

L U M B R E



Contratos entre piezas: qué espera cada componente de las demás

Materia: Desarrollo de Software con IA

07/10/2026

Antes de empezar

Las piezas de un sistema no se hablan por intuición: se hablan por contrato. Un contrato dice qué datos entran, cuáles salen y qué pasa si algo va mal. Cuando tú escribes el contrato, la IA no puede improvisar una forma distinta en cada pieza. Aquí aprendes a firmar esos acuerdos en lenguaje natural antes de pedir una línea de código.

QUÉ APRENDERÁS

- Escribir contratos simples (qué entra, qué sale) entre pantallas, lógica y datos.

Piezas que se hablan con reglas claras

Cuando la IA construye el frontend por un lado y el backend por otro, tienen que **ponerse de acuerdo en la forma** de lo que se pasan. A ese acuerdo se le llama contrato: qué le llega a una pieza, qué devuelve, y qué ocurre si falla. Sin contrato, cada pieza inventa su propia forma y todo se rompe en la integración.

Las tres preguntas de todo contrato

Si las tres respuestas están escritas, la IA no tiene dónde improvisar.

Pregunta	Qué define	Ejemplo en «buscar contactos»
¿Qué entra?	Los datos que recibe la pieza	un texto de búsqueda (puede venir vacío)
¿Qué sale?	Lo que devuelve en el caso bueno	una lista de contactos con nombre y teléfono
¿Qué pasa si falla?	El comportamiento ante el error	lista vacía con un aviso, no un pantallazo

El contrato por escrito

No hace falta código: basta una ficha en lenguaje natural, o un ejemplo de la forma que tendrán los datos. Esta es la ficha de un endpoint descrita como contrato:

Contrato de una pieza

```
{\n  "pieza": "GET /contactos?query=",\n  "entra": {\n    "query": "texto\nopcional; si viene vacio devuelve todos"\n  },\n  "sale_bien": {\n    "contactos":\n    [ { "id": 1, "nombre": "Ana", "telefono": "55..." } ],\n    "total": 1\n  },\n  "sale_mal": {\n    "error": "mensaje legible",\n    "contactos": []\n  }\n}
```

Por qué el contrato te protege

Regla de oro

Un contrato escrito convierte «suena raro» en «no cumple la ficha». Puedes pedirle a la IA que revise UNA pieza contra el contrato sin conocer las demás. Esa es la base de probar por partes.

Prueba

1. El frontend muestra «NaN» en el total porque esperaba total pero el backend devolvió count. ¿Qué faltó fijar?

- ☐ El color del número.
- ☐ El contrato del nombre y la forma de los datos que salen.
- ☐ El dominio del sitio.
- ☐ La hoja de estilos.

2. Un buen contrato debe decir también qué ocurre cuando algo falla.

- ☐ Verdadero
- ☐ Falso

Detecta el contrato incompleto

ACTIVIDAD
INTERACTIVA
DISPONIBLE EN
LUMBRE

3. Revisas la ficha de una pieza antes de pedir su código. Señala la línea que deja un hueco por el que la IA improvisará:



4. Elige una pieza de tu proyecto y escribe su contrato con las tres preguntas (entra, sale bien, sale mal). Enséñaselo a la IA y pide que confirme que la construirá exactamente así.

Si no sabes qué debe devolver una pieza al fallar, todavía no sabes construir esa pieza.

Referencia curricular

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

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



<https://empresa.lumbre.lat/object/fsi-m05-0a02>

Material educativo generado desde Lumbre · lumbre.lat