



📌 Case Técnico – Desenvolvedor Full-Stack

Processo Seletivo Dizevolv

Prazo de Entrega: dia 28/01, às 10hrs

Objetivo do Desafio

Este desafio tem como objetivo avaliar a capacidade do(a) candidato(a) de:

- Executar um projeto **end-to-end** (Frontend + Backend)
- Interpretar corretamente um **PRD técnico real**
- Aplicar conceitos de **arquitetura SaaS e multi-tenancy**
- Desenvolver uma aplicação **funcional, segura e escalável**
- Seguir rigorosamente **padrões de stack, arquitetura e boas práticas**

O desafio simula um **cenário real de desenvolvimento de um produto SaaS**, com foco em **segurança, escalabilidade, organização técnica e governança de dados**.

Contexto do Desafio

Você deverá desenvolver o **MVP de uma Plataforma SaaS de Gestão Operacional e Compliance para Clínicas**, chamada **ClinicOps**.

A **ClinicOps** é uma plataforma **multi-tenant** voltada para clínicas médicas, odontológicas e de saúde integrada, com o objetivo de **centralizar a gestão operacional**, garantindo:

- Segregação total de dados entre clínicas
- Controle de acesso por perfil
- Rastreabilidade de ações

- Auditoria e compliance (LGPD)
- Governança e escalabilidade SaaS

A plataforma será utilizada por:

- **Admin Master (Plataforma)** – equipe proprietária do SaaS
 - **Admin Tenant (Clínica)** – gestores da clínica
 - **Usuários Operacionais** – médicos, atendentes e profissionais de saúde
-

Definição do Case via PRD (Obrigatório)

Ponto de Partida Único: PRD Oficial

O case será apresentado e explicado **exclusivamente** por meio do PRD abaixo.

O candidato **NÃO deverá criar, alterar ou reinterpretar o PRD**.

O PRD já está previamente definido e faz parte do desafio.

Sua responsabilidade consiste em:

- Copiar o PRD fornecido
 - Utilizá-lo como base **integral** para o desenvolvimento do projeto
 - Anexar o PRD ao material final de entrega, **sem modificações conceituais**
-

Documento de Requisitos do Produto (PRD) – Versão Final

Nome do Projeto: ClinicOps

1. Visão Geral (Overview)

O projeto consiste na construção de uma **Plataforma SaaS Multi-tenant de Gestão Operacional para Clínicas**, denominada **ClinicOps**, que será o MVP do curso “**Do 0 ao App**”.

A plataforma permitirá que clínicas (tenants) realizem:

- Cadastro e gestão de profissionais de saúde

- Gestão de pacientes
- Registro de atendimentos e procedimentos
- Armazenamento de documentos sensíveis
- Controle de acesso por perfil de usuário
- Auditoria completa das ações realizadas no sistema

Todo o sistema deverá operar sob um **modelo SaaS escalável**, garantindo **segurança, isolamento de dados e conformidade com LGPD**.

Pilares do Produto

- **Segurança Multi-tenant (RLS)**
Segregação total de dados entre clínicas, utilizando Row Level Security no banco de dados.
- **Modelo de Assinatura SaaS**
A plataforma é monetizada por meio de planos recorrentes, cobrados por clínica.
- **Gestão Hierárquica de Usuários**
Dashboards distintos para:
 - **Admin Master (Plataforma)**
 - **Admin da Clínica (Tenant)**
 - **Usuários Operacionais**

2. Metas do Produto

- Implementar **multi-tenancy segura com RLS (Supabase)**
- Integrar **Stripe** para cobrança recorrente da assinatura SaaS
- Criar um **Dashboard Master** para gestão da plataforma
- Desenvolver **Landing Page e Página de Pricing** focadas em conversão
- Garantir **criptografia de dados sensíveis**
- Implementar **e-mails transacionais automatizados**

3. Stack Tecnológica e Integrações (Referencial do Produto)

Frontend

- React + Vite

Banco de Dados

- Supabase (PostgreSQL, Auth, RLS, Storage)

Pagamentos

- Stripe

E-mails

- Brevo

Hosting

- Netlify

UI

- shadcn/ui
-

4. Funcionalidades Essenciais

Público / Marketing

- Landing Page institucional
- Página de Pricing

Autenticação

- Cadastro de clínicas
- Login de usuários
- Recuperação de senha

Dashboard do Usuário

- Visualização de pacientes
- Registro de atendimentos e procedimentos
- Upload e visualização de documentos

Admin Dashboard (Tenant)

- Gestão de usuários da clínica
- Gestão de pacientes
- Visualização de uso do plano
- Controle de permissões

Dashboard Master (Plataforma)

- Gestão de clínicas
- Gestão de planos
- Visualização de métricas globais
- Controle de assinaturas

Auditoria

- Registro de ações sensíveis
 - Logs de criação, edição e exclusão
-

5. Modelo de Dados

Tabelas Obrigatórias

- planos
- clínicas
- usuários
- pacientes
- atendimentos
- documentos
- uso recursos
- auditoria

→ Obrigatório **implementar RLS em todas as tabelas utilizando `clinica_id`.**

6. Segurança e Compliance

- Criptografia de dados sensíveis
 - RLS sempre ativo
 - Auditoria completa de ações
 - Rate Limiting
 - Conformidade com LGPD
 - Boas práticas de segurança de aplicações SaaS
-

Importante

Este PRD **não deve ser recriado**, apenas utilizado como base integral para o desenvolvimento.

2. Padrão de Stacks e Requisitos Técnicos (Obrigatório)

Frontend

- Framework: React ou Next.js (App Router)
- Linguagem: TypeScript
- Estilização: Tailwind CSS
- UI: shadcn/ui
- Mobile: React Native + Expo

Backend

- Runtime: Node.js (LTS)
- API: Next.js API Routes ou Edge Functions
- BaaS: Supabase (PostgreSQL, Auth, Storage)
- Edge Functions: Supabase Edge Functions (Deno)

Infraestrutura

- Versionamento: GitHub
 - Hospedagem Frontend: Vercel
 - CDN / Segurança: Cloudflare
 - Backend: Supabase Cloud
-

3. Uso do MCP e Publicação

O projeto deverá ser desenvolvido utilizando o **MCP do Supabase**.

A entrega deve contemplar:

- Frontend 100% funcional
 - Backend 100% funcional
 - Implementação completa no Supabase
 - Projeto publicado e acessível (**URL funcional**)
-

Entregáveis

O candidato deverá entregar:

- Link do repositório GitHub
 - URL da aplicação publicada
 - PRD anexado (sem alterações)
 - Evidências de funcionamento:
 - Prints **ou**
 - Vídeo curto demonstrativo
-

Critérios de Avaliação

- Fidelidade ao PRD
 - Qualidade da arquitetura
 - Implementação correta de segurança e RLS
 - Organização e clareza do código
 - Funcionamento completo da aplicação
 - Clareza e profissionalismo na entrega
-

Boa sorte!