



Claude Code Telegram Bot

Puente técnico avanzado que permite ejecutar y controlar Claude Code (la CLI agéntica de Anthropic) de forma remota mediante Telegram. Diseñado para ingenieros de software, DevOps y arquitectos de sistemas, facilita la interacción con bases de código, diagnósticos rápidos y aplicación de correcciones urgentes sin necesidad de terminal física, manteniendo la persistencia de sesión por proyecto y contexto de trabajo.

[Visitar Sitio Oficial](#) [Preguntar a ChatGPT](#) [Preguntar a Claude](#) [Preguntar a Grok](#)

Contenido del Dossier

- [Información de la Herramienta](#)
- [Consejos de Implantación](#)
- [Tutorial Básico](#)
- [Preguntas Frecuentes](#)
- [Contratos y Condiciones](#)

INFORMACIÓN DE LA HERRAMIENTA

Qué y para quién es

Claude Code Telegram Bot es un puente técnico avanzado que permite ejecutar y controlar **Claude Code** (la CLI de IA agéntica de Anthropic) de forma remota a través de una interfaz de Telegram. Está diseñado específicamente para ingenieros de software, DevOps y arquitectos de sistemas que necesitan interactuar con su base de código, realizar diagnósticos rápidos o aplicar correcciones urgentes sin necesidad de estar frente a una terminal física o abrir una VPN al servidor de desarrollo.

Principal ventaja profesional

En mi opinión profesional, la capacidad de **persistencia de sesión automática por proyecto** es el factor diferencial. A diferencia de otros bots simples, este mantiene el contexto del hilo de trabajo; puedes iniciar un refactor complejo en la oficina y continuarlo exactamente donde lo dejaste desde el móvil mientras te desplazas, con Claude recordando los archivos leídos y las pruebas ejecutadas.

Para quién no es

No es una herramienta para perfiles de gestión o usuarios finales sin conocimientos técnicos. Quienes busquen un simple "chat con PDF" o busquen una interfaz puramente visual se sentirán frustrados. Tras probarlo, queda claro que si el usuario no comprende conceptos como sandboxing, webhooks o no está cómodo delegando la ejecución de comandos bash a una IA, la curva de desconfianza será demasiado alta en un entorno corporativo.

Funcionalidades clave

- **Modo Agéntico Conversacional:** Permite hablar con el código en lenguaje natural; Claude analiza, edita archivos y ejecuta tests de forma autónoma.
- **Niveles de Verbosidad Ajustables:** Control real sobre cuánta información de los procesos en segundo plano (razonamiento de la IA) queremos ver en el chat.
- **Seguridad Multicapa:** Incluye sandboxing de directorios (no puede salir de la carpeta aprobada), lista blanca de IDs de Telegram y validación de comandos peligrosos.
- **Automatización Basada en Eventos:** Servidor de webhooks integrado para recibir notificaciones de GitHub (PRs, Issues) y que Claude genere resúmenes o revisiones automáticas.
- **Transcripción de Voz:** Soporte para enviar instrucciones por audio procesadas localmente o vía API (Whisper/Mistral).

Precios

- **Versión Gratuita:** El software es Open Source bajo licencia MIT. No hay coste por la herramienta en sí.
- **Costes Operativos:** El usuario debe asumir el coste de la API de Anthropic (Claude) y, opcionalmente, OpenAI/Mistral si utiliza transcripción de voz.
- **Gestión de Gasto:** El bot permite configurar un `CLAUDE_MAX_COST_PER_USER` para evitar sorpresas en la facturación mensual de la empresa.

Perfil del usuario

- Empresas de desarrollo de software con equipos distribuidos.
- Administradores de sistemas que gestionan despliegues en remoto.
- Programadores que mantienen proyectos personales o profesionales en servidores domésticos/nubes privadas.
- Perfiles de Soporte Técnico de Nivel 3.

Nivel técnico requerido

- **Uso:** Medio. Requiere saber interactuar con una CLI aunque sea a través de chat.
- **Instalación:** Alto. Requiere conocimientos de Python 3.11+, gestión de entornos virtuales (Poetry/uv) y configuración de tokens de Telegram y APIs.
- **Competencias necesarias:** Familiaridad con entornos Unix/Linux, Git y gestión de variables de entorno `.env`.

Ejemplos de uso profesional

- **Corrección de bugs en movilidad:** Recibir una alerta de error, pedir al bot que lea los logs del servidor y que aplique un "hotfix" inmediato.
- **Revisión de Pull Requests:** Configurar un webhook para que, al abrirse una PR, el bot envíe a Telegram

un análisis crítico del código generado.

- **Auditoría de seguridad rápida:** Solicitar un escaneo nocturno de dependencias mediante un cron job programado en el bot que reporte vulnerabilidades detectadas.

Uso y distribución

- **Versión web:** No disponible (interfaz vía Telegram Web).

- **Versión escritorio/móvil:** A través de la aplicación oficial de Telegram en cualquier plataforma.

- **CLI:** Instalación mediante uv o pip desde el repositorio de GitHub.

Open source

Proyecto alojado en GitHub bajo licencia MIT, lo que permite su modificación y uso comercial sin restricciones significativas, siempre que se mantenga el aviso de copyright.

Integraciones

- **Facilidad de integración:** Nivel técnico alto para configuraciones personalizadas.

- **API propia:** Incluye un servidor Fast API para recibir webhooks.

- **Nativas:** Integración profunda con la CLI de GitHub (gh) y Git. Permite listar repositorios, clonar y gestionar ramas directamente desde el chat.

Notas finales

Veredicto técnico

Como profesional, valoro esta herramienta como una **utilidad de nicho pero de altísima eficiencia**. No es un juguete; es un terminal remoto vitaminado con IA. Para una consultora tecnológica o una startup, compensa con creces el esfuerzo de instalación inicial por la soberanía técnica que otorga sobre el código en situaciones de emergencia.

Información legal y licencias

Licencia MIT. El usuario es propietario intelectual de los cambios realizados por la IA, pero debe ser consciente de que el código se envía a los servidores de Anthropic. Es vital revisar las políticas de privacidad de datos de la empresa antes de usarlo con código propietario sensible.

Fuentes consultadas:

- [Sitio web oficial \(GitHub\)](#)

- [Documentación técnica de Claude Code](#)

- [Repositorio y Releases](#)

CONSEJOS DE IMPLANTACIÓN

Aplicación profesional

Según mi experiencia, esta herramienta es ideal para **agencias de desarrollo boutique, equipos DevOps y startups tecnológicas** con una cultura de "disponibilidad remota". No es solo un chatbot; es un nodo de control operativo. Permite que un arquitecto de sistemas gestione incidentes graves o revise código crítico desde su móvil sin necesidad de desplegar un entorno de trabajo completo. En mi opinión profesional, el presupuesto necesario es mínimo (costes de API de Anthropic según uso), pero el valor en **ahorro de tiempo de respuesta (MTTR)** ante incidentes es incalculable.

Madurez digital requerida

- **Usuarios y equipo:** Nivel avanzado. Deben estar familiarizados con el uso de terminales (CLI), Git y, lo más importante, entender las limitaciones y "alucinaciones" potenciales de una IA agéntica al manipular código real.
- **Empresa:** Media-Alta. La organización debe contar con políticas claras de seguridad de datos (ya que el código viaja a las API de Anthropic) y una infraestructura que permita ejecutar un proceso persistente en un servidor o máquina siempre encendida (Linux/Mac).

Plan orientativo de implantación

Pasos necesarios y estimaciones

- **Evaluación inicial (1-2 días):** Identificar los repositorios críticos que estarán en la APPROVED_DIRECTORY. Es vital definir qué nivel de acceso tendrá el bot para evitar riesgos innecesarios.
- **Configuración técnica (2-4 horas):** Instalación de Python 3.11+, configuración de Claude CLI (claude auth login) o API Keys, y creación del bot en Telegram vía @BotFather.
- **Prueba de concepto (1 semana):** Despliegue en un entorno de "sandbox" o con un proyecto no crítico para que los desarrolladores validen la respuesta del bot y la utilidad de los comandos /verbose.
- **Ajuste de seguridad y límites (1 día):** Configuración estricta de ALLOWED_USERS y CLAUDE_MAX_COST_PER_USER para evitar sorpresas en la facturación.
- **Formación y despliegue final:** Capacitación sobre el "Modo Agéntico" frente al "Modo Clásico" y activación de webhooks para notificaciones de GitHub.

Necesidades de formación del equipo

Es fundamental que el equipo aprenda a interactuar con Claude mediante **instrucciones de contexto**. Deben conocer cómo usar el comando /repo para saltar entre proyectos y cómo interpretar las trazas de razonamiento (pensamiento de la IA) que el bot muestra en Telegram antes de ejecutar un comando bash.

Perfiles necesarios

- **Perfiles técnicos:** Un ingeniero DevOps o un desarrollador Senior para la instalación y securización del servidor donde resida el bot.
- **Personal externo:** No es estrictamente necesario, dado que es un proyecto Open Source bien documentado.

Retorno de la inversión (ROI)

- **Tiempos:** Reducción drástica en el tiempo de "context switching". Resolver una duda sobre la base de código o aplicar un fix rápido desde Telegram puede pasar de 15 minutos (abrir portátil, VPN, IDE) a menos de 2 minutos.
- **KPIs:** Número de incidencias resueltas de forma remota, reducción del tiempo de revisión de Pull Requests (usando webhooks) y control de gasto mensual de API frente a horas de desarrollador ahorradas.

Otros

- **Seguridad Multicapa:** Lo que más me gusta es el sistema de 5 capas de defensa que incluye el proyecto (aislamiento de directorios, validación de patrones peligrosos tipo rm -rf y lista blanca de IDs). Al usarlo, te das cuenta de que la seguridad no es un añadido, sino la base del diseño.
- **Modo Verbo:** Mi consejo es empezar con /verbose 2 para entender cómo "piensa" la IA y qué herramientas utiliza (Grep, LS, Edit), y pasar a /verbose 1 una vez establecida la confianza.
- **Soberanía Técnica:** El bot soporta transcripción de voz local vía whisper.cpp, lo cual es excelente para quienes buscan privacidad máxima y ahorro en costes de API de terceros.

TUTORIAL BÁSICO

Esta guía detalla cómo configurar y optimizar el entorno para utilizar **Claude Code** de forma remota a través de **Telegram**, utilizando la integración desarrollada por RichardAtCT. Esta herramienta permite interactuar con tu código mediante lenguaje natural desde cualquier lugar, manteniendo sesiones persistentes.

Instalación

Para una correcta implementación, es fundamental seguir un flujo de dependencias específico.

1. Prerrequisitos del sistema:

- Servidor o máquina con **Python 3.11+**.
- **Claude Code CLI** instalado oficialmente en el sistema (`curl -fsSL https://claude.ai/install.sh | bash`).
- Un token de bot de Telegram (obtenido vía [@BotFather](#)).

2. Instalación del Bot (vía `uv` - Recomendado):

```
bash
uv tool install git+https://github.com/RichardAtCT/claude-code-telegram@latest
```

Según mi experiencia, usar uv garantiza que el entorno esté aislado y no genere conflictos con otras librerías de Python del sistema.

3. Configuración crítica (.env):

- Define APPROVED_DIRECTORY: Es la ruta absoluta donde residen tus proyectos. Por seguridad, el bot solo tiene acceso a lo que esté dentro de esta carpeta.
- Añade tu ALLOWED_USERS: Es imprescindible poner tu ID de Telegram (puedes obtenerlo escribiendo a [@userinfobot](#)) para evitar que terceros usen tu crédito de Claude.

Uso en el día a día

- **Modo Agente (Default):** No necesitas comandos. Simplemente escribe "Analiza el archivo main.py y añade validación de errores". Claude leerá, propondrá cambios y los aplicará tras tu confirmación.
- **Control de verbosidad:** Usa `/verbose 1` para ver solo qué herramientas usa (lectura, edición, bash) o `/verbose 0` para una respuesta limpia.
- **Gestión de contextos:** Si trabajas en varios repositorios, usa `/repo` para listar y cambiar entre ellos sin perder el hilo de la conversación.

Trucos de experto

- **Autenticación sin Keychain (Remote Mac):** Al usar el bot por SSH en un Mac, el acceso al llavero del sistema suele fallar. **En mi opinión profesional**, lo más eficiente es omitir la autenticación del CLI y usar directamente una API Key de Anthropic en el .env (`ANTHROPIC_API_KEY=sk-ant...`). Esto evita errores de permisos en sesiones remotas.
- **Flujo GitHub:** Si tienes configurado gh cli en tu servidor, puedes pedirle al bot: "Crea una rama para el fix, sube los cambios y abre un PR". Claude ejecutará los comandos por ti.
- **Mensajes de voz:** Si habilitas `ENABLE_VOICE_MESSAGES=true` y configuras un proveedor (Mistral o OpenAI), puedes enviar instrucciones complejas por voz mientras caminas y el bot las transcribirá y ejecutará.

Posibles problemas/incidencias

- **Bloqueo del Keychain:** Si corres el bot en una Mac remota mediante SSH, verás fallos de autenticación silenciosos. Se soluciona usando `make run-remote` o configurando `security unlock-keychain` en tu perfil de shell.
- **Timeout en operaciones largas:** Para refactorizaciones masivas, aumenta `CLAUDE_TIMEOUT_SECONDS` a 600 en el .env. El valor por defecto (300) puede quedarse corto si Claude debe analizar cientos de archivos.
- **Límites de coste:** Al usarlo por Telegram es fácil "pasarse" de presupuesto. Configura `CLAUDE_MAX_COST_PER_USER=X.X` para tener un control estricto del gasto diario.

Otros

- **Seguridad:** El bot implementa protección contra Path Traversal. Intentar acceder a archivos fuera del APPROVED_DIRECTORY provocará un bloqueo automático de la acción.
- **Automatización Extra:** Puedes habilitar el servidor de Webhooks del bot para recibir notificaciones

proactivas de GitHub (Push, Pull Requests) y que Claude te haga un resumen del cambio directamente en el chat de Telegram.

PREGUNTAS FRECUENTES

¿Qué es exactamente Claude Code Telegram Bot?

Es una herramienta técnica de código abierto que actúa como interfaz remota para Claude Code, la CLI de inteligencia artificial agéntica de Anthropic. Permite a desarrolladores y administradores de sistemas interactuar con su base de código, ejecutar comandos y realizar correcciones en servidores desde la aplicación de Telegram, sin necesidad de acceso físico a la terminal.

¿Para qué sirve en un entorno profesional?

Su utilidad principal radica en la gestión remota de proyectos de software. Permite realizar diagnósticos de errores, aplicar 'hotfixes' urgentes, revisar Pull Requests mediante webhooks y ejecutar pruebas automáticas. Al mantener la persistencia de sesión, un profesional puede retomar una tarea de refactorización compleja iniciada en la oficina directamente desde su dispositivo móvil.

¿Cuánto cuesta y qué costes operativos implica?

El software es gratuito y se distribuye bajo la licencia MIT. Sin embargo, el uso profesional conlleva costes externos: el consumo de la API de Anthropic (Claude) por el procesamiento de tokens y, opcionalmente, el coste de APIs de transcripción de voz como Whisper. El bot incluye funciones de control de gasto para limitar el presupuesto máximo por usuario.

¿Es código abierto y dónde puedo descargarlo?

Sí, el proyecto es totalmente Open Source bajo licencia MIT. El código fuente, las versiones y la documentación técnica están disponibles públicamente en su repositorio oficial de GitHub (RichardAtCT/claude-code-telegram), permitiendo auditorías de seguridad y modificaciones personalizadas.

¿Cumple con la normativa de privacidad y seguridad de datos?

El bot incluye medidas de seguridad como sandboxing de directorios para limitar el acceso al sistema de archivos y listas blancas de IDs de Telegram. No obstante, al utilizar modelos de lenguaje externos, el código se envía a los servidores de Anthropic. Las empresas deben validar que el uso de estas APIs de terceros cumple con sus políticas internas de privacidad y cumplimiento regulatorio.

¿Qué nivel técnico se requiere para su implementación?

El nivel de instalación es alto. Se requiere experiencia avanzada en administración de sistemas, específicamente en Python 3.11+, gestión de entornos virtuales (Poetry o uv), configuración de variables de entorno (.env) y manejo de tokens de API y webhooks. No está diseñado para usuarios finales sin conocimientos de programación o gestión de servidores.

¿Es una tecnología segura para ejecutar comandos en un servidor?

Aunque ofrece una capa de seguridad con validación de comandos peligrosos y restricción de carpetas, el bot delega la ejecución de comandos bash a la IA. Se recomienda su uso en entornos controlados y por perfiles técnicos que comprendan las implicaciones de otorgar permisos de ejecución a un agente de IA en producción.

¿Qué integraciones nativas ofrece?

Se integra profundamente con Git y la CLI de GitHub (gh), permitiendo clonar repositorios, gestionar ramas y analizar cambios. También incluye un servidor FastAPI para recibir notificaciones automáticas de eventos externos e integrarse en flujos de CI/CD ya existentes.

CONTRATOS Y CONDICIONES

Opinión inicial

Tras analizar la arquitectura técnica de este puente entre Telegram y Claude Code, clasifico el riesgo legal de esta herramienta como **alto** para cualquier empresa española. Técnicamente, estamos ante una "interfaz no oficial" que actúa como intermediaria. Según los documentos consultados sobre el uso de la API de Anthropic y los términos de Telegram, el principal conflicto reside en la pérdida de control del perímetro de seguridad: el código fuente de la empresa sale del entorno seguro corporativo para viajar por los servidores de Telegram (empresa con sede fuera de la UE y políticas de privacidad cuestionables en el ámbito profesional) antes de llegar a la IA. En mi opinión profesional, aunque la licencia MIT permite su uso comercial, el despliegue de esta herramienta sin un túnel de cifrado extremo a extremo y sin un acuerdo de procesamiento de datos (DPA) con Telegram vulnera principios básicos del RGPD.

Principales recomendaciones

- Prohibir estrictamente el uso de esta herramienta con repositorios que contengan datos de carácter personal, claves de API o secretos industriales.
- Realizar una Evaluación de Impacto relativa a la Protección de Datos (EIPD) antes de su implementación, dado que la herramienta permite la ejecución de comandos bash de forma remota, lo que supone un vector de riesgo crítico de seguridad.
- Configurar el bot exclusivamente en servidores privados bajo control de la empresa y limitar el acceso mediante una "lista blanca" de IDs de Telegram verificados.
- Desactivar la función de transcripción de voz si se utiliza a través de APIs de terceros para evitar la transferencia adicional de registros biométricos o voz a otros proveedores.

Ley de Inteligencia Artificial (AI Act)

- El uso de Claude Code mediante este bot se clasifica generalmente como de **riesgo bajo o limitado**, al ser una herramienta de asistencia técnica para programación. Sin embargo, si el bot se utiliza para generar código que tome decisiones automatizadas sobre personas físicas, la clasificación podría elevarse.
- Es obligatorio informar a los empleados que están interactuando con un sistema de IA y no con un humano, cumpliendo con el deber de transparencia del AI Act.

Privacidad y protección de datos

- **Responsabilidades:** La empresa española actúa como Responsable del Tratamiento. El desarrollador del bot (repositorio GitHub) no asume ninguna responsabilidad legal por fugas de datos.
- **Ubicación de los datos:** Los datos transitan por la infraestructura de Telegram (global) y se procesan en los servidores de Anthropic (principalmente EE.UU.).
- **Transferencia internacional:** Existe una transferencia internacional de datos de alto riesgo. Al usar Telegram como interfaz, los metadatos y el código enviado no están protegidos bajo los marcos de adecuación estándar de la UE de la misma forma que una instancia corporativa aislada.
- **Derechos ARCO:** La gestión de derechos (Acceso, Rectificación, Cancelación, Oposición) se vuelve compleja ya que el historial de conversaciones queda almacenado tanto en el servidor local del bot como en la nube de Telegram.

Propiedad intelectual

- **Propiedad de datos:** El código enviado a través del bot sigue siendo propiedad de la empresa, pero según las condiciones de la API de Anthropic, es fundamental verificar si se ha optado por la exclusión de entrenamiento de modelos para evitar que el código propietario alimente futuras versiones de la IA.
- **Propiedad del resultado:** Bajo la legislación española, las obras generadas íntegramente por una IA no gozan de protección de derechos de autor, aunque el programador que dirige la herramienta puede reclamar la propiedad sobre el resultado final tras una edición humana significativa.

Usos y prohibiciones

- **Usos prohibidos:** Tratamiento de bases de datos de clientes reales, acceso a entornos de producción con datos sensibles de salud o financieros, y bypass de medidas de seguridad corporativas (shadow IT).
- **Usos admitidos:** Refactorización de código genérico, generación de pruebas unitarias en entornos de desarrollo aislados (sandboxing) y documentación de funciones técnicas.

Seguridad y certificaciones

- **Seguridad:** La herramienta carece de certificaciones tipo SOC2 o ISO 27001 por sí misma. La seguridad

depende íntegramente de la configuración del administrador (variables de entorno y permisos de sistema).

- **Riesgo técnico:** La ejecución de comandos bash vía Telegram es una vulnerabilidad crítica si el token del bot es comprometido.

Otros

- Es imperativo recordar que el uso de Telegram para fines profesionales en España debe estar regulado por una política de uso de herramientas de comunicación interna, ya que el uso de cuentas personales para interactuar con el bot de la empresa puede derivar en problemas de cumplimiento laboral y de protección de datos.

Fuentes consultadas:

- [Condiciones legales de Anthropic](#)
- [Política de Privacidad de Telegram](#)
- [Licencia MIT del proyecto en GitHub](#)
- [Reglamento General de Protección de Datos \(RGPD\)](#)

Para más información y herramientas:

Explora look4.tools para descubrir las mejores soluciones tecnológicas del mercado.

[Inicio](#) [Todas las herramientas](#) [Categorías](#)

Este documento ofrece recomendaciones generadas mediante análisis humano y sistemas de IA automatizados. La información tiene carácter meramente informativo y no constituye asesoramiento legal, profesional ni garantía de resultados. Las marcas, logotipos y nombres comerciales pertenecen a sus respectivos propietarios y se utilizan únicamente con fines identificativos.