Prueba Técnica – Gestión de Actas y Compromisos

Construir un módulo independiente que permita gestionar actas, compromisos y gestiones, con autenticación por rol, protección de archivos y consumo de API propia.

Requisitos Generales

- Stack obligatorio: Django (backend) + React (frontend)
- No debe tocar APIs ni bases de datos del proyecto original.
- Debe crear su propia base de datos, endpoints y modelos.
- La base debe ser SQLite y correr en local sin despliegues.
- El frontend debe conectarse exclusivamente a su backend.

Roles a Implementar

- Administrador: puede ver todas las actas, compromisos y gestiones.
- Usuario Base:solo puede ver las actas donde participa (como creador o responsable).

Autenticación

- Login clásico (`correo`,`contraseña`).
- Al iniciar sesión, el frontend debe guardar el token o mantener sesión.
- El backend debe exponer el rol del usuario para controlar el acceso.
- Las rutas deben estar protegidas en frontend y backend.

Funcionalidades Obligatorias

1. Login de usuario

- Endpoint `/login/` que devuelva datos y rol.
- Base de datos precargada con:

- Usuario Admin

- Usuario Base

- Frontend: formulario de login + lógica para redirección.

2. Panel de Actas (`/actas/`)

- Tabla con: título, estado, fecha, compromisos y botón de detalle.
- Filtros: por estado, título y fecha.
- Datos servidos desde backend.

3. Detalle de Acta (`/actas/:id/`)

- Mostrar los datos completos del acta.
- Listar compromisos asociados.
- Mostrar el PDF del acta (solo si esta autenticado).
- Botón para agregar gestión si el usuario tiene permisos.

4. Crear Gestión

- Endpoint `/gestiones/`.
- Formulario con: fecha, descripción y archivo adjunto.
- Validaciones:

  - Solo archivos `.pdf` o `.jpg`

  - Máximo 5MB.

- Guardar el archivo físicamente y exponer va `/media/<archivo>` protegida.

5. Protección de Archivos

- Solo usuarios autenticados pueden acceder a los archivos.
- Usar vista tipo `/media/<archivo>` que retorne 403 si no está autenticado.

Entregables

1. Proyecto funcional: backend Django + frontend React.

2. Base de datos precargada con usuarios y ejemplos.

3. Instrucciones claras en el README en el repositorio de github (adjuntar el respectivo repositorio para el proyecto)

4. Video explicativo de máximo 5 minutos mostrando:

   - Que hizo.

   - Como se navega.