





Gestão de Saúde Ocupacional (SaaS)

Um sistema SaaS B2B desenvolvido para **Consultorias de Saúde e Segurança do Trabalho**. O software gerencia o ciclo de vida financeiro e operacional de exames ocupacionais (ASO), desde o cadastro do funcionário até a emissão da fatura consolidada para a empresa cliente.

Visão Geral

O sistema atua como um hub administrativo, permitindo que a consultoria registre exames realizados em clínicas parceiras, aplique regras de precificação complexas (tabelas de exceção por cliente) e automatize o fechamento financeiro mensal.

Principais Módulos

-  **Administrativo:** Gestão da própria consultoria (Tenant) e controle de acesso da equipe.
-  **Comercial:** Gestão de contratos, tabelas de preços personalizadas e protocolos de exames por cargo (PCMSO).
-  **Operacional:** Registro de atendimentos, cálculo automático de exames obrigatórios e vigência do ASO.
-  **Financeiro:** Agrupamento de atendimentos, geração de faturas mensais e controle de inadimplência.

Stack Tecnológica

A arquitetura foi pensada para performance, tipagem estrita e velocidade de desenvolvimento.

- **Framework:** [Next.js 15](#) (App Router & Server Actions)
- **Linguagem:** [TypeScript](#)
- **Banco de Dados & Auth:** [Supabase](#) (PostgreSQL)
- **Estilização:** [Tailwind CSS](#)
- **Componentes UI:** [shadcn/ui](#) (Radix UI)
- **Validação:** [Zod](#)
- **PDF:** [@react-pdf/renderer](#)

Pré-requisitos

Antes de começar, certifique-se de ter instalado:

- Node.js 18+ ou superior.
- Uma conta no [Supabase](#).

Instalação e Configuração

1. Clone o repositório:

```
git clone [https://github.com/seu-usuario/sso-consultoria.git](https://github.com/seu-usuario/sso-consultoria.git)
cd sso-consultoria
```

2. Instale as dependências:

```
npm install
# ou
yarn install
```

3. Configuração do Ambiente (.env): Crie um arquivo `.env.local` na raiz do projeto e adicione as chaves do seu projeto Supabase:

```
NEXT_PUBLIC_SUPABASE_URL=sua_url_aqui
NEXT_PUBLIC_SUPABASE_ANON_KEY=sua_key_aqui
```

4. Banco de Dados: Acesse o SQL Editor do Supabase e execute o script de migração localizado em `/docs/database/schema.sql` (ou consulte a documentação técnica).

5. Rodando o projeto:

```
npm run dev
```

O sistema estará acessível em `http://localhost:3000`.

Estrutura do Projeto

A organização de pastas segue o padrão de **Features** para melhor escalabilidade:

```
/src
  /app                                # Rotas (Next.js App Router)
    /(auth)                          # Telas de Login/Recuperação
    /(dashboard)                     # Área logada (Admin, Comercial, Operacional, Financeiro)
  /components
    /ui                              # Componentes base (shadcn/ui)
    /features                        # Componentes específicos de negócio (ex: AsoForm)
  /lib
    /validations                     # Schemas do Zod
    supabase.ts                      # Cliente de conexão
  /services                          # Server Actions (Lógica de Banco de Dados)
  /types                             # Definições de Tipos TypeScript
```

Documentação

A documentação completa das regras de negócio e modelagem de dados encontra-se na pasta /docs :

- [Modelagem de Dados e Banco \(PDF\)](#)
- [Especificação Funcional - Administrativo](#)
- [Especificação Funcional - Comercial](#)
- [Especificação Funcional - Operacional](#)
- [Especificação Funcional - Financeiro](#)

Contribuição

Este é um projeto proprietário. Pull requests são aceitos apenas da equipe interna.

Licença

Copyright © 2026 Gaj. Todos os direitos reservados.