

GUÍA PRÁCTICA DE PROMPTING

Cómo crear herramientas digitales con Claude y Codex, aunque no seas programador.

De una idea a una herramienta propia, con instrucciones claras, contexto y un método de trabajo.

I

Intención

Qué problema resolvemos y para qué.

D

Diseño

Usuarios, funciones, datos y pantallas.

E

Ejecución

Reglas, validaciones, permisos e integraciones.

A

Aseguramiento

Pruebas, errores y criterios de aceptación.

Índice

01	Entender qué podemos crear	Intención
02	El método IDEA	Método
03	Cómo estructurar un buen prompt	Diseño
04	El contexto: enseñar a la IA cómo funciona mi negocio	Intención · Diseño
05	Cómo utilizar archivos y referencias	Diseño
06	Cómo promptear por voz	Intención
07	Cómo trabajar con Claude y Codex	Ejecución
08	Cómo construir herramientas reales	IDEA completo
09	Cómo iterar un prompt y corregir resultados	Aseguramiento
10	Cómo reducir iteraciones y optimizar tokens	Ejecución
11	Pruebas, seguridad y publicación	Aseguramiento
12	Mi proyecto paso a paso	Cuaderno
A	Checklist general del proyecto	Anexo
B	Laboratorio personal de prompts	Anexo
C	Glosario básico	Anexo

Videos demostrativos de la guía

La docente agregará los enlaces. Cada video se vincula con una sección y una actividad. El contenido escrito funciona aunque todavía no hayas visto el video.

N.º	Título	Dónde
01	De una idea hablada a un proyecto	Capítulo 6
02	Cómo analizar una planilla, un CRM o una plataforma	Capítulo 5

N.º	Título	Dónde
03	Cómo redactar un prompt inicial	Capítulo 3
04	Cómo interpretar una respuesta incompleta	Capítulo 9
05	Cómo iterar y mejorar instrucciones	Capítulo 9
06	Cómo trabajar con Claude o Codex sobre un proyecto	Capítulo 7
07	Cómo probar y corregir una función	Capítulo 11

Cómo usar esta guía

No necesitás ser programador para empezar a desarrollar herramientas digitales con inteligencia artificial. Sí necesitás aprender a explicar qué querés construir, cómo funciona tu negocio, qué reglas debe respetar el sistema y cómo comprobar que el resultado sea correcto.

Esta guía es un cuaderno de trabajo. Te acompaña desde que identificás un problema hasta que planificás, construís, probás y mejorás tu herramienta. Podés usarla en clase y también como referencia cuando trabajes por tu cuenta.

Pensar antes de pedir

Ordená el problema y el objetivo antes de escribir el primer prompt.

Documentar antes de construir

Lo que no está escrito, la IA lo va a suponer.

Validar antes de publicar

Que el código exista no significa que funcione bien.

Qué vas a encontrar en cada capítulo

Bloque	Para qué sirve
Qué vas a aprender	Los objetivos del capítulo.
Ejemplo	Un caso realista para entender el concepto.
Prompt listo para usar	Instrucciones que podés copiar y adaptar. Los textos entre [corchetes] se reemplazan con tu información.
Atención	Advertencias de seguridad, límites y errores frecuentes.
Ejercicio	Una actividad para aplicar el tema a tu proyecto, con espacio para escribir.
Video demostrativo	Un espacio para el enlace al video de la docente y su actividad asociada.
Antes de avanzar	Lo que debería quedar resuelto antes del capítulo siguiente.

UNA ACLARACIÓN IMPORTANTE

Las funciones de Claude y Codex cambian según la versión, el plan contratado y el entorno donde los uses (chat, aplicación de escritorio, terminal, editor de código). Cuando esta guía menciona una función específica, verificá en la documentación oficial vigente si está disponible para vos.

Del problema a la herramienta: las 10 etapas

IDEA organiza cómo pensás tu proyecto. Las diez etapas organizan cuándo hacés cada cosa. No son una secuencia rígida: una prueba puede mostrarte que falta una regla y obligarte a volver al diseño. Eso es normal. Lo importante es que cada cambio relevante quede documentado.

#	Etapas	Pregunta clave	IDEA
1	Descubrimiento del problema	¿Qué me hace perder tiempo, dinero o clientes?	I
2	Relevamiento del proceso actual	¿Cómo lo hago hoy, paso a paso?	I
3	Recopilación de contexto y referencias	¿Qué materiales, reglas y ejemplos tengo?	I · D
4	Definición de requisitos	¿Qué tiene que hacer sí o sí la herramienta?	D
5	Diseño funcional	¿Qué pantallas, datos y recorridos va a tener?	D
6	Evaluación técnica	¿Con qué tecnología, dónde se aloja y cuánto cuesta?	E
7	Construcción por etapas	¿Qué parte construyo primero?	E
8	Pruebas y correcciones	¿Funciona en casos normales, con errores y en los límites?	A
9	Preparación para la publicación	¿Es seguro usarla con datos reales?	A
10	Documentación y mantenimiento	¿Cómo la actualizo y quién se ocupa?	A

Entender qué podemos crear

QUÉ VAS A APRENDER

- Qué tipo de herramientas se pueden desarrollar con ayuda de Claude y Codex.
- Por qué no siempre conviene reemplazar una plataforma completa.
- Cómo elegir una tarea propia para empezar.

Claude y Codex son asistentes de inteligencia artificial que pueden ayudarte a analizar un problema, ordenar requisitos, escribir código, revisarlo y explicarlo. No adivinan cómo funciona tu negocio ni reemplazan tus decisiones: trabajan con lo que les explicás.

Con un buen método, una persona sin formación técnica puede construir herramientas útiles y acotadas. Estos son algunos ejemplos:

Herramienta	Qué resuelve · Ejemplo
Cotizador de precios	Calcula un presupuesto a partir de opciones. Una imprenta que cotiza según cantidad, papel y terminación.
Sistema de costos	Suma insumos, horas y gastos para conocer el costo real. Una pastelería que calcula el costo por torta.
Agenda y turnero	Muestra horarios libres y evita superposiciones. Un consultorio de kinesiología.
CRM a medida	Ordena clientes, oportunidades y seguimientos. Una consultora que registra propuestas y etapas comerciales.
Directorio digital	Publica fichas con filtros y búsqueda. Una cámara de comercio con sus empresas socias.
Formulario inteligente	Cambia las preguntas según las respuestas y valida datos. Un estudio contable que recibe consultas.
Inscripción con cupos	Controla vacantes y listas de espera. Un taller de cerámica con grupos de 12 personas.
Panel de gestión	Reúne indicadores y pendientes en una pantalla. Una tienda que sigue pedidos y entregas.
Automatización de tareas	Ejecuta pasos repetitivos. Generar un recordatorio cuando una factura está por vencer.
Conector de procesos	Hace que dos herramientas existentes se comuniquen. Pasar las respuestas de un formulario a una planilla y avisar por correo.

No siempre hay que reemplazarlo todo

Un error habitual es querer construir “un sistema completo” desde el primer día. Muchas veces conviene empezar por una parte del proceso:

- **Desarrollar un módulo:** agregar un cotizador a la planilla que ya usás, en lugar de reemplazar tu sistema de ventas.
- **Conectar herramientas:** enviar los datos de tu formulario a tu CRM sin cargarlos a mano.
- **Automatizar una tarea:** generar el resumen semanal de turnos en vez de armarlo vos.

Empezar chico reduce riesgos, costos y tiempo, y te permite comprobar rápido si la herramienta sirve.

EJEMPLO

Laura tiene un centro de estética y quería “una app para todo”. Al analizar su semana, descubrió que el mayor problema era confirmar turnos por WhatsApp uno por uno. Su primer proyecto fue una agenda que muestra horarios libres y genera el mensaje de confirmación. El resto quedó para una segunda etapa.

Errores frecuentes

- Elegir un proyecto demasiado grande para empezar.
- Pensar en la tecnología antes que en el problema.
- Copiar una herramienta ajena sin preguntarse si sirve para tu forma de trabajar.

EJERCICIO · ELEGÍ TU TAREA

Pensá en tu última semana de trabajo y respondé:

1. ¿Qué tarea repetís seguido y te consume tiempo?

2. ¿Qué errores aparecen en esa tarea?

3. ¿Qué tipo de herramienta de la tabla se parece a lo que necesitás?

4. ¿Qué parte podrías resolver primero, sin reemplazar todo?

ANTES DE AVANZAR

- ☐ Elegí una tarea concreta para trabajar durante el curso.
- ☐ Definí si es un módulo, una conexión o una automatización.

QUÉ VAS A APRENDER

- Los cuatro componentes del método IDEA.
- La diferencia entre una instrucción vaga y una precisa.
- Cómo aplicar cada componente a tu proyecto.

IDEA es una forma de organizar el trabajo. No es un lenguaje de programación ni una función de Claude o Codex: es una lista ordenada de preguntas que te ayuda a no olvidar nada importante antes de pedir que se construya algo.

I**Intención**

Problema, objetivo, resultado esperado.

D**Diseño**

Usuarios, funciones, datos, pantallas, recorrido.

E**Ejecución**

Reglas, validaciones, permisos, integraciones.

A**Aseguramiento**

Pruebas, errores, criterios de aceptación.

I · Intención

Definí qué problema querés resolver, para qué y cómo vas a saber que lo lograste. Si la intención no está clara, la IA va a completar los huecos con suposiciones.

Instrucción vaga

Haceme un sistema para mi negocio.

Instrucción precisa

Tengo una pastelería. Hoy calculo los precios en una planilla y tardo 20 minutos por pedido. Quiero una herramienta que, al cargar los productos y las cantidades, me muestre el costo y el precio sugerido en menos de un minuto.

ACTIVIDAD · MI INTENCIÓN

Completá: “Hoy [cómo lo hago], eso me genera [problema]. Quiero [resultado] para [beneficio]”.

D • Diseño

Describí quién va a usar la herramienta, qué va a poder hacer, qué datos necesita y por qué pantallas pasa. No hace falta dibujar con precisión: alcanza con explicar el recorrido.

- **Usuarios:** quiénes la usan y qué rol tienen (dueña, empleado, cliente).
- **Funciones:** qué acciones permite (cargar, editar, buscar, exportar).
- **Datos:** qué información se guarda (nombre, fecha, precio, estado).
- **Pantallas y recorrido:** por dónde entra cada usuario y qué ve en cada paso.

Instrucción vaga	Instrucción precisa
Que tenga una parte para clientes y otra para mí.	Hay dos usuarios. El cliente elige productos y ve el total, sin ver costos. Yo, como administradora, cargo insumos, edito precios y veo el margen de cada pedido.

ACTIVIDAD • MI DISEÑO

Anotá dos usuarios, tres funciones y cinco datos que tu herramienta necesita.

E • Ejecución

Establecé las reglas que el sistema debe respetar siempre. Acá aparecen las condiciones del negocio, las validaciones (comprobaciones automáticas antes de guardar un dato), los permisos y las conexiones con otras herramientas.

Instrucción vaga	Instrucción precisa
Que no se puedan cargar cosas mal.	La cantidad debe ser un número entero mayor que cero. Un pedido no se puede guardar sin fecha de entrega. Solo la administradora puede modificar el margen. Si un insumo no tiene precio, mostrar un aviso y no calcular.

ACTIVIDAD · MIS REGLAS

Escribí tres reglas que tu herramienta nunca debe romper, con la forma “Siempre...” o “Nunca...”.

A · Aseguramiento

Definí cómo vas a comprobar que funciona. Los **criterios de aceptación** son condiciones verificables: si se cumplen, la función está lista; si no, hay que corregir.

Instrucción vaga	Instrucción precisa
Fijate que ande bien.	Con 2 kg de harina a \$1.000 el kilo y un recargo del 40 % sobre el costo (regla confirmada por la dueña), el precio sugerido debe dar \$2.800. Si cargo cantidad 0, debe aparecer el mensaje “La cantidad debe ser mayor que cero”.

ACTIVIDAD · MIS CRITERIOS

Escribí dos criterios de aceptación con datos concretos: “Si hago X, debe pasar Y”.

Errores frecuentes

- Saltar de la intención a la construcción sin pasar por el diseño.
- Dejar las reglas “en la cabeza” y no escribirlas.
- Dar por buena una función porque “se ve bien”, sin probarla con datos.

ANTES DE AVANZAR

- ☐ Completé las cuatro actividades IDEA para mi proyecto.
- ☐ Tengo al menos un criterio de aceptación con datos concretos.

Cómo estructurar un buen prompt

QUÉ VAS A APRENDER

- Los once elementos que puede tener una instrucción.
- Cuándo usar todos y cuándo alcanza con pocos.
- Cómo completar la plantilla maestra.

Un **prompt** es la instrucción que le das a la IA. Cuanto más claro sea, menos tiene que adivinar. No todos los prompts necesitan todos los elementos: para una pregunta puntual alcanza con dos o tres. Para un proyecto que involucra datos, usuarios y reglas, conviene una definición completa.

Elemento	Qué responde	Ejemplo breve
Rol	¿Desde qué perspectiva debe trabajar la IA?	Actuá como analista funcional.
Contexto	¿Qué tiene que saber de mi negocio?	Tengo un taller con 3 profesoras.
Problema	¿Qué no funciona hoy?	Se superponen inscripciones.
Objetivo	¿Qué quiero lograr?	Controlar cupos automáticamente.
Usuarios	¿Quién la usa y con qué permisos?	Alumnas y administradora.
Funciones	¿Qué debe permitir hacer?	Inscribirse, ver vacantes, cancelar.
Datos	¿Qué información se guarda?	Nombre, correo, curso, fecha.
Reglas	¿Qué condiciones se cumplen siempre?	Máximo 12 personas por grupo.
Restricciones	¿Qué límites hay que respetar?	Sin costos mensuales; debe verse bien en el celular.
Resultado esperado	¿Qué quiero recibir y en qué formato?	Una propuesta funcional en lista.
Criterios de aceptación	¿Cómo sé que está bien?	La inscripción 13 pasa a lista de espera.

CUÁNTO DETALLE HACE FALTA

Consulta simple (“¿qué es una base de datos?”): objetivo y resultado esperado.

Tarea acotada (“revisá esta fórmula”): contexto, objetivo, datos y resultado esperado.

Proyecto (“construí un turnero”): los once elementos, idealmente apoyados en una ficha de contexto (capítulo 4).

Plantilla maestra

Copíá esta estructura, reemplazá lo que está entre corchetes y borrá los bloques que no apliquen.

PROMPT LISTO PARA USAR · PLANTILLA MAESTRA

Rol: Actuá como [analista funcional / desarrollador / revisor].

Contexto: Mi negocio es [descripción breve]. Hoy trabajo así: [proceso actual].

Problema: [qué no funciona o qué me hace perder tiempo].

Objetivo: Quiero [qué debe lograr la herramienta].

Usuarios: [quiénes la usan y qué puede hacer cada uno].

Funciones: [lista de acciones].

Datos: [qué información se carga, se guarda o se muestra].

Reglas: [condiciones que siempre deben cumplirse].

Restricciones: [presupuesto, tecnología, plazos, dispositivos, lo que no se debe tocar].

Resultado esperado: Entregame [propuesta / plan / código / lista de preguntas] en formato [lista / tabla / pasos].

Criterios de aceptación: Voy a considerar que está bien si [condiciones verificables].

Antes de empezar, si falta información importante, haceme las preguntas necesarias. Diferenciá lo que sabés por mi información de lo que estás suponiendo.

Plantilla rellenable

Rol Ej.: analista funcional	
Contexto Ej.: centro de estética, 2 profesionales	
Problema Ej.: turnos superpuestos	
Objetivo	
Usuarios	
Funciones	

Datos	
Reglas	
Restricciones	
Resultado esperado	
Criterios de aceptación	

Errores frecuentes

- Escribir todo en un solo párrafo largo, sin separar elementos.
- Pedir el resultado sin decir en qué formato lo querés.
- Olvidar las restricciones: la IA puede proponer tecnologías con costo mensual.
- No invitar a la IA a preguntar cuando falta información.

▶ VIDEO DEMOSTRATIVO 03

Cómo redactar un prompt inicial

Enlace	[Agregar enlace del video]
Qué vas a aprender	A pasar de una idea a un prompt inicial usando la plantilla maestra.
Aplicá lo aprendido	Completá la plantilla rellenable con tu proyecto y registrala en el Laboratorio de prompts.

ANTES DE AVANZAR

- ☐ Completé la plantilla rellenable con mi proyecto.
- ☐ Identifiqué qué elementos todavía no sé completar.

04

El contexto: enseñar a la IA cómo funciona mi negocio

QUÉ VAS A APRENDER

- Por qué la IA no conoce tu negocio aunque “sepa mucho”.
- Cómo armar una ficha de contexto reutilizable.
- Cómo mantenerla actualizada entre conversaciones y sesiones.

La IA conoce mucha información general, pero no sabe cómo cotizás, qué clientes priorizás, qué herramientas usás ni qué excepciones aceptás. Si no se lo explicás, va a usar criterios genéricos que pueden no servirte.

La **ficha de contexto** es un documento breve que describe tu negocio y tu proyecto. La escribís una vez, la mejorás con el tiempo y la compartís al inicio de cada trabajo importante.

EJEMPLO · SIN CONTEXTO Y CON CONTEXTO

Sin contexto: “Armá un cotizador de eventos”. La IA propone precio por persona y un recargo por fin de semana, que quizás no apliques.

Con contexto: “Cobro un valor base por salón, más un adicional por cada hora extra después de las 4 horas. El catering lo cotiza un proveedor externo y yo sumo un porcentaje que defino en cada evento”. Ahora la propuesta respeta tu forma de trabajar.

Plantilla · Ficha de contexto

Descripción del negocio Qué hacés, para quién, tamaño del equipo.	
Proceso actual Paso a paso, como lo hacés hoy.	
Problema	
Objetivo	

Herramientas utilizadas Planillas, CRM, mensajería, web.	
Información disponible Qué datos ya tenés y dónde están.	
Reglas de trabajo	
Usuarios y permisos	
Restricciones	
Resultado esperado	
Referencias disponibles	
Decisiones pendientes Lo que todavía no definiste.	
Versión y fecha	

Mantener el contexto vivo

- **Actualizala cuando cambie algo importante:** una regla nueva, un precio, un usuario. Anotá la fecha y qué cambió.
- **Guardala como archivo:** un documento de texto que puedas adjuntar o pegar al inicio de cada sesión.
- **Usá archivos de instrucciones del proyecto cuando el entorno lo permita.** Algunas herramientas leen un archivo con instrucciones permanentes ubicado en la carpeta del proyecto (por ejemplo, Claude Code usa CLAUDE.md y Codex usa AGENTS.md). Otras ofrecen “proyectos” con documentos persistentes. Verificá en la documentación vigente qué admite tu plan y tu entorno.

ATENCIÓN

No todas las conversaciones ni todos los entornos conservan la información. Una conversación nueva puede empezar sin nada de lo que explicaste antes. Ante la duda, volvé a compartir la ficha de contexto.

PROMPT LISTO PARA USAR · COMPLETAR MI FICHA DE CONTEXTO

Te comparto la ficha de contexto de mi negocio. Revisala y decime:

1. Qué información falta para poder diseñar [nombre de la herramienta].
2. Qué partes son ambiguas o se pueden interpretar de más de una manera.
3. Qué reglas de negocio te parece que existen pero no están escritas.

No propongas soluciones todavía. Respondé con una lista de preguntas ordenadas por importancia.

[Pegar ficha de contexto]

Errores frecuentes

- Suponer que la IA “ya sabe” porque se lo contaste en otra conversación.
- Escribir el contexto con jerga interna sin explicarla.
- No registrar las decisiones pendientes: la IA las va a resolver por su cuenta.

ANTES DE AVANZAR

- ☐ Tengo una primera versión de mi ficha de contexto, con fecha.
- ☐ Listé las decisiones pendientes.
- ☐ Sé dónde voy a guardar la ficha para reutilizarla.

Cómo utilizar archivos y referencias

QUÉ VAS A APRENDER

- Qué puede analizar la IA a partir de cada tipo de material.
- Qué no puede inferir con certeza.
- Cómo preparar y compartir referencias de forma segura.

Un archivo bien elegido vale más que muchas explicaciones. Pero cada material muestra solo una parte de la realidad: una captura de pantalla muestra cómo se ve una pantalla, no qué reglas hay detrás.

Material	Qué se puede analizar	Qué no se puede inferir con certeza
Planillas	Columnas, fórmulas visibles, tipos de datos, cálculos.	Por qué se usa cada fórmula; excepciones que aplicás a mano.
Documentos	Procedimientos, políticas, condiciones comerciales.	Si siguen vigentes o se aplican en la práctica.
Capturas de pantalla	Campos, botones, organización visual, textos.	Qué pasa al hacer clic; validaciones; datos que no se ven.
Formularios	Preguntas, campos obligatorios, opciones.	Qué se hace con las respuestas después.
Sitios de referencia	Estructura, recorrido, funciones visibles.	Cómo funciona por dentro; costos; si es adecuado para tu caso.
CRM y plataformas	Etapas, campos, exportaciones, configuraciones visibles.	Automatizaciones ocultas, límites del plan, integraciones disponibles.
Ejemplos de datos	Formatos, casos habituales, problemas de calidad.	Casos raros que no aparecen en la muestra.
Código y documentación técnica	Estructura, funciones, dependencias, posibles errores.	Cómo se comporta en producción si no se ejecuta ni se prueba.

NUNCA COMPARTAS

Contraseñas, credenciales de acceso, claves privadas, claves de API, datos bancarios o información confidencial que no sea necesaria para la tarea. Si un archivo las contiene, eliminalas o reemplazalas por valores de ejemplo antes de compartirlo.

Lista de preparación de referencias

- ☐ Identifiqué los materiales útiles.
- ☐ Eliminé o anonimicé datos personales innecesarios.
- ☐ Verifiqué que puedo compartir los archivos.
- ☐ Expliqué qué debe analizar la IA.
- ☐ Definí qué resultado espero recibir.
- ☐ Pedí que diferencie hechos comprobados de suposiciones.

PROMPT LISTO PARA USAR · AUDITAR UNA HERRAMIENTA EXISTENTE SIN MODIFICARLA

Actuá como analista funcional. Te comparto [planilla / capturas / exportación / código] de la herramienta que uso hoy para [tarea].

No modifiques nada ni propongas cambios todavía. Solo analizá y entregame:

1. Qué hace la herramienta, paso a paso.
2. Qué datos maneja y cómo se relacionan.
3. Qué reglas o cálculos detectás.
4. Qué problemas, inconsistencias o riesgos ves.
5. Qué no podés saber con certeza a partir de este material.

Separá tu respuesta en tres grupos: **hechos comprobados** en el material, **suposiciones** y **preguntas** que necesitás que yo responda.

Errores frecuentes

- Compartir una planilla completa con datos reales de clientes cuando alcanzaba con diez filas de ejemplo.
- Enviar capturas sin explicar qué hay que mirar.
- Tomar como cierto lo que la IA “dedujo” de una imagen.

► VIDEO DEMOSTRATIVO 02

Cómo analizar una planilla, un CRM o una plataforma

Enlace

[Agregar enlace del video]

Qué vas a aprender

A pedir un análisis de una herramienta existente separando hechos, suposiciones y preguntas.

Aplicá lo aprendido

Prepará una referencia con la lista anterior y usá el prompt de auditoría.

EJERCICIO · MIS REFERENCIAS

Listá tres materiales que podés compartir y qué esperás que la IA analice en cada uno.

ANTES DE AVANZAR

- ☐ Tengo mis referencias preparadas y sin datos sensibles.
- ☐ Hice una auditoría de mi herramienta actual, si existe.

Cómo promptear por voz

QUÉ VAS A APRENDER

- A empezar un proyecto contando tu idea con tus propias palabras.
- Tres prompts listos para dictar.
- A completar la información faltante con preguntas sucesivas.

No hace falta tener todo claro para empezar. Muchas personas explican mejor hablando que escribiendo. Podés dictar tu idea tal como te sale, con desorden y repeticiones, y pedirle a la IA que la organice.

Según el dispositivo, podés usar el dictado del teclado del celular, el dictado del sistema operativo o el modo de voz de la aplicación, si está disponible en tu plan. El resultado es el mismo: texto que la IA puede procesar.

CONSEJOS PARA DICTAR

- Contá cómo trabajás hoy, como si se lo explicaras a alguien que empieza a trabajar con vos.
- Mencioná ejemplos reales: “la semana pasada me pasó que...”.
- No te preocupes por el orden: el prompt se encarga de eso.
- Revisá la transcripción antes de enviarla: los nombres y números suelen transcribirse mal.

PROMPT PARA DICTAR 1 · ORDENAR UNA IDEA DESORGANIZADA

Te voy a contar una idea de forma desordenada. Tu tarea es organizarla, no construir nada todavía. Después de escucharme, entregame: el problema que quiero resolver, el objetivo, quiénes usarían la herramienta, qué debería hacer, qué información maneja y qué dudas detectás. Si algo no queda claro, marcalo como pendiente en lugar de suponerlo. Mi idea es esta: [dictar].

PROMPT PARA DICTAR 2 · CONVERTIR UNA EXPLICACIÓN EN UNA ESPECIFICACIÓN FUNCIONAL

Actuá como analista funcional. Con lo que te voy a explicar, armá una especificación funcional organizada con el método IDEA: intención, diseño, ejecución y aseguramiento. Incluí usuarios y permisos, funciones, datos, reglas, validaciones y criterios de aceptación. Al final, listá las preguntas que necesitás que responda antes de construir. No inventes reglas de negocio: si falta una, preguntá. Mi explicación es esta: [dictar].

PROMPT PARA DICTAR 3 · ANALIZAR UNA HERRAMIENTA EXISTENTE

Te voy a describir cómo uso hoy [planilla / sistema / plataforma] para [tarea]. Analizá mi descripción y decime qué partes del proceso funcionan, cuáles generan errores o demoras, qué se podría automatizar y qué conviene mantener como está. Diferenciá lo que te estoy contando de lo que estás interpretando. No propongas una herramienta nueva todavía. Mi descripción es esta: [dictar].

Preguntas sucesivas

Después del primer ordenamiento, pedile a la IA que te entreviste. Así vas completando lo que falta sin tener que pensarlo todo de antemano.

PROMPT LISTO PARA USAR · ENTREVISTA GUIADA

Haceme preguntas de a una para completar la especificación de mi proyecto. Empezá por las que más afectan el resultado. Después de cada respuesta, actualizá un resumen breve y pasá a la siguiente pregunta. Cuando consideres que hay información suficiente para una primera versión, avisame y mostrá el resumen completo.

Errores frecuentes

- Enviar la transcripción sin revisarla.
- Pedir que construya directamente a partir de un audio desordenado.
- Responder “lo que te parezca” a preguntas sobre reglas del negocio.

► VIDEO DEMOSTRATIVO 01

De una idea hablada a un proyecto

Enlace

[Agregar enlace del video]

Qué vas a aprender

A dictar una idea y convertirla en una especificación funcional con preguntas sucesivas.

Aplicá lo aprendido

Completá el registro de esta página con tu propio proyecto.

EJERCICIO · REGISTRO DE MI IDEA HABLADA

Mi explicación inicial (resumen o transcripción)

Preguntas que surgieron

Versión organizada del proyecto

ANTES DE AVANZAR

- ☐ Tengo una versión organizada de mi idea.
- ☐ Respondí o registré como pendientes las preguntas importantes.

QUÉ VAS A APRENDER

- Para qué tipo de tareas conviene usar cada herramienta.
- El ciclo de siete pasos para trabajar por etapas.
- Prompts específicos para cada paso.

Claude y Codex pueden ayudarte a analizar requisitos, leer archivos, planificar cambios, escribir código y revisarlo. Lo que pueden hacer en concreto depende del producto, el plan y el entorno: no es lo mismo una conversación en el navegador que un asistente con acceso a la carpeta de tu proyecto o a una terminal.

VERIFICÁ ANTES DE EMPEZAR

¿Puede leer archivos? ¿Puede crear o modificar archivos en tu computadora? ¿Puede ejecutar código y pruebas? ¿Conserva información entre sesiones? Las respuestas cambian según la versión. Revisá la documentación oficial y, ante la duda, preguntale a la propia herramienta qué puede hacer en ese entorno.

El ciclo de trabajo por etapas

Pedir “hacé todo” suele producir resultados difíciles de revisar. Trabajar por etapas te permite controlar cada paso y detectar errores a tiempo.

1

Analizar

2

Planificar

3

Aprobar

4

Implementar

5

Probar

6

Revisar

7

Documentar

El paso 3 es tuyo: nada se construye hasta que vos apruebes el plan.

1 · ANALIZAR

Leé la ficha de contexto y [los archivos del proyecto]. Explicame con tus palabras qué entendiste del objetivo, qué existe hoy y qué falta. No escribas código. Terminá con las preguntas que necesitás resolver.

2 · PLANIFICAR

Proponé un plan de desarrollo por etapas para [función o herramienta]. Para cada etapa indicá qué se construye, qué archivos se crean o modifican, cómo se va a probar y qué riesgos ves. Si hay alternativas técnicas, compará costos, complejidad y mantenimiento. No implementes nada hasta que apruebe el plan.

3 · APROBAR

Apruebo el plan con estos cambios: [cambios]. Empezá solamente por la etapa 1. Al terminar, detenete y mostrame qué hiciste antes de seguir.

4 · IMPLEMENTAR

Implementá la etapa [número] del plan aprobado. No modifiques funciones que ya funcionan fuera del alcance de esta etapa. Al terminar, listá los archivos que cambiaste y explicá cada cambio en lenguaje sencillo.

5 · PROBAR

Proponé casos de prueba para lo que implementaste: casos normales, errores esperables y situaciones límite. Si en este entorno podés ejecutar pruebas, ejecutalas e informá el resultado real. Si no podés, indicalo claramente y decime cómo las pruebo yo.

6 · REVISAR

Revisá lo construido como si fueras otra persona. Buscá errores, reglas de negocio no respetadas, problemas de seguridad, datos sin validar y partes difíciles de mantener. Ordená los hallazgos por gravedad. No corrijas nada todavía.

7 · DOCUMENTAR

Actualizá la documentación del proyecto con lo realizado en esta etapa: qué funciones existen, qué reglas aplican, qué decisiones tomamos, qué quedó pendiente y cómo se prueba. Escribilo para que otra persona pueda continuar el trabajo.

Errores frecuentes

- Saltear la aprobación y dejar que implemente todo de una vez.
- Aceptar que “ya está probado” sin ver qué pruebas se hicieron.
- No pedir la lista de archivos modificados.

► VIDEO DEMOSTRATIVO 06

Cómo trabajar con Claude o Codex sobre un proyecto

Enlace

[Agregar enlace del video]

Qué vas a aprender

A recorrer el ciclo analizar-planificar-aprobar-implementar sobre un proyecto real.

Aplicá lo aprendido

Pedí un plan por etapas para tu proyecto y anotalo en el plan de desarrollo del capítulo 12.

ANTES DE AVANZAR

- ☐ Verifiqué qué puede hacer la herramienta en mi entorno.
- ☐ Tengo un plan por etapas aprobado por mí.

Cómo construir herramientas reales

QUÉ VAS A APRENDER

- A aplicar el método completo en cuatro casos frecuentes.
- Qué preguntar, qué reglas definir y cómo probar cada herramienta.
- A adaptar un caso modelo a tu negocio.

Los cuatro casos son didácticos: los datos y las reglas son ejemplos. Cada caso sigue la misma estructura para que puedas compararlos y elegir el más cercano a tu proyecto.

ATENCIÓN

La IA no debe inventar fórmulas comerciales, criterios fiscales, políticas de cancelación ni condiciones legales. Todas esas reglas tienen que salir de vos o de un profesional que las valide.

Sistema de costos y cotizador

Problema inicial	Un taller de carpintería cotiza muebles a mano. Cada presupuesto tarda y dos personas pueden dar precios distintos para el mismo trabajo.
Contexto que necesita la IA	Lista de materiales y precios, cómo se calculan las horas de trabajo, gastos fijos, criterio de margen, impuestos que se suman (confirmados por el contador), formato del presupuesto actual.
Preguntas que deben responderse	¿El margen se calcula sobre el costo o sobre el precio de venta? ¿Cómo se reparten los gastos fijos? ¿Se redondean los precios? ¿Quién actualiza los precios de materiales y cada cuánto? ¿Cuánto tiempo es válido un presupuesto?

PROMPT INICIAL

Actúa como analista funcional. Tengo [negocio] y quiero un cotizador que calcule el costo y el precio de [productos o servicios]. Te comparto la planilla que uso hoy y la explicación de mi método de cálculo. Primero, documentá mi método: variables, fórmulas, orden de los cálculos y ejemplos. **No inventes fórmulas ni criterios fiscales:** si algo no está claro, preguntame. Después mostrame un ejemplo resuelto paso a paso para que yo confirme que el cálculo es correcto.

Resultado funcional esperado	Pantalla para elegir producto, medidas y materiales; cálculo de costo directo, gastos asignados, margen y precio final; presupuesto descargable; listado de materiales editable solo por administración.
Reglas principales	Las fórmulas son las documentadas y aprobadas por el usuario. Ningún material sin precio entra al cálculo. Las cantidades deben ser mayores que cero. Cada presupuesto guarda los precios vigentes al momento de emitirlo. Solo administración modifica precios y márgenes.

PROMPT DE CONSTRUCCIÓN POR ETAPAS

Con el método de cálculo aprobado, construí el cotizador en estas etapas, deteniéndote al final de cada una: 1) carga y edición de materiales; 2) cálculo de costo de un producto; 3) aplicación de gastos y margen; 4) generación del presupuesto; 5) historial de presupuestos. En cada etapa, verificá el cálculo con el ejemplo resuelto que aprobé y mostrame el resultado.

Caso de prueba	Resultado esperado
Producto con el ejemplo resuelto aprobado	El precio coincide exactamente con el cálculo manual.
Cantidad igual a cero o negativa	No calcula y muestra un mensaje claro.
Material sin precio cargado	Aviso de material incompleto; no emite presupuesto.
Cambio de precio de un material	Los presupuestos anteriores conservan el precio original.
Usuario sin permiso intenta editar el margen	La acción se bloquea, también desde el servidor.

Errores frecuentes: confundir margen sobre costo con margen sobre precio; redondear en pasos intermedios; dejar que la IA “estime” impuestos; no guardar la versión de precios de cada presupuesto.

ADAPTALO A TU NEGOCIO
¿Qué cotizás? ¿Cuáles son tus variables? Escribí tu fórmula como la aplicás hoy.

Agenda de turnos a medida

Problema inicial	Un centro de estética con dos profesionales agenda por mensaje. Hay turnos superpuestos, olvidos y cancelaciones de último momento.
Contexto que necesita la IA	Servicios y duración de cada uno, tiempo de preparación entre turnos, horarios por profesional, días no laborables, política de modificación y cancelación, si se cobra seña, canales de aviso.
Preguntas que deben responderse	¿El cliente elige profesional o se asigna? ¿Con cuánta anticipación se puede reservar o cancelar? ¿Qué pasa con la seña si cancela? ¿Hay servicios que requieren una cabina o equipo específico? ¿Qué zona horaria se usa?

PROMPT INICIAL

Actuá como analista funcional. Quiero una agenda de turnos para [negocio]. Te comparto los servicios con su duración, los horarios de cada profesional y mis reglas de cancelación. Diseñá el funcionamiento de la agenda: cómo se calcula la disponibilidad, cómo se evita la superposición, qué pasa al modificar o cancelar, y cuándo se envía cada aviso. Listá las reglas que detectás y las preguntas que necesitás que responda. No construyas todavía.

Resultado funcional esperado	Vista de horarios libres por servicio y profesional; reserva con datos mínimos del cliente; modificación y cancelación según política; registro de seña; avisos de confirmación y recordatorio; vista diaria para el equipo.
Reglas principales	Un turno ocupa la duración del servicio más el tiempo de preparación. Dos turnos de la misma profesional no pueden solaparse. La disponibilidad se vuelve a validar al confirmar y al modificar , no solo al mostrar horarios. Un turno cancelado libera el horario. Fuera del plazo de cancelación, se aplica la política definida por el negocio.

PROMPT DE CONSTRUCCIÓN POR ETAPAS

Construí la agenda en etapas, deteniéndote al final de cada una: 1) servicios, profesionales y horarios; 2) cálculo de disponibilidad; 3) reserva con validación de solapamiento en el servidor; 4) modificación y cancelación; 5) señas; 6) avisos. En la etapa 3, demostrá con pruebas que dos reservas simultáneas para el mismo horario no pueden confirmarse ambas.

Caso de prueba	Resultado esperado
Turno de 60 min a las 10:00 y otro a las 10:30 con la misma profesional	El segundo se rechaza por solapamiento.
Turno que termina a las 11:00 con 15 min de preparación	El siguiente disponible es a las 11:15.
Dos personas reservan el mismo horario al mismo tiempo	Solo una reserva se confirma.

Caso de prueba	Resultado esperado
Modificar un turno a un horario ocupado	Se rechaza y el turno original se mantiene.
Cancelar fuera de plazo	Se aplica la política definida y queda registrado.

Errores frecuentes: validar disponibilidad solo en la pantalla; olvidar el tiempo de preparación; no contemplar feriados; prometer avisos por mensajería sin revisar costos y requisitos del proveedor.

ADAPTALO A TU NEGOCIO

Listá tus servicios con su duración y tu política de cancelación.

CRM adaptado a un negocio

Problema inicial	Una consultora de comunicación sigue a sus clientes en planillas, correos y notas sueltas. Se pierden briefs, cambios pedidos y decisiones acordadas.
Contexto que necesita la IA	Etapas comerciales reales (contacto, reunión, propuesta, negociación, cliente activo, cerrado), datos de la ficha de cliente, cómo se evalúa una oportunidad, estructura del brief, roles del equipo.
Preguntas que deben responderse	¿Qué dato identifica a un cliente de forma única? ¿Quién puede ver información económica? ¿Cuándo una oportunidad cambia de etapa? ¿Qué se registra en el historial? ¿Cuánto tiempo se conservan los datos?

PROMPT INICIAL

Actuá como analista funcional. Quiero un CRM para [negocio] que respete mi forma de trabajar. Te comparto mis etapas comerciales, la ficha de cliente que uso y un brief de ejemplo (anonimizado). Diseñá: entidades y datos, recorrido de una oportunidad, registro de modificaciones y decisiones, tareas y recordatorios, roles y permisos. Proponé un criterio para evitar clientes duplicados y explicame ventajas y riesgos. No construyas todavía.

Resultado funcional esperado	Fichas de clientes y contactos; tablero de oportunidades por etapa; evaluación de cada oportunidad; briefs vinculados; registro de modificaciones solicitadas; tareas con responsable y fecha; historial de decisiones con autor y fecha.
Reglas principales	Antes de crear un cliente se buscan coincidencias (por ejemplo, CUIT o correo) y se avisa si existe. Cada cambio de etapa queda en el historial. Las decisiones no se borran: se corrigen con una nueva entrada. Los permisos se controlan en el servidor. Solo se guardan los datos personales necesarios, con acceso restringido.

PROMPT DE CONSTRUCCIÓN POR ETAPAS

Construí el CRM en etapas, deteniéndote al final de cada una: 1) usuarios, roles y permisos; 2) clientes y contactos con control de duplicados; 3) oportunidades y etapas; 4) briefs y modificaciones; 5) tareas; 6) historial de decisiones. En cada etapa, probá que un usuario sin permiso no pueda ver ni modificar lo que no le corresponde.

Caso de prueba	Resultado esperado
Crear un cliente con un CUIT ya registrado	Aviso de posible duplicado con enlace al registro existente.
Mover una oportunidad de “Propuesta” a “Negociación”	Queda registrado quién, cuándo y desde qué etapa.
Usuario sin rol comercial abre montos de una propuesta	Acceso denegado.

Caso de prueba	Resultado esperado
Intentar borrar una decisión del historial	No se permite; se ofrece agregar una corrección.

Errores frecuentes: copiar las etapas de un CRM genérico; ocultar botones sin bloquear permisos en el servidor; guardar datos personales “por las dudas”; no definir qué pasa con clientes duplicados ya existentes.

ADAPTALO A TU NEGOCIO

Escribí tus etapas comerciales y qué condición marca el paso de una a otra.

Directorio digital

Problema inicial	Una red de emprendedoras tiene un listado en una planilla compartida. Está desactualizado, no se puede buscar por rubro y cualquiera puede modificarlo.
Contexto que necesita la IA	Campos de cada ficha, categorías y subcategorías, filtros necesarios, quién carga y quién aprueba, qué datos son públicos y cuáles privados, frecuencia de actualización.
Preguntas que deben responderse	¿Cada integrante edita su propia ficha? ¿Los cambios requieren aprobación? ¿Qué pasa con fichas sin actualizar en un año? ¿Se pueden tener varias categorías? ¿Hay consentimiento para publicar los datos?

PROMPT INICIAL

Actuá como analista funcional. Quiero un directorio digital para [organización]. Te comparto la estructura de la planilla actual (sin datos reales) y las categorías que usamos. Diseñá: campos de la ficha, separación entre datos públicos y privados, categorías y filtros, perfiles de usuario, flujo de alta, actualización y aprobación. Señalá qué decisiones sobre privacidad y consentimiento tengo que tomar yo.

Resultado funcional esperado	Página pública con buscador y filtros; fichas con datos públicos; panel privado para que cada integrante edite su ficha; cola de aprobación para administración; aviso de fichas desactualizadas.
Reglas principales	Solo se publican fichas aprobadas. Los datos privados nunca se envían a la página pública. Cada integrante edita solo su propia ficha. Los cambios en campos sensibles vuelven a aprobación. Se registra fecha de última actualización.

PROMPT DE CONSTRUCCIÓN POR ETAPAS

Construí el directorio en etapas, deteniéndote al final de cada una: 1) estructura de fichas y categorías; 2) carga inicial desde la planilla de ejemplo; 3) página pública con búsqueda y filtros; 4) acceso privado y edición de la propia ficha; 5) aprobaciones; 6) avisos de actualización. En la etapa 3, demostrá que ningún dato privado aparece en la página pública ni en sus respuestas del servidor.

Caso de prueba	Resultado esperado
Ficha nueva sin aprobar	No aparece en la búsqueda pública.
Filtrar por dos categorías	Muestra solo las fichas que cumplen el criterio acordado.
Integrante intenta editar una ficha ajena	Acceso denegado.
Revisar el código de la página pública	No contiene teléfonos ni correos marcados como privados.

Errores frecuentes: ocultar datos privados con diseño pero enviarlos igual al navegador; no definir si los filtros suman o restringen; publicar datos sin consentimiento; no prever qué hacer con fichas abandonadas.

ADAPTALO A TU NEGOCIO

¿Qué datos de tu directorio serían públicos y cuáles privados?

- ANTES DE AVANZAR
- ☐

Elegí el caso más parecido a mi proyecto.
- ☐

Adapté sus reglas y casos de prueba a mi negocio.

09

Cómo iterar un prompt y corregir resultados

QUÉ VAS A APRENDER

- Un procedimiento de siete pasos para corregir sin empezar de cero.
- A detectar si falta contexto, una regla o una instrucción.
- A pedir cambios específicos que no rompan lo que ya funciona.

La primera respuesta casi nunca es la definitiva. Iterar es parte del trabajo, no una señal de que lo hiciste mal. Lo importante es que cada corrección te acerque al objetivo y no genere errores nuevos.

1. **Compará** el resultado con el objetivo y los criterios de aceptación.
2. **Identificá** la diferencia concreta: qué esperabas y qué obtuviste.
3. **Determiná la causa:** ¿faltó contexto, una regla o una instrucción clara?
4. **Pedí un cambio específico**, con un ejemplo si es posible.
5. **Protegé lo que funciona:** indicá qué no se debe modificar.
6. **Volvé a probar** con los mismos casos que fallaron, y con los que ya funcionaban.
7. **Documentá** la versión validada en tu laboratorio de prompts.

► VIDEO DEMOSTRATIVO 04

Cómo interpretar una respuesta incompleta

Enlace

[Agregar enlace del video]

Qué vas a aprender

A comparar una respuesta con el objetivo y encontrar qué falta (pasos 1 a 3).

Aplicá lo aprendido

Tomá una respuesta propia y completá los pasos 1 a 3 en el registro de iteraciones.

Ejemplo comparativo

1. Primera instrucción	"Agregá a mi agenda la posibilidad de cancelar turnos".
2. Respuesta incompleta	Se agregó un botón "Cancelar" que borra el turno. No hay plazo mínimo, no queda registro y no se libera la seña según la política.
Diagnóstico	Faltaron reglas: plazo, qué pasa con la seña y que el turno no se borre sino que cambie de estado.

3. Iteración	“La cancelación funciona, pero necesito tres ajustes, sin modificar la reserva ni la vista diaria: (a) el turno no se borra, pasa a estado ‘Cancelado’ con fecha y quién canceló; (b) si faltan menos de 24 horas, mostrar aviso de que la seña no se reintegra; (c) el horario queda libre para nuevas reservas. Mostrame los archivos que cambiás y proponé pruebas para los tres puntos”.
4. Resultado esperado	Turno cancelado con historial, aviso según plazo, horario disponible y pruebas superadas para los tres casos. La reserva sigue funcionando como antes.

PROMPT LISTO PARA USAR · CORRECCIÓN PUNTUAL

Esperaba: [resultado esperado]. Obtuve: [resultado actual]. La diferencia es: [diferencia concreta]. Creo que falta [contexto / regla / instrucción]: [explicación]. Corregí solamente esto. No modifiques [funciones que ya funcionan]. Antes de cambiar, explicame en dos o tres líneas qué vas a hacer.

► VIDEO DEMOSTRATIVO 05

Cómo iterar y mejorar instrucciones

Enlace	[Agregar enlace del video]
Qué vas a aprender	A escribir una iteración específica y validar el resultado (pasos 4 a 7).
Aplicá lo aprendido	Usá el prompt de corrección puntual y registrá la versión final en el Laboratorio de prompts.

Errores frecuentes

- Responder “no funciona, arreglalo” sin explicar qué esperabas.
- Pedir varias correcciones distintas en el mismo mensaje.
- Probar solo lo que se corrigió y no lo que ya funcionaba.
- No guardar la versión del prompt que finalmente dio resultado.

EJERCICIO · MI PRIMERA ITERACIÓN

Esperaba

Obtuve

Faltó (contexto / regla / instrucción)

Mi pedido de cambio

10

Cómo reducir iteraciones innecesarias y optimizar tokens

QUÉ VAS A APRENDER

- Qué son los tokens, en términos sencillos.
- Por qué optimizar no significa escribir siempre menos.
- Ocho hábitos para trabajar con menos idas y vueltas.

Los **tokens** son los fragmentos en que la IA divide el texto para procesarlo: pueden ser palabras completas, partes de palabras o signos. Todo cuenta: lo que escribís, los archivos que compartís, el historial de la conversación y lo que la IA responde. Según el plan, los tokens se relacionan con límites de uso o con costos.

Optimizar no es escribir prompts cortos. Un prompt largo y claro puede ahorrar cinco correcciones. Lo que se busca es evitar ambigüedades, información irrelevante, repeticiones y errores que se podían prever.

Hábito	Cómo aplicarlo
Compartir el contexto relevante desde el principio	Empezá con la ficha de contexto en lugar de ir agregando datos a medida que aparecen errores.
Usar documentos de requisitos	Un documento con funciones, reglas y criterios evita repetir lo mismo en cada mensaje.
Pedir una tarea concreta por etapa	“Implementá la etapa 2” en lugar de “hacé todo el sistema”.
No pedir una reconstrucción ante un error puntual	Corregí lo que falla y protegé lo que funciona (capítulo 9).
No pedir información innecesaria	Si no vas a leer una explicación extensa, pedí un resumen o solo la lista de cambios.
Mantener instrucciones de proyecto	Si el entorno lo permite, guardá las reglas permanentes en un archivo de instrucciones o en la configuración del proyecto.
Definir el formato de respuesta	“Respondé en una tabla con tres columnas” o “máximo diez puntos”.
Pedir aclaraciones cuando falten decisiones	Es más barato responder una pregunta que deshacer una suposición equivocada.

PROMPT LISTO PARA USAR · CERRAR UNA SESIÓN DE TRABAJO

Resumí esta sesión en un máximo de 15 puntos para retomarla en una conversación nueva: objetivo, decisiones tomadas, reglas acordadas, qué funciona, qué está pendiente y próximo paso. No incluyas explicaciones que ya no sean necesarias.

ATENCIÓN

Cada herramienta, modelo y plan maneja los tokens de forma distinta. Estas prácticas ayudan a trabajar con más orden, pero no garantizan un ahorro determinado ni resultados idénticos en todas las herramientas.

ANTES DE AVANZAR

- ☐ Tengo un documento de requisitos o una ficha de contexto lista para compartir.
- ☐ Elegí dos hábitos de la tabla para aplicar en la próxima sesión.

Pruebas, seguridad y publicación

QUÉ VAS A APRENDER

- Cómo comprobar que una herramienta funciona antes de usarla con información real.
- La diferencia entre una prueba propuesta, una ejecutada y un resultado verificado.
- Qué revisar antes de publicar.

PRINCIPIO BÁSICO

Que la IA genere código no garantiza que sea seguro, correcto ni apto para producción. Si la herramienta va a manejar datos personales, pagos o información sensible, pedí la revisión de un profesional antes de publicarla.

Propuesta, ejecución y verificación

Tipo	Qué significa	¿Alcanza para publicar?
Prueba propuesta	La IA describe qué habría que probar.	No. Es solo un plan.
Prueba ejecutada	La prueba se corrió y hay un resultado registrado.	Todavía no: falta comparar con lo esperado.
Resultado verificado	Vos (u otra persona) comprobaste que el resultado coincide con el criterio de aceptación.	Sí, para esa función.

Probá siempre tres tipos de casos: **normales** (el uso habitual), **errores** (datos incorrectos o faltantes) y **límites** (el último cupo, el último minuto del día, el valor máximo). Usá datos de prueba, nunca datos reales de clientes.

PROMPT LISTO PARA USAR · REVISIÓN PREVIA A PUBLICAR

Revisá el proyecto antes de publicarlo. Verificá: validaciones del lado del servidor, control de permisos, manejo de errores, credenciales expuestas en el código, datos sensibles en lugares públicos, respaldos y configuración de producción. Para cada punto indicá: estado (correcto / con problemas / no verificable), evidencia concreta y acción recomendada. Si no podés verificar algo en este entorno, decilo explícitamente.

Lista de control antes de publicar

Aspecto	Qué comprobar	Observaciones
☐ Datos y almacenamiento	Dónde se guardan, quién accede, si persisten al recargar.	
☐ Fórmulas y cálculos	Coinciden con ejemplos resueltos a mano.	
☐ Campos obligatorios	No se puede guardar sin ellos.	
☐ Errores y mensajes	Son claros y no muestran información técnica interna.	
☐ Permisos	Cada rol ve y hace solo lo que le corresponde.	
☐ Validaciones en el servidor	Las reglas se controlan también en el servidor, no solo en la pantalla.	
☐ Integraciones	Qué pasa si el servicio externo falla o tarda.	
☐ Copias de respaldo	Existen, son periódicas y probaste restaurar una.	
☐ Credenciales y variables de entorno	Las claves no están escritas en el código; se guardan como variables de entorno (valores de configuración fuera del código).	
☐ Costos de funcionamiento	Alojamiento, base de datos, dominio, envíos de mensajes, servicios externos.	
☐ Configuración de producción	Modo de depuración desactivado, conexión segura (HTTPS), datos de prueba eliminados.	

► VIDEO DEMOSTRATIVO 07

Cómo probar y corregir una función

Enlace	[Agregar enlace del video]
Qué vas a aprender	A pasar de una prueba propuesta a un resultado verificado.
Aplicá lo aprendido	Completá el checklist de pruebas y el registro de errores del capítulo 12.

Errores frecuentes

- Probar con datos reales de clientes.
- Pegar una clave de acceso en el chat para “que lo configure”.
- Publicar sin saber cuánto va a costar mantener la herramienta funcionando.

- Dar por verificada una prueba que solo fue propuesta.

ANTES DE AVANZAR

- ☐ Completé la lista de control con observaciones.
- ☐ Cada función crítica tiene un resultado verificado.

12

CAPÍTULO 12 · CUADERNO

Mi proyecto paso a paso

Estas fichas reúnen todo lo que trabajaste en la guía. Completalas en orden la primera vez y volvé a ellas cada vez que el proyecto cambie. Si trabajás en digital, podés pegar textos directamente en cada espacio o copiar la estructura a tu propio documento.

1. Ficha de idea
2. Ficha de contexto
3. Herramientas actuales
4. Archivos y referencias
5. Mapa del proceso actual
6. Lista de requisitos
7. Reglas de negocio
8. Decisiones pendientes
9. Plan de desarrollo
10. Registro de prompts
11. Registro de iteraciones
12. Checklist de pruebas
13. Registro de errores
14. Mejoras futuras
15. Evaluación final

1 · Ficha de idea

Nombre del proyecto	
Problema que resuelve	
Quién lo sufre hoy	
Resultado esperado	
Tipo (módulo / conexión / automatización / herramienta nueva)	
Cómo sabré que funcionó	

2 · Ficha de contexto

Usá la plantilla del capítulo 4. Registrá acá las versiones para saber cuál es la vigente.

Versión	Fecha	Qué cambió	Dónde está guardada

Versión	Fecha	Qué cambió	Dónde está guardada

3 • Inventario de herramientas actuales

Herramienta	Para qué la uso	Quién la usa	¿Se mantiene, se conecta o se reemplaza?

4 • Inventario de archivos y referencias

Archivo o referencia	Qué debe analizar la IA	¿Anonimizado?	¿Puedo compartirlo?

5 • Mapa del proceso actual

Describí cada paso como ocurre hoy. Marcá dónde se pierde tiempo o aparecen errores.

Paso	Qué pasa	Quién	Con qué	Problema detectado
1				
2				

Paso	Qué pasa	Quién	Con qué	Problema detectado
3				
4				
5				
6				

6 · Lista de requisitos

Requisito (la herramienta debe...)	Usuario	Prioridad (alta / media / baja)	Criterio de aceptación

7 · Reglas de negocio

Regla (siempre / nunca / si... entonces...)	Ejemplo con datos	Confirmada por

Regla (siempre / nunca / si... entonces...)	Ejemplo con datos	Confirmada por

8 · Decisiones pendientes

Decisión	Opciones	Afecta a	Decidido (fecha)

9 · Plan de desarrollo

Etapas	Qué se construye	Cómo se prueba	Estado
1			
2			
3			
4			
5			

10 · Registro de prompts

Índice rápido de tus prompts. El detalle de cada uno va en el Laboratorio personal de prompts.

Fecha	Nombre del prompt	Etapas	¿Validado?

11 · Registro de iteraciones

N.º	Diferencia detectada	Causa (contexto / regla / instrucción)	Cambio pedido y resultado
1			
2			
3			
4			

12 · Checklist de pruebas

Caso de prueba	Tipo (normal / error / límite)	Resultado esperado	Estado (propuesta / ejecutada / verificada)

Caso de prueba	Tipo (normal / error / límite)	Resultado esperado	Estado (propuesta / ejecutada / verificada)

13 · Registro de errores

Fecha	Qué pasó y cómo reproducirlo	Gravedad	Solución y verificación

14 · Lista de mejoras futuras

Mejora	Por qué sería útil	Prioridad

15 · Evaluación final del proyecto

¿La herramienta resuelve el problema inicial? ¿En qué medida?	
---	--

¿Se cumplieron los criterios de aceptación?	
Qué funcionó mejor del método	
Qué haría distinto en el próximo proyecto	
Quién mantiene la herramienta y cada cuánto se revisa	
Costos recurrentes identificados	
Próximo paso	

Checklist general del proyecto

Reutilizable para cada proyecto. Marcá cada ítem y registrá el estado de la etapa: **pendiente, en curso** o **completa**.

Etapa	Ítems	Estado	Observaciones
1. Definir	☐ Identifiqué el problema. ☐ Expliqué el proceso actual. ☐ Definí el resultado esperado.		
2. Contextualizar	☐ Documenté el negocio y sus reglas. ☐ Reuní las herramientas y referencias. ☐ Identifiqué restricciones y riesgos.		
3. Especificar	☐ Definí usuarios y permisos. ☐ Enumeré funciones. ☐ Identifiqué datos y validaciones. ☐ Resolví o documenté preguntas pendientes.		
4. Planificar	☐ Revisé la propuesta funcional. ☐ Evalué alternativas técnicas y costos. ☐ Aprobé el alcance inicial.		
5. Construir	☐ Implementé por etapas. ☐ Revisé los cambios. ☐ Conservé las funciones que ya trabajaban correctamente.		
6. Validar	☐ Probé los casos normales. ☐ Probé errores y situaciones límite. ☐ Verifiqué los resultados. ☐ Registré las correcciones pendientes.		
7. Publicar	☐ Revisé seguridad y configuración. ☐ Verifiqué almacenamiento y respaldos. ☐ Identifiqué costos recurrentes. ☐ Documenté el mantenimiento.		

Laboratorio personal de prompts

Una ficha por prompt importante. El objetivo no es acumular prompts, sino entender qué funcionó, por qué, y qué aprendiste para el próximo proyecto. Imprimí o duplicá esta ficha las veces que necesites.

FICHA N.º ____

Nombre del prompt	
Proyecto	
Etapas	
Objetivo	
Contexto utilizado	
Archivos o referencias	
Prompt inicial	
Respuesta obtenida (resumen)	
Qué funcionó	
Qué faltó o salió mal	

Qué voy a cambiar	
Nueva versión del prompt	
Resultado de la iteración	
¿Validé el resultado?	☐ Sí ☐ No ☐ Parcialmente
Prompt final reutilizable	
Aprendizaje para futuros proyectos	

Glosario básico

Prompt	Instrucción que le das a la IA.
Contexto	Información sobre tu negocio y tu proyecto que la IA necesita para responder bien.
Requisito	Algo que la herramienta debe hacer o cumplir.
Especificación funcional	Documento que describe qué hace la herramienta, para quién y con qué reglas, sin entrar en el código.
Regla de negocio	Condición propia de tu actividad que el sistema debe respetar. Ej.: “no se aceptan reservas con menos de 2 horas de anticipación”.
Validación	Comprobación automática de que un dato es correcto antes de guardarlo.
Servidor	Computadora remota donde funciona la herramienta y se guardan los datos. Las validaciones importantes deben hacerse ahí, no solo en la pantalla del usuario.
Base de datos	Lugar ordenado donde se guarda la información. Parecida a una planilla, pero preparada para muchos usuarios y reglas.
Integración	Conexión entre dos herramientas para que intercambien información.
Criterio de aceptación	Condición verificable que indica que una función está lista.
Token	Fragmento de texto que la IA procesa. Se relaciona con límites de uso o costos.
Variable de entorno	Valor de configuración (como una clave de acceso) que se guarda fuera del código para protegerlo.
Producción	La versión de la herramienta que usan personas reales con datos reales.
Copia de respaldo	Copia de los datos que permite recuperarlos si algo falla.

Pensar antes de pedir. Documentar antes de construir. Validar antes de publicar.



Relaciones Públicas