

## Módulo 4. Casos avanzados de GPT: custom actions



☰ 1. ¿Qué son las custom actions [acciones personalizadas] y para qué sirven?

☰ 2. Posibilidades que habilitan las custom actions

☰ 3. Cómo se configuran las custom actions en GPT personalizados

☰ Referencias

☰ Descarga en PDF

# 1. ¿Qué son las custom actions [acciones personalizadas] y para qué sirven?

---

En términos simples, son extensiones que permiten que nuestro GPT tome acciones o consulte información fuera del propio ChatGPT. Hasta ahora, la interacción con un GPT no salía del entorno de ChatGPT, es decir, todas las respuestas eran en texto, imágenes o archivos, pero sin influir en aplicaciones externas. Mediante las custom actions, podemos conectar al asistente con API de terceros, bases de datos, servicios web u otras aplicaciones, de forma que pueda, por ejemplo, buscar datos en tiempo real o interactuar con sistemas externos (OpenAI, 2023, <https://shorturl.at/9DUnx>). En esencia, una custom action actúa como un «brazo operativo» del GPT: el modelo analiza la solicitud del usuario, decide si es necesario usar alguna de las acciones disponibles y, de ser así, formula una llamada a la API externa definida, incorporando la respuesta de esa API en su respuesta final al usuario (Brigham Young University, s. f., <https://shorturl.at/6eeSq>). Todo esto ocurre de manera transparente para el usuario, que simplemente ve a su asistente resolviendo tareas que van más allá de la conversación estática. Esto convierte al GPT en un verdadero asistente digital proactivo, capaz de integrarse con las herramientas y datos de la organización para ofrecer soluciones más completas.

CONTINUAR

## 2. Posibilidades que habilitan las custom actions

---

Las posibilidades que se habilitan con las acciones personalizadas son muy amplias y continuamente están surgiendo nuevos casos de uso a medida que la comunidad las explora (OpenAI, 2023, <https://shorturl.at/9DUnx>). A continuación, destacamos algunas categorías de funciones que un GPT personalizado puede realizar al aprovechar custom actions, ilustradas con ejemplos útiles en contextos profesionales.

### Consulta y extracción de información dinámica —

cualquier dato disponible vía API puede ser traído a la conversación. Por ejemplo, un asistente de **business intelligence** [inteligencia de negocios] podría conectarse a una API financiera para obtener en tiempo real la cotización de acciones o monedas o a una API del clima para informar el pronóstico en distintas ciudades cuando se lo pida el usuario. Otro caso: un **asistente de recursos humanos** podría usar una acción para consultar el saldo de vacaciones o información de nómina de un empleado desde el sistema HR de la empresa en tiempo real. Esto soluciona la limitación de la información estática en los archivos que le cargamos al GPT como conocimiento, pudiendo ahora, a través de las acciones, consultar datos actualizados en aplicaciones externas.

### Integración con bases de datos y sistemas internos —

un GPT empresarial puede tener acciones para leer o escribir en las bases de datos de la organización. Por ejemplo, un **asistente de soporte técnico** podría, dado un ID de cliente proporcionado en la pregunta del usuario, realizar una consulta a la base de datos de clientes y recuperar el historial de tickets de ese cliente para brindar una respuesta más informada. También podría registrar automáticamente una nueva incidencia en el sistema después de la conversación.

### Control de herramientas ofimáticas y productividad —

muchos profesionales usan hojas de cálculo, gestores de tareas, calendarios y otras herramientas en su día a día. Con acciones personalizadas, podemos conectar el GPT a estos servicios. Un ejemplo concreto es la integración con **Google Sheets**: mediante la API de Google Sheets, el asistente podría leer datos de una hoja (por ejemplo, extraer el valor de ciertas celdas para incluirlo en una respuesta) o agregar nuevas filas con información que el usuario le proporcione. Imagina un **asistente de ventas** al que le dictas nuevos prospectos y automáticamente los agrega a una hoja de cálculo compartida. De forma similar, con una acción hacia la API de **Google Calendar**, el GPT puede crear eventos o listar tus próximas reuniones en agenda. Incluso se han implementado GPT que interactúan con **Notion, Trello** u otras aplicaciones de productividad, permitiendo, por ejemplo, extraer notas

desde Notion o crear tarjetas en Trello usando solo comandos en lenguaje natural. Todo esto es posible porque prácticamente **todas estas aplicaciones modernas ofrecen API** y con las custom actions podemos poner esas API al servicio de nuestro asistente.

### Automatización de comunicaciones (correo, chat, notificaciones) —

uno de los usos más llamativos es habilitar al GPT para que envíe comunicaciones en nuestro nombre. Un caso práctico: un asistente personal con acceso a la API de Gmail u Outlook podría redactar y enviar correos electrónicos. OpenAI, a través de su biblioteca de acciones de ejemplo, ha demostrado que es viable conectar un GPT a Gmail con OAuth seguro, permitiéndole **leer, redactar y enviar e-mails** desde la cuenta del usuario (Westmoreland, 2024, <https://shorturl.at/iFexz>). Con las instrucciones adecuadas, el modelo incluso puede resumir correos largos o adjuntar información relevante antes de enviar. De igual manera, se podría integrar con API de mensajería instantánea (por ejemplo, Slack o Microsoft Teams) para que el asistente envíe un mensaje a un canal de equipo cuando se le pida (imaginemos: «**GPT, avisa en el canal #proyectos que el informe está listo**» y que el GPT use la acción de «Slack» para publicar ese aviso automáticamente).

### Asistentes especializados por dominio —

combinando las capacidades anteriores, es posible crear asistentes altamente especializados para sectores profesionales específicos. Por ejemplo, en marketing, un GPT podría conectarse a la API de una plataforma de automatización de marketing para extraer métricas de una campaña y luego enviar un reporte. En logística, un GPT podría consultar la API de un sistema de tracking de envíos para informar al instante dónde está un paquete y luego quizás notificar vía e-mail al cliente esa información. En finanzas, un GPT analista podría traer datos de distintas fuentes (bolsa, tipo de cambio, bases internas) y presentarlos consolidados al usuario. Cada dominio tiene sus fuentes de datos y servicios clave y, gracias a las *custom actions*, un asistente puede integrarse con ellos y proporcionar valor agregado. En muchos casos, estas acciones reemplazan tener que consultar manualmente varias aplicaciones: el asistente lo hace por el usuario.

En resumen, las *custom actions* habilitan a los GPT personalizados a convertirse en agentes de acción y consulta en el entorno profesional. Las tareas antes fragmentadas entre distintas aplicaciones ahora se pueden orquestar desde una conversación unificada. Esto supone un aumento en eficiencia (el usuario ahorra tiempo y reduce errores al delegar tareas mecánicas al asistente) y también en inteligencia contextual (el GPT puede combinar su generación de lenguaje con datos frescos de fuentes fiables, produciendo respuestas más exactas y útiles). Estas posibilidades desarrolladas son solo la punta del iceberg, ampliándose las mismas a la imaginación y necesidades de cada usuario.

CONTINUAR

### 3. Cómo se configuran las custom actions en GPT personalizados

---

**La configuración de una *custom action* puede parecer técnica, pero OpenAI ha proporcionado herramientas para simplificar el proceso. En esencia, configurar una acción consiste en describirle al GPT cómo acceder a una API externa. A continuación, presentamos una guía paso a paso, integrando el uso del asistente de configuración de OpenAI, la definición del esquema OpenAPI correspondiente, la gestión de parámetros, autenticación y pruebas.**

Antes de sumergirnos en los pasos técnicos, es importante aclarar un concepto clave para comprender cómo funcionan las custom actions: ¿qué es una API?

Una API (sigla en inglés de application programming interface, [interfaz de programación de aplicaciones]) es un conjunto de reglas que permite que diferentes programas o servicios se comuniquen entre sí. Podríamos imaginarla como un menú en un restaurante: tú (el usuario) le pides al mozo un plato y él lleva tu pedido a la cocina. No necesitas saber cómo se cocina, solo qué opciones están disponibles. La API es ese «mozo» que traduce tu pedido en una acción entendible por el sistema.

En el caso de los GPT, cuando configuramos una custom action, lo que hacemos es enseñarle al asistente cómo hablar con una API externa. Por ejemplo, si queremos que consulte información en una hoja de cálculo, le indicamos cómo pedirle a la API de Google Sheets que le devuelva esos datos.

Cuando hacemos un «pedido» a través de una API, utilizamos lo que se conoce como métodos HTTP, que son como las distintas formas de interactuar con el sistema. Los más comunes son los siguientes.

## GET —

se usa para **consultar o leer** información. Por ejemplo, un GPT que pregunta a una API cuántos productos quedan en stock estaría haciendo una solicitud GET.

## POST —

se usa para **crear o enviar** nueva información. Si el GPT crea un nuevo evento en un calendario o agrega una fila en una hoja de cálculo, probablemente está usando un POST.

## PUT o PATCH —

se usan para **actualizar** algo existente. Por ejemplo, si el GPT cambia el estado de un ticket de soporte de «pendiente» a «resuelto», se estaría usando uno de estos métodos.

## DELETE —

se usa para **eliminar** información.

Todo este «lenguaje común» se define mediante algo llamado esquema OpenAPI, que es como una guía que le explica al GPT qué opciones hay disponibles, qué necesita para hacer un pedido válido (por ejemplo, un ID de producto, una fecha, etc.) y qué tipo de respuesta puede esperar. El esquema OpenAPI está generalmente en formato JSON, por lo que puede tener cierta complejidad desarrollarlo, pero con la ayuda del asistente ActionsGPT proporcionado por OpenAI, el proceso resulta más sencillo incluso para usuarios sin conocimiento técnico.

**Figura 1. ActionsGPT**



## ActionsGPT

Por ChatGPT 

Helps you create OpenAPI specifications from documentation, code examples, cURL commands, or just a description of how to use an API.

Create a spec for  
this API: curl -G  
https://api.stripe.c...

Add this endpoint  
to the spec: const  
response = await...

Make a spec to call  
the endpoint at  
https://api.openai....

Here's the  
documentation:  
https://platform.o...

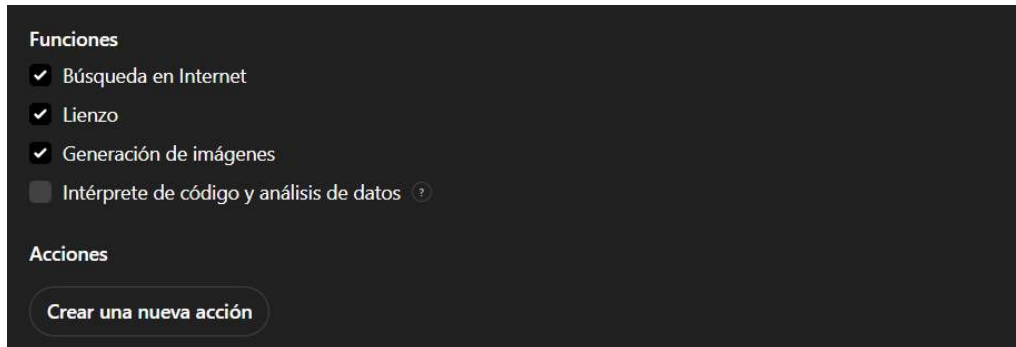
+ Pregunta lo que quieras



Fuente: captura de pantalla de ChatGPT (<https://shorturl.at/6iTDQ>).

Con esta idea en mente, vamos a ver cómo se configuran estas acciones paso a paso.

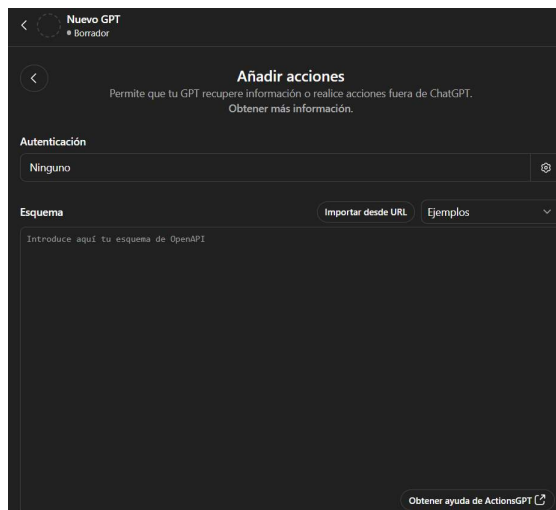
## Iniciar la creación del GPT y abrir el configurador de acciones



**Figura 2. Funciones**

Primero, debemos tener un GPT personalizado creado o en proceso de creación. Esto se hace desde la sección «Create a GPT» [crear un GPT] en la interfaz de ChatGPT (OpenAI, s. f., <https://shorturl.at/zEzsz>). Para agregar una custom action, vamos a la pestaña «Configure» [configurar] de nuestro GPT y desplazamos hacia abajo hasta encontrar la sección «Actions» [acciones] (OpenAI, s. f., <https://shorturl.at/zEzsz>). Allí habrá un botón o enlace que dice «Create new action» [Crear nueva acción]. Al hacer clic, se abrirá una interfaz similar a la mostrada en la figura anterior, con campos para «Autenticación» y «Esquema» (schema) (Brigham Young University, s. f., <https://shorturl.at/GeeSq>). Nota: si estamos usando el «Assistant Builder» conversacional, también podríamos indicarle algo como «quiero que este GPT pueda conectar con X servicio para hacer Y», y el asistente nos ayudará a preparar la acción, pero de todos modos terminaremos revisando o llenando los detalles en la pestaña de configuración.

## Definir la conexión y autenticación (si aplica)



**Figura 3. Añadir acciones**

En la ventana de «Add actions» [añadir acciones], lo primero que vemos es el apartado de «Authentication» [autenticación]. Aquí debemos especificar cómo se autentica el GPT frente a la API externa, en caso de que sea necesario. Este paso de autenticación es opcional porque no todas las API lo requieren, pero es crítico hacerlo bien cuando aplique para garantizar que el GPT tenga el acceso necesario. Hay dos métodos comunes: API Key (clave de API) y OAuth 2.0.

- Si la API no requiere autenticación (por ejemplo, una API pública o un endpoint abierto), podemos dejar «Authentication: None», es decir, dejarla en blanco.
- Si la API exige una clave secreta (API Key) que el GPT debe incluir en sus llamadas, elegiremos la opción de API Key y proporcionaremos la clave en el formulario. Por lo general, el GPT Builder nos pedirá ingresar el valor de la clave y quizá el formato de cabecera o parámetro donde debe usarse (por ejemplo, «Header “Authorization: Bearer [API\_KEY]”» o similar).
- Si la acción va a acceder a datos personales de cada usuario (por ejemplo, leer el e-mail o calendario del usuario), entonces se utiliza OAuth. En este caso, configuraremos los detalles de OAuth para la integración con el servicio (por ejemplo, con Google, proporcionar el Client ID y Client Secret de nuestra app, URL de autorización y token, y los scopes de acceso requeridos) (Brigham Young University, s. f., <https://shorturl.at/6eeSq>). Con OAuth, cuando un usuario quiera usar el GPT, se le pedirá que inicie sesión y otorgue permiso a la aplicación (OpenAI) para acceder a su cuenta de ese servicio. De esa manera, cada usuario conecta su propia cuenta (por ejemplo, su propio Gmail) y el GPT actuará en nombre de esa persona con su autorización.

¿Cómo decidir entre API Key y OAuth? Una buena regla es: si la acción accede a un recurso compartido o general (por ejemplo, consultar una API de clima o una base de datos común de la empresa), usar API Key es más sencillo y todos los usuarios del GPT usarán la misma credencial/API Key. En cambio, si la acción accede a datos privados de cada usuario (correo personal, calendario personal, archivos personales), se debe usar OAuth para que cada uno autentique su cuenta por separado (Brigham Young University, s. f., <https://shorturl.at/6eeSq>). OpenAI admite ambos métodos en la configuración de acciones. Este paso de autenticación es opcional porque no todas las API lo requieren, pero es crítico hacerlo bien cuando aplique para garantizar que el GPT tenga el acceso necesario.

## Proporcionar el esquema OpenAPI de la acción

Este es el corazón de la custom action: describir la API que el GPT podrá invocar. OpenAI utiliza el estándar OpenAPI Specification (anteriormente conocido como Swagger) para estructurar esta descripción (Sniffin, 2023, <https://shorturl.at/trtAF>). Si no estás familiarizado, un esquema OpenAPI es básicamente un documento (en formato JSON o YAML) que lista los endpoints disponibles de una API con sus métodos (GET, POST, etc.), parámetros de entrada y formatos de respuesta. Para la acción personalizada, podemos seguir dos caminos: escribir o pegar el esquema manualmente en el cuadro de texto, o usar la opción Import from URL si tenemos la URL donde está publicada la especificación. Muchas API públicas proveen su archivo OpenAPI (o un archivo de manifiesto de plugin en el caso de haber desarrollado previamente un plugin de ChatGPT) que se puede importar directamente (OpenAI, s. f., <https://shorturl.at/zEzsz>). Por ejemplo, si quisiéramos conectar una API meteorológica como Open-Meteo, podríamos copiar su especificación OpenAPI. Si no tenemos uno predefinido, podemos escribir un esquema básico describiendo solo lo que necesitamos. Esta redacción es recomendable realizarla apoyándose en el asistente mencionado. Explicando lo que quieres que haga el GPT a través de la acción y proporcionando los detalles de la API de la aplicación externa, el asistente puede generar el esquema listo para que lo pegues y lo pruebes.

¿Qué debe incluir el esquema? Al menos, la URL base del servicio (campo servers con la URL del API) y, dentro de paths, cada endpoint que queramos permitir. Por ejemplo, supongamos que nuestra acción es consultar una hoja de Google Sheets a través de una API gateway propia; podríamos tener un endpoint /getsheet de tipo GET que requiere un parámetro de hoja o rango y devuelve datos en JSON. En el esquema lo describimos con su método, parámetros (nombre, tipo, ubicación — query, path, body—, si son obligatorios o no) y las posibles respuestas (al menos un 200 con esquema de respuesta). Cuanto más completo y preciso sea el esquema, mejor podrá el GPT usar la acción correctamente. El modelo leerá esta especificación para saber cuándo y cómo llamar a la acción. Por ejemplo, si el usuario pregunta «¿Qué reuniones tengo mañana?» y hemos configurado un endpoint de calendar /events con un parámetro fecha, el GPT detectará que para responder debe llamar a /events pasando la fecha de mañana, obtener la lista y luego formatearla en la respuesta. Consejo: empieza definiendo solo lo necesario. No hace falta describir toda la API, solo las operaciones que nuestro asistente realmente va a usar. Esto reduce la complejidad.

En la interfaz, tras pegar el esquema OpenAPI en el campo «Schema», el sistema de OpenAI suele validar que esté bien formado (sintaxis JSON/YAML correcta, campos necesarios presentes). Si algo está mal, te avisará para que lo corrijas. OpenAI

también requiere en algunos casos agregar un enlace a una política de privacidad si vas a publicar el GPT públicamente y este usa acciones debido al tema de cumplimiento de datos externos (Sniffin, 2023, <https://shorturl.at/trtAE>). Para acciones internas o de uso privado en tu empresa, esto no suele ser necesario, pero es bueno tenerlo en cuenta si planeas compartir el GPT con otros.

Si bien en este paso desarrollamos de forma técnica el contenido del esquema, no es necesario entender al 100 % todos los conceptos si no nos especializamos en desarrollo. Como se mencionó anteriormente, si este es tu caso, basta con probar interactuar con el asistente para que te genere el esquema.

## Configurar parámetros y pruebas iniciales

Una vez que hemos definido el esquema, es muy útil aprovechar la ventana para probar el GPT que está disponible mientras lo configuramos. Ahí mismo podemos pedirle que ejecute la acción y ver si funciona. Recomendamos construir también en la sección de «Instrucciones del GPT» algunas indicaciones sobre cómo utilizar la acción. Por ejemplo, indicarle que utilice la acción «enviarMails» cuando se le solicite que envíe un correo por Gmail. Además, una buena práctica es deshabilitar las capacidades redundantes si queremos que use nuestra acción personalizada. Por ejemplo, si nuestro GPT tiene una acción para buscar en una base de conocimientos interna, podemos desactivar la capacidad de web browsing [navegación web] así el modelo no intentará buscar en internet general para esa respuesta (Brigham Young University, s. f., <https://shorturl.at/6eeSq>).

Una vez que se haya probado en la sección de configuración, guardamos los cambios y volvemos a la interfaz de conversación con nuestro GPT personalizado para probar la versión final. Un primer test recomendado es hacer una pregunta sencilla que deba invocar la acción. Por ejemplo, si configuramos un GPT «Asistente del clima» con un endpoint de pronóstico, podríamos preguntarle: «¿Cómo está el clima en Córdoba ahora mismo?». Al enviar esa pregunta, deberíamos observar que el asistente quizá nos pida confirmación para usar la acción (la primera vez suele aparecer una alerta indicando «Este GPT quiere usar una acción externa, ¿permitir?», especialmente si es un dominio nuevo) (Daegs, 2024, <https://shorturl.at/WqzwD>). Tras permitirlo, el GPT hará la llamada a la API y nos dará la respuesta, por ejemplo: «En Córdoba ahora mismo hay 22 °C con cielo despejado» (los datos traídos son de la API). Si la respuesta del GPT parece alucinada o no coincide, podría ser que la acción no se ejecutó correctamente, por lo que debemos ajustar la acción si algo anda mal o el prompt para que el GPT entienda que debe utilizar la acción. La iteración es normal en esta etapa: rara vez todo funciona perfecto al primer intento.

**Vale la pena mencionar que configurar custom actions es probablemente la parte más técnica de la creación de GPT personalizados, pero también la más poderosa (Lau, 2025, <https://shorturl.at/SSzuz>). Si bien OpenAI ha simplificado el proceso con esta interfaz visual, sigue siendo fundamental entender qué hace la API que estamos conectando. Un error en la configuración puede**

**llevar a respuestas incorrectas o fallos en la llamada. Sin embargo, con paciencia y pruebas iterativas, incluso profesionales sin experiencia en programación pueden lograr integraciones útiles. OpenAI provee documentación oficial y ejemplos de acciones ya hechas (en su OpenAI Cookbook hay bibliotecas de acciones para Gmail, Google Calendar, etc. que incluyen instrucciones y esquemas listos para usar). Esto puede ser muy útil para implementar alguno de los ejemplos que publicaron o ponerlos en práctica solo para practicar y familiarizarse con la herramienta (Westmoreland, 2024, <https://shorturl.at/iFexz>). Apoyarse en estos recursos y en la comunidad (foros de desarrolladores, tutoriales actualizados) es buena idea para aprender las mejores prácticas.**

Una opción de gran valor que veremos más adelante es conectar estas acciones a herramientas de automatización *no-code*. Estas herramientas permiten conectar fácilmente miles de aplicaciones laborales en flujos de trabajo para automatizarlos por completo. Por lo que, al conectar GPT con estos flujos, abrimos un gran abanico de posibilidades para automatizar procesos.

### **Nota aclaratoria sobre uso de IA**

Este material fue asistido con herramientas de IA generativa para tareas de borrador, síntesis, reescritura y apoyo en la organización de contenidos. Cada sección fue revisada, editada y validada por el equipo humano, que verificó la precisión conceptual, la coherencia pedagógica y las fuentes citadas. Se invita a contrastar con las referencias bibliográficas incluidas y la documentación oficial. Dado que los modelos de IA evolucionan con rapidez, ciertas especificaciones técnicas podrían actualizarse; este texto refleja el estado del conocimiento al momento de su elaboración.

CONTINUAR

## Referencias

---

**Brigham Young University.** (2025). *Creating GPT Actions*. GenAI BYU. <https://genai.byu.edu/creating-gpt-actions>

**Daegs. [@Daegs].** (29 de junio de 2024). *Ayuda para depurar las acciones de CustomGPT que no se conectan a mi servidor* [Publicación en Reddit]. Reddit. [https://www.reddit.com/r/ChatGPTCoding/comments/1dr44fr/help\\_debug\\_customgpt\\_actions\\_not\\_hitting\\_my\\_server/?captcha=1](https://www.reddit.com/r/ChatGPTCoding/comments/1dr44fr/help_debug_customgpt_actions_not_hitting_my_server/?captcha=1)

**Lau, J.** (15 de agosto de 2025). *How to create a custom GPT: A beginner's guide*. Zapier. <https://zapier.com/blog/custom-gpt/>

**OpenAI.** (s. f.). *Creating a GPT*. OpenAI. <https://help.openai.com/en/articles/8554397-creating-a-gpt>

**OpenAI.** (6 de noviembre de 2023). *Presentamos GPT*. OpenAI. <https://openai.com/es-419/index/introducing-gpts/>

**Sniffin, A.** (12 de noviembre de 2023). *Building an OpenAI GPT with your API: A step-by-step guide*. Medium. <https://alexsniffin.medium.com/building-an-openai-gpt-with-your-api-a-step-by-step-guide-70168bef00e7>

**Westmoreland, A.** (24 de julio de 2024). *GPT Actions library – Gmail*. OpenAI Cookbook. [https://cookbook.openai.com/examples/chatgpt/gpt\\_actions\\_library/gpt\\_action\\_gmail](https://cookbook.openai.com/examples/chatgpt/gpt_actions_library/gpt_action_gmail)

CONTINUAR

## Descarga en PDF

---