

1. Documento de Análisis y Requerimientos (PRD)

Este documento define **qué** estamos construyendo y los límites del sistema.

A. Requerimientos Funcionales (RF)

- **RF-1: Ingesta Multimodal (Telegram Bot):** El sistema debe recibir fotos y videos vía Webhook desde Telegram.
- **RF-2: Vinculación por Metadatos:** El bot debe solicitar o identificar el # de Orden para asociar los archivos correctamente.
- **RF-3: Gestión de Almacenamiento R2:** Los archivos deben subirse automáticamente a Cloudflare R2 usando el protocolo S3.
- **RF-4: Galería de Cliente:** Generación de una vista web (pestaña única) ligera con miniaturas (thumbnails) para que el cliente visualice su evidencia.
- **RF-5: Notificación Omnicanal:** Envío automático/manual de la liga de evidencia por WhatsApp (Evolution API), Telegram o Correo.
- **RF-6: Dashboard Administrativo:** Panel interno para que el personal del taller vea, elimine o reenvíe evidencias.

B. Requerimientos "Lo que siempre falta" (Acceso y Seguridad)

- **RF-7: Control de Acceso (Auth):** Implementación de login para empleados del taller (Roles: Admin, Recepcionista, Mecánico).
- **RF-8: Seguridad de la Liga del Cliente:** Las URLs enviadas a los clientes deben ser firmadas (Pre-signed URLs) o usar un hash único para que nadie pueda adivinar la liga de otro carro.
- **RF-9: Logs de Auditoría:** Registro de quién subió qué video y a qué hora.

C. Requerimientos No Funcionales (RNF)

- **RNF-1: Independencia Total:** La base de datos de ev.errautomotriz.online debe ser independiente de la principal para evitar caídas en cascada.
- **RNF-2: Performance de Carga:** Los videos deben cargarse mediante "streaming" o carga progresiva para no saturar el celular del cliente.
- **RNF-3: Resiliencia:** Si el envío automático falla, el sistema debe permitir el reintento manual.

2. Diseño Técnico (User Stories para el Senior Lead Dev)

Este es el documento que le pasarás a tu Agente AI. Está redactado para que entienda el flujo lógico y la integración.

Historia de Usuario Épica: Sistema de Evidencias Desacoplado

Contexto: Estamos construyendo `ev.errautomotriz.online`. El stack es PHP/MariaDB en Hostinger con almacenamiento en Cloudflare R2.

Story 1: El Webhook de Telegram

Como sistema de evidencias, **quiero** procesar mensajes de un Bot de Telegram, **para** extraer archivos multimedia y el número de orden del mensaje.

- **Criterio de Aceptación:** El script debe recibir el JSON de Telegram, validar que sea un archivo permitido (JPG, MP4), descargar el archivo temporalmente y extraer el `#orden` de los comentarios del mensaje.

Story 2: El Puente R2 (S3 Protocol)

Como backend del sistema, **quiero** subir los archivos procesados al bucket de Cloudflare R2, **para** no ocupar espacio en el hosting web de Hostinger.

- **Criterio de Aceptación:** Usar la librería AWS SDK para PHP. El archivo debe quedar organizado por `/año/mes/orden/nombre_archivo.mp4`.

Story 3: Sincronización de Datos (The Bridge)

Como sistema independiente, **quiero** consultar los datos del cliente (teléfono, nombre) desde `errautomotriz.online`, **para** saber a quién notificar sin duplicar la base de datos.

- **Criterio de Aceptación:** Implementar una API Read-Only muy sencilla en el sistema principal que reciba el `#orden` y devuelva el JSON del cliente.

Story 4: Generador de Galería "Lightweight"

Como cliente, **quiero** recibir un link que abra una galería optimizada, **para** ver el estado de mi vehículo sin descargar archivos pesados.

- **Criterio de Aceptación:** La galería debe mostrar miniaturas. Al dar clic, el video se reproduce desde el CDN de Cloudflare.

3. Estrategia de Independencia (Nexus6 Style)

Para cumplir con tu regla de que los sistemas vivan separados:

1. **Bases de Datos:** `db_ordenes` y `db_evidencias` deben ser distintas. La de evidencias solo guardará: `id`, `orden_id`, `path_r2`, `tipo_evidencia` y `fecha`.

2. **Comunicación:** El sistema de evidencias le "pregunta" al sistema principal mediante una **API Key** interna. Si el sistema principal está caído, el de evidencias puede seguir recibiendo videos de Telegram y guardarlos, simplemente esperará a que el principal vuelva para saber a qué cliente pertenecen.
3. **Seguridad:** El sistema de evidencias no tiene permisos de escritura en la base de datos principal. Es un "consumidor" de información.