

OS PEDIDOS PRONTOS PRA IA MONTAR O TEU *Supabase*

O Supabase não é só banco de dados. Junto vêm o login dos usuários, o lugar dos arquivos e as funções no servidor. Aqui tem 4 pedidos prontos pra colar no Lovable, no Base44 ou no Claude Code e ligar cada parte do jeito seguro, e o quadro do plano grátis e do Pro.

PASSO 01 / 05

Liga o *login* por e-mail OFICIAL

O login por e-mail e senha já vem ligado nos projetos do Supabase, e por padrão a pessoa precisa confirmar o e-mail antes de entrar. O pedido abaixo faz a IA montar as telas e ligar tudo no teu app.

PEDIDO PRONTO

Liga o login por e-mail e senha do Supabase Auth no meu app. Quero as telas de criar conta, entrar, sair e esqueci minha senha, em português. Depois do login, a pessoa vai pra página inicial do app; sem login, ela não vê nenhuma página interna. Usa só a chave pública (publishable/anon) no app e me explica em português simples cada arquivo que tu mudou.

supabase.com/docs/guides/auth/passwords

PASSO 02 / 05

Cria o lugar dos *arquivos*, com acesso só do dono OFICIAL

O Storage guarda arquivo e foto. Sem regra, ninguém consegue subir nada; com a regra certa, cada pessoa vê e mexe só nos arquivos dela.

PEDIDO PRONTO

Cria no Supabase Storage um bucket privado chamado arquivos pra cada usuário guardar as fotos e os documentos dele. Escreve as policies na tabela storage.objects pra que cada pessoa logada só consiga enviar, ver, trocar e apagar os próprios arquivos (comparando o owner_id com o id do usuário logado). Ninguém sem login acessa nada. Me mostra as policies em SQL e me explica cada uma em português simples antes de aplicar.

supabase.com/docs/guides/storage/security/access-control

PASSO 03 / 05

Liga o RLS: cada usuário vê *só o que é dele* OFICIAL

RLS (segurança por linha) é a portaria do banco. Ligado, ninguém lê nada pela chave pública até existir uma regra. A regra mais comum é: cada usuário vê só as linhas dele.

PEDIDO PRONTO

Confere todas as tabelas do meu projeto no Supabase. Em cada uma, liga o Row Level Security (RLS) e cria as policies pra que cada usuário logado só consiga ver, criar, editar e apagar as linhas que são dele (comparando a coluna do dono com `auth.uid()`). Se alguma tabela não tem coluna de dono, me avisa antes de mudar. Me mostra o SQL de cada tabela e me explica em português simples o que cada policy libera.

supabase.com/docs/guides/database/postgres/row-level-security

PASSO 04 / 05

Tira a *chave de API* do app e passa pra uma função no servidor

OFICIAL

Chave de API é a senha que o app usa pra falar com outras ferramentas (IA, pagamento, e-mail). Ela não pode ficar no código que vai pro navegador. No Supabase, ela fica guardada como segredo e só a função no servidor (Edge Function) lê.

PEDIDO PRONTO

Procura no meu app qualquer chave de API secreta que esteja no código que vai pro navegador (IA, pagamento, e-mail ou a `service_role` do Supabase). Move cada chamada que usa essas chaves pra uma Supabase Edge Function, guarda as chaves como secrets do projeto (`supabase secrets set`) e lê com `Deno.env.get` dentro da função. O app passa a chamar a função, nunca a ferramenta direto. Me lista o que tu mudou e me lembra de trocar as chaves que já ficaram expostas.

Cuidado: A chave `service_role` passa por cima de todas as regras de acesso: ela só pode ficar no servidor, nunca no navegador.

supabase.com/docs/guides/functions/secrets

QUADRO DOS PLANOS

O plano grátis e o Pro, com os poréns

OFICIAL

Grátis (Free): até 2 projetos ativos, 500 MB de banco por projeto, 50 mil usuários ativos por mês, 1 GB de arquivos. **Porém:** o projeto grátis pausa depois de 1 semana sem uso.

Pro: a partir de US\$ 25 por mês, com o primeiro projeto incluído e US\$ 10 por mês de crédito de servidor. Cada projeto a mais paga o servidor dele (a partir de US\$ 10 por mês): 2 projetos dão US\$ 35 por mês.

supabase.com/pricing

O QUE A GENTE CONFERIU EM 16/10/2026	FONTE
Grátis: limite de 2 projetos ativos, 500 MB de banco, 50.000 usuários ativos por mês, 1 GB de arquivos	supabase.com/pricing
Projeto grátis pausa depois de 1 semana sem uso ("paused after 1 week of inactivity")	supabase.com/pricing
Pro a partir de US\$ 25/mês, primeiro projeto incluído, projetos a mais a partir de US\$ 10/mês, US\$ 10/mês de crédito de servidor	supabase.com/pricing
Login por e-mail ligado por padrão, com confirmação de e-mail nos projetos hospedados	supabase.com/docs/guides/auth/passwords
Policies do Storage na tabela storage.objects, comparando o dono com o usuário logado	supabase.com/docs/guides/storage/security/access-control
Com RLS ligado, nada fica acessível pela chave pública até existir uma policy; auth.uid() é o id de quem pede	supabase.com/docs/guides/database/postgres/row-level-security
Segredos com supabase secrets set, lidos com Deno.env.get; a service_role passa por cima do RLS e nunca vai pro navegador	supabase.com/docs/guides/functions/secrets

Quer aprender a usar tudo isso do zero, em português? **Entra na lista da comunidade Prosperando com IA:** aliceallgayer.com.br/lista?de=prosperando