

ARQUITECTURA PRO CON CLAUDE CODE

BASE DE DATOS, REGIÓN Y KUBERNETES



LO QUE SE PREGUNTA UN ARQUITECTO

Claude Code te arma una aplicación completa, pero si no le das tus números, elige la base de datos, el servidor y la región por costumbre. A veces acierta, pero estas tres preguntas hacen que decidan tus datos.

PREGUNTA	QUÉ DECIDE	CÓMO SE RESPONDE
¿LEE O ESCRIBE?	La base de datos y cómo escala	Cuenta cuántas lecturas hay por cada escritura en tus pantallas y procesos principales
¿DÓNDE?	La región del servidor y de la base	El país desde donde entra la mayoría, y qué ley protege sus datos
¿CUÁNTO OPERAS?	Si necesitas Kubernetes o no	Cuántos servicios tienes y cuántas personas los mantienen

BIBLIOTECA

SE LEE MUCHO

Mucha gente lee los mismos libros: haces copias y dejas los más pedidos en la entrada.

OFICINA DE CORREOS

SE ESCRIBE MUCHO

En Navidad llegan miles de cartas: no las clasificas en la puerta, las apilas y las ordenas atrás, a tu ritmo.

¿SE LEE MUCHO O SE ESCRIBE MUCHO?

CATÁLOGO · BLOG · PANEL

SI SE LEE MUCHO

Postgres va perfecto para empezar. Cuando crece, mides cuál consulta es la lenta y agregas una **caché** (Redis) para pocos datos muy pedidos, o **réplicas de lectura** para muchas consultas variadas. Las réplicas van unos segundos atrasadas: lo que alguien acaba de guardar se lee en la principal.

MÉTRICAS · EVENTOS · CHAT

SI SE ESCRIBE MUCHO

Una **cola** que recibe todo y **trabajadores** que lo guardan a su ritmo, así un pico no tumba la base. Los trabajadores toleran recibir el mismo mensaje dos veces, y lo que falla varias veces va a una cola aparte.

TABLA PARA ELEGIR TU BASE

TUS DATOS SE PARECEN A...	MODELO	EJEMPLOS DE MOTOR
PAGOS, RESERVAS	Relacional (tiene que cuadrar)	Postgres, MySQL
CATÁLOGO VARIABLE	Documental, o JSON en Postgres	MongoDB, Postgres con jsonb
SESIONES, CONTADORES	Clave-valor en memoria	Redis
EVENTOS, MÉTRICAS	Columnar o wide-column	ClickHouse, Cassandra
RELACIONES	Grafo (recomendaciones)	Neo4j

Por dentro, Postgres ordena los datos en un árbol, que sirve para buscar. Cassandra anota todo al final y ordena después, entonces aguanta muchísimas escrituras. La caché, en cambio, no sirve para datos que cambian a cada rato. Si no sabes cuál elegir, empieza con Postgres.

¿DÓNDE ESTÁN TUS USUARIOS?

La luz en la fibra óptica viaja a unos 200.000 km/s. De Santiago a Virginia hay unos 8.091 km en línea recta: la ida y vuelta mínima es de unos 81 ms, y en la práctica es más.



1

Servidor y base, en la misma región. Si están en países distintos, cada consulta paga el viaje: 20 consultas seguidas de 100 ms son 2 segundos solo en viajes (ejemplo).

2

La región, cerca de la mayoría de tus usuarios. Muchas plataformas crean tus funciones o tu base en Estados Unidos si no eliges otra cosa: revísalo al crear el proyecto.

3

Lo estático, en una CDN. Imágenes, JavaScript y CSS se sirven desde cerca de cada usuario.

TAMBIÉN ES UNA DECISIÓN LEGAL

En Europa, el Reglamento General de Protección de Datos (RGPD) pone reglas para transferir datos personales fuera de la Unión. En Chile, la Ley 21.719, vigente desde el 1 de diciembre de 2026, también regula la transferencia internacional. Si guardas datos de personas, revisa qué exige la ley de tu país antes de elegir región.

Conocimiento general, no asesoría legal. Para datos sensibles (salud, finanzas, menores), consulta con alguien especializado.

¿CUÁNTO PUEDES OPERAR?

Kubernetes es como un director de orquesta para tus contenedores: si uno se cae lo vuelve a levantar, si llega más tráfico agrega copias, y actualiza tu aplicación sin cortar el servicio.

Ojo, no hace tu aplicación más rápida y no es para empezar, porque además del clúster alguien tiene que mantener sus versiones, permisos, redes y monitoreo.

TU SITUACIÓN	LO QUE CONVIENE
UNA APP, EQUIPO CHICO	Plataforma administrada (Railway, Render, Fly.io o similar) y un monolito ordenado por módulos
PICOS O TAREAS SUELTAS	Funciones serverless para esas partes
MUCHOS SERVICIOS	Kubernetes, idealmente administrado por tu proveedor de nube, con un equipo que pueda mantenerlo

Y con los servicios pasa lo mismo que con la base: separa un módulo en su propio servicio solo cuando tenga una razón medible, como un tráfico muy distinto al resto o un equipo propio que lo despliega por su cuenta.

TUS NÚMEROS Y DOS ALTERNATIVAS

Claude Code conoce la teoría, pero no tus números. Paso 1: escríbelos en el CLAUDE.md del proyecto (los de abajo son de ejemplo).

```
## Números del negocio
```

- Usuarios: 5.000 al mes (meta a un año: 50.000)
- País de la mayoría: Chile
- Patrón: 90 % lecturas (catálogo), 10 % escrituras (reservas)
- Datos personales: sí (nombre, correo, teléfono)
- Equipo: 2 personas, sin experiencia operando servidores
- Presupuesto de infraestructura: US\$ 50 al mes

Paso 2: en modo plan, pídele alternativas antes de programar.

Lee la sección «Números del negocio» del CLAUDE.md.

Antes de escribir código, propón la arquitectura de este proyecto respondiendo:

1. ¿El sistema se lee más o se escribe más? Separa por módulo si cambia.
2. Qué base de datos conviene para cada módulo y por qué, según ese patrón.
3. Cuándo agregarías caché, réplicas de lectura o una cola, y qué métrica te diría que llegó el momento.
4. En qué región van el servidor y la base, considerando dónde están los usuarios y qué ley protege sus datos.
5. Dónde lo despliegas (plataforma administrada, serverless o Kubernetes), considerando el tamaño del equipo.

Dame dos alternativas con su costo mensual aproximado, su complejidad y sus riesgos, y recomienda una. Si te falta un dato, pregúntamelo en vez de suponerlo. No escribas código todavía.

GUÁRDALA EN EL PROYECTO

Paso 3: pídele que la escriba en docs/decisiones/0001-base-de-datos.md. Así la próxima sesión no la vuelve a discutir.

```
# 0001 · Base de datos: Postgres con una réplica de lectura

## Contexto
90 % lecturas, 5.000 usuarios al mes en Chile, equipo de 2 personas.

## Decisión
Postgres en la misma región que el servidor. Una réplica de lectura
cuando la consulta del catálogo pase de 200 ms en el percentil 95.

## Alternativas descartadas
- Caché con Redis: el catálogo cambia varias veces por hora.
- Kubernetes: una sola app y nadie para operar el clúster.

## Consecuencias
Las lecturas de la réplica pueden ir unos segundos atrasadas.
```

CHECKLIST ANTES DE ESCALAR

- | | |
|---|---|
| ☐ Mediste cuál es la parte lenta, no la supusiste. | ☐ Sabes cuántas lecturas hay por escritura. |
| ☐ Servidor y base en la misma región. | ☐ La decisión está escrita, con lo descartado. |
| ☐ Región cerca de tus usuarios y dentro de la ley. | ☐ Hay alguien que opere lo que vas a agregar. |

Sígueme en Elkis Dev para más desarrollo de software, IA y tecnología, explicados simple.