AUTOSETTER™

CEREBRO DE LLAMADAS™

El sistema que convierte cada llamada de Zoom o Meet en inteligencia comercial accionable, sin tocar nada.

MANUAL v1.0

EDICIÓN COMUNIDAD

Índice

1. Qué es el Cerebro de Llamadas
2. El truco que multiplica este manual (Claude Code como tutor)
3. El flujo del sistema en 6 pasos
4. Lo que necesitas antes de empezar
5. Crear el bot de Telegram
6. La API de Claude (el cerebro real)
7. Google Cloud y permisos OAuth
8. Las 4 carpetas de Google Drive
9. Conseguir el refresh token
10. Deploy en Railway
11. Conectar Fathom al sistema
12. La primera llamada de prueba
13. El Maestro Mensual de 30 días
14. Errores comunes y cómo arreglarlos
15. Prompts copy-paste para Claude Code
16. FAQ — preguntas reales
17. Cierre

1. Qué es el Cerebro de Llamadas

Una llamada de venta de 45 minutos contiene oro: objeciones reales, frases textuales del cliente, momentos donde el closer se equivocó, momentos donde acertó. Todo eso normalmente muere ahí. Nadie lo escucha de nuevo. Nadie aprende.

El Cerebro de Llamadas hace que ese oro no se pierda. Cada vez que cuelgas una llamada en Zoom, Meet o Teams, el sistema la coge, la analiza con Claude como si fuera un coach de ventas exigente, y te entrega un Google Doc completo en menos de 60 segundos. Más un aviso a Telegram con el resumen.

Y lo mejor: tú no haces nada. Cuelgas y te vas a otra cosa.

Lo que el sistema hace por ti

- Detecta solo si la llamada es de venta, de cliente o de equipo.
- Analiza objeciones reales, frases clave, errores concretos del closer.
- Te da scripts exactos para mejorar la próxima vez.
- Lo guarda todo en su carpeta de Drive correspondiente.
- Te avisa por Telegram con el link al análisis.
- Cada 30 días genera un informe maestro con patrones del mes.

EL CASO REAL

Llamada de prueba real: closer suelta el precio de 200€/mes sin descubrimiento. Cliente dice 'me parece caro, lo pienso'. Closer acepta sin rebatir. El Cerebro detectó el error, dio el script exacto que debía haber usado, y propuso una fecha concreta de cierre para la próxima llamada. Eso lo hizo Claude en 8 segundos.

2. El truco que multiplica este manual

Antes de meterte con todo el setup, lee esto. Te va a ahorrar HORAS si lo aplicas.

Este manual no es un PDF para leer en sofá. Está pensado para algo más poderoso: subirlo a Claude Code y convertirlo en TU TUTOR PERSONAL durante toda la instalación.

Qué es Claude Code

Claude Code es una app de Anthropic que tienes en claude.com/code. La descargas, la abres, y te encuentras un chat donde puedes hablarle en español como a un amigo programador.

Pero lo brutal viene cuando le das contexto. Si le subes este manual y el código del Cerebro, Claude Code se transforma en un tutor que conoce el sistema entero, sabe en qué paso estás y te ayuda en vivo.

Cómo convertirlo en tu tutor

1. Descarga Claude Code desde claude.com/code (gratis con cuenta Pro, $20/mes).
2. Ábrelo y abre una carpeta nueva en tu Escritorio llamada cerebro-llamadas.
3. Sube este manual al chat (arrastra el .docx al chat).
4. Mándale este prompt inicial:

PROMPT INICIAL · CÓPIALO TAL CUAL

Voy a montar el sistema descrito en este manual. Quiero que me guíes como un tutor: sigue el manual paso por paso, espera a que termine cada sección, y solo entonces pasa a la siguiente. Si me atasco con algo, ayúdame a debuggear leyendo el manual. Si me equivoco en algo, dímelo antes de seguir. Empecemos por la sección 3.

Qué cambia con Claude Code como tutor

- No tienes que descifrar nada técnico solo: Claude te lo traduce en cristiano.
- Si un paso falla, Claude lee los logs contigo y te dice qué cambiar.
- Edita archivos por ti si se lo pides. No copias ni pegas código.
- Recuerda en qué paso estás. No te pierdes.
- Mantiene contexto de TODO. Sabe lo que has hecho 30 minutos antes.

PROMPTS QUE VAS A USAR MUCHO

Al final del manual (sección 15) tienes una lista de prompts copy-paste para situaciones concretas: 'qué hago si Railway crashea', 'cómo cambio el formato del Doc', 'mi webhook no llega'. Vete a esa sección cuando te atasques.

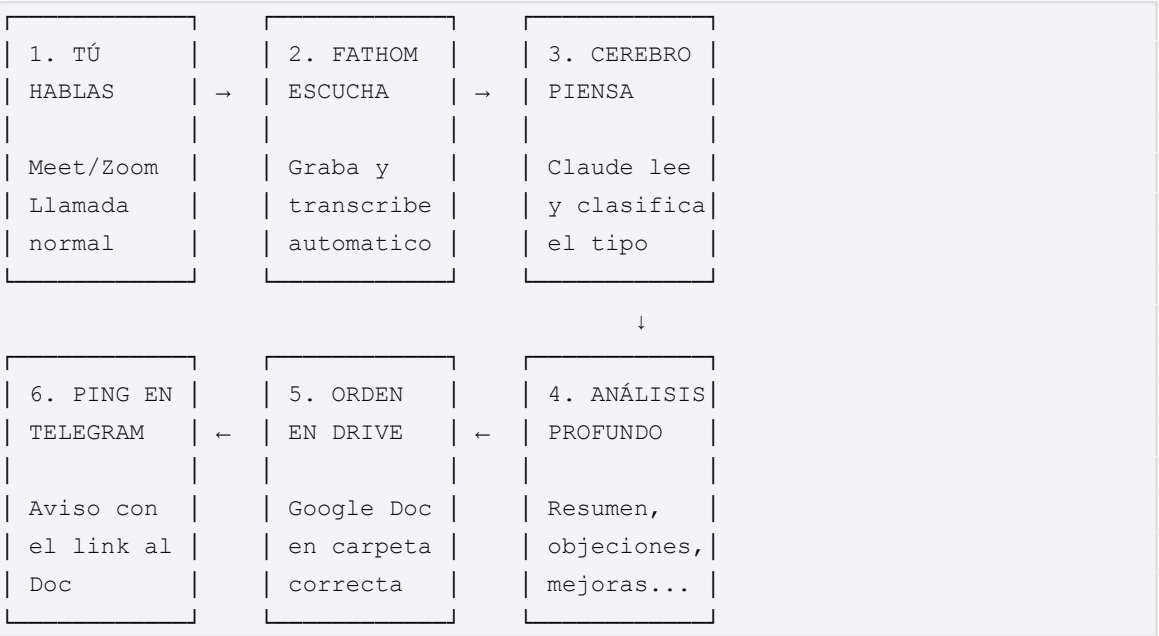
SIN CLAUDE CODE TAMBIÉN FUNCIONA

Si no quieres pagar Pro, puedes usar Cursor (gratis) o VSCode (gratis). Funcionan pero requieren saber un mínimo de inglés técnico. Mi recomendación fuerte: Claude Code. Te ahorra el 80% del setup y entiende español perfectamente.

3. El flujo del sistema en 6 pasos

Esto es lo que pasa entre que cuelgas la llamada y recibes el ping en Telegram. Todo automático, sin intervención humana.

Diagrama del flujo



De colgar la llamada al Doc en tu mano: ~60 segundos.

Cero clicks. Cero copy-paste. Cero olvidos.

Qué pasa en cada paso

Paso 1 · Tú hablas

Llamada normal en Google Meet, Zoom o Teams. Activas Fathom como siempre.

Paso 2 · Fathom escucha

Fathom graba, transcribe y al colgar manda un webhook (un aviso automático) al servidor que vamos a montar.

Paso 3 · El Cerebro piensa

Claude Haiku (rápido y barato) lee los primeros segundos para detectar si es VENTA, CLIENTE o EQUIPO.

Paso 4 · Análisis profundo

Claude Sonnet (más potente) analiza la transcripción entera y devuelve resumen, objeciones, errores, mejoras con script exacto, frases clave y próximos pasos.

Paso 5 · Todo ordenado en Drive

El análisis se convierte en un Google Doc profesional y se guarda en la carpeta del tipo correspondiente.

Paso 6 · Ping en Telegram

Te llega un mensaje al móvil con resumen de 2 líneas + link directo al Doc completo.

POR QUÉ ES BRUTAL

El sistema no solo resume. Te dice exactamente qué deberías haber dicho cuando el cliente puso una objeción. Eso es entrenamiento real para tu equipo de closers, generado solo, en cada llamada.

4. Lo que necesitas antes de empezar

Antes de tocar nada, ten esto preparado. Si te falta alguna cuenta, créala. Te lleva 10 minutos en total.

Cuentas necesarias

- Google — para Drive, Cloud y Gmail. Gratis.
- Anthropic (Claude) — el cerebro analítico. Mete $5 mínimo.
- Fathom — la herramienta que graba. Free vale al inicio.
- Railway — donde vive el servidor. Free trial $5 + recomendado meter $5 más.
- GitHub — para tener el código. Gratis.
- Telegram — para recibir avisos. Gratis.
- Claude Code (recomendado) — para que Claude edite el código por ti.

SALDO MÍNIMO QUE NECESITAS CARGAR

Anthropic: $5 mínimo (sin saldo, las llamadas a Claude fallan). Ve a console.anthropic.com > Plans & Billing > Add credits.

RAILWAY TAMBIÉN

Railway tiene free trial de $5 que dura ~1 mes. Te recomiendo cargar $5 extra para que no se apague el servidor en mitad del primer mes.

Coste real al mes

- Anthropic: ~$10 con 100 llamadas/mes.
- Railway: ~$5 con uso bajo.
- Google: $0.
- Fathom: $0 (free) o $19 si quieres ilimitado.
- Telegram: $0.

TOTAL: ~$15/mes con 100 llamadas. ~$34/mes si quieres Fathom Pro.

5. Crear el bot de Telegram

El bot es quien te va a mandar los avisos al móvil. Necesitas dos cosas: el TOKEN del bot y tu CHAT_ID (a quién avisa).

Paso 1 · Crear el bot con BotFather

5. Abre Telegram en el móvil.
6. Busca @BotFather (es el bot oficial de Telegram).
7. Dale a Iniciar / Start.
8. Escribe /newbot y dale a enviar.
9. BotFather te pide un nombre. Pon: Cerebro de Llamadas Bot.
10. Te pide un username que acabe en 'bot'. Ejemplo: cerebrollamadas_tunombre_bot.
11. BotFather te devuelve un TOKEN, algo así: 1234567890:AAAAxxxxxxxxxx.
12. Copia ese token entero. Lo necesitas en Railway luego.

EL TOKEN ES SECRETO

Este token es como la llave de tu bot. No lo compartas con nadie y no lo subas a GitHub. Si se filtra, vuelve a BotFather y revócalo con /revoke.

Paso 2 · Conseguir tu Chat ID

El TOKEN dice 'qué bot escribe'. El CHAT_ID dice 'a quién escribe'. Tu CHAT_ID es un número personal que identifica tu chat de Telegram.

13. En Telegram, busca @userinfobot (es un bot público).
14. Dale a Start.
15. Te responde con tu User ID al instante, algo así:

```
Id: 123456789
First name: Nico
```

16. Ese número es tu TELEGRAM_CHAT_ID. Cópialo.

PRUEBA QUE TODO FUNCIONA

Antes de seguir, mándale '/start' o 'hola' a TU bot (el que creaste con BotFather). Si no le has escrito antes, no podrá mandarte avisos. Esto es una norma anti-spam de Telegram.

6. La API de Claude (el cerebro real)

Claude es el modelo que va a analizar cada llamada. Para que el servidor pueda hablar con Claude, necesitas una API Key (una llave).

Paso 1 · Cargar saldo (CRÍTICO)

17. Entra en console.anthropic.com y haz login.
18. Ve a Settings > Plans & Billing > Add credits.
19. Mete $5 mínimo. Con $10 te duran 100+ llamadas tranquilamente.

SIN SALDO NO FUNCIONA

Si tu cuenta tiene $0, todas las llamadas a Claude van a fallar. Antes de generar la API Key, mete saldo primero. Si ya tienes saldo de antes, sáltate este paso.

Paso 2 · Crear la API Key

20. En console.anthropic.com, ve a Settings > API Keys.
21. Click en Create Key.
22. Nombre: cerebro-llamadas. Click Create.
23. Te muestra la key. Empieza por sk-ant-api03-...
24. Cópiala YA. Solo se ve una vez. Si cierras sin copiar, créala de nuevo.
25. Pégala en tu nota de claves. La necesitas en Railway luego.

Cuánto vas a gastar de verdad

- Detectar tipo de llamada (Claude Haiku): ~$0.0002 por llamada. Céntimos.
- Análisis profundo (Claude Sonnet): ~$0.10 por llamada.
- 100 llamadas/mes ≈ $10. 50 llamadas/mes ≈ $5.

TIP DE AHORRO

El sistema usa dos modelos a posta. Haiku para detectar el tipo (rapido y baratísimo) y Sonnet solo para el análisis profundo. Si usaras Sonnet para todo, gastarías el doble por nada.

7. Google Cloud y permisos OAuth

Este es el paso más largo del manual. 20 minutos. Pero solo se hace una vez, así que respira y hazlo siguiendo cada paso.

Lo que hacemos aquí: dar permiso a tu sistema para crear Google Docs en TU Drive sin que tengas que iniciar sesión cada vez.

Paso 1 · Crear el proyecto en Google Cloud

26. Entra en console.cloud.google.com.
27. Arriba a la izquierda hay un selector de proyecto. Click > New Project.
28. Nombre: cerebro-llamadas. Click Create.
29. Espera 10 segundos a que se cree.
30. Selecciona el proyecto recién creado en la barra superior. Asegúrate de que pone 'cerebro-llamadas' arriba.

TIENES QUE ESTAR DENTRO DEL PROYECTO

Todo lo que hagas a partir de aquí tiene que ser DENTRO de este proyecto. Si la barra de arriba no pone 'cerebro-llamadas', selecciónalo. Te puedes ahorrar 30 minutos de bucle por equivocarte aquí.

Paso 2 · Activar las APIs (Docs + Drive)

Tu sistema va a usar dos APIs de Google: la de Docs (para crear documentos) y la de Drive (para guardarlos en carpetas).

31. Abre console.cloud.google.com/apis/library/docs.googleapis.com y dale al botón azul ENABLE.
32. Abre console.cloud.google.com/apis/library/drive.googleapis.com y dale al botón azul ENABLE.
33. Verifica que en cada API aparece un check verde de 'API habilitada'.

Paso 3 · OAuth Consent Screen

Esto le dice a Google: 'voy a hacer una app que pedirá acceso a Drive en mi cuenta'.

34. Entra en console.cloud.google.com/apis/credentials/consent.
35. User Type: selecciona External. Dale a Create.
36. App name: cerebro-llamadas.

37. User support email: tu email (sale en el desplegable).
38. Developer contact: tu email.
39. App logo, App domain: déjalos en blanco. Save and continue.
40. Scopes: NO añadas nada. Save and continue.
41. Test users: añade tu email principal de Google (donde están las carpetas).
42. Save and continue. Y luego Back to Dashboard.

AÑADIR TUS TEST USERS ES CRÍTICO

Si no añades tu email aquí, cuando vayas a autorizar la app en el paso 8 Google te bloqueará con 'Access blocked'. Si vas a compartir el sistema con tu equipo, añade aquí también sus emails. Hasta 100 test users permitidos.

Paso 4 · OAuth Client ID

Esto es como la 'tarjeta de identidad' que tu app le enseña a Google.

43. Entra en console.cloud.google.com/apis/credentials.
44. Arriba: + CREATE CREDENTIALS > OAuth client ID.
45. Application type: Desktop app.
46. Name: cerebro-llamadas-cli. Click CREATE.
47. Te aparece un popup con Client ID y Client Secret.
48. Copia los dos a tu nota de claves. Los necesitas en Railway luego.

8. Las 4 carpetas de Google Drive

El sistema necesita 4 carpetas en tu Drive: una para cada tipo de llamada y una para los informes mensuales.

Crear la estructura

49. Abre drive.google.com con tu cuenta de Google.
50. Crea una carpeta principal llamada AUTOSETTER (o el nombre que quieras).
51. Dentro, crea estas 4 subcarpetas:

- VENTAS — aquí van las llamadas de venta con prospectos.
- CLIENTES — aquí van las llamadas con clientes ya activos.
- EQUIPO — aquí van las reuniones internas.
- MAESTROS — aquí van los informes maestros mensuales.

Conseguir el ID de cada carpeta

Cada carpeta de Drive tiene un ID único. El sistema necesita esos IDs para saber dónde guardar cada cosa.

52. Abre cada carpeta haciendo doble click.
53. Mira la URL del navegador. Verás algo así:

```
https://drive.google.com/drive/folders/1ABC...XYZ
```

54. La parte después de /folders/ es el ID. Cópialo.
55. Repite con las 4 carpetas. Anota los 4 IDs.

NO HACE FALTA COMPARTIR LAS CARPETAS

Como vamos a usar OAuth con tu cuenta personal, el sistema actuará como tú. Las carpetas siguen siendo solo tuyas. No las compartas con nadie ni cambies permisos.

9. Conseguir el refresh token

Este token es lo que permite a tu servidor crear Docs en tu Drive sin que tú tengas que iniciar sesión cada vez. Se hace UNA vez y dura para siempre.

Con Claude Code o terminal

56. Abre Claude Code (o terminal) en la carpeta del proyecto.
57. Comprueba que tienes Python 3 instalado. En terminal:

```
python3 --version
```

58. Si pone 'Python 3.x.x', perfecto. Si no, instálalo desde python.org/downloads.
59. Instala la librería necesaria:

```
pip3 install google-auth-oauthlib
```

60. Abre el archivo scripts/get_refresh_token.py del proyecto.
61. En las primeras líneas hay placeholders PEGA_TU_CLIENT_ID_AQUI y PEGA_TU_CLIENT_SECRET_AQUI.
62. Reemplázalos con los valores que copiaste en el paso 7.4.
63. Guarda y ejecuta en terminal:

```
python3 scripts/get_refresh_token.py
```

64. Se abre tu navegador. Selecciona la cuenta de Google donde están tus carpetas.
65. Te dirá 'Google no ha verificado esta app'. Click en 'Configuración avanzada' > 'Ir a cerebro-llamadas (no seguro)'.
66. Acepta los permisos (Drive + Docs). Es seguro: es TU propia app autorizándose contra TU cuenta.
67. La terminal te imprime el REFRESH TOKEN. Cópialo.

SI TE SALE 'ACCESS BLOCKED'

Significa que no añadiste tu email como Test User en el paso 7.3. Vuelve, añádete como Test User, y repite este paso desde el principio.

10. Deploy en Railway

Railway es donde va a vivir tu servidor. Coge tu código de GitHub y lo monta en una máquina que escucha 24/7. Cada vez que cambies algo en el código y hagas push a GitHub, Railway redespliega solo.

Paso 1 · Login y conectar GitHub

68. Entra en railway.app y haz Login with GitHub.
69. Si es la primera vez, autoriza a Railway a leer tus repos.

Paso 2 · Crear el proyecto desde el repo

70. Botón + New Project > Deploy from GitHub repo.
71. Selecciona tu repo cerebro-llamadas (el que subiste de la skill anterior).
72. Si no aparece, pulsa 'Configure GitHub App' y dale acceso al repo.
73. Railway empieza a desplegar. VA A FALLAR la primera vez. Es normal: aún no le pasamos las claves.
74. Espera a que termine con 'Failed' o 'Crashed' en rojo. No pasa nada.

Paso 3 · Meter todas las variables

Aquí es donde le pasas las claves a Railway para que el servidor pueda hablar con todo.

75. Click en el servicio (la card del proyecto).
76. Arriba pestaña Variables.
77. Raw Editor (o Bulk import).
78. Pega TODO esto, sustituyendo cada VALOR_AQUI por la clave correspondiente:

```
ANTHROPIC_API_KEY=sk-ant-api03-...
TELEGRAM_TOKEN=1234567890:AAAAxxxxxx
TELEGRAM_CHAT_ID=123456789
GOOGLE_OAUTH_CLIENT_ID=12345-xxx.apps.googleusercontent.com
GOOGLE_OAUTH_CLIENT_SECRET=GOCSPX-xxxxx
GOOGLE_OAUTH_REFRESH_TOKEN=1//03xxxxx
FOLDER_VENTAS=1ABC...XYZ
FOLDER_CLIENTES=1DEF...UVW
FOLDER_EQUIPO=1GHI...RST
FOLDER_MAESTROS=1JKL...OPQ
CRON_TOKEN=cadena_aleatoria_que_generes
```

79. Para CRON_TOKEN, genera uno aleatorio en terminal:

```
python3 -c "import secrets; print(secrets.token_urlsafe(24))"
```

80. Save. Railway redespliega automáticamente.

Paso 4 · Cargar saldo en Railway

81. Click en tu avatar (arriba derecha) > Account Settings > Billing.
82. Add Payment Method > tarjeta.
83. Add Credits > mete $5 extra.

POR QUÉ ES CRÍTICO

Railway tiene free trial de $5 que dura ~1 mes con uso bajo. Si se acaba sin saldo, tu servidor se apaga. Con $5 extra tienes 2-3 meses tranquilos.

Paso 5 · Generar el dominio público

84. En tu servicio, pestaña Settings > Networking.
85. Click en Generate Domain.
86. Te da una URL tipo tu-app-production-xxxx.up.railway.app.
87. Apuntala. La necesitas en el siguiente paso para Fathom.

VERIFICAR QUE ESTÁ VIVO

Abre tu URL en el navegador. Deberías ver: '🧠 Cerebro de Llamadas v4 activo - by AUTOSETTER™'. Si lo ves, el servidor está respirando. Si ves un error, ve a Logs y mira qué variable falta.

11. Conectar Fathom al sistema

Fathom necesita saber dónde mandarte cada llamada cuando termine. Le damos tu URL de Railway con el path /webhook/fathom al final.

Configurar el webhook

88. Entra en Fathom (fathom.video) y haz login.
89. Click en tu avatar > Settings.
90. Busca la sección Integrations o Webhooks.
91. Click en Add Webhook.
92. URL: pega tu URL de Railway + /webhook/fathom al final. Ejemplo:

```
https://tu-app-production-xxxx.up.railway.app/webhook/fathom
```

93. Trigger / Event: selecciona 'Meeting recorded' o 'Transcript ready'.
94. Save.

EL /webhook/fathom NO SE OLVIDA

Si pones solo la URL de Railway sin /webhook/fathom al final, Fathom mandará el aviso a un sitio donde no hay nadie escuchando. No funciona nada. Asegúrate de pegar la URL completa con el path.

12. La primera llamada de prueba

El momento de la verdad. Vamos a hacer una llamada corta para verificar que todo el flujo funciona de principio a fin.

Hacer la llamada

95. Abre Google Meet contigo mismo (meet.google.com > Nueva reunión).
96. Conecta Fathom (debería conectar solo, si no, dale a 'Join meeting' en la app de Fathom).
97. Empieza a grabar (botón rojo en Fathom).
98. Habla 1-2 minutos. Para que Claude detecte bien el tipo, di frases claras:
- VENTA: 'Te ofrezco este servicio por X euros al mes...'
- CLIENTE: 'Tenemos un problema con la herramienta que has contratado...'
- EQUIPO: 'Vamos a decidir las prioridades de la próxima semana...'
99. Cuelga la llamada.

Esperar el ping (1-5 minutos)

Fathom tarda entre 1 y 5 minutos en procesar la grabación y mandar el webhook a tu servidor. Cuando lo manda, el sistema:

100. Recibe el aviso de Fathom.
101. Claude Haiku detecta el tipo de llamada.
102. Claude Sonnet hace el análisis profundo.
103. El sistema crea un Google Doc en la carpeta correspondiente.
104. Te llega un mensaje a Telegram con el resumen y el link al Doc.

TE LLEGÓ EL PING

Si te llega el mensaje de Telegram con un link funcional al Doc, EL SISTEMA ESTÁ VIVO. Felicidades, acabas de montar lo que hace 2 años costaba 5.000 € de desarrollo.

Si no llega en 10 minutos

Algo falló en algún punto. Vamos a debuggear:

105. Abre Railway > tu servicio > pestaña Logs.
106. Busca un mensaje que empiece por '📨 Webhook recibido'.

107. Si NO aparece: Fathom no está mandando el webhook. Revisa la URL del paso 10.

108. Si SÍ aparece pero seguido de '❌ Error': mira el error específico. Los más comunes:

- 'permission denied / 403' → faltó habilitar las APIs en el paso 7.2.
- 'Claude Error' → falta saldo en Anthropic (paso 6.1).
- 'invalid_grant' → mal el refresh token, repite paso 8.

13. El Maestro Mensual de 30 días

Esta es la joya oculta del sistema. Cada 30 días, sin que tú pidas nada, el Cerebro genera un Doc Maestro que agrega todas las llamadas del mes y te entrega patrones, métricas y wins.

Qué contiene el Maestro

- Total de llamadas procesadas (separadas por tipo).
- Tasa de cierre de ventas y motivos de no-cierre más frecuentes.
- Top 5 objeciones más repetidas en el mes.
- Patrones de los closers que más cierran y los que menos.
- Lecciones de gestión de clientes (problemas recurrentes, satisfacción).
- Productividad de las reuniones de equipo (tareas creadas, decisiones tomadas).
- Alertas: clientes en riesgo, ventas que se enquistan, temas que se repiten en reuniones sin resolver.

Cómo se activa solo

Cada día a las 9:00 (hora de Madrid), el servidor chequea si han pasado 30 días desde el último Maestro. Si sí, lo genera automáticamente y te avisa por Telegram con el link.

Tú no haces nada. Llega cuando toca.

Si quieres forzar uno ahora (para probar)

En tu terminal, sustituye TU_DOMINIO y TU_CRON_TOKEN por los valores que tengas en Railway:

```
curl -X POST "https://TU_DOMINIO/cron/maestro?token=TU_CRON_TOKEN&force=1"
```

En unos segundos te llega un Telegram con el link al Doc Maestro.

EL PRIMER MAESTRO PUEDE SALIR VACÍO

Si aún no has procesado llamadas, el primer Maestro sale en blanco. Es normal. A partir de la próxima llamada real, se va guardando un JSON oculto en cada carpeta que el siguiente Maestro agregará todo.

14. Errores comunes y cómo arreglarlos

Errores que se viven cuando montas este sistema por primera vez. Te los documento todos para que no pierdas 2 horas en lo mismo que perdí yo.

Error 1 · 'My servidor responde pero no llega el Doc a Drive'

Comprueba en este orden:

- Que TÚ estás como Test User en OAuth consent (paso 7.3).
- Que las 2 APIs (Docs + Drive) están habilitadas (paso 7.2).
- Que el GOOGLE_OAUTH_REFRESH_TOKEN está bien copiado en Railway.

Error 2 · 'Telegram me llega pero sin link al Doc'

Probablemente el Doc falló al crearse y el sistema usa el link de Fathom como fallback.

- Ve a Railway > Logs y busca '❌ Error Drive'.
- El mensaje exacto te dice qué falla (usualmente OAuth).

Error 3 · 'Cada llamada se procesa 3 veces'

Fathom reintenta el webhook si tarda en responder.

- Verifica que /webhook/fathom devuelve HTTP 200 rápido en los logs.
- Si pasa, revisa que no tocaste el Procfile.

Error 4 · 'Mi worker se muere por timeout'

El procesamiento dura más de lo que Fathom espera.

- El sistema ya procesa en background para evitar esto.
- Si pasa, revisa que el Procfile no se ha modificado.

Error 5 · 'Necesito rotar las claves'

- Anthropic: console.anthropic.com > API Keys > Revoke > Create new.
- Telegram: @BotFather > /token > seleccionar bot > Revoke current token.
- Google OAuth: borra el OAuth Client y crea uno nuevo. Repite el paso 8.
- Sube las nuevas claves a Railway > Variables.

Error 6 · 'El JSON de Google se lee como vacío'

Este es típico del primer deploy.

- Comprueba que la variable de entorno se llama exactamente GOOGLE_CREDENTIALS_JSON.
- Pega el JSON completo, no solo una línea.
- Quita espacios al principio o final del valor pegado.

15. Prompts copy-paste para Claude Code

Esta es tu sección de rescate. Cuando algo no funcione o quieras modificar el sistema, vete aquí, copia el prompt que toque, y pégaselo a Claude Code. Él hace el resto.

Prompts para debuggear errores

Si Railway crashea al desplegar

PROMPT · RAILWAY CRASHED

Mi deploy en Railway está en CRASHED. Aquí están los últimos 30 líneas de los Deploy Logs: [PEGA LOS LOGS]. Lee el error, dime qué falla y arréglalo en el código. Cuando termines, haz commit y push.

Si el webhook de Fathom no llega

PROMPT · WEBHOOK NO LLEGA

He hecho una llamada con Fathom y han pasado 10 minutos sin recibir nada en Telegram. Mira los logs de Railway (te los paso) y dime si el webhook está llegando o no. Si no llega, posibles causas: URL mal en Fathom, webhook desactivado o servidor caído. Logs: [PEGA LOGS].

Si el Google Doc no se crea

PROMPT · NO SE CREA EL DOC

Me llega el ping de Telegram pero sin link al Google Doc, o el link es el de Fathom. Eso significa que la creación en Drive falla. Revisa main.py, mira los logs (te los paso) y dime qué error de Google está saliendo. Logs: [PEGA LOGS].

Si Claude no analiza bien la llamada

PROMPT · ANÁLISIS POBRE

El análisis que está dando Claude de mis llamadas es genérico o le falta profundidad. Quiero que ajustes el prompt PROMPT_VENTA (o PROMPT_CLIENTE, PROMPT_EQUIPO según corresponda) en main.py para que sea más exigente. Lo que me falta concretamente: [DESCRIBE LO QUE FALTA].

Prompts para personalizar

Cambiar el formato del Google Doc

PROMPT · FORMATO DEL DOC

Quiero cambiar el formato de los Google Docs que genera el sistema. Específicamente: [DESCRIBE LO QUE QUIERES]. Ejemplo: 'quiero una tabla con las objeciones', 'quiero más espacio entre secciones', 'quiero un emoji distinto en cada heading'. Edita las funciones build_html_venta/cliente/equipo en main.py y haz push.

Cambiar el destino de los avisos

PROMPT · DE TELEGRAM A SLACK

Quiero que los avisos lleguen a Slack en vez de Telegram. Reemplaza la función mandar_telegram por mandar_slack usando un webhook de Slack. Necesito que me digas qué tengo que crear en Slack y qué variable de entorno tengo que añadir a Railway.

Cambiar el modelo de Claude

PROMPT · CAMBIAR MODELO

Anthropic ha sacado un modelo nuevo llamado [NOMBRE]. Cámbialo en main.py donde corresponda (MODEL_DETECCION o MODEL_ANALISIS) y haz push. Avísame si hay algún cambio en la API que requiera más ajustes.

Prompts para entender mejor el sistema

Que te explique cómo funciona algo

PROMPT · EXPLÍCAMELO

No entiendo qué hace [PARTE DEL CÓDIGO O CONCEPTO]. Explícamelo como si tuviera 10 años y no supiera nada de programación. Usa analogías con cosas del día a día.

Que te haga un resumen del estado actual

PROMPT · DAME EL ESTADO

Dame un resumen de cómo está montado mi sistema ahora mismo: qué archivos importantes hay, qué hace cada uno, qué variables tiene Railway, y qué se podría mejorar. Hazlo como si fueras mi CTO explicándome el sistema.

La regla de oro con Claude Code

Cuando algo no funciona, NO intentes arreglarlo solo si nunca has programado. Pega el error tal cual a Claude Code y dile 'arréglalo'. Tarda 30 segundos. Si lo intentas solo sin saber, puedes romper algo y tardar 2 horas en deshacerlo.

EL SECRETO

Claude Code es brutal porque NO se cansa, NO se enfada, y SIEMPRE tiene contexto. Habla con él como con un amigo programador con el que estás aprendiendo. Pregunta lo que no entiendas. Pídele que te explique antes de cambiar algo. Es tu copiloto, úsalo así.

16. FAQ — preguntas reales

¿Cuánto cuesta esto al mes de verdad?

Con 100 llamadas al mes: ~$15 ($10 Anthropic + $5 Railway). Sin Fathom Pro, totalmente gratis su parte. Con Fathom Pro (ilimitado): ~$34/mes.

¿Mis llamadas las verá alguien?

No. Anthropic no usa datos de API para entrenar, tu transcripción se procesa y se descarta. Google Drive es tu Drive personal. Fathom ya tiene tus llamadas porque es quien las graba. El servidor (Railway) es tuyo. Nadie más que tú accede.

¿Puedo compartirlo con mi equipo?

Sí, pero cada uno necesita estar como Test User en tu OAuth (paso 7.3). Si vais a ser más de 10-15 personas, mejor que cada uno monte su propio sistema (es lo mismo, gratis los demás servicios).

¿Funciona con Zoom o Microsoft Teams?

Funciona con lo que Fathom soporte: Google Meet, Zoom, Teams. Cualquier llamada que Fathom grabe, el sistema la procesa.

¿Y si Anthropic saca un modelo mejor?

Cambias MODEL_DETECCION o MODEL_ANALISIS en main.py por el nuevo nombre. Push, redeploy. 30 segundos.

¿Puedo cambiar Telegram por Slack o WhatsApp?

Sí. Para Slack: necesitas un Incoming Webhook de Slack. Para WhatsApp Business: API de Meta. Pídeselo a Claude Code: 'reemplaza mandar_telegram por mandar_slack', te lo hace.

¿Cuánto tarda el setup completo?

Primera vez: 90 minutos siguiendo el manual. Segunda vez (sabiendo lo que haces): 20 minutos.

¿Necesito saber programar?

Cero. Si sabes copiar y pegar, lo tienes. Si te atascas con algo, le pides a Claude Code que te lo arregle y lo hace por ti.

NICO ACERO · AUTOSETTER™

Tus llamadas trabajan para ti, no contra ti.

Cuelga. Vete a hacer otra cosa. El Doc llega solo.

@autosetter.ai

MANUAL v1.0 · CEREBRO DE LLAMADAS™ · 2026