

L U M B R E



Proyecto 2 · Agenda de contactos: tu primera aplicación CRUD

Materia: Desarrollo de Software con IA

07/10/2026

Antes de empezar

El proyecto donde tu app deja de ser adorno: los datos tienen que sobrevivir al recargar, y eso obliga a backend y base de datos —las dos piezas que M6 y M7 explicaron y aquí se montan de verdad—. La misión: agenda de contactos con crear, listar, buscar, editar y borrar, con el borrado suave como rey de la casa y validación en los dos lados. Incluye el momento formativo garantizado: borrar algo que no debías y descubrir si tu protocolo de respaldo sirve o era decoración.

QUÉ APRENDERÁS

- Construir crear-leer-actualizar-borrar con datos persistentes, validaciones y búsqueda.

La misión

Una agenda de contactos profesional: **crear** (nombre, teléfono, correo, notas, favorita), **listar** ordenado y con búsqueda instantánea, **editar** sin recargar, **borrar**... con dos niveles (quitar de la vista / eliminar de verdad con confirmación tipada). Los datos viven en una base real, no en la memoria del navegador: recarga la página y todo sigue ahí. Backend y base de datos los arma la IA; el modelo de datos, las validaciones y el protocolo de borrado los escribes tú en la especificación.

La ruta, seis tiempos

ACTIVIDAD INTERACTIVA DISPONIBLE EN LUMBRE

1. Ruta del proyecto 2



El prompt del modelo de datos (el que va primero)

El prompt que fija las reglas de negocio ANTES de que la IA las "proponga"

Vamos a construir la agenda de contactos del proyecto 2 de mi taller, por capas.\nCapa 1: base de datos y backend. Capa 2 la pidamos cuando esta este verde.\n\nModelo de datos (decidido por mi, no lo cambies sin discutirlo):\n- tabla contactos: id, nombre (obligatorio), telefono (10 digitos, opcional),\n correo (formato validado, opcional), notas (texto largo, opcional), favorita\n (booleano, default falso), creado_el, actualizado_el, eliminado_el (nulo = vivo)\n- reglas de negocio:\n * "borrar" de la vista = marcar eliminado_el (soft delete); la lista solo muestra\n donde eliminado_el es nulo\n * "eliminar de verdad" requiere confirmacion con la palabra SI en el frontend y\n solo esta disponible si el contacto esta borrado de la vista primero\n * un mismo correo no puede existir dos veces vivo; en borrados si puede (historial)\n\nEndpoints como contrato (definir exito Y error de cada uno):\nGET /contactos?q=&favoritas= | POST | PUT /:id | DELETE /:id (soft) | PURGE /:id (real)\n\nImplementa con [stack del documento maestro], con validacion EN EL SERVIDOR de todas\nlas reglas (el frontend es UX, no defensa). Pruebas automatizadas por endpoint: caso\nfeliz + al menos 2 casos de error por endpoint. Muestra el diff, corre las pruebas,\nespera mi revision antes de la capa 2.

Qué entrega este proyecto

La diferencia entre este proyecto y el 1 es una pregunta que cualquier usuario hace: ¿y si cierro la laptop? La respuesta —aquí siguen tus datos— es exactamente lo que M6+M7 construyeron.

Entregable	Barra alta de este proyecto
ESPECIFICACION.md	el modelo de datos con tipos y reglas de negocio, y los endpoints como contrato con casos de error
HALLAZGOS.md	la búsqueda que fallaba con acentos, la validación que solo existía en el formulario...
PRUEBAS.md	incluye la prueba del borrado: el escenario destructivo que corriste contra la base de desarrollo y su resultado
App publicada	datos que sobreviven recargar desde otro dispositivo, con búsqueda útil

Prueba

2. Un usuario llama «borrar» al botón rojo de tu agenda. Según el modelo de este proyecto, ¿qué hace el sistema?

- ☐ DELETE de la fila: lo que el usuario pidió, directo.
- ☐ Marcar eliminado_el: sale de la vista pero queda recuperable; el borrado real exige pasos adicionales.
- ☐ Ocultar la fila con CSS hasta mañana.
- ☐ Vaciar la tabla de contactos.

3. La validación del teléfono a 10 dígitos puesta en el formulario HTML (required + pattern) ya cumple la regla del modelo: no hace falta repetir en el servidor.

- ☐ Verdadero
- ☐ Falso

4. Orden de ataque: especificación de 25 minutos, luego el prompt de capa 1 y nada más hasta que esté verde. La prueba del borrado (paso 5) no es opcional: es la escena madre de este proyecto —documenta en HALLAZGOS.md lo que descubriste al intentar romper tu propia app.

Una base de datos que sobrevive a todo es la que sobrevivió primero a tus ataques de prueba.

Referencia curricular

Materia:	Desarrollo de Software con IA
----------	-------------------------------

Versión interactiva completa (con actividades, videos y autoevaluación):



<https://empresa.lumbre.lat/object/fsi-pj-oa02>

Material educativo generado desde Lumbre · lumbre.lat