



samplemed

**Exercício
Samplemed**

Passando aqui para te parabenizar! Você passou na nossa etapa de entrevistas, e agora estamos no **TESTE TÉCNICO**.

Essa etapa é importante para avaliarmos sua experiência e organização durante o desenvolvimento;

1) A ideia não é ver se você vai acertar no bit. Queremos ver como se organiza e implementação da solução (boas práticas, etc).

2) Caso você tenha qualquer tipo de dúvida, pode entrar em contato conosco.

>>> SEGUO O TESTE

Contexto

Faça um MVP (Produto mínimo viável) de um blog baseado no documento abaixo

<https://book.cakephp.org/2/en/tutorials-and-examples/blog/blog.html>

Com base nessas especificações, apresente uma pequena MVP que no mínimo insira posts no banco de dados através de uma tela com chamada a API e liste esses posts. Se preocupe mais com o código interno das APIs (organização, consistência, etc) do que com as telas.

Parte 1 - Arquitetura

Além da tabela de posts desenhe outras tabelas necessárias para o funcionamento correto do sistema de um blog (tabela de usuários, etc). Pode ser um arquivo com o schema, um DBML (<https://www.dbml.org/>), ou mesmo uma imagem (procure colocar este artefato junto com o projeto - git, etc).

Você não precisa utilizar todas essas tabelas no MVP, só queremos entender sua organização e conhecimento na arquitetura do banco de dados.

Considerações

- **Organização:** Agrupe os módulos em contextos de forma que essa associação seja algo que faça lógica no mundo real, se possível utilize todos os nomes de funções e commits em inglês.
- **Repositório:** O histórico de commits deve ser claro e representarem uma mudança unitária do projeto.

Detalhe em um documento a parte os motivos de determinadas tomadas de decisão e da codificação:

- **Resiliência:** O que fazer para mitigar possíveis erros e controlar os possíveis erros recebidos da API?
- **Performance:** Quais boas práticas são aplicadas em banco de dados e no código para garantir performance?
- **Segurança:** Como garantir segurança para as APIs do sistema?
- **Simultaneidade:** Como trabalhar com simultaneidade se milhares de requisições forem solicitadas simultaneamente?

Compartilhe o código do exercício em um repositório git.

>>> FIM DO TESTE

Muito sucesso para você e até breve!

Um abraço,

—