

# CHECKPESO — Project Files

This document contains:

- UI examples and component templates using **shadcn/ui** and **Next.js (App Router)**
  - Supabase table structure (SQL DDL)
  - `README.md` (professional) for the GitHub repo
  - The full project prompt in **.txt** and **README.md** formats (for copy/paste)
- 

## 1. UI (shadcn) — Next.js + TypeScript snippets

These are base components and layout to get a complete UI using shadcn/ui, Tailwind CSS, and the Next.js App Router. Use them as a starting point and extend them per your needs.

### 1.1 Install (quick)

```
# from project root
npm install next react react-dom typescript tailwindcss postcss autoprefixer
npm install @supabase/supabase-js axios zod react-hook-form zustand
# shadcn stack
# follow shadcn/ui install instructions: https://ui.shadcn.com/
npm install @shadcn/ui
# charts
npm install recharts
# exports
npm install xlsx jspdf
```

### 1.2 Project layout (excerpt)

```
/app
  /dashboard
    page.tsx
  /calculos
    page.tsx
  /relatorios
    page.tsx
  /configuracoes
    page.tsx
  layout.tsx
  globals.css
/components
  ui/
    button.tsx
```

```

    card.tsx
    page-header.tsx
    sidebar.tsx
  cards/
    kpi-card.tsx
  charts/
    line-chart.tsx
    bar-chart.tsx
/lib
  supabaseClient.ts
  sheets.ts
  calculos.ts
/sql
  supabase_schema.sql
/public
  logo.png
/README.md

```

### 1.3 components/ui/sidebar.tsx (shadcn + next/link)

```

'use client'
import Image from 'next/image'
import Link from 'next/link'
import { Sidebar as ShadSidebar } from '@shadcn/ui' // illustrative

export default function Sidebar(){
  return (
    <aside className="w-72 bg-white dark:bg-slate-900 h-screen sticky top-0">
      <div className="p-4 flex items-center gap-3">
        <Image src="/logo.png" alt="logo" width={40} height={40} />
        <div className="font-bold">CHECKPESO - GDM</div>
      </div>

      <nav className="p-4 flex flex-col gap-2">
        <Link className="p-2 rounded hover:bg-slate-100" href="/
dashboard">Dashboard</Link>
        <Link className="p-2 rounded hover:bg-slate-100" href="/
calculos">Cálculo</Link>
        <Link className="p-2 rounded hover:bg-slate-100" href="/
relatorios">Relatórios</Link>
        <Link className="p-2 rounded hover:bg-slate-100" href="/
configuracoes">Configurações</Link>
      </nav>
    </aside>
  )
}

```

## 1.4 components/ui/page-header.tsx

```
'use client'
import React from 'react'

export default function PageHeader({ title, subtitle }: {title:string,
subtitle?:string}){
  return (
    <header className="p-4 border-b bg-white dark:bg-slate-900">
      <div className="max-w-7xl mx-auto">
        <h1 className="text-2xl font-semibold">{title}</h1>
        {subtitle && <p className="text-sm text-slate-500">{subtitle}</p>}
      </div>
    </header>
  )
}
```

## 1.5 components/cards/kpi-card.tsx

```
'use client'
import React from 'react'

export default function KpiCard({title, value, delta}:
{title:string,value:string|number,delta?:string}){
  return (
    <div className="p-4 bg-white rounded-lg shadow-sm">
      <div className="text-xs text-slate-500">{title}</div>
      <div className="mt-2 text-2xl font-bold">{value}</div>
      {delta && <div className="text-sm text-slate-400">{delta}</div>}
    </div>
  )
}
```

## 1.6 Chart wrapper example ( components/charts/line-chart.tsx )

```
'use client'
import { LineChart, Line, XAxis, YAxis, Tooltip, ResponsiveContainer } from
'recharts'

export default function SimpleLine({data}:{data:any[]}){
  return (
    <div style={{width:'100%', height:300}}>
      <ResponsiveContainer>
        <LineChart data={data}>
          <XAxis dataKey="date" />

```

```

        <YAxis />
        <Tooltip />
        <Line type="monotone" dataKey="perda_kg" stroke="#10b981"
strokeWidth={2} />
      </LineChart>
    </ResponsiveContainer>
  </div>
)
}

```

## 2. Supabase: tabela SQL (estrutura)

Run these SQL statements in Supabase SQL Editor. Adjust roles/policies as needed.

```

-- 1. filiais
CREATE TABLE public.filiais (
  id uuid DEFAULT gen_random_uuid() PRIMARY KEY,
  nome text NOT NULL,
  codigo text,
  created_at timestamptz DEFAULT now()
);

-- 2. config_tabelas (S1..S4 entries)
CREATE TABLE public.config_tabelas (
  id uuid DEFAULT gen_random_uuid() PRIMARY KEY,
  modelo text NOT NULL, -- 'S1' 'S2' 'S3' 'S4'
  quantidade_min bigint,
  quantidade_max bigint,
  quantidade_retorno bigint,
  descricao text,
  updated_at timestamptz DEFAULT now()
);

-- 3. usuarios
CREATE TABLE public.usuarios (
  id uuid DEFAULT auth.uid() PRIMARY KEY,
  email text,
  nome text,
  role text DEFAULT 'user',
  created_at timestamptz DEFAULT now()
);

-- 4. registros_peso
CREATE TABLE public.registros_peso (
  id uuid DEFAULT gen_random_uuid() PRIMARY KEY,
  data_registro date NOT NULL,
  filial_id uuid REFERENCES public.filiais(id) ON DELETE SET NULL,

```

```

quantidade_recebida bigint NOT NULL,
peso_liquido_por_cx numeric(12,4) NOT NULL,
modelo_tabela text NOT NULL,
quantidade_tabela bigint NOT NULL,
peso_bruto_analise numeric(12,4) NOT NULL,
tara_caixa numeric(12,4) NOT NULL,
fornecedor text,
nota_fiscal text,
observacoes text,
peso_programado numeric(18,4),
peso_analise numeric(18,4),
peso_real numeric(18,4),
perda_kg numeric(18,4),
perda_cx numeric(18,4),
divergencia_percent numeric(10,4),
created_at timestamptz DEFAULT now()
);

-- 5. logs_envio_whatsapp
CREATE TABLE public.logs_envio_whatsapp (
  id uuid DEFAULT gen_random_uuid() PRIMARY KEY,
  registro_id uuid REFERENCES public.registros_peso(id) ON DELETE CASCADE,
  telefone text,
  mensagem text,
  enviado boolean DEFAULT false,
  created_at timestamptz DEFAULT now()
);

-- Indexes (useful)
CREATE INDEX idx_registros_data ON public.registros_peso(data_registro);
CREATE INDEX idx_registros_filial ON public.registros_peso(filial_id);

```

### 3. README.md (professional) — full content

# CHECKPESO – GDM

Aplicativo web para controle e cálculo de pesagens de cargas.

## Visão Geral

CHECKPESO é um sistema construído com Next.js (App Router) que registra pesagens, executa cálculos para detectar perdas e exporta relatórios para Google Sheets, XLS, PDF e WhatsApp. O projeto integra Supabase como banco de dados e usa a conta de serviço do Google para gravar backups em uma planilha.

## Tecnologias

- Next.js (App Router)
- React + TypeScript

- Tailwind CSS + shadcn/ui
- Supabase (DB + Auth + Realtime)
- Google Sheets API (Service Account)
- Recharts (gráficos)
- SheetJS, jsPDF (exportações)

## ## Estrutura do Projeto

Descrevemos a estrutura de pastas e arquivos no repositório.

## ## Instalação

### 1. Clone o repositório

```
``bash
git clone https://github.com/obedysjr-sys/apppesagem.git
cd apppesagem
```

### 2. Instale dependências

```
npm install
```

3. Configurar variáveis de ambiente Crie um arquivo `.env.local` com as variáveis necessárias (exemplo abaixo). **NÃO** comite suas chaves!

```
NEXT_PUBLIC_SUPABASE_URL=your_supabase_url
NEXT_PUBLIC_SUPABASE_ANON_KEY=your_supabase_anon_key
SUPABASE_SERVICE_ROLE=your_service_role_key # keep secret, do not commit
GOOGLE_SERVICE_ACCOUNT_EMAIL=checkpeso@checkpeso.iam.gserviceaccount.com
GOOGLE_SERVICE_ACCOUNT_JSON_PATH=./checkpeso-1533bb5e50f1.json
NEXT_PUBLIC_GOOGLE_SHEET_ID=10RITRzBYzQ5x6-DZxdJjX5Txu0AdIwMC0oKpJmXidY4
```

### 1. Rodar em desenvolvimento

```
npm run dev
```

## Banco de dados (Supabase)

Execute o arquivo `sql/supabase_schema.sql` no SQL Editor do Supabase para criar as tabelas necessárias.

## Deploy

- Configure as mesmas variáveis de ambiente no Vercel (ou outro provedor).

## Segurança

- **Nunca** envie o `SUPABASE_SERVICE_ROLE` em repositórios públicos.
- Mantenha a chave JSON da conta de serviço em local seguro e ignorado pelo git (`.gitignore`).

## Funcionalidades Principais

- Registro de pesagens com cálculo automático
- Dashboard com KPIs e gráficos interativos
- Exportação XLS/PDF/HTML
- Envio de resumo por WhatsApp
- CRUD para tabelas (S1..S4)

## Contribuição

- Abra issues e PRs no repositório
- Siga o padrão de commits e mantenha testes

## Licença

MIT

---

# 4. PROMPT COMPLETO (txt) – versão pronta para colar

```text

PROMPT: CHECKPESO – GDM (Next.js)

Objetivo: Gerar um aplicativo Next.js responsivo com integração Supabase + Google Sheets, interface com shadcn/ui, cálculos de pesagem, registros, relatórios (XLS, PDF, HTML) e envio via WhatsApp.

[...use the full prompt from the earlier assistant message, updated for Next.js...]

(Include all requirements: project repo, supabase keys stored in env, google service account JSON path, fuso horario America/Bahia, default filial, tabela S4 rules, calculations formulas, UI components list, charts and KPIs, CSV/PDF/XLS generation, offline mode, realtime via Supabase)

Note: The `txt` block above is a placeholder; the full prompt is included below in the README-ready block for easy copy/paste.

---

## 5. README-ready PROMPT (readme.md) — full content for your repo (copy/paste)

# Prompt: CHECKPESO – GDM (Next.js)

## Objetivo

Criar um aplicativo web responsivo (Next.js + TypeScript) com as seguintes características:

- Integração com Supabase (principal) e Google Sheets (backup)
- UI completa baseada em `shadcn/ui` e Tailwind CSS
- Páginas: Dashboard, Cálculo, Relatórios, Configurações
- Regras da Tabela S4 e cálculos automatizados
- Exportação XLS/PDF/HTML e compartilhamento via WhatsApp
- Deploy no Vercel / ambiente com variáveis de ambiente

## Repositório

<https://github.com/obedysjr-sys/apppesagem.git>

## Variáveis de ambiente

Preencha o arquivo `.env.local` com:

NEXT\_PUBLIC\_SUPABASE\_URL=... NEXT\_PUBLIC\_SUPABASE\_ANON\_KEY=... SUPABASE\_SERVICE\_ROLE=...

# usar com cuidado

GOOGLE\_SERVICE\_ACCOUNT\_EMAIL=checkpeso@checkpeso.iam.gserviceaccount.com

GOOGLE\_SERVICE\_ACCOUNT\_JSON\_PATH=./checkpeso-1533bb5e50f1.json

NEXT\_PUBLIC\_GOOGLE\_SHEET\_ID=10RITRzBYzQ5x6-DZxdJjX5TxuOAdIwMCOoKpJmXidY4

## Requisitos funcionais (resumo)

- Default filial: TRIELO CD SIMÕES FILHO BA
- Data default: hoje (timezone America/Bahia)
- Modelo S4 default
- Cálculos: programado, analise, real, perda\_kg, perda\_cx, divergencia%
- Exportações e relatórios com paginação
- CRUD de tabelas e logs de envio

## Observações de segurança

- Não commitar chaves privadas
- Manter o `checkpeso-1533bb5e50f1.json` em local seguro e ignorado pelo git

---

## 6. Próximos passos (como eu posso te ajudar agora)

- Posso gerar os arquivos reais (components, app pages) para você no formato de código pronto.
- Posso gerar o arquivo SQL (sql/supabase\_schema.sql) e adicioná-lo ao repositório.
- Posso gerar o README.md como arquivo separado para download.

---

*Fim do arquivo.*