamento das avaliações de um produto, está se tornando cada vez mais lenta e custosa, impactando a experiência dos usuários.

Você foi contratado para construir a primeira versão desse sistema, garantindo que ele seja funcional, performático e sustentável tecnicamente.

Requisitos do Desafio

- Criar uma API REST com:
 - CRUD de produtos
 - CRUD de avaliações de produtos
 - Endpoint para listar as avaliações de um produto (com paginação)
 - Endpoint para retornar os top 10 produtos mais bem avaliados
- O banco deverá ser populado automaticamente com um volume significativo de dados:
 - 20.000 produtos
 - Cada produto com entre 10.000 a 50.000 avaliações
 - Use ferramentas como DoctrineFixturesBundle ou comandos personalizados (ex: bin/console) para gerar os dados de forma programada.

Observação: Este volume deve causar lentidão nas consultas, expondo gargalos de performance para que você possa resolvê-los.

Tecnologias Obrigatórias

- PHP + Symfony

Desafio Técnico - Desenvolvedor Júnior

- MySQL
- Docker (Dockerfile + docker-compose)
- Redis (devera ser utilizado para otimizar o sistema conforme sua analise)
- Testes automatizados (pelo menos para os fluxos principais)

Critérios de Avaliação

- Organização do código e clareza da estrutura
- Design da API e aderência a boas práticas REST
- Tempo de resposta da API em endpoints críticos
- Estratégias aplicadas para melhorar performance
- Qualidade dos testes automatizados
- Clareza e objetividade na documentação do projeto