Guía Completa de Instalación y Configuración del Proyecto Synaps

Este documento describe detalladamente todos los pasos necesarios para preparar el entorno de desarrollo del proyecto Synaps, incluyendo instalación de dependencias (Node.js, React, PHP, Composer), configuración de Docker, levantamiento de contenedores y conexión entre frontend y backend.

# 1. Requisitos Previos

* Git instalado: https://git-scm.com/
* Docker y Docker Compose instalados: https://docs.docker.com/desktop/
* Node.js (v18+) y npm: https://nodejs.org/
* Composer: https://getcomposer.org/

# 2. Clonar Repositorio

* git clone https://github.com/tuusuario/synaps.git
* cd synaps

# 3. Configuración del Backend (Laravel)

* Ubicación: Synaps-back/laravel
* Crear el archivo .env basado en .env.example y configurar las variables como se indicó previamente (APP\_URL, DB\_HOST, etc.).
* Generar la APP\_KEY: docker exec -it synaps-back php artisan key:generate
* Dar permisos correctos:
* docker exec -it synaps-back chown -R www-data:www-data /var/www/html
* docker exec -it synaps-back chmod -R 755 /var/www/html
* Limpiar cachés:
* docker exec -it synaps-back php artisan route:clear
* docker exec -it synaps-back php artisan config:cache

# 4. Configuración del Frontend (React)

* Ubicación: carpeta raíz del frontend (crear o clonar si es necesario)
* npm install
* npm start (luego de verificar que el backend esté disponible en http://localhost:8010)

# 5. Configuración de Docker

* Verifica que docker-compose.yml esté correctamente configurado con puertos únicos:
* Laravel backend: 8010:80
* MariaDB: 3307:3306
* Redis: 6380:6379
* PhpMyAdmin: 8083:80
* Levantar entorno:
* docker-compose up --build
* En caso de errores previos:
* docker-compose down --volumes --remove-orphans
* docker volume prune (con precaución)

# 6. Verificación de Rutas

* docker exec -it synaps-back php artisan route:list
* Asegurarse que la ruta /api/hello aparezca en la lista.

# 7. Acceso y Pruebas

* Frontend: http://localhost:3000
* Backend API: http://localhost:8010/api/hello
* PhpMyAdmin: http://localhost:8083 (root/rootpassword)

# 8. Consideraciones Finales

* Evita usar `php artisan serve` cuando Laravel se ejecuta bajo Apache en Docker.
* Revisa que cada contenedor use puertos únicos para evitar conflictos con otros proyectos.
* El Redis WebSocket Bridge requiere que exista un package.json válido en su carpeta para iniciar correctamente.