

# Taller 3 - Web Movil

---

## Grupo y equipo

- Grupo 1 - Equipo: SmartCoders

## Integrantes

- Bastian Salinas - 21.848.994-K
- Benjamin Cuello - 21.682.135-1
- Benjamin Salas - 21.758.667-4
- Tomas Guerra - 21.664.344-5

## Descripcion del proyecto

Aplicacion web movil construida con Next.js 14 que permite visualizar registros almacenados en una base de datos PostgreSQL mediante un dashboard operativo. Se utiliza Prisma como ORM y Redux Toolkit para manejar el estado global.

El dashboard incluye:

- Tabla de registros con filtros por categoria, estado, rango de fechas, orden y busqueda de texto.
- Metricas agregadas (cantidad, activos, ingresos, avance).
- Al menos cinco graficos (tendencia, categorias, estado operativo, radar de desempeno, velocidad semanal) usando Recharts.
- Vista de detalle por registro, disponible como modal en el dashboard y como pagina dedicada [/records/\[id\]](/records/[id]).
- Tema claro y oscuro con selector en el header, persistente entre recargas.
- Diseno mobile first y responsive para movil, tablet y escritorio.

## Tecnologias utilizadas

### Frontend y arquitectura

- Next.js 14 (App Router)
- React
- TailwindCSS
- Recharts para graficos

### Backend y base de datos

- API REST dentro de Next.js (rutas </app/api/records> y </backend/api/records>)
- Prisma ORM
- PostgreSQL

## Instalacion y ejecucion local

## Requisitos previos

- Node.js 20 o superior
- PostgreSQL en ejecucion (local o en contenedor)
- Variable de entorno `DATABASE_URL` apuntando a la base de datos, por ejemplo:  
`postgresql://postgres:postgres@localhost:5432/taller3_db?schema=public`

## Ejecucion con Docker

Este proyecto incluye configuracion para Docker y Docker Compose.

1. Asegurarse de tener Docker y Docker Compose instalados.
2. Desde la raiz del proyecto, ejecutar:

```
docker compose up --build
```

Esto levantara:

- Un contenedor `db` con PostgreSQL.
- Un contenedor `web` con la aplicacion Next.js, que sincroniza el esquema con Prisma, ejecuta el seed y levanta el modo desarrollo.

3. Acceder al dashboard en:

```
http://localhost:3000
```

## Pasos para entorno local sin Docker

1. Clonar el repositorio:

```
git clone <url-del-repositorio>
cd webmovil-taller3
```

2. Crear el archivo `.env` en la raiz del proyecto (si no existe) con la variable de conexion:

```
DATABASE_URL="postgresql://postgres:postgres@localhost:5432/taller3_db?
schema=public"
```

3. Instalar dependencias:

```
npm install
```

4. Sincronizar el esquema de la base de datos con Prisma:

```
npm run prisma:sync
```

5. Ejecutar el seed para cargar datos de ejemplo:

```
npm run prisma:seed
```

6. Levantar el servidor de desarrollo:

```
npm run dev
```

7. Abrir el navegador en:

```
http://localhost:3000
```