

Prompt Detalhado para App PWA - MinhaGrana

Crie um aplicativo PWA (Progressive Web App) completo e responsivo chamado "MinhaGrana", para cadastro, visualizacao e analise de despesas pessoais. O app deve utilizar PostgreSQL como banco de dados principal e oferecer uma experiencia fluida com suporte offline, autenticao segura e dashboard analitico.

Funcionalidades principais

1. Autenticacao de Usuario

- Tela de login e cadastro de conta (e-mail e senha).
- Armazenar usuarios na tabela users:

```
CREATE TABLE users (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  email TEXT UNIQUE NOT NULL,  
  password_hash TEXT NOT NULL,  
  created_at TIMESTAMPTZ DEFAULT NOW()  
);
```

- Usar JWT ou sessao segura para proteger rotas e recursos privados.
- Permitir logout e recuperacao de senha (opcional).

2. Cadastro de Despesas

- Tela para cadastrar despesas com os campos:
 - nome (string)
 - valor (float)
 - data (date)
 - categoria (string escolha ou adicao livre)
 - tipo: unica, semanal ou mensal (recorrencia)
 - observacoes (texto opcional)

```
CREATE TABLE expenses (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  user_id UUID REFERENCES users(id),  
  name TEXT NOT NULL,  
  amount NUMERIC(10, 2) NOT NULL,  
  date DATE NOT NULL,  
  category TEXT,  
  recurrence TEXT CHECK (recurrence IN ('none', 'weekly', 'monthly')) DEFAULT 'none',  
  notes TEXT,  
  created_at TIMESTAMPTZ DEFAULT NOW())
```

Prompt Detalhado para App PWA - MinhaGrana

);

3. Categorias Personalizaveis

- Tela para gerenciar categorias por usuario.
- CRUD completo: adicionar, editar, excluir.

```
CREATE TABLE categories (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  user_id UUID REFERENCES users(id),  
  name TEXT NOT NULL,  
  created_at TIMESTAMPTZ DEFAULT NOW()  
);
```

4. Dashboard de Acompanhamento

- Grafico de pizza por categoria (mes atual)
- Grafico de linha da evolucao diaria no mes
- Indicadores: total mensal, total semanal, projecao mensal
- Filtros por periodo: mes, semana, personalizado
- Dados calculados com SQL otimizado ou views

5. Recorrencia de Lancamentos

- Despesas recorrentes devem ser clonadas no inicio de cada periodo.
- Ex: "Academia R\$120,00 no dia 10 de cada mes".

6. Exportacao de Dados

- Botao para exportar despesas em CSV ou Excel.

Recursos Tecnicos

- Frontend: React ou outro SPA compativel com PWA
- Backend: Node.js ou equivalente com PostgreSQL
- Estilo: TailwindCSS ou Material UI
- Graficos: Chart.js ou Recharts
- Offline: Cache local + Service Worker
- Responsivo e mobile-first