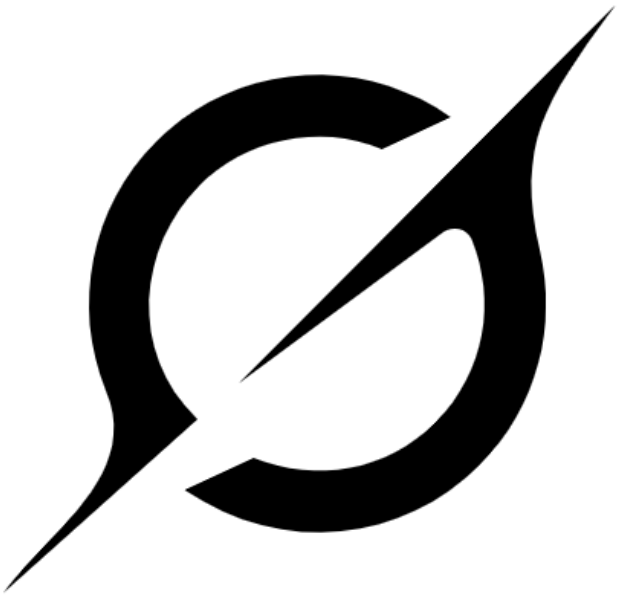
# Saturn Studio – Grok



## ¿Qué es este módulo?

**Grok** es una inteligencia artificial desarrollada por **xAI**, diseñada para ofrecer respuestas útiles y veraces, inspirada en *The Hitchhiker's Guide to the Galaxy* y en *J.A.R.V.I.S.* de Iron Man.
Su objetivo es acelerar el descubrimiento científico humano y ampliar nuestra comprensión colectiva del universo.

**¿Cómo obtener las credenciales? Haz clic aquí**

## Descripción de los comandos

### Listar Modelos

Lista los modelos disponibles en Grok.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Grok necesaria para listar los modelos. | credential |
| Asignar resultado a variable | Variable donde se almacenará el resultado.<br>**Ejemplo:**<br>{"models":[{"id":"grok-2-1212","name":"grok-2-1212"},{"id":"grok-3","name":"grok-3"}]} | {var} |

### Completar Texto

Completa un texto utilizando un modelo de Grok.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Grok necesaria para completar el texto. | Mis credenciales de Grok |

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Modelo | Modelo de Grok que se utilizará para completar el texto. | Model |
| Máx. tokens | Número máximo de tokens a generar. | 300 |
| Temperatura | Controla la creatividad del modelo: valores altos generan respuestas más creativas, y valores bajos más conservadoras. (Rango: 0 a 2, por defecto 1) | 1 |
| Texto | Texto base a completar. | Texto |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“text”:”¡Hola! La inteligencia artificial es un campo que busca crear sistemas capaces de realizar tareas que normalmente requieren inteligencia humana.”} | {var} |

## Generar Imagen

Genera una imagen utilizando un modelo de Grok.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Grok necesaria para generar la imagen. | Mis credenciales de Grok |
| Modelo | Modelo de Grok que se usará para generar la imagen. | Model |
| Prompt | Descripción o instrucción para generar la imagen. Sé lo más detallado posible. | Genera una imagen de un gato en el espacio |
| Formato de respuesta | Formato de respuesta para la imagen. Puede ser **URL** o **Base64**. | URL |
| Número de imágenes | Cantidad de imágenes a generar. | 2 |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“data”:[{“url”:”https://imgen.x.ai/xai-imgen/sample.jpeg”,”revised_prompt”:”Una fotografía de alta resolución de un perro verde en un bosque luminoso.”}]} | {var} |

## Interpretar Imagen

Analiza e interpreta una imagen utilizando un modelo de Grok.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Grok necesaria para interpretar la imagen. | Mis credenciales de Grok |
| Modelo | Modelo de Grok que se utilizará para el análisis. | Model |
| Máx. tokens | Número máximo de tokens a generar. | 300 |
| Temperatura | Controla la creatividad del modelo: valores altos generan respuestas más creativas, y valores bajos más conservadoras. (Rango: 0 a 2, por defecto 1) | 1 |
| Prompt | Instrucción para describir o analizar la imagen. | Describe la imagen en detalle |

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Formato de imagen | Formato de la imagen a enviar al modelo: **Archivo** o **URL**. | Archivo o URL |
| URL | URL de la imagen a analizar. Si se selecciona “Archivo local”, este campo será ignorado. | https://example.com/image.jpg |
| Ruta del archivo | Ruta local del archivo de imagen. Si se selecciona “URL”, este campo será ignorado. | image.jpg |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“text”:”Un perro verde se sienta tranquilo en un bosque soleado y frondoso, destacando por su pelaje brillante.”} | {var} |

---

# Saturn Studio – ClickUp



## ¿Qué es este módulo?

ClickUp es una plataforma de gestión del trabajo y productividad que permite organizar tareas, proyectos y objetivos en un solo lugar.
Ofrece herramientas para planificación, control del tiempo, automatizaciones, documentos colaborativos y gestión de equipos, todo en un entorno altamente personalizable.

**¿Cómo obtener las credenciales? Haz clic aquí**

## Descripción de los comandos

# Suscribirse a Webhook

Suscribe un webhook de ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para suscribirse al webhook. | credential |

## Esperar un evento

Espera un evento de ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para esperar un evento. | credential |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“event”:”taskCreated”,”history_items”:[{“id”:”5123456789012345678″,”type”:1,”field”:”assignee_add”}]} | {var} |

## Listar espacios de trabajo

Lista todos los espacios de trabajo en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para listar los espacios de trabajo. | credential |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“teams”:[{“id”:”9034647833″,”name”:”SaturnStudio”}]} | {var} |

## Listar todas las tareas

Lista todas las tareas de un espacio de trabajo en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para listar las tareas. | credential |
| Workspace | Espacio de trabajo desde el cual se listarán las tareas. | SaturnStudio |
| Space | Espacio desde el cual se listarán las tareas. | SaturnStudio |
| Folder | Carpeta desde la cual se listarán las tareas. | SaturnStudio |
| Lists | Lista desde la cual se listarán las tareas. | SaturnStudio |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“tasks”:[{“id”:”task-001″,”name”:”Design Landing Page”,”status”:”in progress”}]} | {var} |

## Listar subtareas

Lista todas las subtareas de una tarea en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para listar las subtareas. | credential |

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Workspace | Espacio de trabajo desde el cual se listarán las subtareas. | SaturnStudio |
| Space | Espacio desde el cual se listarán las subtareas. | SaturnStudio |
| Folder | Carpeta desde la cual se listarán las subtareas. | SaturnStudio |
| Lists | Lista desde la cual se listarán las subtareas. | SaturnStudio |
| Tarea principal | Tarea padre desde la cual se listarán las subtareas. | SaturnStudio |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“subtasks”:[{“id”:”st-001″,”name”:”Create wireframe”,”status”:”in progress”}]} | {var} |

## Listar usuarios

Lista todos los usuarios de un espacio de trabajo en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para listar los usuarios. | credential |
| Workspace | Espacio de trabajo desde el cual se listarán los usuarios. | SaturnStudio |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“users”:[{“id”:”st-001″,”name”:”Alice Johnson”}]} | {var} |

## Crear una tarea

Crea una tarea en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para crear la tarea. | credential |
| Workspace | Espacio de trabajo desde el cual se creará la tarea. | SaturnStudio |
| Space | Espacio desde el cual se creará la tarea. | SaturnStudio |
| Folder | Carpeta desde la cual se creará la tarea. | SaturnStudio |
| Lists | Lista desde la cual se creará la tarea. | SaturnStudio |
| Tarea principal | Si se selecciona, la tarea se creará como subtarea. | SaturnStudio |
| Nombre de la tarea | Nombre de la tarea a crear. | SaturnStudio |
| Contenido | Contenido de la tarea. | SaturnStudio |
| Asignado a | Usuario al que se asignará la tarea. | John Doe |
| Prioridad | Prioridad de la tarea (Urgente, Alta, Normal, Baja). | Urgente |
| Fecha de inicio | Fecha de inicio (AAAA-MM-DD). | 2025-05-01 |
| Fecha de vencimiento | Fecha de vencimiento (AAAA-MM-DD). | 2025-05-01 |

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“id”:”st-123abc”,”name”:”New SaturnStudio Subtask”} | {var} |

## Actualizar una tarea

Actualiza una tarea existente en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para actualizar la tarea. | credential |
| ID de la tarea | ID de la tarea a actualizar (puede ser subtarea). | SaturnStudio |
| ID de tarea principal | ID de la tarea padre (opcional). | SaturnStudio |
| Nombre de la tarea | Nombre de la tarea a actualizar. | SaturnStudio |
| Contenido | Contenido actualizado de la tarea. | SaturnStudio |
| Agregar asignado | ID del usuario que se asignará a la tarea. | 154785393 |
| Quitar asignado | ID del usuario que se eliminará de la tarea. | 154785393 |
| Prioridad | Prioridad de la tarea (Urgente, Alta, Normal, Baja). | Alta |
| Fecha de inicio | Fecha de inicio (AAAA-MM-DD). | 2025-05-01 |
| Fecha de vencimiento | Fecha de vencimiento (AAAA-MM-DD). | 2025-05-01 |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“id”:”task-xyz789″,”name”:”Updated Task”} | {var} |

## Subir archivo adjunto a una tarea

Sube un archivo adjunto a una tarea en ClickUp.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de ClickUp necesaria para subir archivos. | credential |
| Workspace | Espacio de trabajo desde el cual se subirá el archivo. | SaturnStudio |
| Space | Espacio desde el cual se subirá el archivo. | SaturnStudio |
| Folder | Carpeta desde la cual se subirá el archivo. | SaturnStudio |
| Lists | Lista desde la cual se subirá el archivo. | SaturnStudio |
| Tarea | Tarea a la que se adjuntará el archivo. | SaturnStudio |
| Archivo local | Archivo que se subirá como adjunto. | seleccionar archivo |
| Asignar resultado a variable | Variable donde se almacenará el resultado. **Ejemplo:** {“id”:”a1b2c3d4″,”name”:”Sample Document.pdf”} | {var} |

# Saturn Studio – Credencial Click Up



## ¿Qué es este módulo?

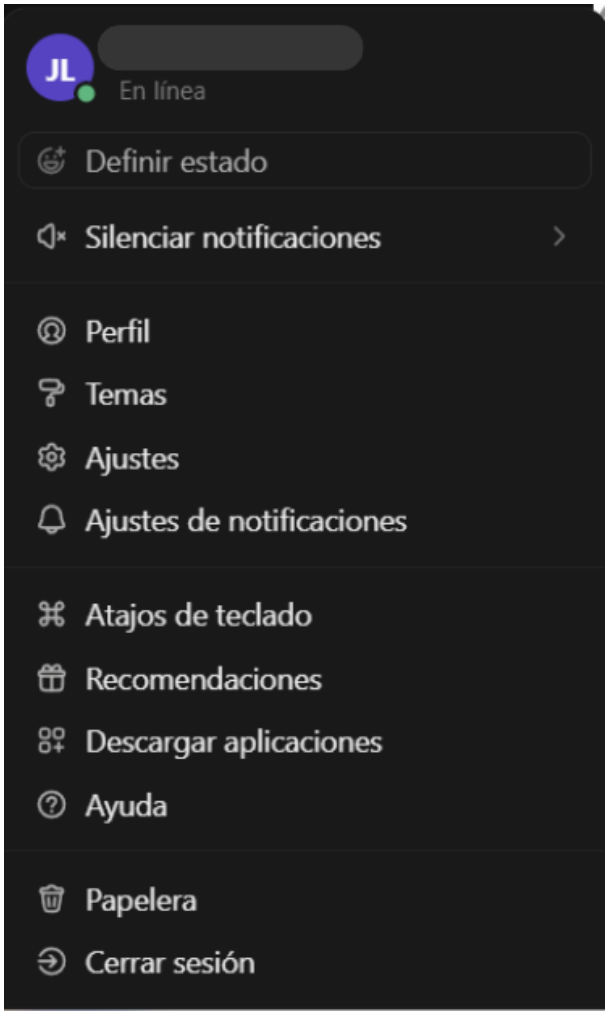
El módulo **ClickUp** permite conectar Saturn Studio con tu cuenta de ClickUp para automatizar tareas, proyectos y flujos de trabajo dentro de tus espacios de trabajo.
Para usar los comandos disponibles, primero deberás crear una **credencial OAuth 2.0** en la plataforma de desarrolladores de ClickUp.

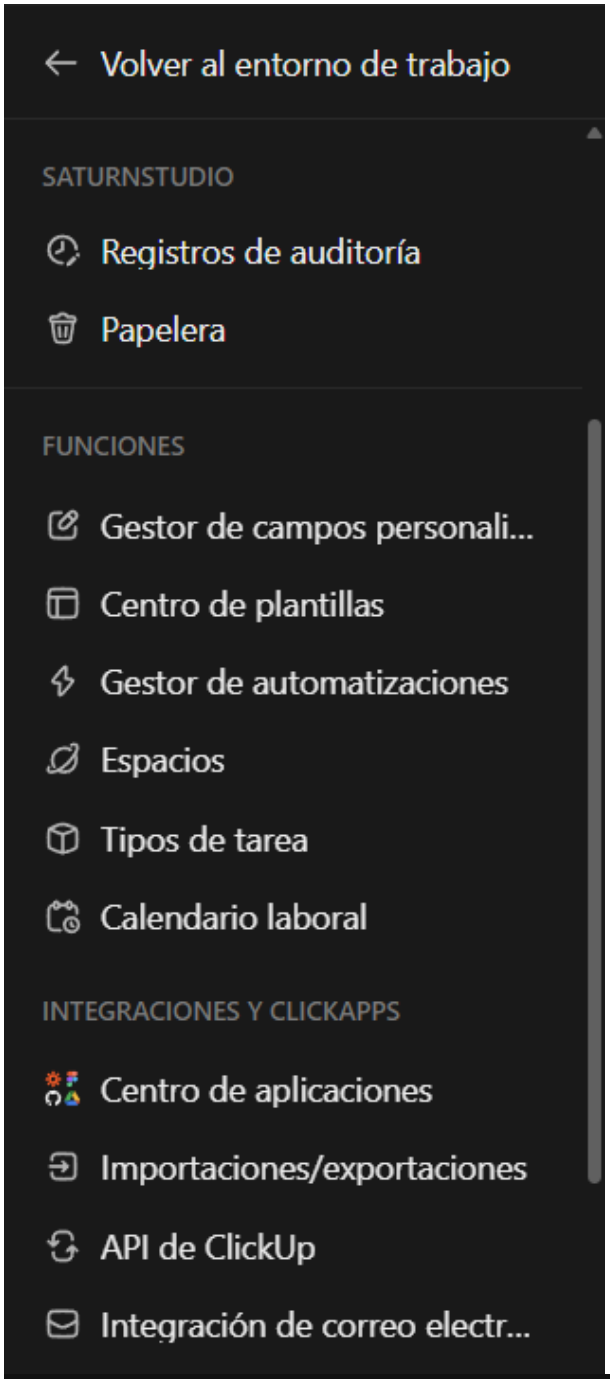
## Cómo generar las credenciales de ClickUp

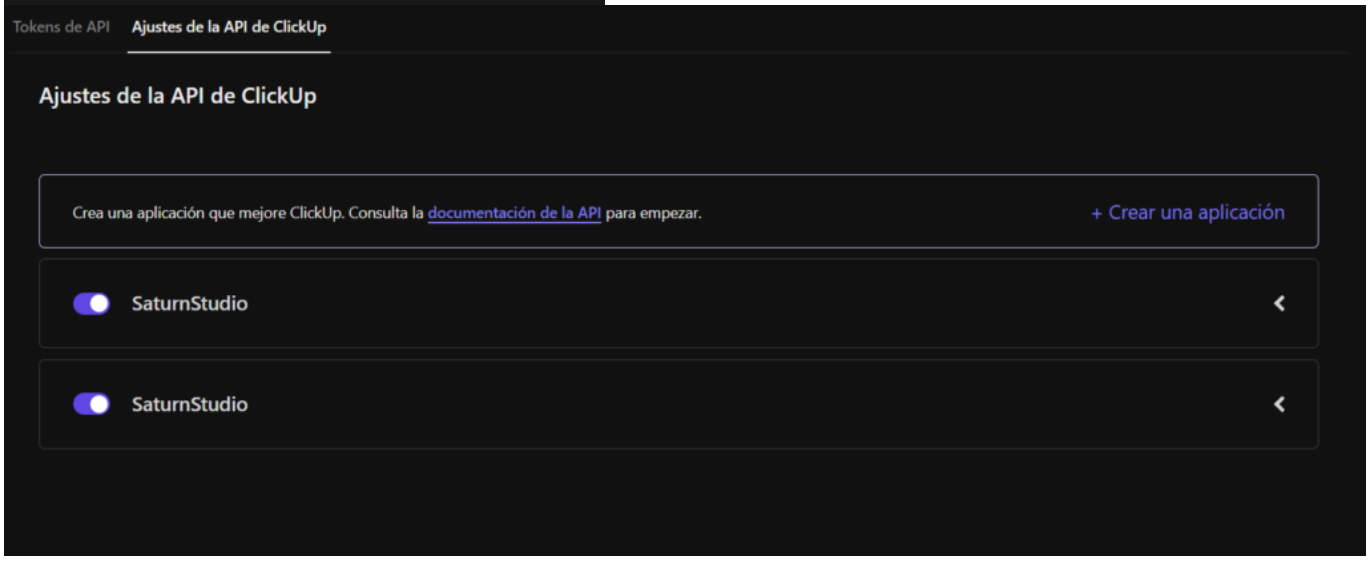
### 1) Ingresar al portal de desarrolladores

Accede a:
 https://developer.clickup.com

Para obtener las credenciales de ClickUp, debes acceder a tu perfil y seleccionar la opción Ajustes. Luego, en el menú lateral, ingresa en API de ClickUp, dirígete a Ajustes de la API y finalmente selecciona Crear una aplicación.


JL
En línea
Definir estado
Silenciar notificaciones
Perfil
Temas
Ajustes
Ajustes de notificaciones
Atajos de teclado
Recomendaciones
Descargar aplicaciones
Ayuda
Papelera
Cerrar sesión


Volver al entorno de trabajo
SATURNSTUDIO
Registros de auditoría
Papelera
FUNCIONES
Gestor de campos personali...
Centro de plantillas
Gestor de automatizaciones
Espacios
Tipos de tarea
Calendario laboral
INTEGRACIONES Y CLICKAPPS
Centro de aplicaciones
Importaciones/exportaciones
API de ClickUp
Integración de correo electr...


Tokens de API
Ajustes de la API de ClickUp
Ajustes de la API de ClickUp
Crea una aplicación que mejore ClickUp. Consulta la documentación de la API para empezar.
+ Crear una aplicación
SaturnStudio
SaturnStudio

## 2) Crear una nueva aplicación

Completa los campos:

| Campo | Descripción | Ejemplo |
|---|---|---|
| App Name | Nombre con el que identificarás tu integración. | Saturn Studio |
| Redirect URL | URL de redirección autorizada. | https://studio.rocketbot.com |
| Description | (Opcional) Breve descripción de la integración. | Integración con Saturn Studio para automatizar tareas |

Haz clic en **Create App**.

## 3) Obtener tus credenciales

Una vez creada la app, ClickUp mostrará los siguientes valores:

| Dato | Descripción | Ejemplo |
|---|---|---|
| **Client ID** | Identificador único de tu aplicación. | a1b2c3d4e5f6g7h8 |
| **Client Secret** | Clave secreta asociada a tu aplicación. | z9y8x7w6v5u4t3s2 |

## 4) Conectar con Saturn Studio

1. Abre el módulo **ClickUp** en Saturn Studio.
2. Haz clic en **New** para crear una nueva credencial.
3. Asigna un nombre (por ejemplo, *Mi ClickUp Credential*).
4. Copia y pega el **Client ID** y **Client Secret** obtenidos en los campos correspondientes.
5. Presiona **Connect to ClickUp**.
6. Autoriza el acceso cuando aparezca la ventana de autenticación.

 ¡Listo! Tu credencial queda registrada y podrás usar todos los comandos del módulo ClickUp.

## Requisitos

- **Tener una cuenta activa de ClickUp.**
- Acceso a ClickUp Developer Platform.
- Contar con permisos de administrador o creador de aplicaciones.

---

# Saturn Studio — Question and Answer Agent (v2)

### Módulo de Inteligencia Artificial — Saturn Studio

Este componente pertenece al módulo **IA** y permite crear **agentes de preguntas y respuestas** capaces de interactuar con el usuario, mantener historial de conversación y **utilizar fuentes externas como Gmail, Google Calendar u otros módulos conectados** para enriquecer sus respuestas mediante herramientas integradas.

### ¿Qué es este módulo?

El **Question and Answer Agent (v2)** es un agente autónomo diseñado para **responder preguntas, redactar contenido y brindar información dinámica** utilizando un modelo de lenguaje configurable (por ejemplo, OpenAI o Hugging Face).

Además, este agente puede **consultar datos de diferentes módulos** dentro de Saturn Studio, como correos, eventos, documentos o registros, combinando esa información con la capacidad de razonamiento del modelo de IA para generar respuestas contextualizadas.

De esta forma, actúa como un **agente inteligente que busca información en las fuentes conectadas y responde con lenguaje natural**.
También puede conservar el contexto entre mensajes si se habilita la opción **Use History**.

### Comandos disponibles

#### Agente de Preguntas y Respuestas (v2)

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| **AI Credential** | Credencial del proveedor de IA. Debe haberse creado previamente en el módulo correspondiente (por ejemplo, OpenAI). [Guía: Saturn Studio — Credencial OpenAI] | OPENAI — test |
| **Model** | Modelo de lenguaje que el agente utilizará para generar respuestas. | gpt-4o |
| **Input Prompt** | Texto o instrucción que el agente recibirá como entrada. Se pueden usar variables como {var_name}. | “Genera un resumen del texto: {documento}” |
| **Use History** | (Opcional) Permite que el agente recuerde mensajes anteriores para mantener contexto. | Activado |
| **ID to identify the history** | Identificador único para almacenar el historial (por ejemplo: chat_id, canal, usuario). | user_001 |

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| **History Size** | Cantidad de mensajes que conservará en la memoria de conversación. | 5 |
| **Data Store** | Almacenamiento de datos persistente opcional para guardar respuestas o contexto. | Test Data Store |
| **Tools (Fuentes de datos)** | Define los **módulos externos** que el agente puede utilizar como fuentes de información para responder (por ejemplo: leer correos desde Gmail, buscar eventos en Google Calendar o consultar datos en Notion). Estas herramientas se conectan al agente y le permiten **recuperar información real antes de generar la respuesta con IA**. | Gmail, Google Calendar |
| **Assign result to variable** | Variable donde se almacenará el resultado de la respuesta generada por el agente. | respuesta |

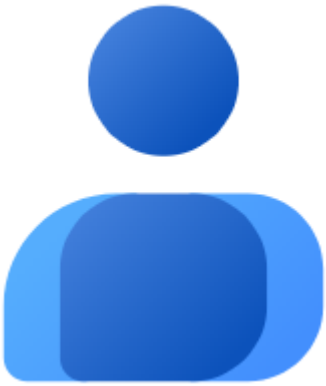
## □ Ejemplo de uso

1. Dirígete al módulo de la IA que desees utilizar (por ejemplo, **OpenAI**) y crea una Credencial de IA siguiendo la guía: *Saturn Studio – Credencial OpenAI*.
2. En el flujo, agrega el comando **Question and Answer Agent (v2)** y conéctalo con los módulos que funcionarán como fuentes (por ejemplo, *Read all email* de Gmail o *Search Events* de Google Calendar).
3. Configura los campos:
    - **AI Credential:** OPENAI – test
    - **Model:** gpt-4o
    - **Input Prompt:** "¿Qué correos recibí hoy?"
    - **Tools:** Gmail
    - **Assign result to variable:** respuesta_agente
4. Ejecuta el flujo.
   El agente leerá la información desde los módulos conectados, procesará el contenido con la IA y devolverá una **respuesta final generada automáticamente**.

## □ Requisitos

- Tener configurada una **credencial válida** del proveedor de IA.
- Disponer de un **modelo compatible** en la cuenta.
- Conectar los **módulos que servirán como fuentes** de información (por ejemplo, Gmail, Calendar, Drive, Notion, etc.).
- (Opcional) Configurar un **Data Store** si se desea mantener memoria de conversación persistente.

# Saturn Studio – Credencial Google Contacts



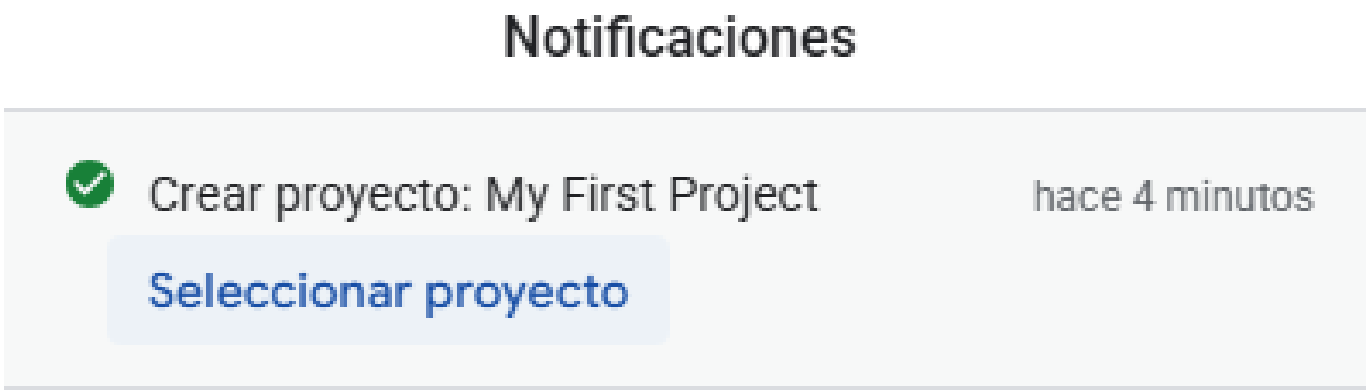
Este documento explica cómo integrar y autenticar el módulo **Google Contacts** dentro de **Saturn Studio**, permitiendo acceder, leer y administrar contactos de Google de forma segura.

## Configuración (OAuth 2.0 con Google Cloud)

Este método brinda mayor control sobre los permisos y seguridad, ideal para entornos de trabajo colaborativos o producción.

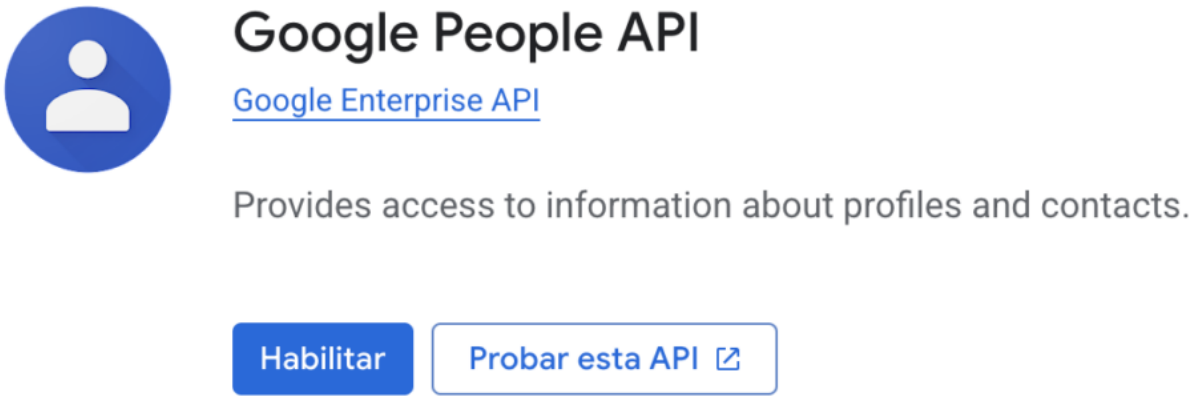
### Crear / elegir proyecto y habilitar APIs

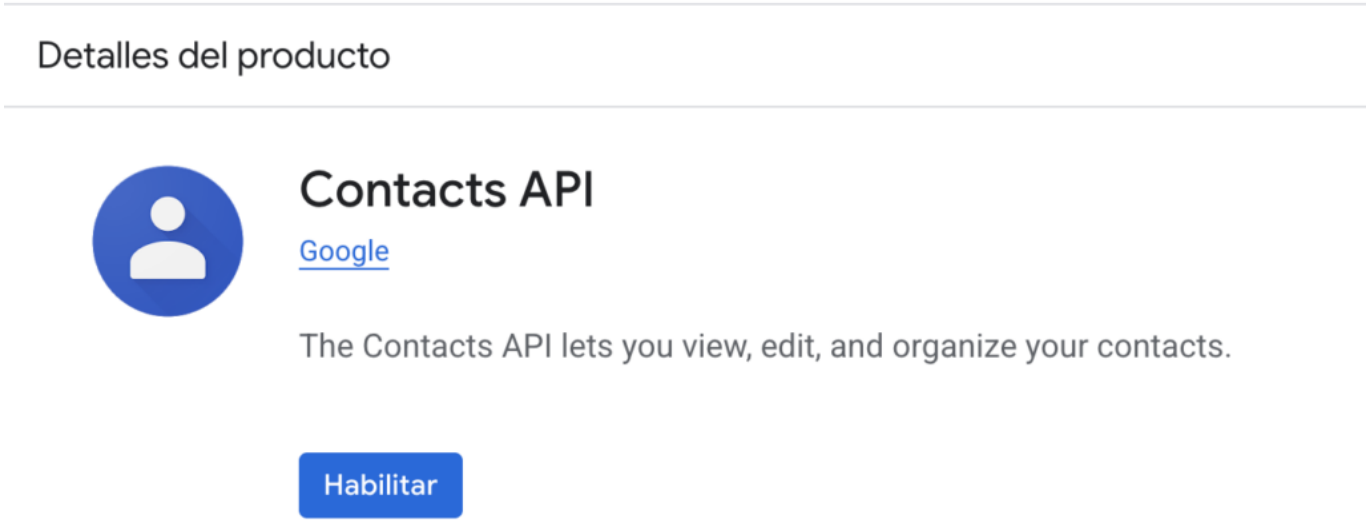
- Ingresa a Google Cloud Console.
- Inicia sesión con tu cuenta de Google.
- Crea un nuevo proyecto o selecciona uno existente desde la barra superior.




- Desde la barra de búsqueda, busca y habilita las siguientes APIs:
  - **People API** (requerida para gestionar los contactos de Google)
  - **Google Drive API** *(opcional, solo si deseas vincular datos o archivos relacionados)*

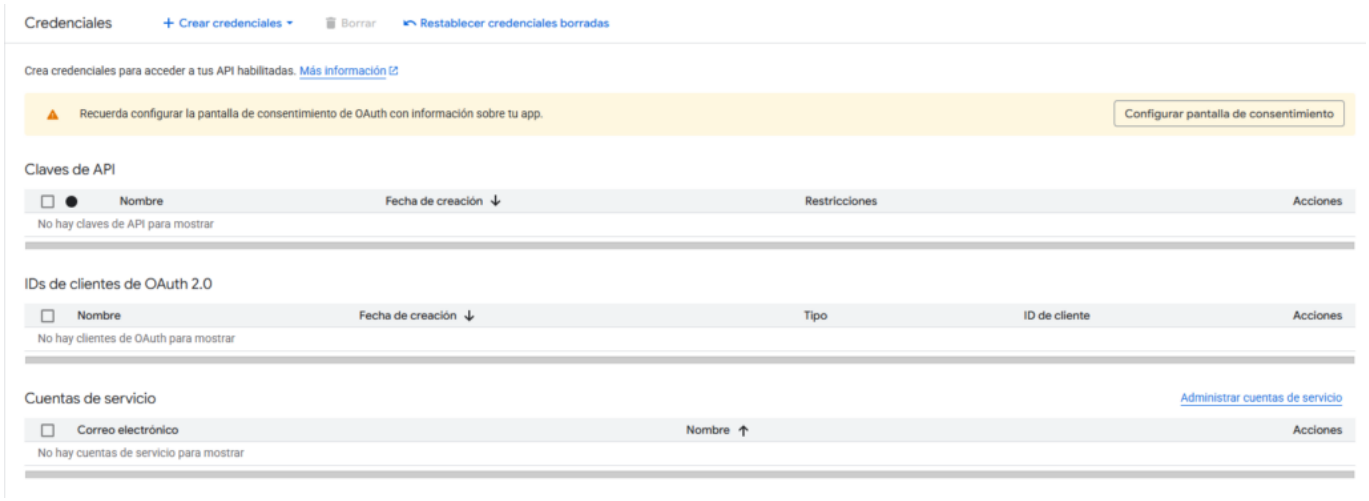
- Verifica que las APIs queden con estado **Habilitado**.
  Si alguna no lo está, la autenticación podría completarse, pero las llamadas devolverán errores del tipo *access_not_configured*.






## Configurar la Pantalla de consentimiento OAuth

- En el menú lateral, entra a:
  **APIs y servicios → Pantalla de consentimiento OAuth → Configurar pantalla de consentimiento.**



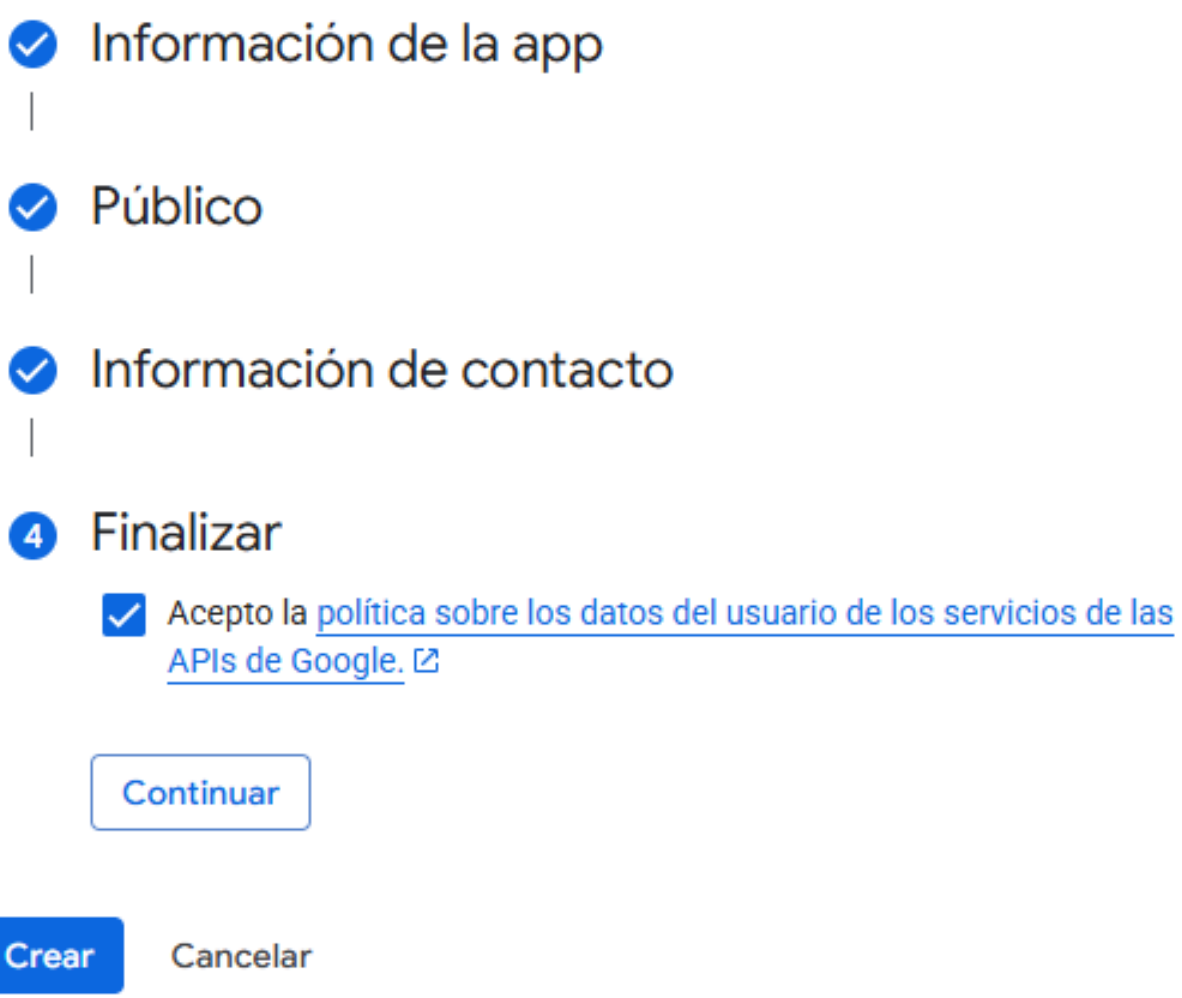

- Completa los campos solicitados:
  - **Nombre de la aplicación**
  - **Correo de soporte**
  - *(Opcional)* Logo, dominio, enlaces de privacidad y términos.

- Elige el tipo de usuario según tu caso:

| Tipo de usuario | Descripción | Recomendación |
|---|---|---|
| **Interno** | Solo usuarios de tu organización (requiere dominio Google Workspace). | Úsalo si tu equipo pertenece al mismo dominio. |
| **Externo** | Permite acceso a cualquier cuenta de Google (modo de prueba). | Ideal para uso personal o con cuentas @gmail.com. |

 **Nota:** en modo *Externo (prueba)* deberás agregar manualmente a los usuarios de prueba, incluido tu propio correo.
Solo esos usuarios podrán autorizar la app mientras esté en prueba.

Guarda los cambios al finalizar.

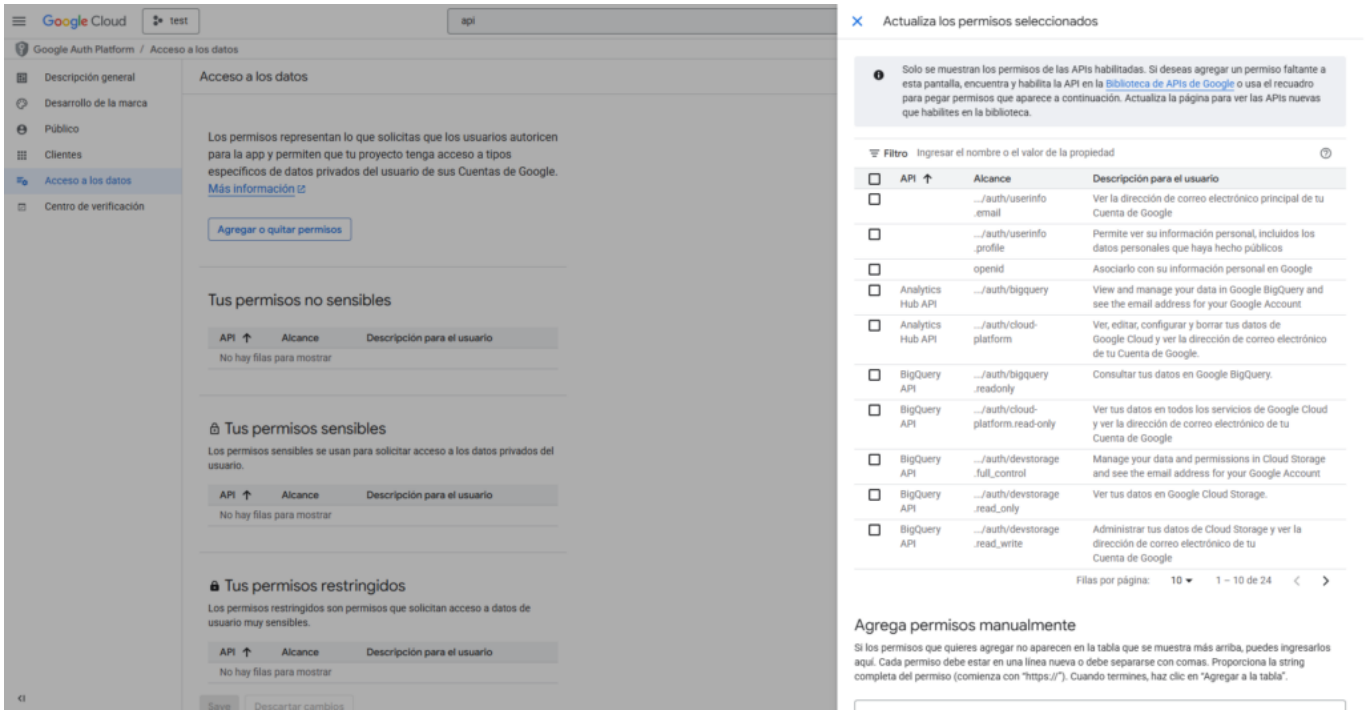



## Agregar scopes (permisos) a la app

1. Desde la misma sección de **Pantalla de consentimiento OAuth**, ve a **Acceso a los datos → Agregar o quitar permisos.**
2. Añade los siguientes scopes recomendados:

| Scope | Descripción |
| --- | --- |
| https://www.googleapis.com/auth/contacts | Permite lectura y escritura de contactos. |
| https://www.googleapis.com/auth/contacts.readonly | Permite solo lectura de contactos. |
| https://www.googleapis.com/auth/userinfo.email | Accede al correo del usuario autenticado. |
| https://www.googleapis.com/auth/userinfo.profile | Accede al perfil básico del usuario. |
| openid | Requerido para la autenticación mediante OAuth 2.0. |

Actualiza y guarda los cambios.




## Crear el Client ID y Client Secret

- Ve a **APIs y servicios → Credenciales → Crear credencial → ID de cliente de OAuth.**
- Completa los datos de la siguiente manera:
  - **Tipo de aplicación:** Aplicación web
  - **Nombre:** *Saturn Studio*
  - **URIs de redirección:** https://studio.rocketbot.com

# Crear ID de cliente de OAuth

Un ID de cliente se usa con el fin de identificar una sola app para los servidores de OAuth de Google. Si la app se ejecuta en varias plataformas, cada una necesitará su propio ID de cliente. Consulta Configura OAuth 2.0 para obtener más información. Obtén más información sobre los tipos de clientes de OAuth.

Tipo de aplicación *
Aplicación web

Nombre *
Saturn Studio

El nombre de tu cliente de OAuth 2.0. Este nombre solo se usa para identificar al cliente en la consola y no se mostrará a los usuarios finales.

Los dominios de los URI que agregues a continuación se incorporarán automáticamente a tu pantalla de consentimiento de OAuth como dominios autorizados.

## Orígenes autorizados de JavaScript

Para usar con solicitudes de un navegador

+ Agregar URI

## URIs de redireccionamiento autorizados

Para usar con solicitudes de un servidor web

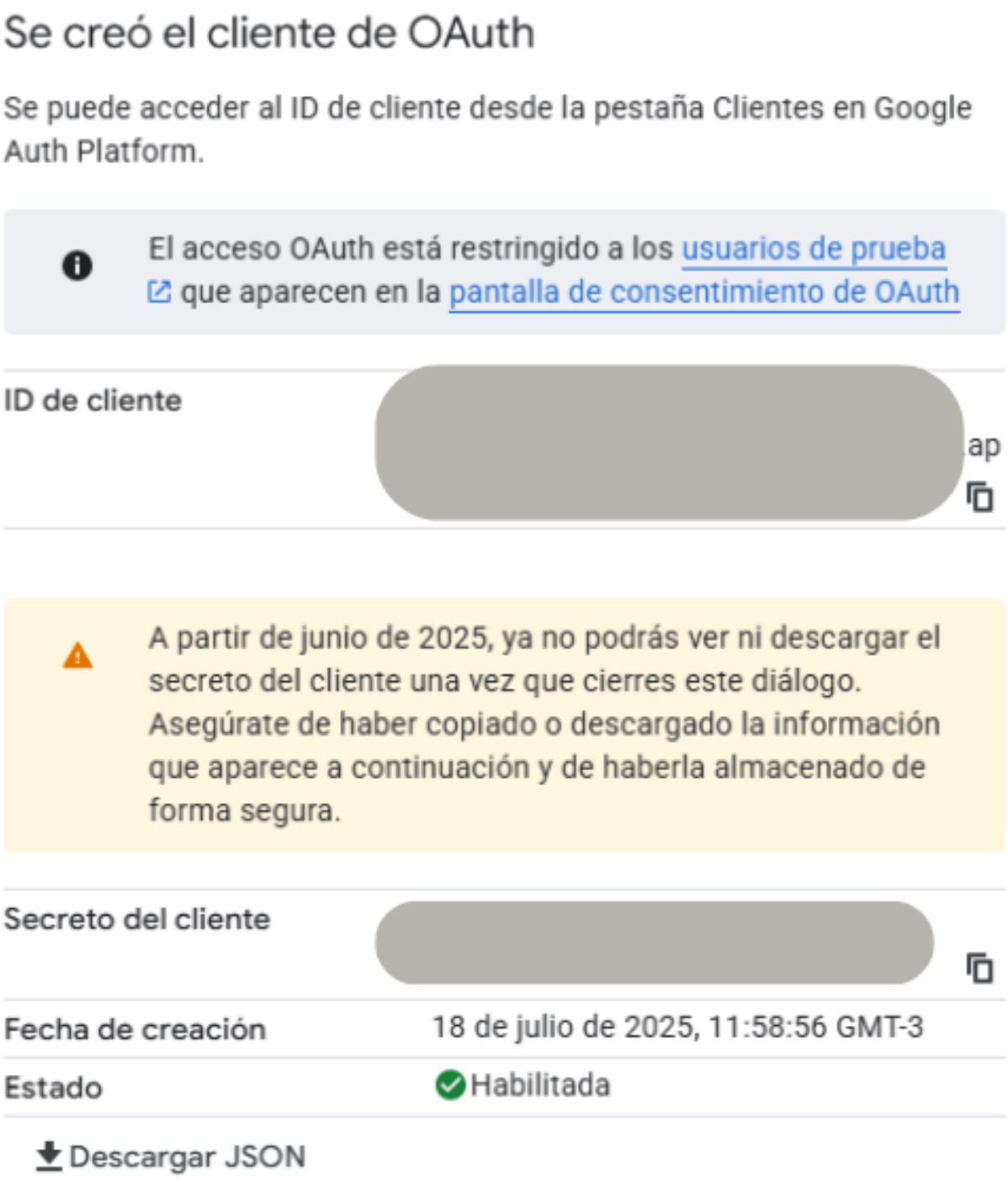
URI 1 *
https://studio.rocketbot.com

+ Agregar URI

Nota: La configuración puede tardar entre 5 minutos y algunas horas en aplicarse

Crear Cancelar

- Al guardar, Google generará:
  - **Client ID**
  - **Client Secret**
    Cópialos y resérvalos.




## Cargar credenciales en Saturn Studio

1. Regresa al módulo **Google Contacts** en Saturn Studio.
2. Selecciona el modo **Avanzado (OAuth)**.
3. Pega el **Client ID** y **Client Secret** obtenidos en los campos correspondientes.
4. Guarda la credencial y realiza la conexión.

 **Listo:** tu credencial avanzada quedará configurada correctamente para el uso completo del módulo Google Contacts.

### **Notas importantes**

- **Usuarios externos (modo de prueba):** agrega los correos de los usuarios en la lista de “Usuarios de prueba”. De lo contrario, no podrán autorizar la app.
- **Interno:** solo disponible con cuentas de dominio empresarial (Google Workspace).
- **Scopes y APIs:** los *scopes* otorgan permisos, pero las *APIs habilitadas* son las que permiten ejecutar las acciones.
- **URIs de redirección:** deben coincidir exactamente con la configuración de Google Cloud.
- **Verificación:** si deseas publicar la app (modo producción) y usas *scopes sensibles*, Google podría solicitar verificación del dominio, logo y políticas de privacidad.

---

# Saturn Studio — Credencial Microsoft Teams (Microsoft Graph)



Este módulo conecta Saturn Studio con **Microsoft Graph** para operar en **Teams** (leer/enviar mensajes, crear chats, gestionar miembros, etc.) usando **OAuth 2.0**.

## 1) Iniciar sesión y registrar la aplicación

1. Entra a **Azure Portal → Microsoft Entra ID → Registros de aplicaciones** (o usa el acceso directo del portal a la lista de apps).
2. Haz clic en **Nuevo registro**.
3. **Nombre**: pon un nombre identificable (ej.: Saturn Studio — Teams).
4. **Tipos de cuentas compatibles** (elige uno):

| Opción (audiencia) | ¿Quién puede iniciar sesión? | Qué pones en Tenant ID en Saturn Studio |
|---|---|---|
| **Inquilino único** (Single-tenant) | Solo usuarios de **tu organización** | **GUID** de tu tenant (Directory (tenant) ID) |
| **Multiinquilino (solo organizaciones)** | Usuarios de **cualquier directorio** (sin cuentas personales) | Usa organizations o el **tenant GUID** que necesites |
| **Multiinquilino + cuentas personales** | Org. + **cuentas personales** (MSA) | Usa common (o un **tenant GUID** si quieres fijar uno) |

> Nota: muchas API de **Teams** no funcionan con **cuentas personales**. Para escenarios de empresa, usa **Inquilino único** o **Multiinquilino (organizational)**.

5. **URI de redirección**
   - **Tipo**: Web
   - **URL**: https://studio.rocketbot.com
     *(Debe coincidir exactamente con lo que muestre el formulario de la credencial en Saturn Studio.)*

6. Haz clic en **Registrar**.

## 2) Obtener identificadores y secreto

En la ficha de la aplicación copia:

- **Application (client) ID** → será tu **Client ID**.
- **Directory (tenant) ID** → será tu **Tenant ID** (si usas single-tenant).

**Guía recomendada para esta sección: Microsoft Learn**

Ve a **Certificados y secretos → Nuevo secreto de cliente**.

- Define la **vigencia** (elige la mayor posible que cumpla tu política).
- **Copia el valor** del secreto inmediatamente (no el ID), se oculta luego.

## 3) Agregar permisos (Microsoft Graph → Delegados)

Ruta: **Permisos de API → Agregar un permiso → Microsoft Graph → Permisos delegados**.

### Esenciales (OIDC)

- openid, offline_access, email.

### Teams (elige solo lo necesario, principio de menor privilegio)

- **Canales**:
  - Channel.ReadBasic.All (ver nombres y descripciones de canales).
  - **Guía recomendada para esta sección:** Microsoft Learn+1
  - **Mensajes de canal**: ChannelMessage.Read.All (leer) y ChannelMessage.Send (enviar).
  - **Guía recomendada para esta sección:** Microsoft Learn+1

- **Chats 1:1 / grupales**:
  - Chat.Read, Chat.ReadWrite, Chat.Create.
  - **Guía recomendada para esta sección:** Graph Permissions+1
  - (Opcional avanzado) Chat.ManageDeletion.All para borrar/restaurar chats — requiere rol admin.
  - **Guía recomendada para esta sección:** Graph Permissions+1

- **Equipos/miembros**:
  - Team.ReadBasic.All, TeamMember.Read.All, TeamMember.ReadWrite.All (agregar/quitar miembros).
  - **Guía recomendada para esta sección:** Microsoft Learn+1

> Muchas operaciones de Teams **no admiten cuentas personales**; para "Delegated (personal Microsoft account)" suele mostrarse **Not supported** en la tabla de permisos.
>
> **Guía recomendada para esta sección:** Microsoft Learn+1

Cuando termines, haz clic en **Conceder consentimiento de administrador** para tu organización (evita que cada usuario deba aprobar permisos).

**Guía recomendada para esta sección:** Microsoft Learn+1

## 4) Cargar la credencial en Saturn Studio (módulo Microsoft Teams)

Completa los campos:

| Campo | Descripción | Ejemplo |
|---|---|---|
| **Client ID** | Application (client) ID de tu app registrada | xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx |
| **Tenant ID** | common, organizations o el **GUID** del tenant (según tu elección en “Tipos de cuentas”) | common |
| **Client Secret** | Valor del secreto de cliente generado | ****** |

Guarda y realiza el **Sign in / Consent** desde el módulo.

---

# Saturn Studio — Credencial iLovePDF

**iLovePDF** permite convertir, comprimir, unir, dividir y editar PDFs. Este módulo conecta Saturn Studio con la **iLovePDF API** para automatizar esas tareas dentro de tus flujos.

## ¿Qué es este módulo?

El módulo se autentica con **Public Key** y **Secret Key** de iLovePDF. Con esas credenciales, Saturn Studio gestiona el flujo estándar de la API (Start → Upload → Process → Download).

La plataforma ofrece una cuenta gratuita con **250 archivos/mes** para

pruebas y desarrollo.

## Cómo generar las credenciales de iLovePDF

1. **Crear cuenta / Iniciar sesión**
   Regístrate o inicia sesión en el portal de desarrolladores para acceder a tu consola.
2. **Acceder al panel (Developer Console)**
   Al registrarte tendrás un **Default Project** ya creado con su **par de claves** (Public Key + Secret Key). Desde la consola puedes **crear nuevos proyectos**; cada proyecto genera su propio par de claves.
3. **Obtener las claves**
   En **API Keys** del proyecto copia tu **Public Key** y **Secret Key**. (La guía "Quick Start" indica obtenerlas desde la sección *API Keys* de la consola).

## Configurar la credencial en Saturn Studio

1. Abre el módulo **iLovePDF** en Saturn Studio.
2. Haz clic en **New** para crear una nueva credencial.
3. Completa los campos:

| Campo | Descripción | Ejemplo |
|---|---|---|
| Credential name | Nombre identificador de la credencial | ilovepdf-prod |
| Public Key | Clave pública del **proyecto** iLovePDF | project_public_key |
| Secret Key | Clave secreta asociada al proyecto | project_secret_key |

4. Guarda la credencial. A partir de aquí podrás usar las acciones de iLovePDF (convertir, unir, dividir, comprimir, etc.) dentro de tus flujos.

## Notas útiles

- Si necesitas más volumen que el plan gratuito, puedes **aumentar el límite mensual** desde *Pricing*.
- La API dispone de librerías oficiales (PHP, .NET, Ruby, Node.js) que ya gestionan la firma JWT con **Public/Secret Key**.
- Guia oficial de [iLoveAPI – A PDF REST API for developers]

# Saturn Studio – Hugging Face



## ¿Qué es este módulo?

Hugging Face es una plataforma de IA de código abierto que ofrece modelos de lenguaje, visión por computadora y más. Facilita el uso e implementación de modelos de *machine learning* de vanguardia dentro de Saturn Studio

**¿Cómo obtener las credenciales? Haz clic aquí.**

## Comandos disponibles

- Chat completion
- Summarize Text
- Translate Text
- Generate Embeddings
- Generate an Image

## Descripción de los comandos

### 1) Chat completion

Completa un chat usando un modelo de Inferencia de Hugging Face.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credential | Credencial de Hugging Face necesaria para completar el chat | Mis credenciales de Hugging Face |
| Model | ID del modelo en Hugging Face para completar el chat | Model |
| Provider | Proveedor para ejecutar la inferencia | cerebras |
| Text | Texto del mensaje a completar | Text |
| Max tokens | Máximo de *tokens* a generar | 300 |
| Temperature | Temperatura de muestreo. Valores altos generan respuestas más creativas; valores bajos, más conservadoras. Rango 0–2, por defecto 1 | 1 |
| Assign result to variable | Variable donde se almacenará el resultado. Ejemplo de salida: {“response”:”Hi there! How can I assist you today?”,”model_used”:”Qwen/Qwen3-32B”,”input_messages”:[{“role”:”user”,”content”:”Hello chat”}],”usage”:{“prompt_tokens”:10,”completion_tokens”:102,”total_tokens”:112}} | {var} |

## 2) Summarize Text

Resume un texto usando un modelo de Inferencia de Hugging Face.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credential | Credencial de Hugging Face necesaria para resumir el texto | Mis credenciales de Hugging Face |
| Model | ID del modelo en Hugging Face para resumir | Model |
| Provider | Proveedor para ejecutar la inferencia | cerebras |
| Text | Texto a resumir | Text |
| Min Length | Longitud mínima del resumen | 300 |
| Max Length | Longitud máxima del resumen | 500 |
| Assign result to variable | Variable donde se almacenará el resultado. Ejemplo de salida: {“response”:”Keep it simple and open-ended…”,”model_used”:”facebook/bart-large-cnn”,”input_text”:”That’s a greeting…”} | {var} |

## 3) Translate Text

Traduce un texto usando un modelo de Inferencia de Hugging Face.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credential | Credencial de Hugging Face necesaria para traducir el texto | Mis credenciales de Hugging Face |
| Model | ID del modelo en Hugging Face para traducir | Model |
| Provider | Proveedor para ejecutar la inferencia | cerebras |
| Text | Texto a traducir | Text |
| Language | Idioma destino de la traducción (por ejemplo: ‘en’ inglés, ‘es’ español, ‘fr’ francés) | en |
| Assign result to variable | Variable donde se almacenará el resultado. Ejemplo de salida: {“response”:”Usos principales NLLB-200 es un modelo de traducción automática…”,”model_used”:”Helsinki-NLP/opus-mt-en-es”,”input_text”:”Primary intended uses NLLB-200 is a machine translation model…”} | {var} |

## 4) Generate Embeddings

Genera *embeddings* para un texto usando un modelo de Inferencia de Hugging Face.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credential | Credencial de Hugging Face necesaria para generar *embeddings* | Mis credenciales de Hugging Face |
| Model | ID del modelo en Hugging Face para generar *embeddings* | Model |
| Text | Texto a procesar | Text |
| Provider | Proveedor para ejecutar la inferencia | cerebras |
| Truncate | Si está habilitado, el texto se truncará al máximo permitido por el modelo. Útil para textos largos | (habilitado) |
| Assign result to variable | Variable donde se almacenará el resultado. Ejemplo de salida: {“response”:[[0.02355957,-0.01794434,0.00296021,0.01513672,…]],”model_used”:”Qwen/Qwen3-Embedding-8B”,”input_text”:”Primary intended uses NLLB-200…”,”dimensions”:4096} | {var} |

## 5) Generate an Image

Genera una imagen a partir de un *prompt* usando un modelo de Inferencia de Hugging Face.

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credential | Credencial de Hugging Face necesaria para generar la imagen | Mis credenciales de Hugging Face |
| Model | ID del modelo en Hugging Face para generar la imagen | Model |
| Prompt | *Prompt* desde el que se generará la imagen | Prompt |
| Provider | Proveedor para generar la imagen | cerebras |
| Assign result to variable | Variable donde se almacenará el resultado. Ejemplo de salida: [{"filename":"saturn_hugging-face_1747255067961.png","id":"2500b30f…","size":3163231,"provider":"hugging-face"}] | {var} |

## Requisitos

- Contar con una **credencial de Hugging Face** válida (User Access Token, formato hf_…).
- ID de modelo válido de Hugging Face y, cuando corresponda, proveedor soportado (por ejemplo, cerebras).

---

# Saturn Studio – Credencial Hugging Face



Este módulo permite integrar **Saturn Studio** con los modelos de inferencia de **Hugging Face**, habilitando tareas como resumen de texto, traducción, entre otros.

## ¿Cómo crear una credencial de Hugging Face en Saturn Studio?

1. Abre el módulo **Hugging Face** en Saturn Studio.
2. Haz clic en **New** para crear una nueva credencial.
3. Completa los campos:

| Campo | Descripción | Ejemplo |
|---|---|---|
| Credential name | Nombre para identificar la credencial. | huggingface-prod |

| Campo | Descripción | Ejemplo |
| --- | --- | --- |
| API Key | Pega tu **User Access Token** de Hugging Face (formato hf_…). | hf_xxxxx... |

4. Haz clic en **Check** para validar la API Key.
5. Presiona **Save credentials** para guardar.

## ¿Cómo obtener la API Key (User Access Token) de Hugging Face?

1. Inicia sesión en tu cuenta de Hugging Face.
2. Ve a **Settings → Access Tokens** y elige **New token**.
3. Selecciona el **rol** de tu token y ponle un nombre:
    - **read**: solo lectura (recomendado si solo harás descargas/uso de inferencia).
    - **write**: lectura + escritura (necesario si vas a subir/editar repos).
    - **fine-grained**: acceso específico a recursos; ideal para producción

4. Cópialo y guárdalo: el token comienza con hf_….
5. Pégalo en el campo **API Key** de la credencial en Saturn Studio y guarda.

> Nota: Los **User Access Tokens** pueden usarse como *bearer token* para llamadas de inferencia (Inference Providers / Inference API).

## Recomendaciones de seguridad

- **Un token por proyecto/ambiente** (p. ej., uno para local, otro para producción) para poder revocarlos sin afectar otros usos.
- **Evita filtrar el token** en repos públicos; si se filtra, **róta** el token desde *Settings → Access Tokens*.

Guia Oficial de Hugging Face

---

# Saturn Studio – Credencial HubSpot



HubSpot es una plataforma CRM que ofrece herramientas para marketing, ventas, atención al cliente y gestión de contenido.
Este módulo de **Saturn Studio** permite interactuar con la API de HubSpot para automatizar tareas como la creación y actualización de contactos, negocios y empresas.

## ¿Qué es este módulo?

Este módulo conecta Saturn Studio con HubSpot mediante autenticación **OAuth 2.0**.
Para utilizarlo, es necesario crear una aplicación en el portal de desarrolladores de HubSpot y obtener las credenciales correspondientes (**Client ID, Client Secret y Auth URL**).

## Cómo generar las credenciales de HubSpot

**Debes ser un superadministrador para instalar una aplicación en una cuenta de HubSpot.**

1. Acceder al portal de desarrolladores de HubSpot
    - Inicia sesión o regístrate en: https://developers.hubspot.com

2. Crear una nueva aplicación
    - En el panel, selecciona **Apps → Create App**.
    - Ingresa un nombre y una descripción para tu aplicación.

3. Configurar la autenticación OAuth
    - Ve a la pestaña **Auth**.
    - En **Redirect URLs**, agrega: https://studio.rocketbot.com.
    - Selecciona los **scopes (permisos)** necesarios para tu integración,

por ejemplo:
- crm.objects.contacts.read
- crm.objects.contacts.write
- crm.objects.companies.read
- crm.objects.companies.write
- *(Agrega más según las funciones que quieras usar)*

4. Registrar la aplicación
   - Guarda la configuración.
   - Copia el **Client ID** y **Client Secret** que se mostrarán en la pestaña **Auth**.

## Configuración de la credencial en Saturn Studio

1. Abre el módulo **HubSpot** en Saturn Studio.
2. Haz clic en **New** para crear una nueva credencial.
3. Completa los campos:

| Campo | Descripción |
|---|---|
| Credential name* | Nombre identificador para la credencial (ej: Hubspot_Prod). |
| Client ID* | Pega el Client ID obtenido en HubSpot. |
| Client Secret* | Pega el Client Secret generado en HubSpot. |
| Auth URL* | Usa la URL de autorización: https://app.hubspot.com/oauth/authorize |

* Campos obligatorios

4. Haz clic en **Connect to HubSpot** y sigue el flujo de autenticación.
5. Guarda la credencial con **Save credentials**

Guia oficial de HubSpot para más información.