GUÍA · 4 CONCEPTOS + WORKSHOP + PLANTILLA

Crea *agentes* persistentes con Claude Managed Agents.

Entiende sus cuatro piezas, el workshop oficial y los controles que debes definir antes de llevar un agente a producción.

AUTOR

Jorge Riquelme

Fundador · Otherway Labs

CONTENIDO

4 conceptos + plantilla

Workshop · decisión · controles

LECTURA

8 minutos

Nivel base · Equipos y builders

— 01 · UBICACIÓN

Ubica Managed Agents en la capa *correcta*.

Claude Managed Agents es un servicio beta de Claude Platform. Está dirigido a quienes construyen agentes propios con API, CLI o SDK.

PRODUCTO	QUÉ RESUELVE	QUIÉN OPERA
Claude Chat y Cowork	Trabajo diario, archivos y aplicaciones dentro de Claude.	Producto listo para ocupar.
Claude Code	Agente de programación que opera en tu proyecto.	Tú inicias y supervisas.
Messages API	Acceso directo al modelo para una aplicación propia.	Tu equipo opera el loop.
Claude Agent SDK	Harness configurable dentro de un proceso propio.	Tu infraestructura lo ejecuta.
Managed Agents	Harness alojado, sesiones con estado y sandbox por sesión.	Anthropic opera el runtime.

ACLARACIÓN

Managed Agents pertenece a Anthropic. Requiere una API key y trabajo técnico. La empresa recibe infraestructura administrada, no un constructor no-code dentro de Cowork.

Reparte las *responsabilidades* antes de construir.

El ahorro viene de sacar infraestructura del código de tu aplicación. El criterio del agente y el riesgo operativo siguen en tu lado.

ANTHROPIC ADMINISTRA

01	Agent loop y harness alojado.
02	Historial de la sesión y eventos.
03	Sandbox cloud y tools integradas.
04	Prompt caching, compaction y reintentos transitorios.
05	Streaming, estados y trazabilidad.

TU EQUIPO DEFINE Y CONTROLA

01	Resultado esperado y condición de término.
02	Contexto, fuentes y calidad de datos.
03	Custom tools, triggers e integraciones.
04	Permisos, allowlists y aprobaciones.
05	Pruebas, costo, retención y gobierno.

CRITERIO

La plataforma entrega infraestructura y controles para producción. Tu agente queda apto para producción después de configurar permisos, evaluaciones, presupuesto, monitoreo y manejo de errores para su caso real.

— 03 · ARQUITECTURA

Cada ejecución ocupa *cuatro* recursos.

El workshop destaca Agent, Environment y Session. La documentación actual suma Events como el cuarto concepto oficial.

01 · AGENT

Configuración reutilizable

Modelo, system prompt, tools, MCP servers y skills. Se crea una vez, se versiona y se referencia por ID.

02 · ENVIRONMENT

Lugar de ejecución

Sandbox cloud o self-hosted. Define red, runtime, archivos, repositorios y dónde corren las tools.

03 · SESSION

Un trabajo concreto

Une Agent y Environment. Conserva historial, estado, archivos y resultados de esa ejecución.

04 · EVENTS

Entradas y trazabilidad

Mensajes, tool calls, resultados y cambios de estado. Se transmiten por SSE y quedan registrados.

CICLO DE UNA SESIÓN

Idle → **Evento o cron** → **Running + tools** → **Idle**

Un evento externo, como un lead nuevo, requiere que tu integración envíe el evento a la sesión. Los scheduled deployments crean sesiones nuevas en un horario definido.

PERSISTENCIA — El historial dura hasta que eliminas la sesión. El estado del sandbox se conserva 30 días desde su creación. La memoria entre sesiones requiere un Memory Store.

El workshop conecta *siete* piezas.

Anthropic prepara una aplicación Streamlit de respuesta a incidentes. El ejercicio completa siete funciones, cada una asociada a una operación de Managed Agents.

01 **setup_agent()**
Crea el agente SRE con instrucciones y tools declaradas.

02 **setup_environment()**
Crea el entorno cloud donde el sandbox ejecutará acciones.

03 **upload_log()**
Sube el archivo por Files API para montarlo en la sesión.

04 **start_session()**
Une Agent, Environment y el log en una sesión.

05 **stream_reply()**
Abre SSE, envía la tarea y muestra cada evento.

06 **handle_tool()**
Ejecuta métricas, deployments y diffs en la aplicación local, y devuelve cada resultado.

07 **delete_session()**
Elimina la sesión y sus datos cuando termina la demo.

LO QUE OCURRE AL EJECUTAR

70.000 líneas
El log se procesa dentro del sandbox.

Logs → métricas → diff
Claude elige el orden según la evidencia disponible.

Causa raíz
Identifica el cambio que agotó el pool de base de datos.

LA DIFERENCIA

El código define capacidades, datos y límites. Claude decide qué herramienta consultar y qué hacer después de observar cada resultado. Ahí aparece la autonomía del agente.

Define tu primer agente en *siete* campos.

Completa esta ficha antes de abrir el editor. Si un campo queda ambiguo, el agente todavía no está listo para una prueba real.

01	**Resultado**	Entregable y condición de éxito.
02	**Inicio**	Mensaje, evento o horario.
03	**Contexto**	Datos, políticas y fuentes válidas.
04	**Tools de lectura**	Qué puede consultar sin cambiar datos.
05	**Tools de escritura**	Qué sistemas puede modificar.
06	**Límites**	Aprobaciones, red, datos y acciones prohibidas.
07	**Evidencia**	Prueba, métrica, presupuesto y cierre.

EJEMPLO · AGENTE COMERCIAL

Inicio y resultado

Nuevo lead. Ficha CRM completa, score y correo en borrador.

Contexto y tools

ICP, historial, web, CRM y email en modo borrador.

Límite y evidencia

Requiere aprobación para enviar. Fuentes visibles y campos obligatorios completos.

Conciliación de facturas

Compara factura, orden de compra y pago. Deja excepciones para revisión humana.

Respuesta a incidentes

Consulta logs, métricas y código. Propone una causa y adjunta la evidencia.

PRIMERA PRUEBA

Ocupa datos de prueba, tools de solo lectura y una sola decisión abierta. Aumenta permisos cuando la evidencia muestre resultados consistentes.

Elige la opción que tu proceso *necesita*.

Un agente administrado tiene sentido cuando el trabajo cambia según lo que encuentra y requiere varias acciones durante minutos u horas.

Trabajo dentro de Claude	**Claude Code o Cowork** Tú o tu equipo ejecutan la tarea en un producto ya construido.
Secuencia fija y control total	**Messages API o Agent SDK** Tu aplicación define y opera el loop, la ejecución y los hooks.
Trabajo largo, stateful y variable	**Claude Managed Agents** Anthropic opera el harness, la sesión y el sandbox.

CHECKLIST ANTES DE PRODUCCIÓN

- ☐ Red limited con allowed hosts explícitos.
- ☐ Aprobación para tools que modifican datos.
- ☐ Presupuesto y condición de término.
- ☐ Eventos, outputs y errores registrados.
- ☐ Evaluación con casos normales y fallos.
- ☐ Retención, borrado y escalamiento humano.

AGOSTO 2026 Managed Agents sigue en beta. Actualmente no es elegible para Zero Data Retention ni cobertura HIPAA BAA. Dreaming y MCP tunnels requieren acceso de research preview.

FUENTES Overview · Migration · Events · Permissions · Workshop oficial

MÁS GUÍAS otherwaylabs.cl/recursos



SOBRE EL AUTOR

Jorge Riquelme

FUNDADOR · OTHERWAY LABS

Ayudo a pymes y profesionales a recuperar tiempo con automatizaciones de IA, con ROI claro y garantizado.

otherwaylabs.cl · LinkedIn: Jorge Riquelme

Tres formas de seguir *avanzando*.

PARA EMPRESAS
GRATIS

Sesión gratuita

¿Hay procesos en tu empresa que *deberían correr solos*? Revisamos si Managed Agents encaja y qué sistema conviene construir primero.

→ otherwaylabs.cl

PARA EQUIPOS

Capacitación

Forma a tu equipo para que use IA *todos los días*. Aprende a definir objetivo, contexto, tools, permisos y pruebas.

→ otherwaylabs.cl/capacitaciones

ACCESO
ANTICIPADO
GRATIS

Grupo WhatsApp

Recibe estos recursos *antes que nadie*. Grupo exclusivo donde publico cada guía, MCP y skill antes de subirla a YouTube o Instagram.

→ chat.whatsapp.com