

# RESUMO COMPLETO - Sistema Gestão de Obras Solar

## CONTEXTO DO PROJETO

Estamos desenvolvendo um sistema web completo para gestão de obras de energia solar, com controle de colaboradores, clientes, ordens de serviço, equipes, EPIs, certificados, DRE por obra, etc.

---

## CREDENCIAIS SUPABASE

- **Project URL:** <https://ebpxqmakimkvqoqwfeeh.supabase.co>
  - **Anon Key:**  
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6ImVicHhxWFraW1rdnFvcXdmZWV0Iwiwcm9sZSI6ImFub24iLCJpYXQiOjE3NjQ0NDYzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzM30.JLRUbRym-Z3qDiOwtiaFa9\_7520aP7q4BpY5OLIICFY
- 

## BANCO DE DADOS (30 TABELAS CRIADAS)

Tabelas principais:

- colaboradores, funcoes, equipes, equipe\_membros
  - clientes, servicos, veiculos
  - ordens\_servico, os\_servicos, os\_colaboradores, os\_custos\_extras
  - certificados, epis
  - usuarios, perfis, permissoes
  - checklists, checklist\_itens, checklist\_respostas
  - empresas\_contratantes
  - Views: vw\_dre\_os (DRE por obra)
- 

## AMBIENTE DE DESENVOLVIMENTO

- **Localização do projeto:** ~/Desktop/gestao-obra-solar
- **Node.js:** v24.11.1
- **Framework:** React + Vite
- **Porta:** localhost:3000
- **Deploy:** Vercel (<https://gestao-obra-solar.vercel.app>)

## Dependências instaladas:

- react, react-dom, react-router-dom
  - @supabase/supabase-js
  - @tanstack/react-query
  - lucide-react (ícones)
  - date-fns (datas)
  - recharts (gráficos)
  - react-hook-form, @hookform/resolvers, zod (formulários)
  - react-hot-toast (notificações)
  - tailwindcss, postcss, autoprefixer
- 

## ESTRUTURA DE PASTAS

```
gestao-oberas-solar/
├── src/
│   ├── components/
│   │   └── PagePlaceholder.jsx
│   ├── contexts/
│   │   └── AuthContext.jsx
│   ├── layouts/
│   │   ├── AuthLayout.jsx
│   │   └── MainLayout.jsx
│   └── lib/
│       └── supabase.js
└── pages/
    ├── auth/
    │   ├── Login.jsx
    │   ├── Register.jsx
    │   └── ForgotPassword.jsx
    ├── cadastros/
    │   ├── Colaboradores.jsx ✅ COMPLETO
    │   ├── ColaboradorForm.jsx ✅ COMPLETO
    │   ├── Clientes.jsx ✅ COMPLETO
    │   ├── ClienteForm.jsx ✅ COMPLETO
    │   ├── Equipes.jsx (placeholder)
    │   ├── EquipeForm.jsx (placeholder)
    │   ├── Funcoes.jsx (placeholder)
    │   ├── Servicos.jsx (placeholder)
    │   ├── Veiculos.jsx (placeholder)
    │   └── EmpresasContratantes.jsx (placeholder)
```

```
|- operacao/
|   |- OrdensServico.jsx (placeholder)
|   |- OrdemServicoForm.jsx (placeholder)
|   |- OrdemServicoDetalhes.jsx (placeholder)
|   |- Calendario.jsx (placeholder)
|   |- Checklists.jsx (placeholder)
|   |   └── ChecklistForm.jsx (placeholder)
|   └── relatorios/
|       |- Relatorios.jsx (placeholder)
|       └── DREObras.jsx (placeholder)
|   └── configuracoes/
|       |- Configuracoes.jsx (placeholder)
|       |- Usuarios.jsx (placeholder)
|       └── Perfis.jsx (placeholder)
|   └── Dashboard.jsx ✅ COMPLETO
|- App.jsx ✅ COMPLETO (todas as rotas)
|- main.jsx ✅
└── index.css ✅
index.html
vite.config.js
tailwind.config.js
postcss.config.js
package.json
```

## O QUE JÁ ESTÁ FUNCIONANDO

### 1. Sistema de Autenticação

- Login/Registro/Logout
- Proteção de rotas
- Contexto de autenticação

### 2. Layout Principal

- Sidebar com menu expansível
- Header com usuário logado
- Responsivo (mobile/desktop)

### 3. Dashboard

- Cards de estatísticas (OS, Colaboradores, Clientes, Obras Hoje)
- Ações rápidas

### 4. Cadastro de Colaboradores (COMPLETO)

- Listagem com busca

- Formulário completo (dados pessoais, contato, endereço)
- Gerenciamento de Certificados (NR-10, NR-35, etc)
- Gerenciamento de EPIs
- Edição e exclusão
- Status de vencimentos (Regular/Vencendo/Vencido)

## 5. Cadastro de Clientes (COMPLETO)

- Listagem com busca
  - Formulário completo (PF/PJ, contato, endereço)
  - Edição e exclusão
- 

## PROBLEMA CONHECIDO

O navegador normal fica com loading infinito na tela de login devido a cache. **Solução temporária:** Usar aba anônima (Cmd + Shift + N) para testar.

---

## PRÓXIMOS PASSOS A DESENVOLVER

1. Formulário de Funções (funções com valor de diária)
  2. Formulário de Equipes
  3. Formulário de Ordens de Serviço
  4. Calendário de obras
  5. Relatório DRE por obra
  6. Checklists
  7. Gestão de Usuários e Perfis
- 

## COMANDOS ÚTEIS

Rodar o projeto localmente:

```
bash  
cd ~/Desktop/gestao-obras-solar  
npm run dev
```

Fazer deploy na Vercel:

```
bash
```

```
cd ~/Desktop/gestao-obras-solar
```

```
vercel --prod
```

## Verificar estrutura:

```
bash
```

```
ls -la src/pages/cadastros/
```

## USUÁRIO DO SISTEMA

- **E-mail:** [carlos@criteriaenergia.com.br](mailto:carlos@criteriaenergia.com.br)
- **URL produção:** <https://gestao-obras-solar.vercel.app>

## INSTRUÇÕES PARA NOVO CHAT

Ao iniciar um novo chat, cole este resumo e diga:

"Estou continuando o desenvolvimento do sistema Gestão de Obras Solar. Aqui está o resumo do que já foi feito. Quero continuar desenvolvendo as próximas funcionalidades."

O Claude terá todo o contexto para continuar de onde paramos!