

# Estructura de Carpetas Profesional para Krypton Inventory

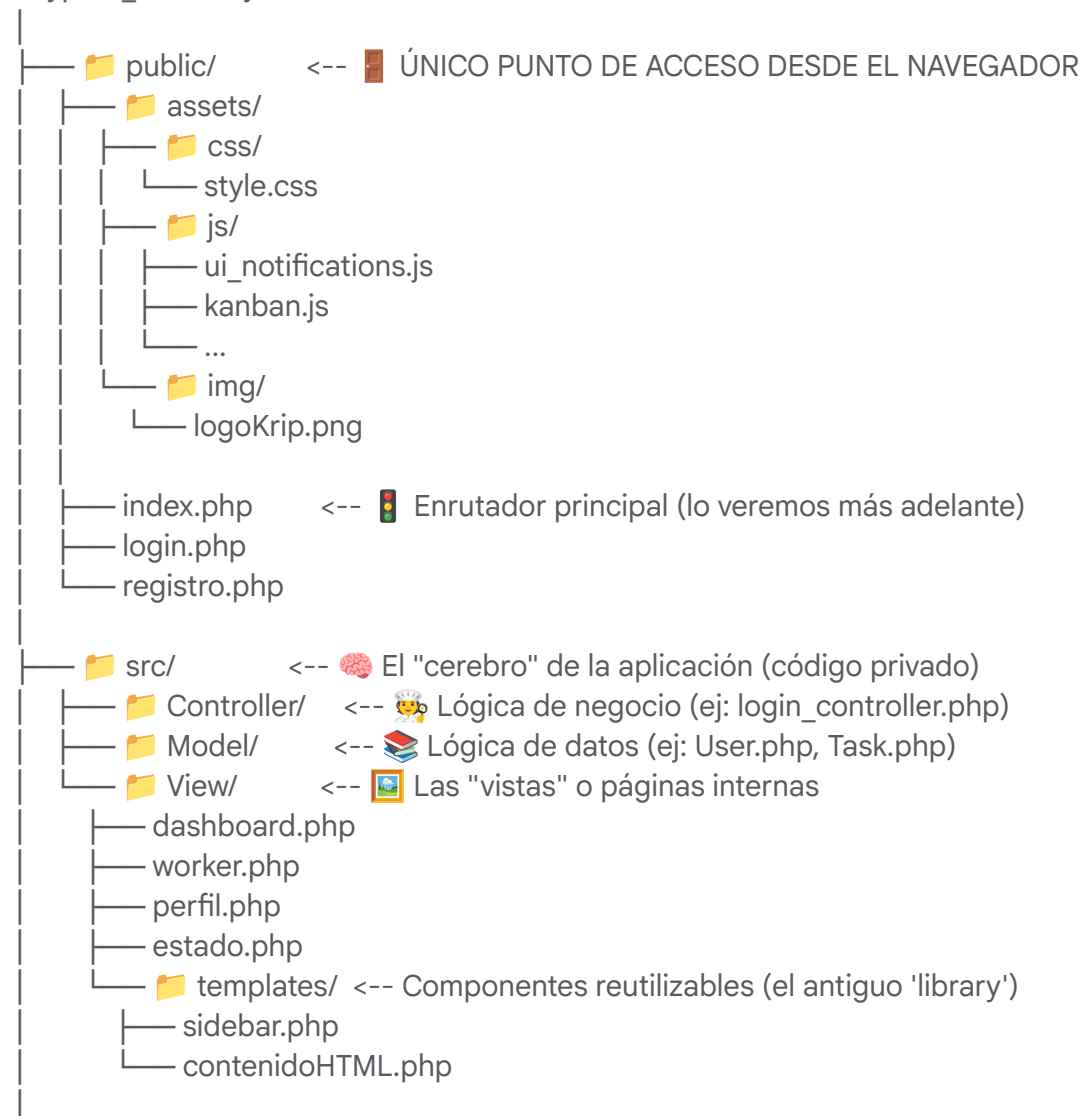
Esta organización está diseñada para ser **clara, segura y escalable**. Sigue un modelo estándar que te facilitará la vida a medida que el proyecto crezca.



## El Principio Clave: Separar lo Público de lo Privado

La idea más importante es tener una única carpeta "pública" que será la única accesible desde el navegador. Todo lo demás (lógica de PHP, contraseñas, configuración) vivirá fuera de esa carpeta, protegido en el servidor.

## Diagrama de la Estructura Recomendada

/krypton\_inventory/



└─  config/ <--  Configuración y llaves (código privado)  
└─ db\_config.php <-- ¡NUESTRO ARCHIVO DE CONEXIÓN VA AQUÍ!

## Explicación de Cada Carpeta:

- **/public/ (Tu Frontend)**
  - **¿Qué es?** Es la única carpeta que un usuario podrá ver desde su navegador. Contiene todos los archivos "estáticos" (CSS, JS, imágenes) y los puntos de entrada iniciales como login.php.
  - **Ventaja:** Máxima seguridad. Nadie podrá acceder directamente a tu lógica de PHP o a tu archivo de configuración de la base de datos escribiendo una URL.
- **/src/ (Tu Backend - El Cerebro)**
  - **¿Qué es?** Aquí vive toda la lógica de tu aplicación. Está organizada siguiendo un patrón muy popular llamado **MVC (Modelo-Vista-Controlador)**.
  - **Controller/:** Contiene los archivos que reciben las peticiones y deciden qué hacer. Por ejemplo, login\_controller.php tomará los datos del formulario de login y le pedirá al Model que verifique al usuario.
  - **Model/:** Contiene la lógica que interactúa directamente con la base de datos. Más adelante, podríamos tener aquí archivos como User.php o Task.php que representen nuestras tablas.
  - **View/:** Contiene las páginas que solo un usuario logueado puede ver (dashboard.php, worker.php, etc.).
  - **View/templates/:** Es el nuevo hogar para tu antigua carpeta library. Es el lugar perfecto para los componentes reutilizables como el menú lateral.
- **/config/ (Tu Backend - La Configuración)**
  - **¿Qué es?** Una carpeta específica para archivos de configuración. Es el lugar más seguro y lógico para tu archivo de conexión a la base de datos.
  - **Ejercicio Anterior:** El archivo db\_config.php que creamos en la Sesión 2 debe ir aquí.

## ¿Cómo se Integra Todo?

Imagina que estás en public/login.php y envías el formulario. La acción del formulario apuntará a un archivo en el controlador, por ejemplo: ../src/Controller/authController.php.

Ese controlador, al estar fuera de la carpeta pública, es seguro. Luego, él se encargará de incluir (require\_once) el archivo de configuración de la base de datos (../config/db\_config.php) y la lógica del modelo para hacer su trabajo.

### Tu Tarea Ahora:

Antes de continuar con la Sesión 2, **reorganiza tus archivos** siguiendo esta estructura. Es un paso pequeño pero que establecerá una base sólida y profesional para todo lo que

construiremos a continuación.

Una vez que tengas tus carpetas y archivos en su nuevo lugar, avísame y retomaremos el ejercicio de crear el archivo `db_config.php` en su ubicación correcta (`/config/`).