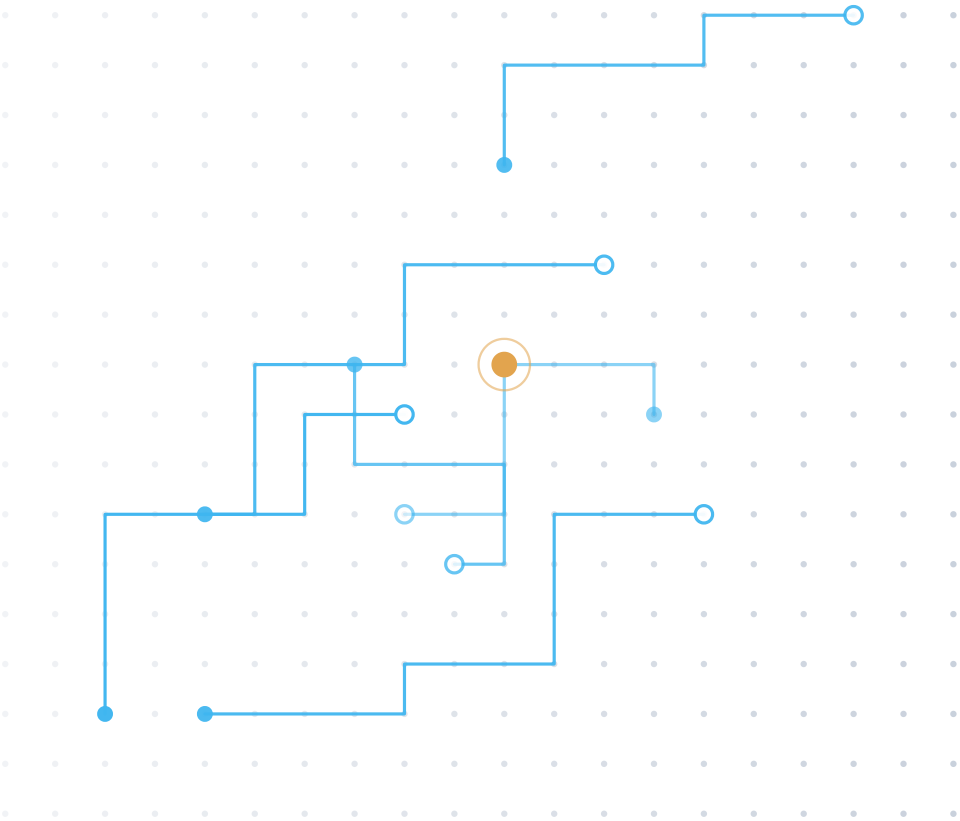


GUÍA PRÁCTICA

El modelo adecuado para cada tarea: paga por criterio, no por lectura

Reparte el trabajo de IA en cuatro niveles y gasta menos acertando más

AUTOR
Mario Chasco · Flowara AI

PARA
Equipos técnicos y dueños de negocio que pagan la factura de la IA

LECTURA
9 min

EDICIÓN
v2.0 · noviembre 2026

ANTES DE EMPEZAR

Qué vas a sacar de esta guía

01 Decidir qué modelo hace cada tarea según cuánto hay que leer y cuánto cuesta un error.

02 Organizar un orquestador y cuatro agentes con trabajos claros y límites escritos.

03 Calcular lo que cuesta una conversación antes de poner precio a una cuota mensual.

04 Escribir encargos que un agente entiende a la primera.

05 Comprobar con casos reales que el reparto acierta, en vez de suponerlo.

CONTENIDO

01 El problema: una arquitecta haciendo fotocopias
02 Cuatro niveles, cuatro trabajos
03 El orquestador lee, decide, delega y verifica
04 Qué tarea va a cada nivel
05 Cuánto cuesta cada nivel
06 Cinco reglas para repartir bien
07 Ejemplo: una función nueva de principio a fin
08 Lista para repartir por niveles

Para: Equipos técnicos y dueños de negocio que pagan la factura de la IA · 9 min de lectura

01

El problema: una arquitecta haciendo fotocopias

Imagina un estudio de arquitectura donde la arquitecta sénior, la que más cobra por hora, pasa la mañana haciendo fotocopias, buscando planos en el archivo y pasando actas a limpio. El estudio paga tarifa de experta por trabajo de becario. Y encima ella llega cansada a la única decisión que de verdad necesitaba su criterio.

Con los modelos de IA pasa lo mismo. Usar el más potente para todo parece lo prudente, pero cuesta dos veces. Primero en la factura: leer cien archivos para encontrar tres líneas es la tarea más cara y la que menos criterio exige. Después en la calidad, porque un contexto lleno de lecturas irrelevantes deja menos atención para lo importante y los errores se esconden entre el ruido.

TABLA 1 **El mismo trabajo, dos maneras de pagarlo**

	Todo con el más potente	Reparto por niveles
FACTURA	Tarifa alta en cada lectura	Tarifa alta solo donde hay riesgo
CALIDAD	Contexto lleno de ruido	Cada modelo ve solo lo suyo
VELOCIDAD	Lento también en lo fácil	Lo mecánico va al más rápido
ERRORES	Se esconden entre lecturas	Un verificador los busca

Lo contrario tampoco funciona. Pedir a un modelo pequeño que decida la arquitectura de un sistema de cobros ahorra unos céntimos y cuesta semanas. La salida es repartir por niveles, como en cualquier equipo bien organizado.

> *Paga por criterio, no por lectura: cada tarea al nivel más bajo que la hace bien.*
>
> LA REGLA DE REPARTO QUE USO EN FLOWARA AI

02

Cuatro niveles, cuatro trabajos

El reparto tiene cuatro perfiles. Cada uno corresponde a un modelo y a un tipo de trabajo, y cada uno tiene cosas que nunca hace. Así los tengo configurados en mi servidor:

01 **Junior, con Haiku 5.5.** Descubrimiento y resúmenes: localizar archivos, leer registros, contar, clasificar mensajes o resumir un documento largo. Solo lee; nunca edita ni opina de diseño.

02 **Ejecutor, con Sonnet 5.5.** Implementación rutinaria y redacción cuando el diseño ya está decidido: una página nueva con una plantilla, una prueba más, la documentación de algo que ya funciona. No resuelve dudas de diseño ni de seguridad.

03 **Senior, con Opus 5.5.** Arquitectura, errores difíciles, todo lo que toque dinero o seguridad y cualquier decisión difícil de deshacer. No se usa para buscar archivos.

04 **Verificador, con Sonnet 5.5.** Revisa cada entrega relevante y responde *aprobado*, *aprobado con avisos* o *no aprobado*. Nunca corrige; si corrigiera, dejaría de ser independiente.

> ✦ IDEA
>
> El verificador no es un lujo. Quien ha escrito algo es el peor revisor de su propio trabajo, sea persona o modelo.

EN UNA FRASE

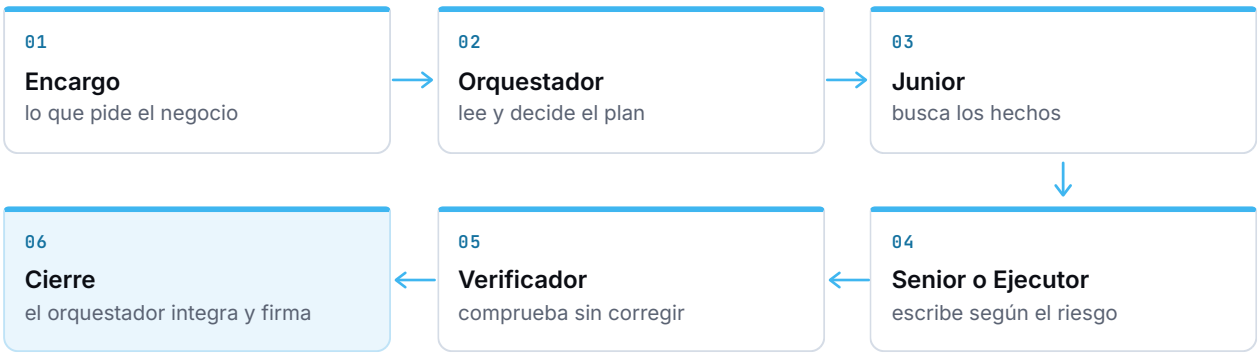
Cada nivel tiene un trabajo y una lista de cosas que no hace.

03

El orquestador lee, decide, delega y verifica

Por encima de los cuatro niveles hay una sesión principal que hace de orquestador. Su papel es el de una jefa de obra: entiende el encargo, decide el plan, reparte las piezas, revisa lo que vuelve e integra el resultado. Lo que no hace es escribir ella misma código ni documentos largos, aunque parezca «solo una línea».

FIGURA 1 **Cómo circula un encargo**



En mi configuración, la sesión principal corre con Fable 5.1, el modelo más capaz y también el más caro de la familia. Precisamente por eso no escribe. Cada línea que redactara costaría cinco veces más que en Sonnet 5.5, y su contexto es lo que más vale del sistema: si lo llena de código, pierde la visión de conjunto.

Hay dos razones más. Delegar obliga a escribir un encargo claro, y eso ya elimina muchos errores. Y separar quién escribe de quién decide hace que cada resultado pase por al menos dos miradas.

Las tareas independientes se reparten en paralelo. Las que tocan los mismos archivos van una detrás de otra, con un punto de control entre ellas.

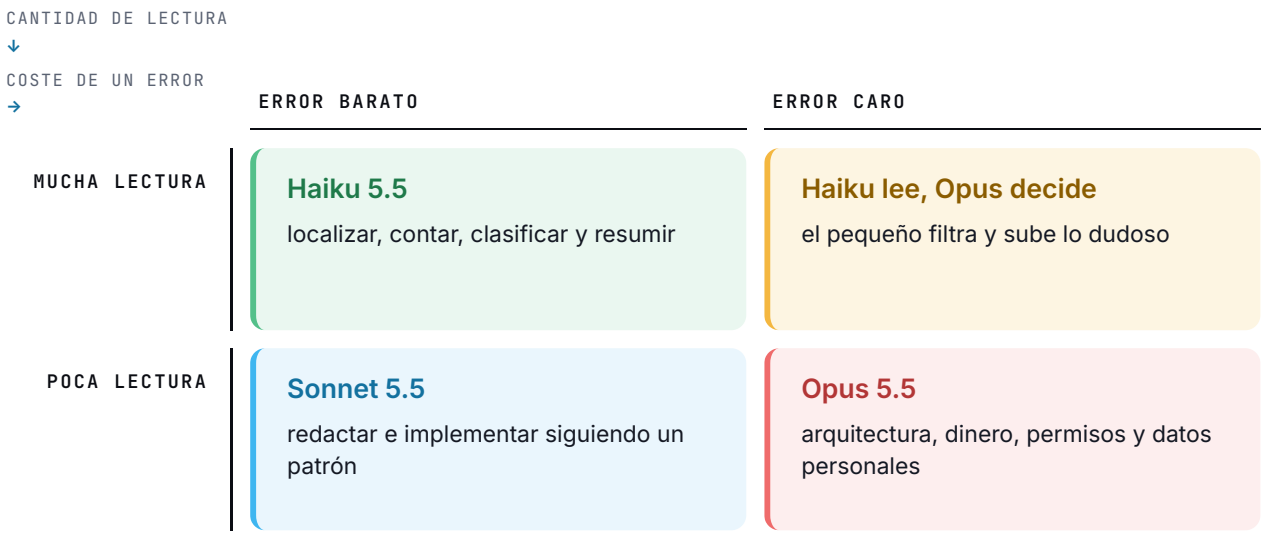
EN UNA FRASE

El orquestador no escribe; reparte el trabajo y firma el resultado.

04 Qué tarea va a cada nivel

Ante la duda, hazte dos preguntas. ¿Cuánto hay que leer? ¿Y cuánto cuesta deshacer un error aquí? Mucha lectura y poco riesgo, abajo. Poca lectura y mucho riesgo, arriba.
Para la segunda pregunta uso un semáforo. Si el error se deshace con un clic, Verde. Si cuesta una tarde de trabajo, Ámbar. Si toca dinero, permisos o datos personales, Rojo, y entonces la tarea va a Opus 5.5 aunque sea corta.

FIGURA 2 **Qué modelo para qué tarea**



Fíjate en que el nivel alto ocupa una sola celda, pero es la que más importa. La celda ámbar es la más rentable: el modelo barato hace toda la lectura y el caro solo ve lo que el barato no tiene claro.

EN UNA FRASE

Mucha lectura va abajo; mucho riesgo va arriba.

05 Cuánto cuesta cada nivel

Los cuatro modelos aceptan un contexto de hasta un millón de tokens, que son unas 555.000 palabras. Lo que cambia es el precio y la velocidad. Estos son los precios públicos de Anthropic a fecha de la guía:

MODELO	ENTRADA / SALIDA, POR MILLÓN DE TOKENS	VELOCIDAD	PARA QUÉ LO USO
Haiku 5.5	desde 0,10 $ / 0,50 $	La más rápida	Clasificar, contar, resumir
Sonnet 5.5	2 $ / 10 $	Rápida	Redactar e implementar
Opus 5.5	4 $ / 20 $	Moderada	Decidir y lo difícil
Fable 5.1	10 $ / 50 $	La más lenta	Orquestar, en mi caso

Dos detalles bajan mucho la factura. Las tareas que pueden esperar, como una revisión nocturna, se pueden mandar por lotes y cuestan la mitad. Y lo que se repite en cada llamada, como las reglas o el catálogo de un negocio, se puede guardar en caché: volver a leerlo cuesta un 10 % del precio de entrada, y un 5 % en Opus 5.5 y Sonnet 5.5.

> ⚠ **AVISO**
>
> Usa estas cifras como orden de magnitud. Los precios cambian; antes de presupuestar nada, comprueba la página oficial de precios de Anthropic: https://platform.claude.com/docs/en/about-claude/pricing

Antes de poner precio a una cuota mensual, mido lo que cuesta una conversación. La cuenta es sencilla: tokens de entrada por su precio, más tokens de salida por el suyo. Por ejemplo, una conversación de 2.000 tokens de entrada y 300 de salida (cifras de ejemplo, no de un cliente) sale a menos de un céntimo de dólar con Sonnet 5.5, y al doble con Opus 5.5. Parece poco hasta que lo multiplicas por las conversaciones de un mes en una peluquería con el WhatsApp lleno.

Para elegir entre dos modelos en una tarea concreta, no me fío de la intuición. Paso los mismos casos reales por los dos y anoto por cada caso si acertó y cuánto costó, con los segundos que tardó al lado. A veces el pequeño acierta igual y responde antes, y entonces no hay nada que pensar.

06 Cinco reglas para repartir bien

Estas reglas evitan casi todos los errores de reparto que he visto.

01 **Nunca bajes de nivel en tareas de criterio.** Si la decisión es difícil de deshacer o toca dinero o seguridad, va al nivel alto aunque sea corta. Ahorrar ahí sale caro.

02 **Nunca subas de nivel en tareas mecánicas.** Buscar, renombrar, formatear o resumir no mejora con un modelo más caro; solo cuesta más.

03 **Digiere los hechos antes de delegar.** El encargo ya lleva lo que el agente necesita: archivos, decisiones tomadas, qué está descartado y qué no debe tocar. Un agente que tiene que redescubrir el contexto gasta el doble y acierta menos.

04 **Pide informes cortos.** Como máximo 250 palabras con lo que ha cambiado, cómo se ha comprobado y lo que queda pendiente. Un informe largo traslada el coste de lectura al orquestador.

05 **Reparte en agentes solo lo grande y divisible.** Cada agente arranca con su propio contexto y vuelve a leer lo que necesita, así que repartir multiplica el consumo de tokens. Compensa en trabajos grandes que se parten en piezas independientes; en un cambio pequeño, sale más caro que hacerlo de una vez.

EJEMPLO

En vez de «arregla el formulario», el encargo dice: «el envío falla cuando falta el teléfono; el archivo es este; no cambies el diseño; devuélveme el cambio y la prueba que lo demuestra».

EN UNA FRASE

Un buen encargo vale más que un modelo más caro.

07 Ejemplo: una función nueva de principio a fin

Una agencia de viajes quiere añadir a su web un formulario de solicitud de presupuesto que avise por correo al equipo. Así se reparte:

FIGURA 3 **Una función nueva, nivel a nivel**

1 **Orquestador**
Lee la petición y fija el alcance: formulario, aviso por correo y nada de pagos.

2 **Junior con Haiku 5.5**
Localiza el formulario de contacto actual y cómo se envían hoy los correos. Devuelve 150 palabras con los archivos exactos.

3 **Senior con Opus 5.5**
Decide qué datos personales se guardan, cuánto tiempo y qué nunca se escribe en los registros. Es corta, pero de criterio.

4 **Ejecutor con Sonnet 5.5**
Implementa el formulario y la plantilla de correo siguiendo ese diseño, con sus pruebas.

5 **Verificador**
Ejecuta las pruebas y comprueba los registros. Responde aprobado con avisos: falta un mensaje de error amable.

6 **Cierre**
El ejecutor añade el mensaje, el verificador repite y el orquestador integra.

El modelo más caro de los cuatro niveles trabajó en una sola pieza, la que tenía riesgo. Todo lo demás lo hicieron niveles más baratos, y con una revisión independiente al final.

EN UNA FRASE

El criterio caro se gasta donde hay riesgo, no donde hay volumen.

08

Lista para repartir por niveles

Usa esta lista para revisar cómo reparte el trabajo con IA tu equipo, o tu propia configuración.

Reparto por niveles, punto por punto

14 PUNTOS

- [ ] Cada tarea tiene un nivel antes de empezar
- [ ] Lo difícil de deshacer va siempre a Opus 5.5
  dinero, permisos y seguridad
- [ ] La lectura masiva va a Haiku 5.5
  localizar, contar y resumir
- [ ] El orquestador solo lee, decide, reparte y verifica
- [ ] Cada encargo lleva objetivo, archivos y descartes
  y lo que no se toca
- [ ] Cada encargo fija el formato del informe
  como máximo 250 palabras

- [ ] Un verificador que no la escribió revisa cada entrega
- [ ] El verificador informa y nunca corrige
- [ ] Mismos archivos, en secuencia
  con un punto de control
- [ ] Agentes en paralelo solo en trabajos grandes y divisibles
- [ ] Coste por conversación medido antes de fijar una cuota
- [ ] Modelos comparados con casos reales
  aciertos, coste y segundos
- [ ] Cada agente tiene una persona con nombre que responde de él
- [ ] Cada error de reparto se convierte en una regla escrita

SIGUIENTE PASO

Diagnóstico gratuito (20 min)

Si quieres ver qué parte de tu negocio se puede automatizar con el riesgo bajo control, reserva un diagnóstico gratuito de 20 minutos.

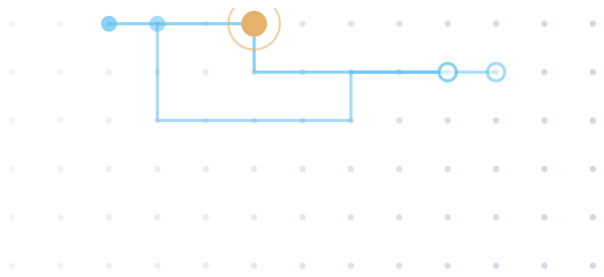
Reservar diagnóstico ›

flowaraai.com/es/contacto

RECURSOS RELACIONADOS

Kit con los cuatro agentes para Claude Code ↗

Vídeo: el cerebro de tu negocio en notas de texto ↗



Mario Chasco · Flowara AI

Sistemas de IA en producción para negocios pequeños.

flowaraai.com
github.com/mariochasco78557
youtube.com/@mariochasco_AI
linkedin.com/in/mario-adrian-chasco-frauca

Puedes compartir esta guía tal cual, citando flowaraai.com.