

Manual de Instalación y Ejecución - Earth Vibe

Esta guía detalla los pasos necesarios para desplegar el entorno de desarrollo completo de Earth Vibe (Backend, Frontend y Mobile).

1. Requisitos Previos

Antes de comenzar, asegúrate de tener instalado lo siguiente:

- **Node.js:** v18 o superior (Recomendado v20 LTS).
- **npm:** Gestor de paquetes de Node.
- **Git:** Para clonar el repositorio.
- **Flutter SDK:** (Opcional, solo si vas a ejecutar la app móvil).
- **Base de Datos:**
 - Acceso a una instancia de **MongoDB** (local o Atlas).
 - (Opcional) SQLite se maneja localmente mediante archivos.

2. Estructura del Proyecto

El repositorio contiene las siguientes carpetas principales:

- **src/backend:** Servidor API (Express/TypeScript).
- **src/frontend:** Aplicación Web (React/Vite).
- **mobile:** Aplicación Móvil (Flutter).

3. Instalación y Ejecución

A. Backend (Servidor API)

El backend es el núcleo del sistema. Debe estar corriendo para que el frontend y la app móvil funcionen correctamente.

1. **Navegar al directorio:**
`cd src/backend`
2. **Instalar dependencias:**
`npm install`
3. **Configuración de Entorno:**
 - Crea un archivo `.env` en la raíz de `src/backend`.
 - Define las variables necesarias (ejemplo):
`PORT=3000`
`MONGODB_URI=mongodb://localhost:27017/earthvibe`

```
JWT_SECRET=tu_secreto_super_seguro  
FRONTEND_URL=http://localhost:5173
```

4. **Ejecutar en modo desarrollo:**

```
npm run dev
```

- El servidor iniciará en `http://localhost:3000`.
- Verás logs indicando la conexión a la base de datos y el inicio de servicios.

B. Frontend (Web App)

Interfaz web para usuarios y administradores.

1. **Navegar al directorio:**

```
cd src/frontend
```

2. **Instalar dependencias:**

```
npm install
```

3. **Ejecutar en modo desarrollo:**

```
npm run dev
```

- La aplicación web estará disponible en `http://localhost:5173` (por defecto).

C. Mobile (App Flutter)

Aplicación para dispositivos móviles.

1. **Navegar al directorio:**

```
cd mobile
```

2. **Obtener dependencias:**

```
flutter pub get
```

3. **Ejecutar la aplicación:**

- Conecta un dispositivo físico o inicia un emulador.
- Ejecuta:

```
flutter run
```

D. Vibe Pod (IoT / Raspberry Pi)

Software de kiosco para el módulo de reciclaje.

1. **Requisitos del Sistema (Raspberry Pi OS):**

```
sudo apt-get update  
sudo apt-get install python3-tk python3-pil.imagetk
```

2. **Navegar al directorio:**

```
cd src/vibepod
```

3. **Instalar dependencias Python:**

```
pip install -r requirements.txt
```

4. **Configuración:**

- Copia el archivo de ejemplo: `cp .env.example .env`
- Edita `.env` y asegura que `API_URL` apunte a la IP de tu servidor Backend (no uses `localhost` si corres en dispositivos distintos).

5. **Ejecutar:**

```
python main.py
```

- La aplicación se abrirá en pantalla completa. Presiona F11 o ESC para salir.

4. Comandos Útiles

| Módulo | Comando | Descripción |
|----------|--------------------------------|-------------------------------------------------------------------------|
| Backend | <code>npm run build</code> | Compila el código TypeScript a JavaScript (carpeta <code>dist</code>). |
| Backend | <code>npm start</code> | Ejecuta el servidor compilado (producción). |
| Backend | <code>npm run gen:proto</code> | Genera archivos JS/TS a partir de definiciones Protobuf. |
| Frontend | <code>npm run build</code> | Genera la versión de producción de la web. |
| Frontend | <code>npm run preview</code> | Vista previa local de la build de producción. |

5. Solución de Problemas

- **Error de conexión a BD:** Verifica que tu URI de MongoDB en el `.env` sea correcta y que el servicio de base de datos esté activo.
- **Puerto ocupado:** Si el puerto 3000 o 5173 está en uso, cambia el puerto en el `.env` o en

la configuración de Vite (`vite.config.ts`).