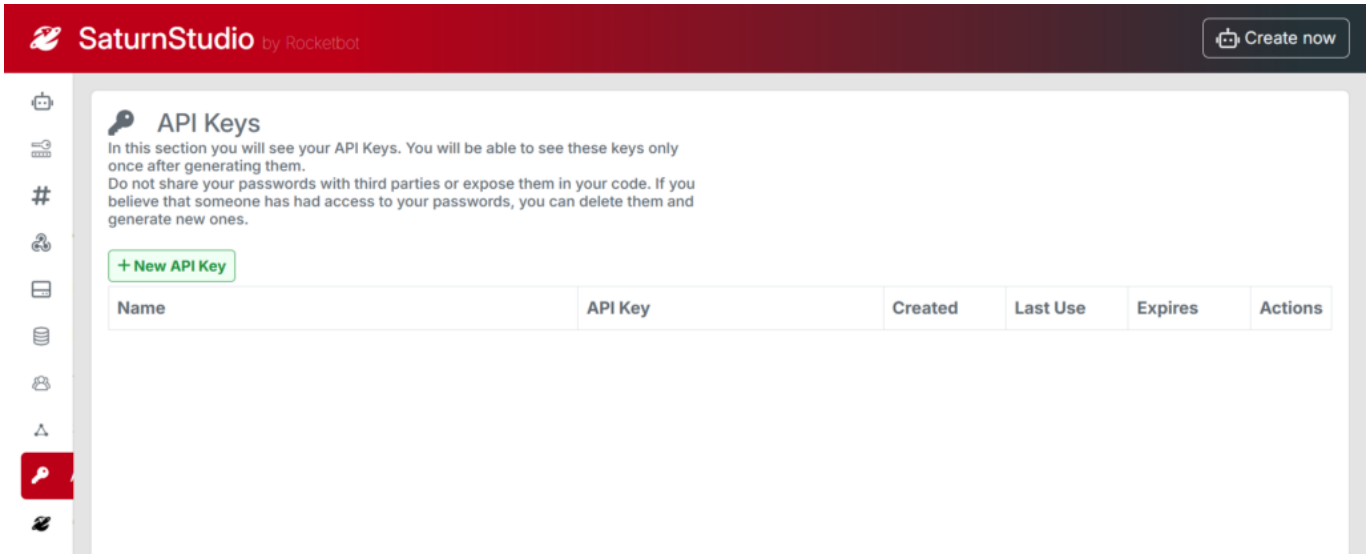
Saturn Studio – API Keys

Las **API Keys** actúan como llaves maestras que permiten que aplicaciones externas se comuniquen de forma segura con **Saturn Studio**. Si necesitas que un software de terceros active un robot o consulte datos de tu cuenta sin darle tu contraseña personal, aquí es donde generas ese acceso controlado.



¿Para qué sirven?

A diferencia de las credenciales de usuario, una API Key está diseñada para la integración entre máquinas. Son esenciales para:

- Conectar Saturn Studio con plataformas de automatización externas .
- Permitir que desarrolladores externos interactúen con tu infraestructura de bots mediante la API de la plataforma.
- Mantener la seguridad: puedes revocar una llave específica en cualquier momento sin afectar tu acceso principal.

Gestión de la Interfaz

Desde el menú lateral (icono de la llave), accederás al panel principal de **API Keys**. Aquí encontrarás un resumen de todas tus llaves activas y su estado de uso.

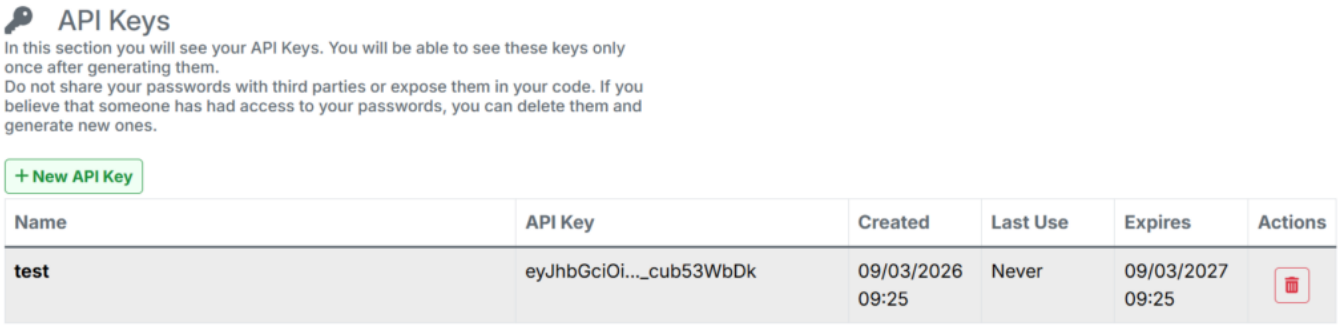


Tabla de Monitoreo

Columna	Función
Name	Nombre descriptivo que asignaste a la llave para identificar su propósito.
API Key	Versión enmascarada del token generado.
Created	Fecha y hora exacta de la creación de la llave.
Last Use	Indica la última vez que una aplicación externa utilizó este token. Ideal para detectar llaves obsoletas.
Expires	Fecha de caducidad. Una vez cumplida, la llave dejará de funcionar automáticamente.
Actions	Botón de papelera para eliminar (revocar) el acceso de forma inmediata.

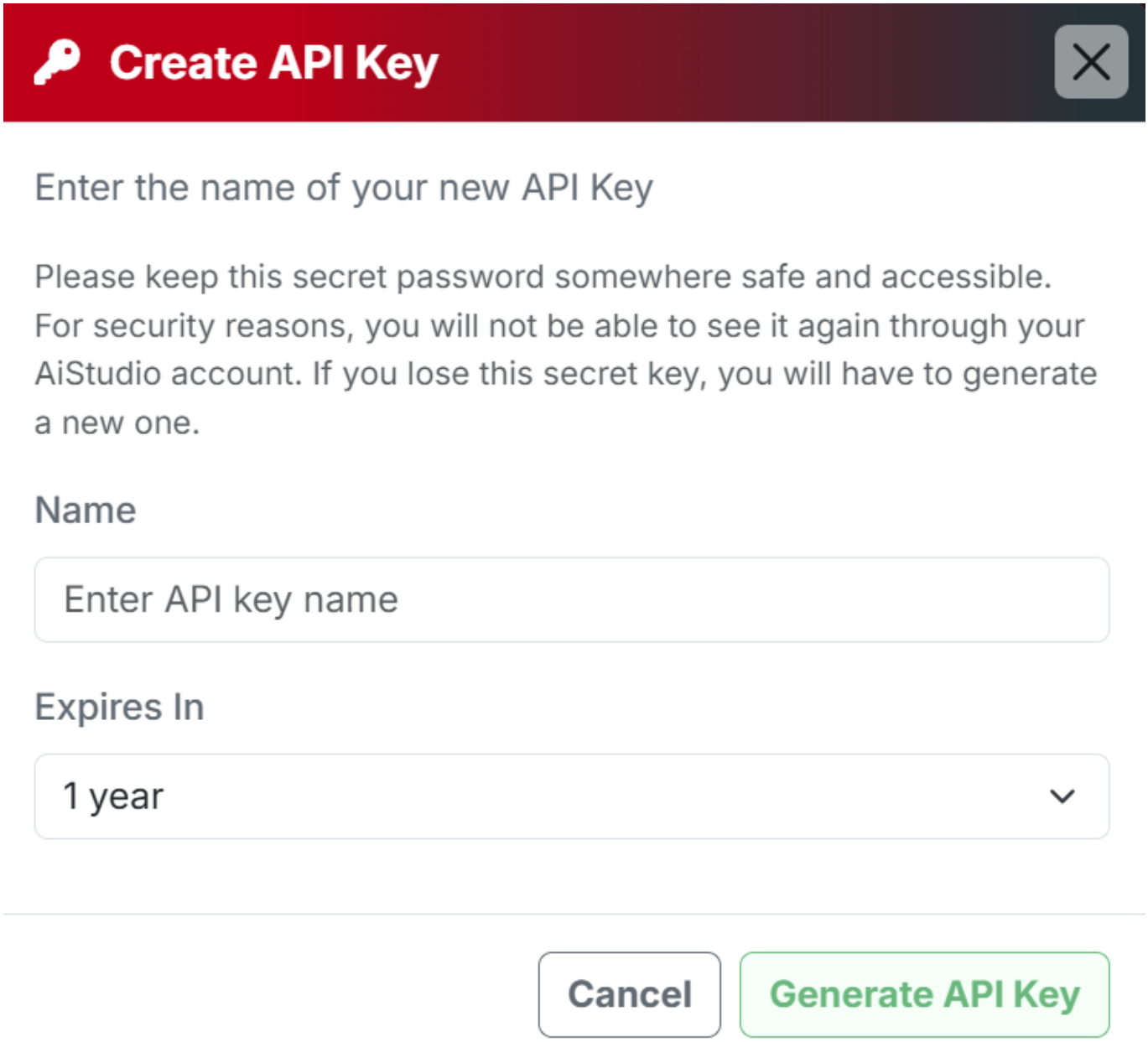
Proceso de Creación

Para generar un nuevo acceso, sigue estos pasos dentro del modal de configuración:

1. Definir Identidad y Caducidad

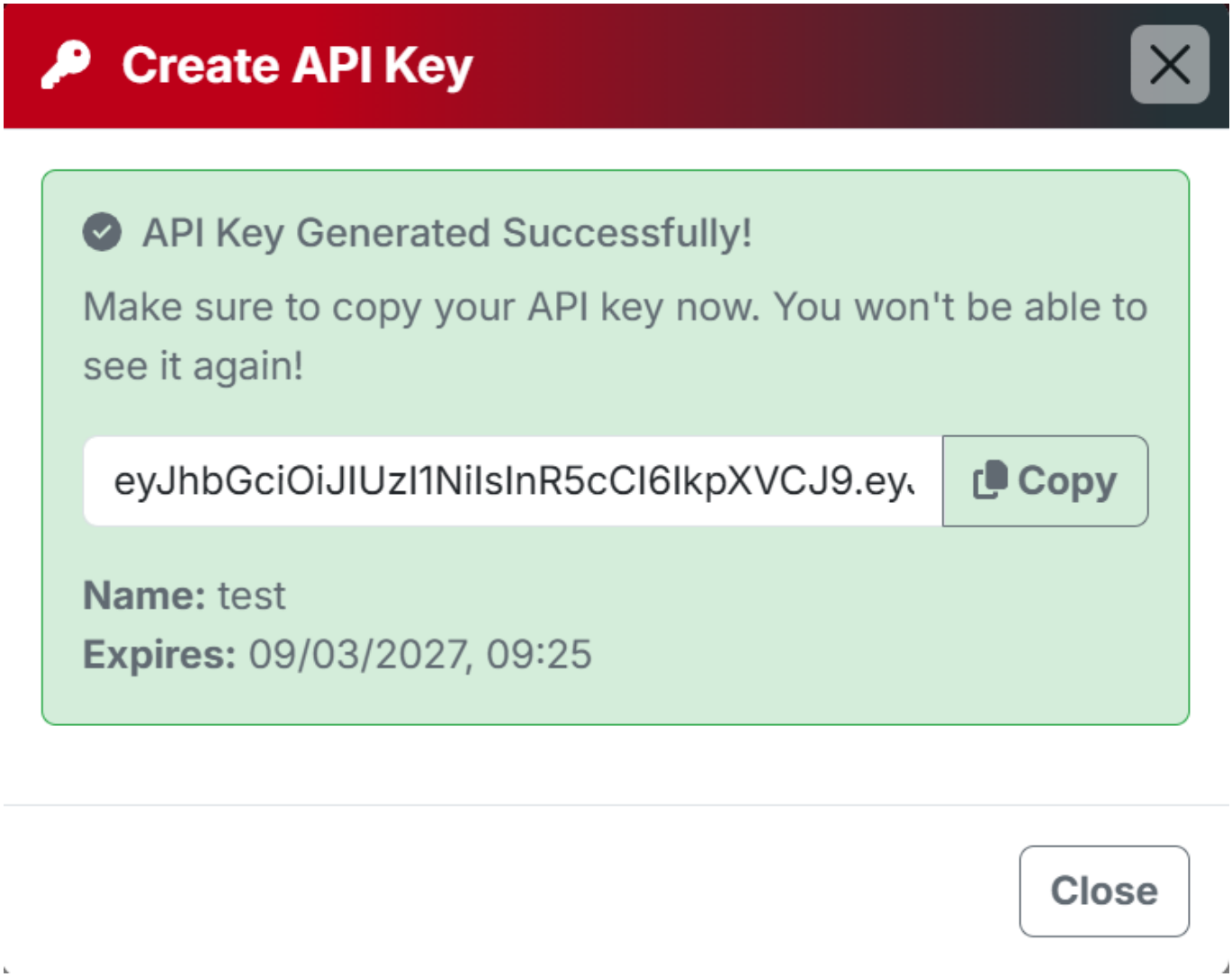
Haz clic en **+ New API Key**. Se abrirá una ventana donde deberás:

- **Name:** Ingresar un nombre único y descriptivo.
- **Expires In:** Seleccionar el tiempo de vida de la llave (desde 1 mes hasta 1 año). Por seguridad, se recomienda establecer fechas de expiración cortas si el uso es temporal.



2. Generación y Resguardo

Al hacer clic en **Generate API Key**, el sistema creará el token alfanumérico.



> **Aviso de Seguridad Crítico:** Esta es la única vez que verás la llave completa. Saturn Studio no almacena este código en texto plano por razones de seguridad. **Cópiala y guárdala en un gestor de contraseñas seguro.** Si la pierdes, deberás borrarla y generar una nueva.

Seguridad y Mantenimiento

- **No compartas tus llaves:** Trata una API Key con la misma precaución que una contraseña. Si sospechas que alguien ha tenido acceso a ella, elimínala de inmediato desde la columna **Actions**.
- **Rotación:** Es una buena práctica crear llaves nuevas y eliminar las antiguas cada cierto tiempo para mantener la integridad de tus integraciones.

Saturn Studio – Markdown PDF



Markdown PDF es una herramienta diseñada para convertir archivos Markdown en documentos PDF, preservando el formato y permitiendo la personalización del estilo y de la salida.

Descripción de los comandos

Markdown a PDF

Convierte un archivo Markdown en un documento PDF.

Parámetro	Descripción	Ejemplo
Markdown	Contenido Markdown a convertir.	# Sample Title\n\nThis is a sample paragraph in Markdown.
Ruta del archivo	Ruta al archivo Markdown local para convertir en PDF.	file.pdf
Nombre	Nombre del archivo PDF de salida.	output
Asignar resultado a variable	Variable donde se almacenará el resultado. Ejemplo: {"extractedValue":[{"name1":"Juan Pérez"},{"name2":"María González"},{"name3":"Carlos López"},{"name4":"Ana Martínez"}]}	{var}

Saturn Studio – Zoom



Zoom es una plataforma de colaboración que integra chat, videollamadas, almacenamiento de archivos y aplicaciones en un solo lugar para mejorar la productividad del equipo.

Descripción de los comandos

Crear Zoom (Create Zoom)

Crea una reunión en Zoom.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Zoom necesaria para crear una reunión.	credential
Subject	Asunto de la reunión.	Rocketbot Meeting
Start Date	Fecha de inicio de la reunión.	2023-10-01
Start Time	Hora de inicio de la reunión.	10:00
End Date	Fecha de finalización de la reunión.	2023-10-01
End Time	Hora de finalización de la reunión.	10:00
Time zone	Seleccionar el tipo de notificación.	Time zone
Attendees	Asistentes de la reunión.	johndoe@example.com, timcook@example.com
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: [{"topic": "subject", "type": 2, "start_time": "2023-10-01T10:00:00Z", "duration": 60, "attendees": ["johndoe@example.com"]}]	{var}

Obtener Zoom (Get Zoom)

Obtiene Zoom desde Microsoft Zoom.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Zoom necesaria para obtener Zoom.	credential

Parámetro	Descripción	Ejemplo
Meeting ID	ID de la reunión a obtener.	Meeting ID
Assign result to variable	Variable donde se almacenará el resultado.	{var}

Eliminar Zoom (Delete Zoom)

Elimina una reunión en Zoom.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Zoom necesaria para eliminar una reunión.	credential
Meeting ID	ID de la reunión a eliminar.	Meeting ID
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {"message": "Meeting deleted successfully", "meetingId": "123456789"}	{var}

Actualizar Zoom (Update Zoom)

Actualiza una reunión en Zoom.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Zoom necesaria para actualizar una reunión.	credential
Meeting ID	ID de la reunión a actualizar.	Meeting ID
Subject	Asunto de la reunión.	New Subject
Start Date	Nueva fecha de inicio de la reunión.	2023-10-02
Start Time	Hora de inicio de la reunión.	11:00
End Date	Fecha de finalización de la reunión.	2023-10-02
End Time	Hora de finalización de la reunión.	11:00
Time zone	Seleccionar la zona horaria.	Time zone
Attendees	Asistentes de la reunión.	johndoe@example.com, timcook@example.com
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: [{"topic": "subject", "type": 2, "start_time": "2023-10-01T10:00:00Z", "duration": 60, "attendees": ["johndoe@example.com"]}]	{var}

Listar Zoom (List Zoom)

Lista las reuniones de Zoom.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Zoom necesaria para enviar un mensaje.	credential
From Date	Fecha de inicio para filtrar reuniones (formato AAAA-MM-DD).	2023-10-02
To Date	Fecha de fin para filtrar reuniones (formato AAAA-MM-DD).	2023-10-31

Parámetro	Descripción	Ejemplo
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: [{"uuid": "PLAC35iLS2ubnCaBYW93q69", "id": "123456789", "host_id": "abcdEFGHijklMNOPqrstUV", "topic": "Rocketbot Meeting", "type": 2, "start_time": "2023-10-01T10:00:00Z", "duration": 60, "timeZone": "America/Argentina/Buenos_Aires", "join_url": "https://zoom.us/j/123456789"}]	{var}

Saturn Studio – Teams & Shared Robots

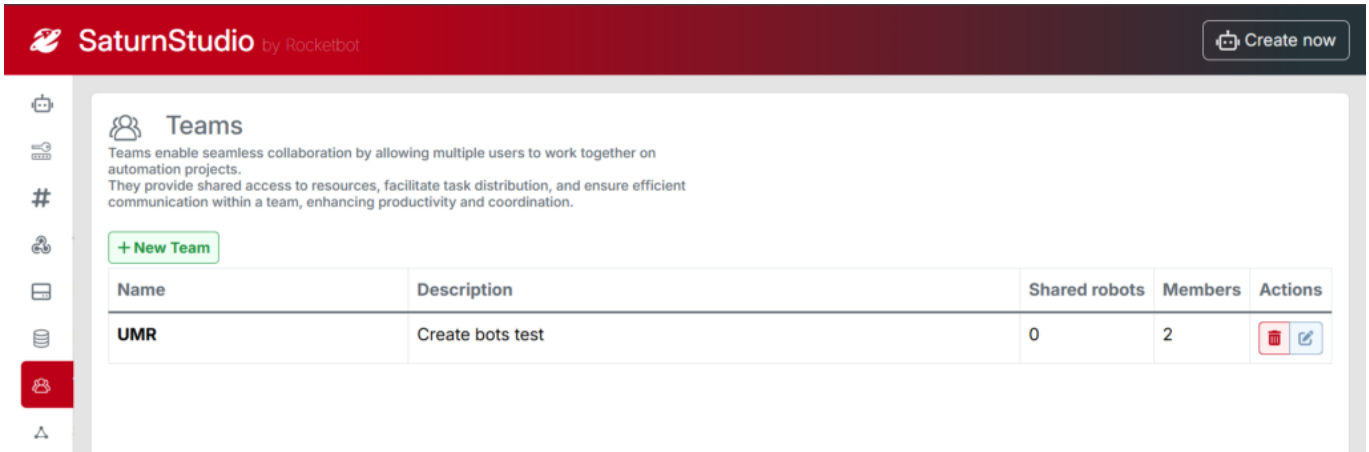
Saturn Studio facilita la colaboración entre desarrolladores mediante la creación de equipos y el uso de robots compartidos. A través del menú lateral, es posible gestionar el acceso a recursos, distribuir tareas y asegurar una comunicación eficiente dentro de un equipo de trabajo, lo que potencia la productividad y la coordinación en proyectos de automatización.

Gestión de Equipos (Teams)

La sección de **Teams** permite invitar a múltiples desarrolladores para trabajar y ejecutar procesos de forma conjunta.

Interfaz de Equipos

En la pantalla principal de Teams se visualiza un listado con la siguiente información:

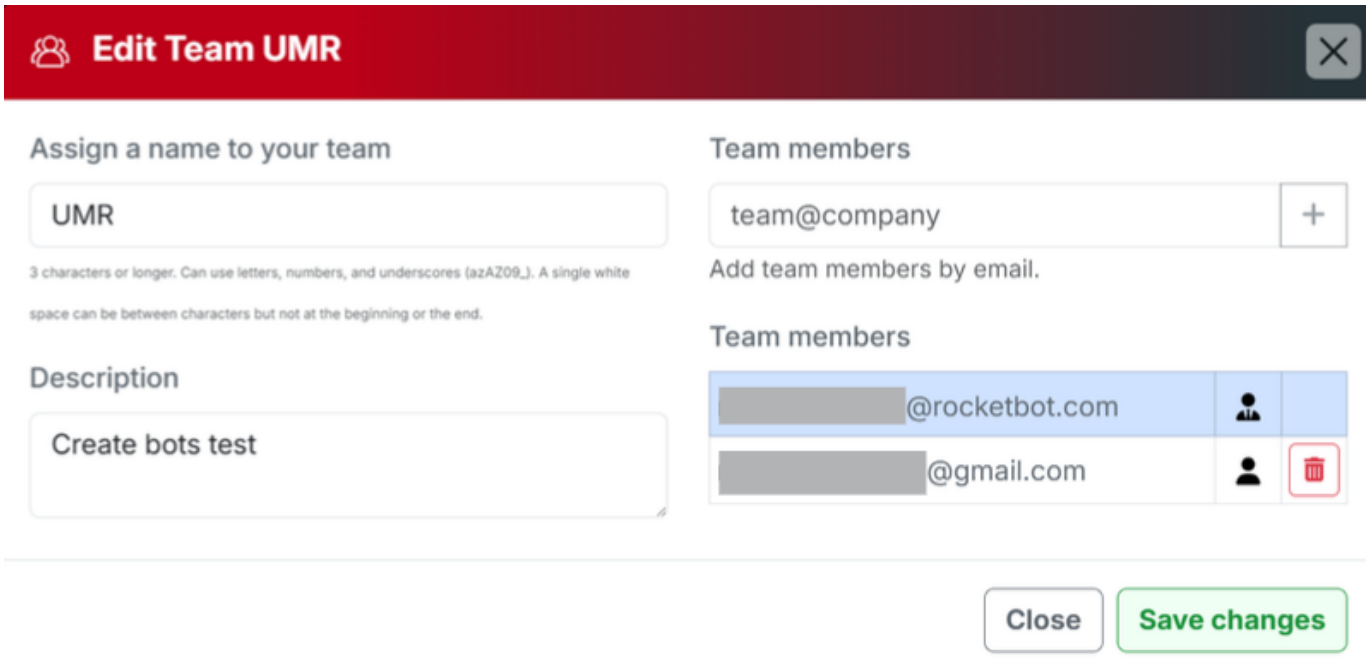


Columna	Descripción
Name	El nombre asignado al equipo.
Description	Una breve explicación de la finalidad del grupo.
Shared robots	Cantidad de procesos compartidos con este equipo.
Members	Número de integrantes que conforman el equipo.
Actions	Permite eliminar el equipo o editar sus elementos (nombre, descripción e integrantes).

Crear y Editar un Equipo

Para configurar un equipo nuevo, debe seguir estos pasos:

1. **Nombre y Descripción**: Asigne un nombre (mínimo 3 caracteres) y una descripción al equipo.
2. **Invitación de Miembros**: Agregue desarrolladores ingresando su correo electrónico en el campo correspondiente.
3. **Lista de Integrantes**: Debajo de la opción de invitación, aparecerá la lista de miembros con permisos para ejecutar los procesos que se compartan en el futuro.

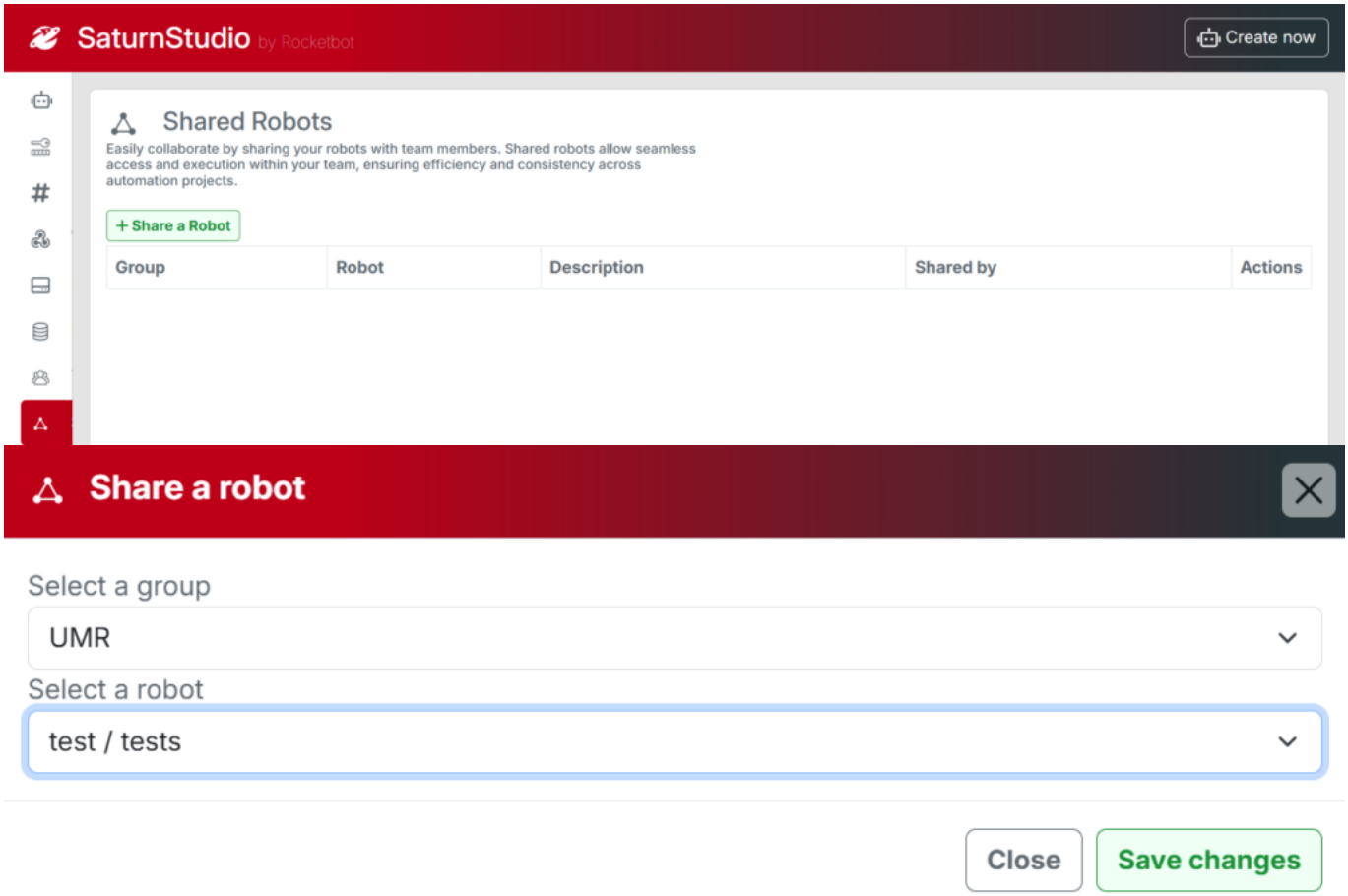


Robots Compartidos (Shared Robots)

El apartado de **Shared Robots** permite que un desarrollador comparta sus robots o procesos con grupos específicos de los cuales es miembro.

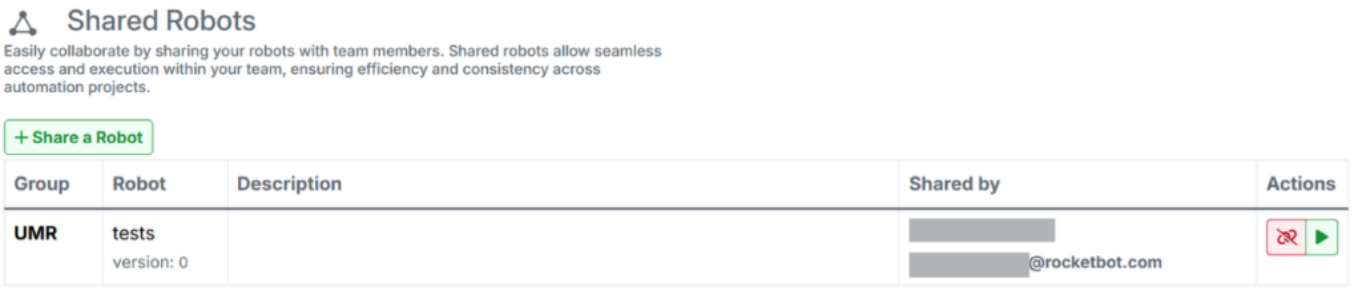
Cómo compartir un Robot

1. Haga clic en el botón **+ Share a Robot**.
2. **Selección de Grupo**: Elija el equipo o grupo donde desea subir el robot.
3. **Selección de Robot**: Elija el proceso específico que quiere compartir y guarde los cambios.



Detalle de la Información Compartida

Una vez compartido, la interfaz mostrará los siguientes datos técnicos:

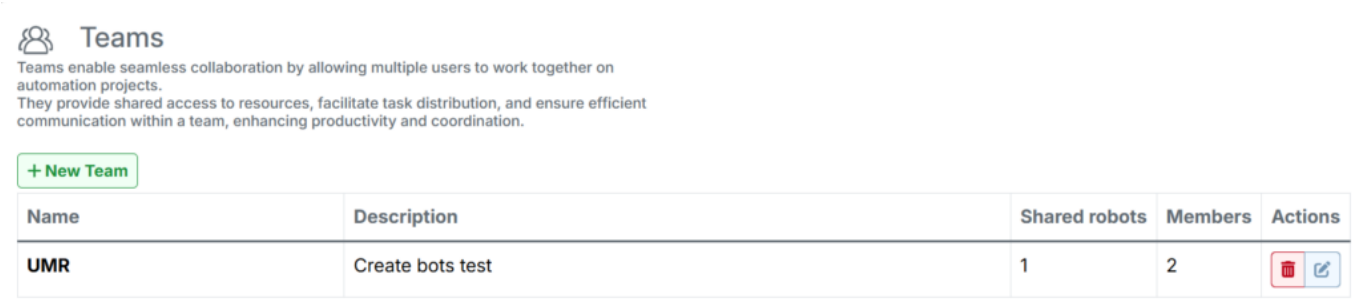


- **Group**: El equipo donde se ha compartido el recurso.
- **Robot**: Nombre del proceso y su número de versión actual.
- **Description**: Información adicional sobre el funcionamiento del bot.
- **Shared by**: Datos del emisor (Nombre y Email) que compartió el proceso.
- **Actions**: Botones para **Ejecutar** el proceso o **Eliminarlo** del grupo compartido.

Visualización General

Al finalizar la creación del equipo y compartir el proceso, el sistema actualiza automáticamente el contador en la sección de **Teams**, permitiendo

visualizar cuántos robots compartidos tiene cada grupo en tiempo real.



Saturn Studio – Webhooks

Los **Webhooks** son herramientas de comunicación en tiempo real que permiten a aplicaciones externas (como Facebook, GitHub o servicios de pago) enviar notificaciones automáticas a **Saturn Studio**. En lugar de que tu robot esté preguntando constantemente si hay novedades, el sistema externo envía la información de manera inmediata a una dirección URL única, lo que dispara la ejecución del bot al instante.

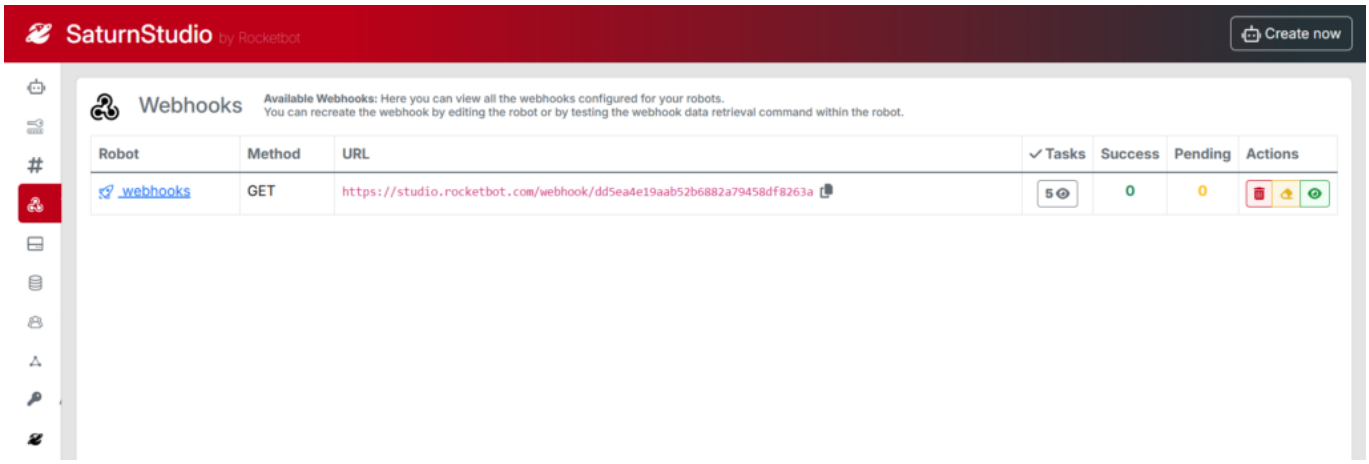
Interfaz de Gestión (Panel Lateral)

Una vez que has configurado y guardado un robot que contenga un comando de Webhook, la información de control aparecerá automáticamente en la sección **Webhooks** del menú lateral izquierdo.

Columna	Descripción
Robot	El nombre del bot que tiene el receptor activo.
Method	El tipo de petición (POST, GET, etc.) que espera recibir.

Columna	Descripción
URL	La dirección única que debes copiar y pegar en la aplicación externa.
Tasks	El contador de cuántas veces se ha ejecutado el flujo por este medio.
Success / Pending	Muestra cuántas solicitudes terminaron bien y cuántas están en proceso.
Actions	Aparecerán las opciones para eliminar el webhook, borrar tareas o ver la cola de tareas.

Visualizar cola de tareas / View tasks queue

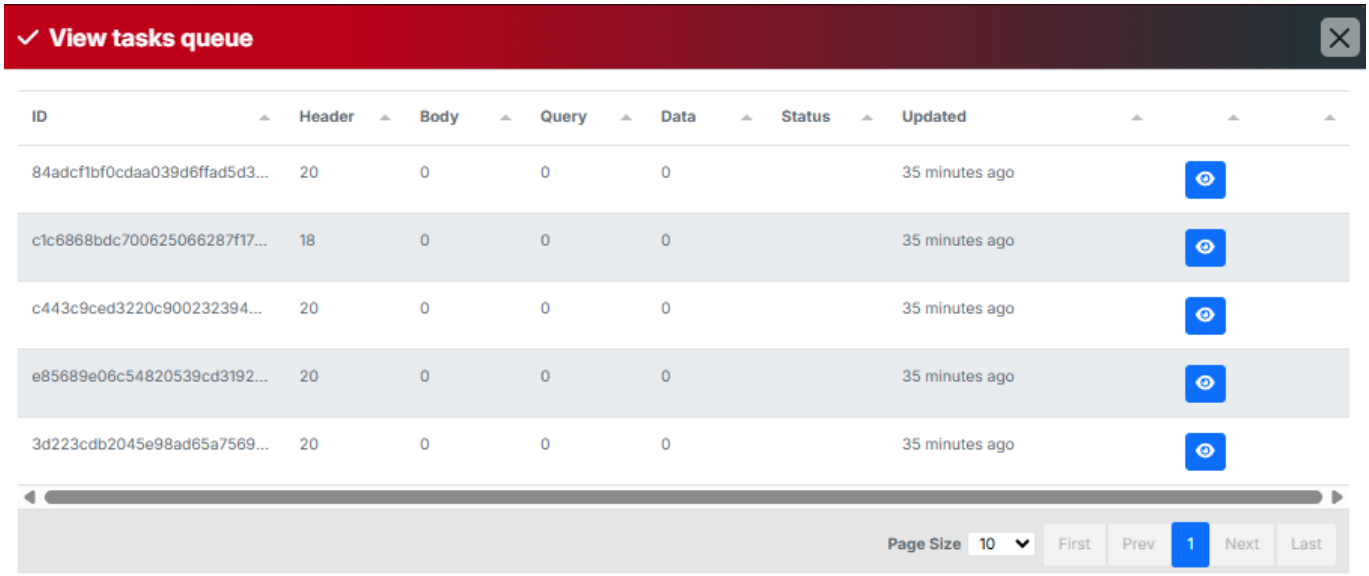


Botones de Acción (Actions)

- **Papelera (Rojo):** Elimina permanentemente el Webhook y su URL del servidor.
- **Escoba/Borrador (Amarillo):** Limpia el historial de tareas y pone los contadores a cero.
- **Ojo (Verde):** Permite ver los detalles técnicos de la configuración actual del Webhook.

Al inspeccionar las ejecuciones de un Webhook, se despliega este modal que lista cada petición recibida de forma individual.

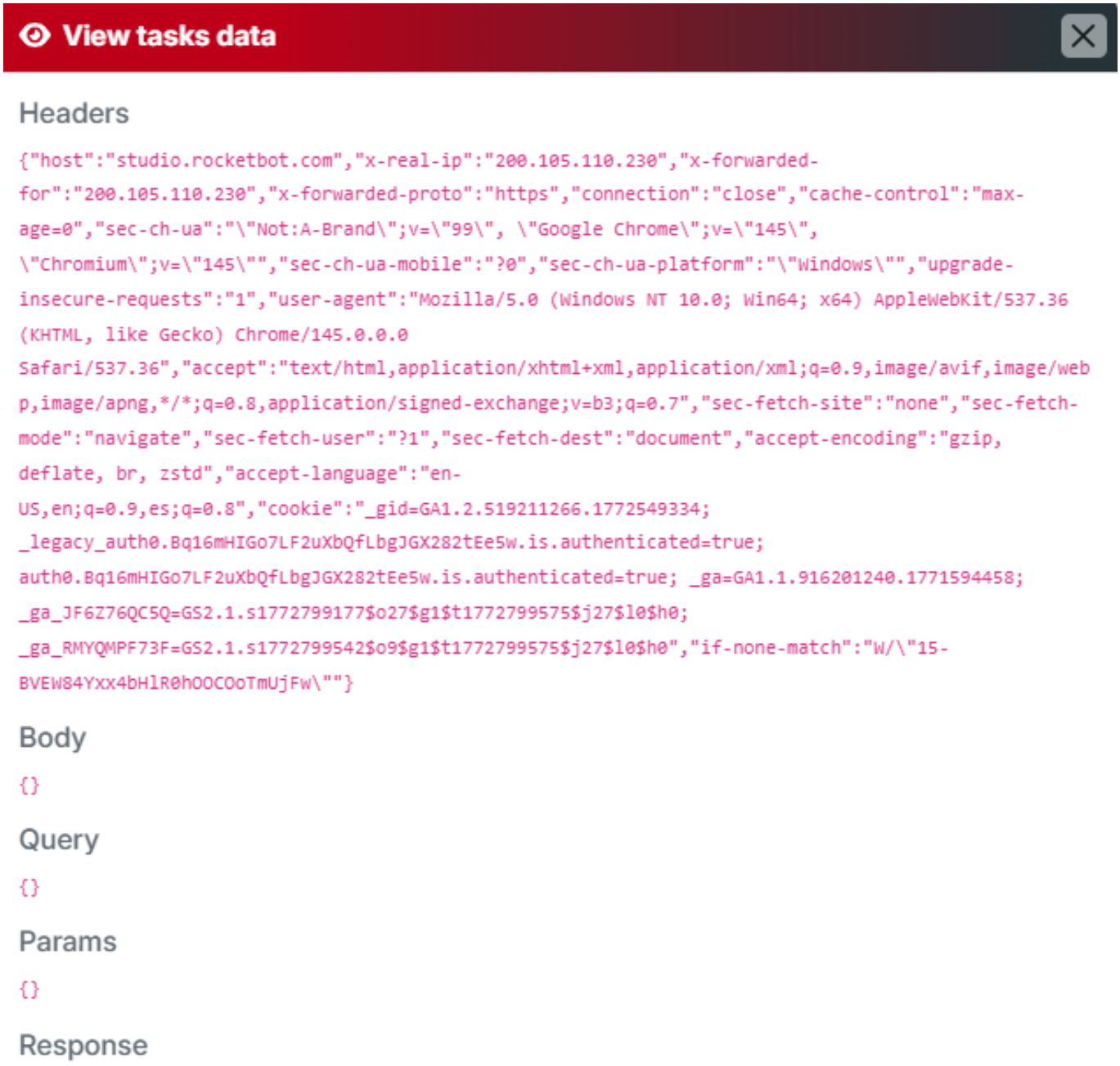
- **ID:** Identificador único alfanumérico de la petición recibida.
- **Header / Body / Query / Data:** Indican la cantidad de parámetros o el peso de la información recibida en cada sección de la petición HTTP.
- **Status:** Muestra si la tarea específica fue completada o si presentó algún error.
- **Updated:** Tiempo transcurrido desde que se recibió la petición.
- **Botón de Ver (Ojo Azul):** Abre el desglose detallado de los datos crudos (raw data) de esa tarea específica.



Detalle de Datos de Tarea (View tasks data)

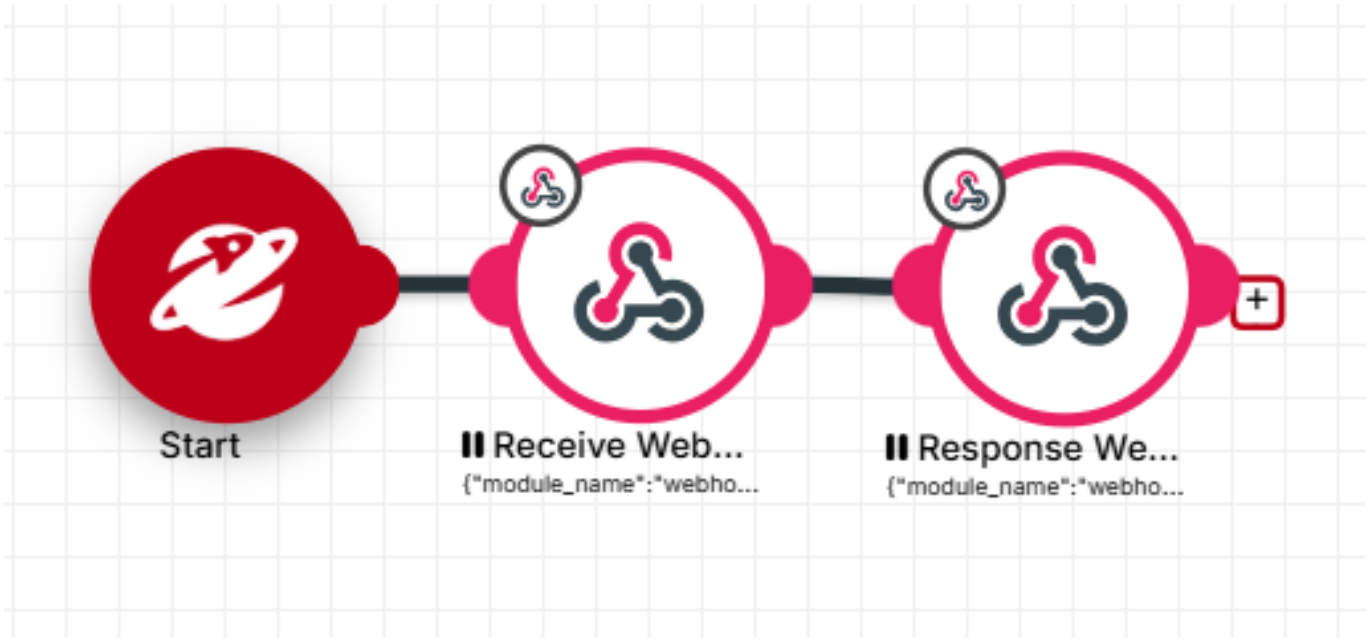
Este panel es fundamental para la depuración (debugging), ya que muestra exactamente qué información envió la aplicación externa a Saturn Studio.

- **Headers:** Contiene los metadatos de la conexión (IP de origen, tipo de navegador, tokens de autenticación, etc.).
- **Body:** El contenido principal del mensaje, generalmente en formato JSON.
- **Query:** Parámetros enviados directamente en la URL (ej: ?id=123).
- **Params:** Parámetros de ruta procesados.
- **Response:** La respuesta que Saturn Studio devolvió al servidor emisor tras recibir el mensaje.



Descripción de los Comandos

Para usar esta tecnología, encontrarás los siguientes módulos dentro de la categoría **Webhooks** del editor.



1. Recibir Webhook (Receive Webhook)

- **Webhook URL**: Es la dirección estática que genera el sistema para tu bot.
- **HTTP Method**: El protocolo de comunicación. Puedes elegir entre **GET, POST, PUT o DELETE**.
- **Response mode**: Define cómo se le avisa al emisor que el dato llegó. Puede ser **Immediately** (al instante), **After robot execution** (al terminar el bot) o de forma manual con el comando de respuesta.
- **Assign result to Variable**: El nombre de la variable donde se guardará toda la información recibida (el cuerpo del mensaje).

2. Responder Webhook (Response Webhook)

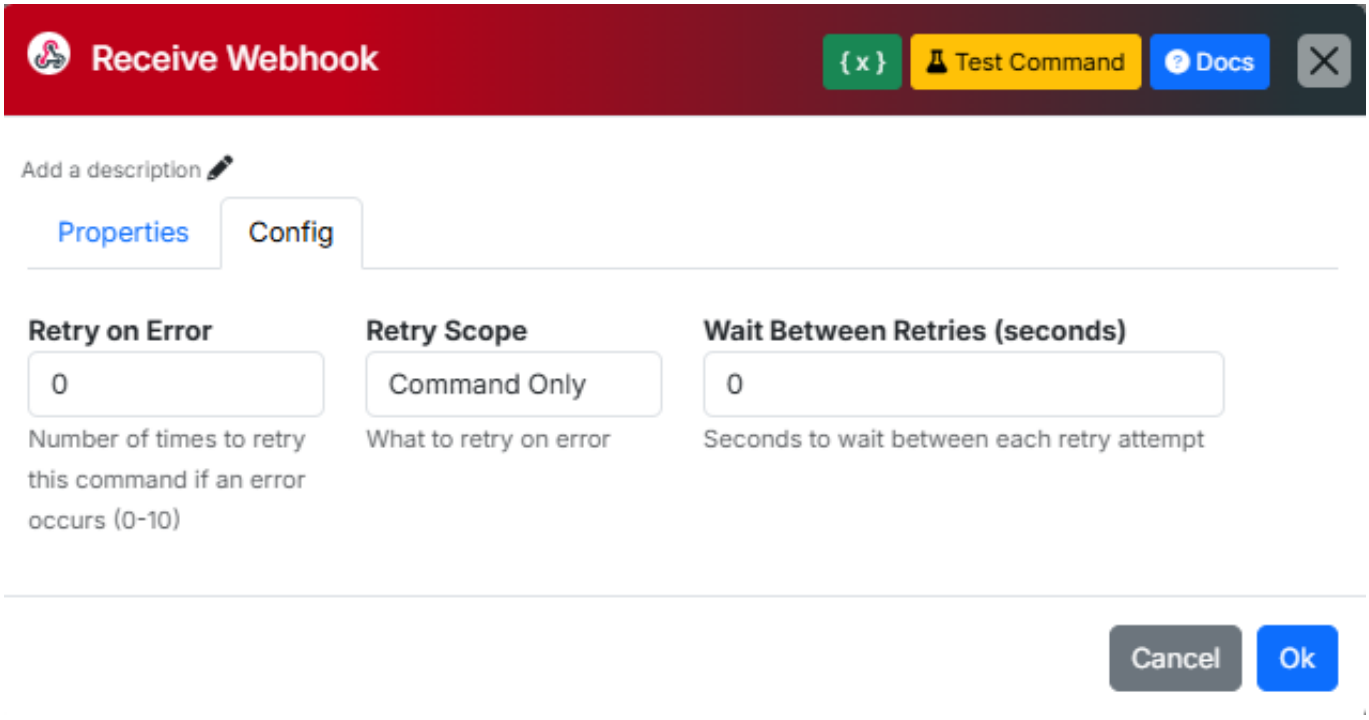
Se utiliza cuando necesitas enviar una respuesta personalizada o específica después de procesar los datos.

- **Task ID**: El identificador de la tarea a la que estás respondiendo.
- **Response**: El texto o JSON que quieres devolver a la aplicación original.
- **Headers**: Encabezados técnicos opcionales para la respuesta.

Configuración Avanzada (Pestaña Config)

Dentro de los módulos, puedes ajustar el comportamiento ante fallos:

- **Retry on Error**: Número de intentos (de 0 a 10) si ocurre un fallo al recibir.
- **Wait Between Retries**: Segundos de espera entre cada intento de recuperación.



Cómo funciona todo en conjunto

1. **En el Editor**: Colocas el módulo **Receive Webhook** al principio de tu robot. Al configurarlo, copias la **URL**.
2. **En la App Externa**: Pegas esa URL en la configuración de la aplicación que quieres conectar (por ejemplo, en las opciones de desarrollador de WhatsApp o Facebook).
3. **El Evento**: Cuando sucede algo afuera, esa app envía un mensaje a la URL.
4. **La Ejecución**: Saturn Studio detecta el mensaje, "despierta" al robot y guarda los datos en la **Variable** que elegiste.
5. **El Monitoreo**: Puedes ir al menú principal, entrar a **Webhooks** y ver cuántas veces ha funcionado y si hubo errores.

Saturn Studio – Global Variables

Las **Variables Globales** son un recurso centralizado diseñado para simplificar la gestión de datos que necesitan ser compartidos entre múltiples procesos. En lugar de configurar la misma información en cada robot, puedes definirla una sola vez y acceder a ella desde cualquier lugar de tu cuenta.

¿Para qué sirven?

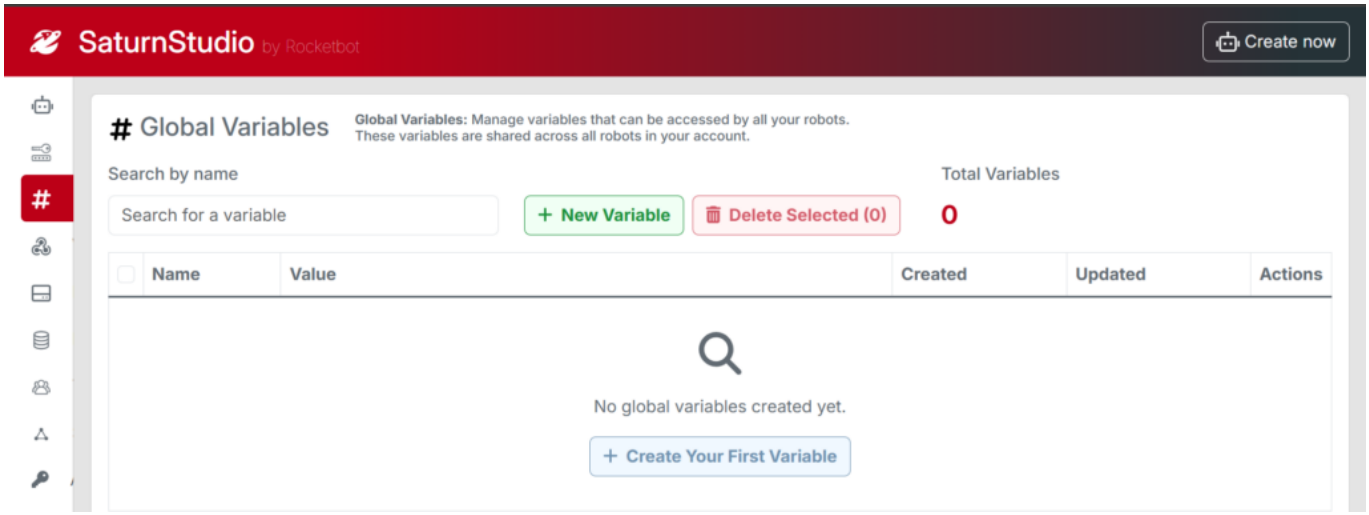
Estas variables están pensadas para almacenar datos constantes o configuraciones maestras que todos tus robots deben conocer. Son ideales

para:

- **URLs de API**: Si la dirección de tu servidor cambia, solo la actualizas en un lugar.
- **Credenciales compartidas o Hosts de Base de Datos**: Mantiene la consistencia técnica en todo tu ecosistema de automatización.
- **Tokens o Flags de sistema**: Información que debe ser idéntica para todos los procesos activos.

Interfaz de Gestión

Desde el panel lateral de Saturn Studio, puedes acceder al gestor de variables globales, donde encontrarás las siguientes herramientas:



Elemento	Función
Search by name	Permite filtrar y encontrar rápidamente una variable específica en una lista extensa.
+ New Variable	Abre el formulario para registrar una nueva variable global.
Delete Selected	Elimina de forma masiva las variables marcadas en la tabla.
Total Variables	Contador en tiempo real del número de variables creadas.

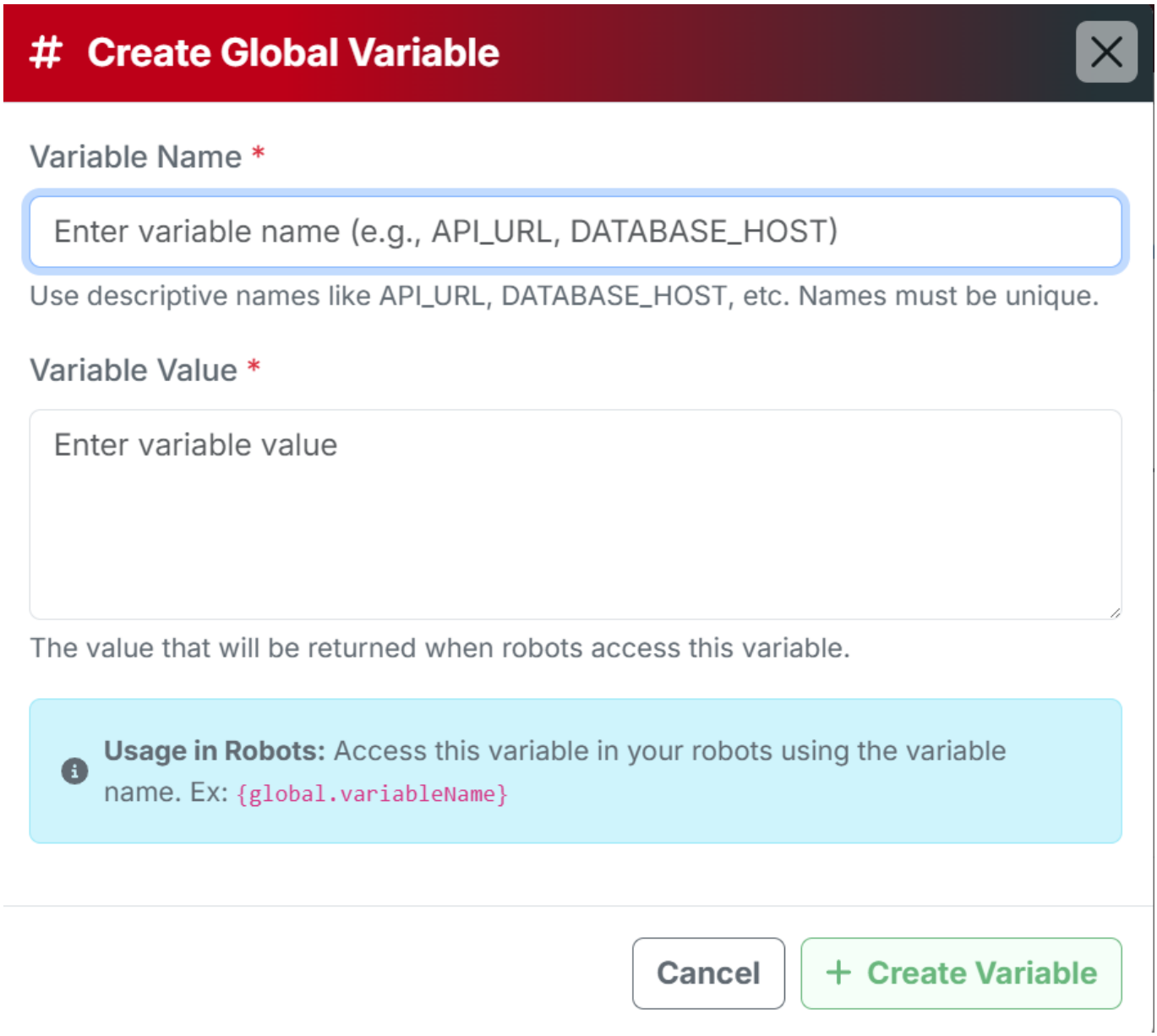
La Tabla de Variables

La tabla principal muestra información detallada para mantener el control sobre los datos:

- **Name**: El identificador único de la variable.
- **Value**: El valor que será devuelto a los robots cuando la invoquen.
- **Created / Updated**: Marcas de tiempo para rastrear cuándo se creó o modificó el dato por última vez.
- **Actions**: Botones para editar o eliminar variables de forma individual.

Cómo Crear una Variable Global

Al hacer clic en **+ New Variable**, se desplegará un modal con los siguientes campos obligatorios:



1. **Variable Name (*)**: Se recomienda usar nombres descriptivos en mayúsculas (ej: API_URL, DATABASE_HOST). **Importante:** Los nombres deben ser únicos dentro de tu cuenta.
2. **Variable Value (*)**: Aquí introduces el dato específico que quieres que tus robots recuperen.
3. **Create Variable**: Guarda el registro y lo hace disponible inmediatamente para todos tus procesos.

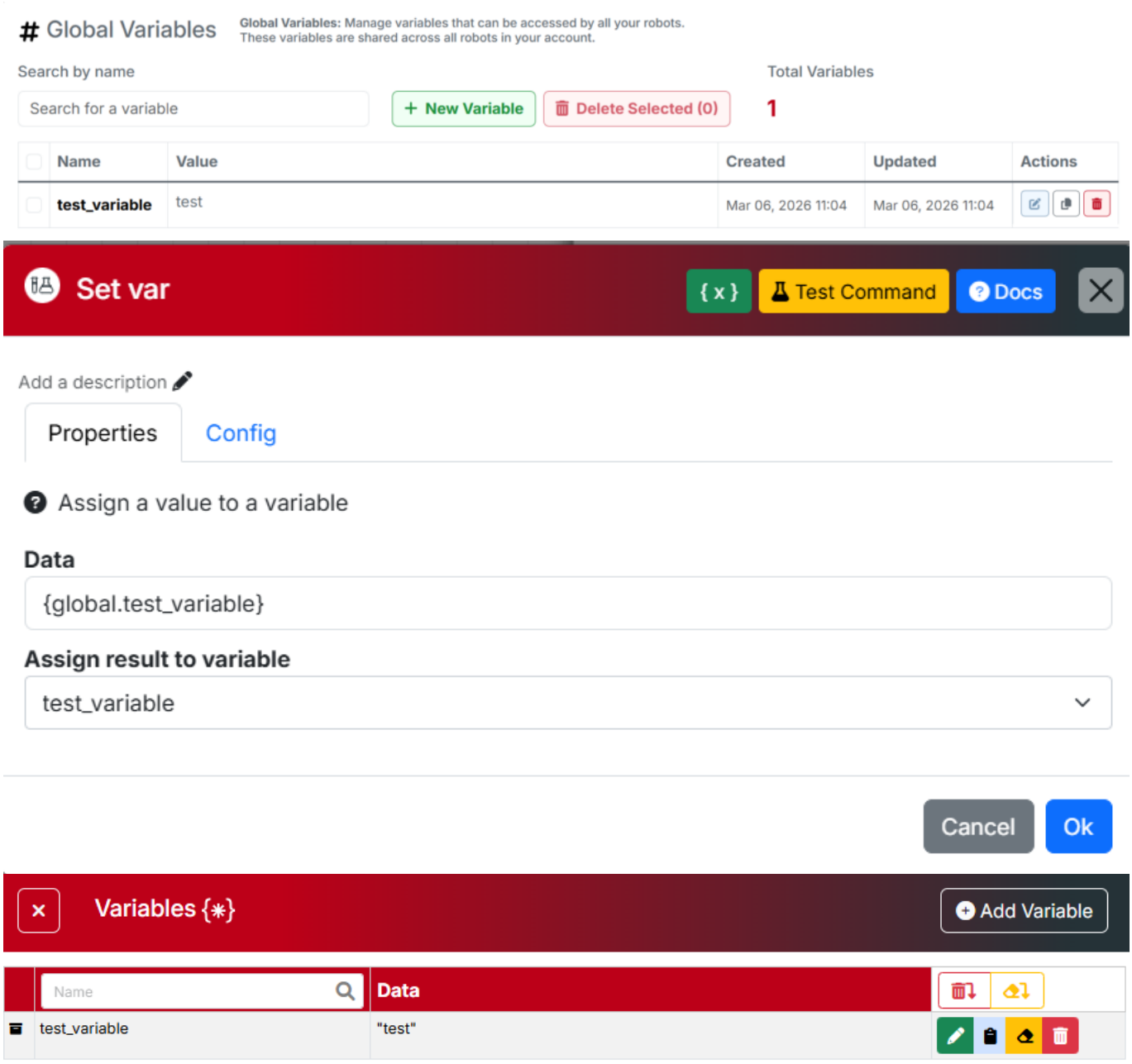
Uso en los Robots (Sintaxis)

Una vez creada, invocar una variable global dentro del editor de cualquier

robot es muy sencillo. Solo debes usar la siguiente estructura:

```
{global.nombreDeTuVariable}
```

Por ejemplo, si creaste una variable llamada API_URL, dentro de tus comandos deberás escribir {global.API_URL} para que Saturn Studio reemplace ese texto por el valor real durante la ejecución.



Saturn Studio – AWS S3



Amazon S3 (Simple Storage Service) es un servicio de almacenamiento en la nube que te permite guardar, proteger y recuperar archivos y datos de forma segura y escalable. Su integración en **Saturn Studio** facilita la gestión automatizada de activos digitales, copias de seguridad y transferencia de archivos a gran escala.

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

Descripción de los Comandos

Cargar Archivo (Upload File)

Sube un archivo local a un contenedor (bucket) específico en AWS S3.

Parámetro	Descripción	Ejemplo
Credential	Credencial de AWS S3 necesaria para cargar el archivo.	My AWS S3 credentials
Bucket Name	Nombre del bucket de S3 donde se guardará el archivo.	bucket_name
File	El archivo local que deseas subir a AWS S3.	Archivo a cargar
Assign result to variable	Variable donde se almacenará el resultado de la carga.	{var}

Descargar Archivo (Download File)

Descarga un archivo desde un bucket de AWS S3 a tu entorno local.

Parámetro	Descripción	Ejemplo
Credential	Credencial de AWS S3 necesaria para descargar el archivo.	My AWS S3 credentials
Bucket Name	Nombre del bucket de S3 donde está almacenado el archivo.	bucket_name
File Name	La clave (ruta) del archivo que se desea descargar de S3.	path/to/file.txt
Assign result to variable	Variable donde se almacenará el archivo descargado o el resultado.	{var}

Obtener Lista de Objetos en un Bucket (Get List of Objects in a Bucket)

Recupera un listado de todos los objetos almacenados dentro de un bucket específico.

Parámetro	Descripción	Ejemplo
Credential	Credencial de AWS S3 necesaria para consultar el bucket.	My AWS S3 credentials
Bucket Name	Nombre del bucket de S3 que se desea listar.	bucket_name
Assign result to variable	Variable donde se almacenará la lista de objetos.	{var}

Obtener Lista de Buckets (Get List of Buckets)

Enumera todos los buckets disponibles asociados a la cuenta de AWS.

Parámetro	Descripción	Ejemplo
Credential	Credencial de AWS S3 necesaria para listar los buckets.	My AWS S3 credentials
Assign result to variable	Variable donde se almacenará el listado de buckets encontrados.	{var}

Saturn Studio – Credencial AWS S3



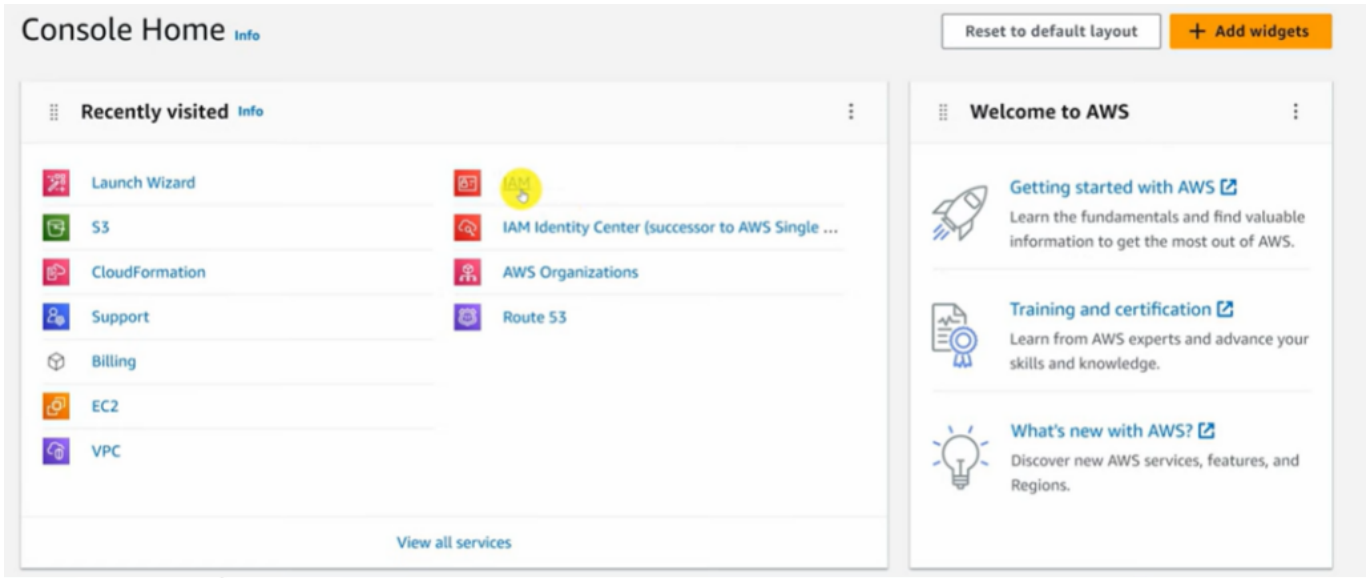
Amazon S3 (Simple Storage Service) es un servicio de almacenamiento en la nube que permite guardar, proteger y recuperar archivos y datos de forma segura y escalable. Su integración en **Saturn Studio** es fundamental para automatizar flujos que requieran el manejo persistente de activos digitales o copias de seguridad.

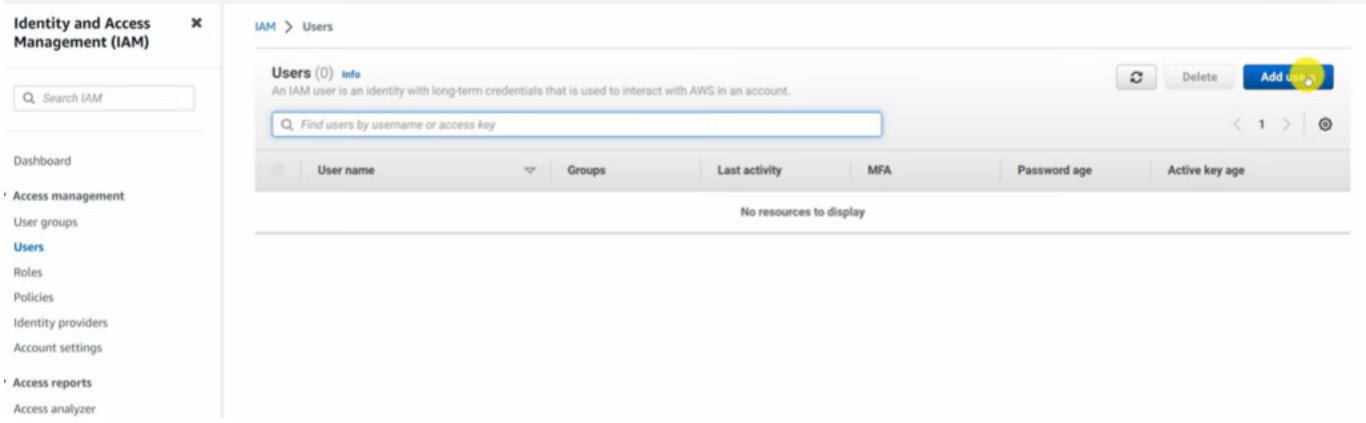
Cómo obtener las llaves de acceso (IAM)

El proceso de obtención de credenciales para S3 es idéntico al de otros servicios de AWS, variando únicamente el permiso específico que se le otorga al usuario.

1. Creación del Usuario en la Consola de AWS

- Inicia sesión en tu Consola de administración de AWS.
- En el buscador superior, selecciona el servicio **IAM**.
- En el menú lateral, dirígete a **Users** (Usuarios) y haz clic en el botón **Add Users**.



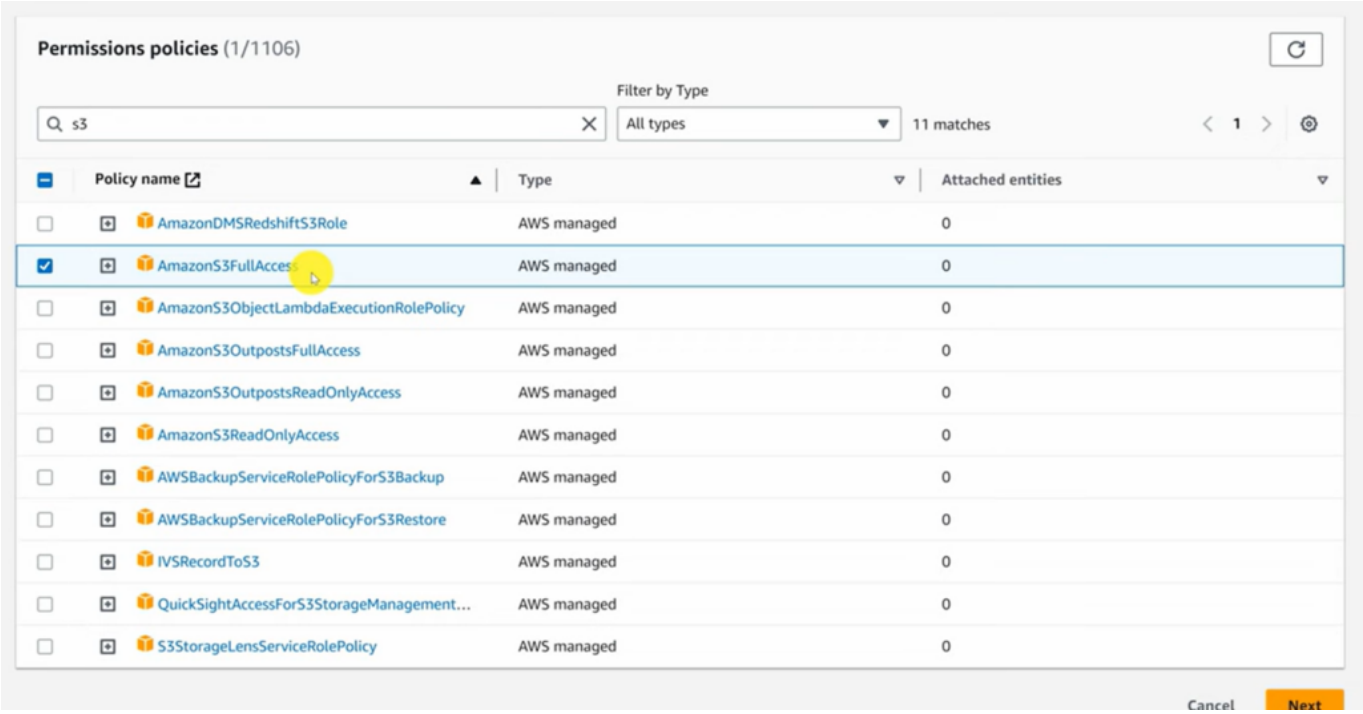


2. Configuración de Permisos

Para que Saturn Studio pueda gestionar tus archivos, el usuario debe tener acceso al servicio de almacenamiento:

- En la pantalla de **Set permissions**, selecciona la opción **Attach policies directly**.
- En el buscador de políticas, escribe **s3** y selecciona la casilla de **AmazonS3FullAccess**.

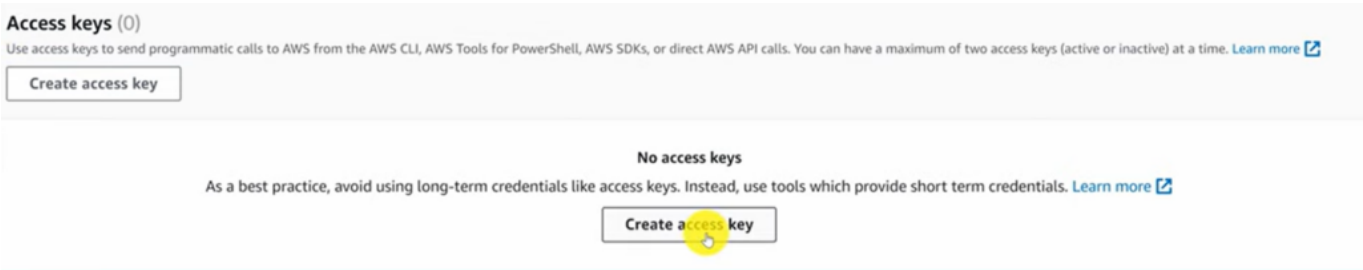
- Finaliza el proceso de creación del usuario.

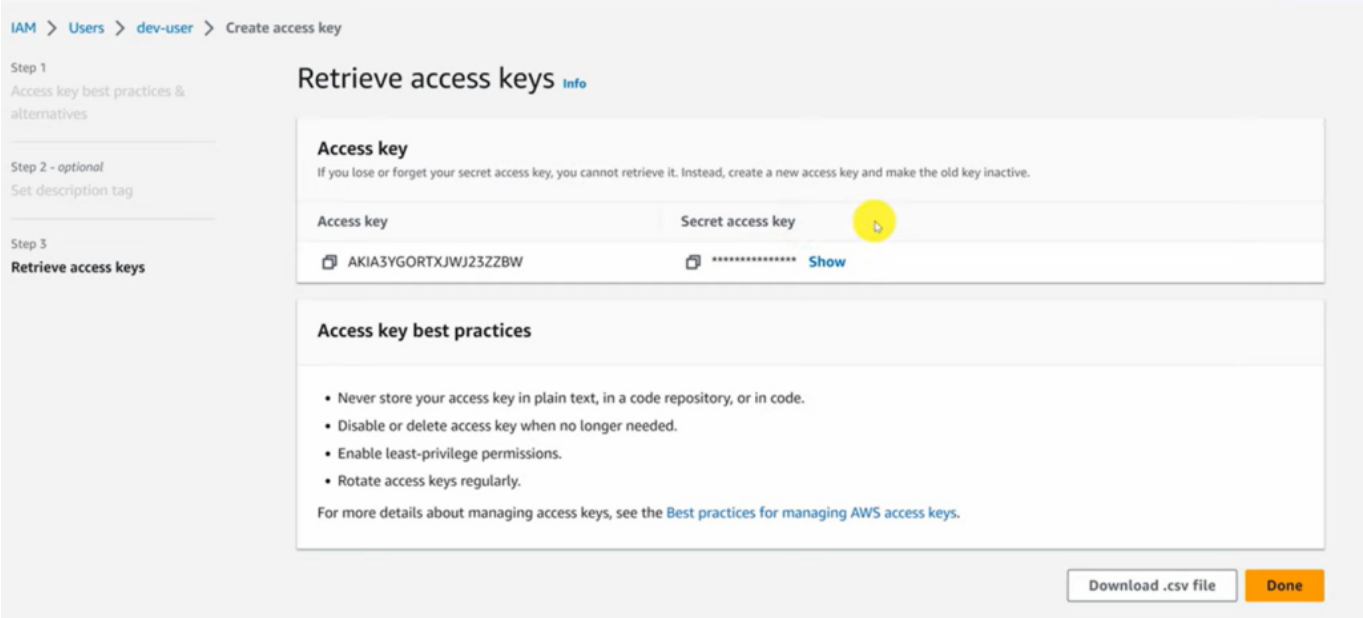


3. Generación de las Access Keys

Una vez que el usuario esté creado, accede a su perfil para generar las llaves:

- Haz clic en la pestaña **Security credentials**.
- Desplázate hasta **Access keys** y haz clic en **Create access key**.
- Selecciona el caso de uso **Other** y asigna una descripción opcional (ej. Saturn_S3_Key).
- **IMPORTANTE**: Se mostrarán tu **Access key** y tu **Secret access key**. Debes copiarlas y pegarlas directamente en el módulo, ya que la llave secreta no volverá a mostrarse por seguridad.





Vinculación en Saturn Studio

Para finalizar la configuración, abre la ventana de credenciales del módulo de **AWS S3** y completa los campos:

1. **Credential name**: Ingresa un nombre para identificar esta conexión.
2. **AWS Region**: Selecciona la región donde se encuentran tus buckets de S3 (ej. us-east-1).
3. **AWS Access Key ID**: Pega el identificador de acceso que copiaste de la consola de AWS.
4. **AWS Secret Access Key**: Pega la llave secreta correspondiente.
5. Haz clic en el botón **Check** para verificar que los permisos de S3 sean válidos.
6. Presiona **Save credentials** para guardar los cambios.

Saturn Studio – Credencial Asana



Asana es una plataforma líder en gestión de trabajo y productividad diseñada para ayudar a los equipos a organizar tareas, proyectos y objetivos en un solo lugar. Al integrarla con **Saturn Studio**, puedes automatizar la creación de tareas y la gestión de flujos de trabajo colaborativos de manera eficiente.

Cómo obtener el Access Token

Para vincular Asana con Saturn Studio, necesitas generar un token de acceso desde la consola de desarrolladores de Asana siguiendo estos pasos:

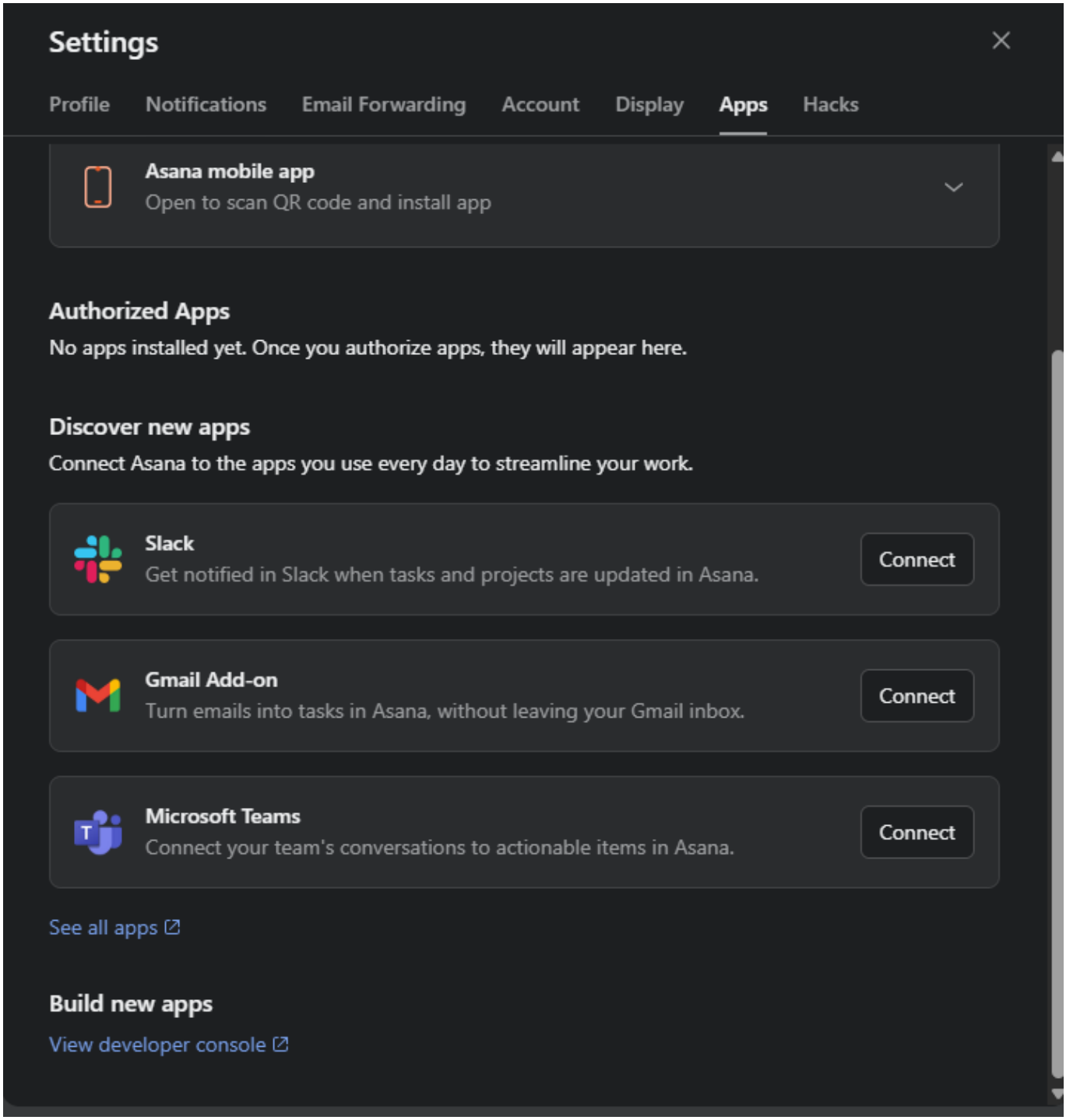
1. Acceso desde Saturn Studio

- Dentro del módulo de Asana en Saturn Studio, localiza y haz clic en el enlace azul **“here”** ubicado debajo del campo de Access Token.
- Esto te redirigirá automáticamente al portal de **Asana para desarrolladores**.
- Inicia sesión con tu cuenta de Asana o regístrate si aún no posees una.



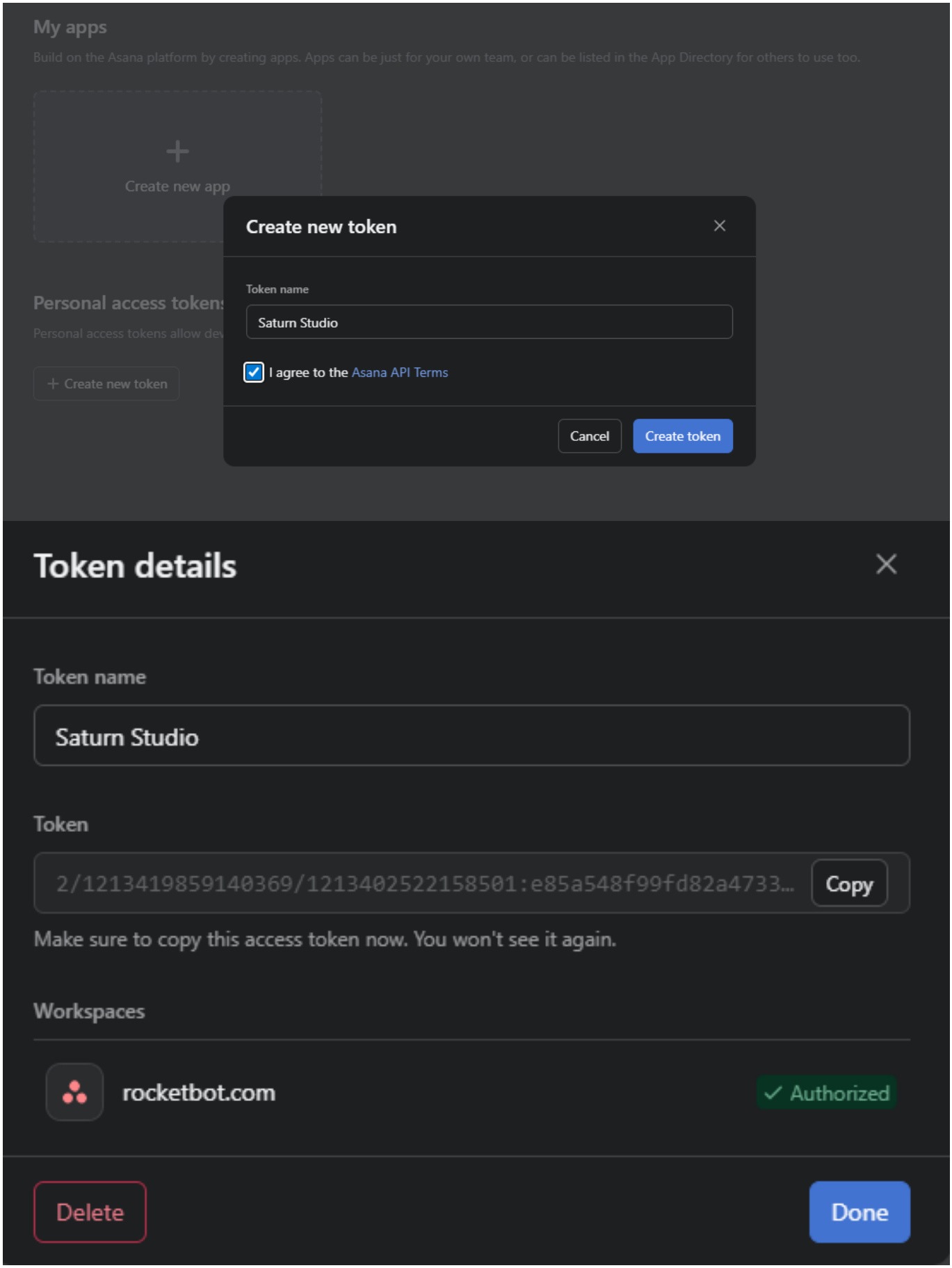
2. Navegación a la Consola de Desarrolladores

- Haz clic en tu **foto de perfil** para acceder a la **Configuración** (Settings).
- Dirígete a la pestaña de **Apps** (Aplicaciones).
- Al final de esta sección, busca y haz clic en el enlace **“View developer console”** (Ver consola de desarrolladores).



3. Generación del Access Token

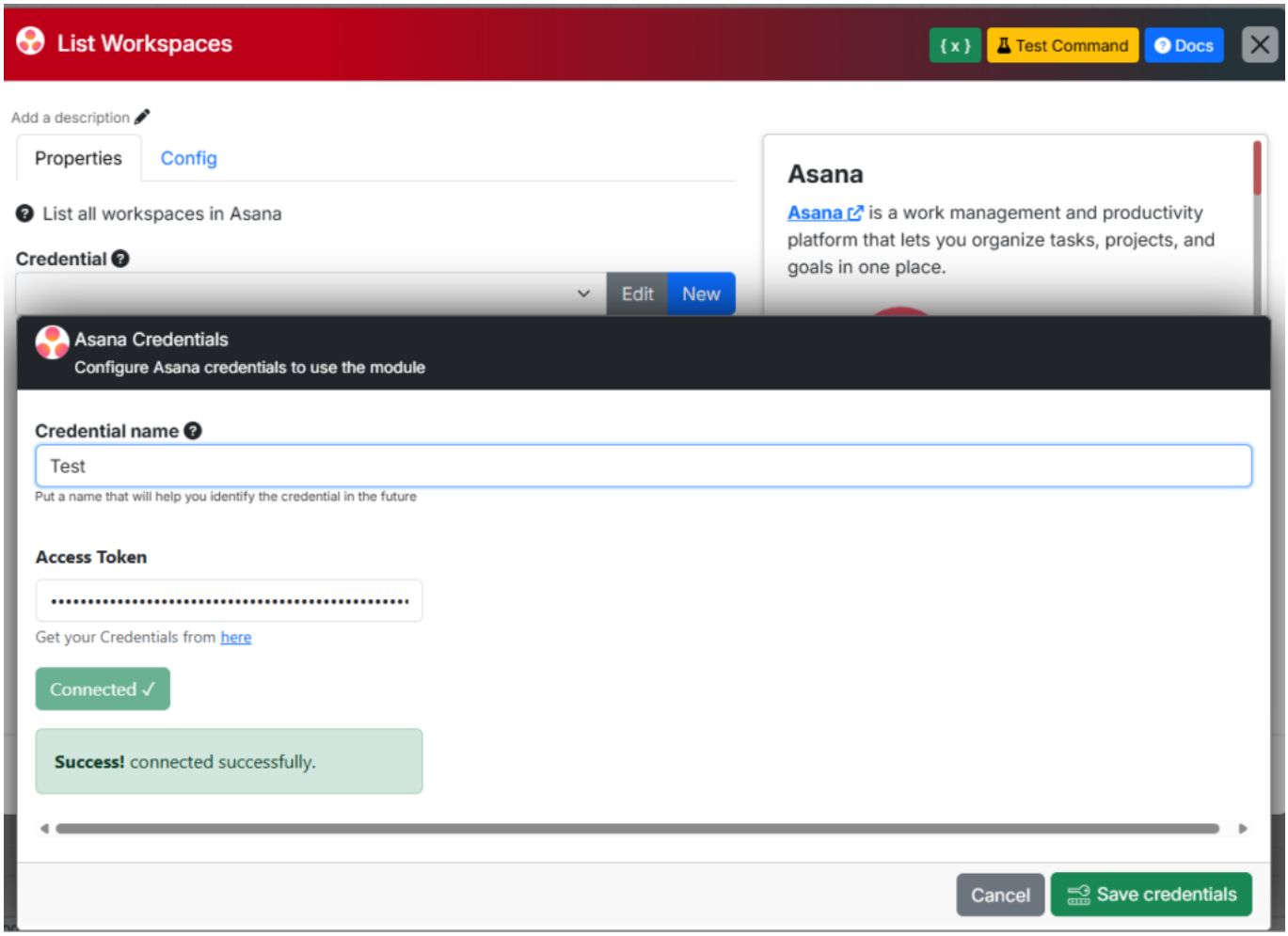
- En la sección de **Personal Access tokens**, haz clic en el botón **“+ Create new token”**.
- **Token name:** Asigna un nombre descriptivo para identificar el token, por ejemplo, “Saturn Studio”.
- Marca la casilla de aceptación de los términos y haz clic en **“Create token”**.
- **Importante:** Se mostrará el token generado. Haz clic en **“Copy”** para copiarlo y guárdalo en un lugar seguro, ya que no se volverá a mostrar completo.



Vinculación en Saturn Studio

Regresa a Saturn Studio para finalizar la configuración de la credencial:

1. En el campo **Credential name**, ingresa un nombre para identificar esta conexión.
2. En el campo **Access Token**, pega el código alfanumérico que acabas de copiar desde Asana.
3. Haz clic en el botón azul **“Connect to Asana”**.
4. Si los datos son correctos, el sistema mostrará un recuadro verde con el mensaje **“Success! connected successfully”** y el estado cambiará a **Connected**.
5. Finalmente, haz clic en **“Save credentials”**.



Saturn Studio – Asana



Asana es una plataforma de gestión de trabajo y productividad que permite organizar tareas, proyectos y objetivos en un solo lugar. Su integración facilita la coordinación de equipos, el seguimiento de plazos y la centralización de la información para mejorar el rendimiento colectivo.

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

Descripción de los Comandos

Listar Espacios de Trabajo (List Workspaces)

Enumera todos los espacios de trabajo disponibles en la cuenta vinculada de Asana.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para listar los espacios de trabajo.	credential
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“Workspaces”:[{“id”:”9034647833″, “gid”:”9034647833″, “name”:”SaturnStudio”}]}	{var}

Listar Todas las Tareas (List All Tasks)

Muestra todas las tareas contenidas en un espacio de trabajo o proyecto específico.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para listar las tareas.	credential
Workspace	Espacio de trabajo desde el cual se listarán las tareas.	My Workspace
Project	Proyecto desde el cual se listarán las tareas (opcional).	Marketing Project
Task Status	Permite filtrar las tareas según su estado de finalización.	

Parámetro	Descripción	Ejemplo
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: [{“id”:”12034567890″,”name”:”name”,”completed”:”true”, “due_on”:”2023-12-31″, “assignee”:”John Doe”, “assignee_id”:”1122334455″}]	{var}

Obtener Espacio de Trabajo (Get Workspace)

Recupera los detalles técnicos de un espacio de trabajo específico.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para obtener los detalles.	credential
Workspace ID	ID único del espacio de trabajo que se desea consultar.	9034647833
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“id”:”9034647833″,”name”:”SaturnStudio”,”is_organization”: “false”, “archived”:”false”}	{var}

Obtener Proyecto (Get Project)

Obtiene la información detallada de un proyecto mediante su identificador.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para obtener los detalles del proyecto.	credential
Project ID	ID del proyecto del cual se desea obtener la información.	12034567890
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“id”:”12034567890″,”name”:”Project”,”archived”:”false”, “owner”:”John Doe”}	{var}

Crear Tarea (Create Task)

Genera una nueva tarea dentro de un espacio de trabajo o proyecto definido.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para crear la tarea.	credential
Workspace	Espacio de trabajo donde se creará la tarea.	My Workspace
Project	Proyecto donde se alojará la tarea.	Marketing Project
Assignee	Usuario a quien se le asignará la tarea.	nombre_usuario
Task Name	Nombre de la tarea a crear.	Revisar diseño de la interfaz
Description	Descripción detallada de las acciones a realizar en la tarea.	Revisar el diseño de la nueva interfaz
Due Date	Fecha de vencimiento (Formato YYYY-MM-DD o DD-MM-YYYY).	31-12-2026
Custom Fields	Campos personalizados en formato JSON.	mi_valor

Parámetro	Descripción	Ejemplo
Assign result to variable	Variable donde se almacenará el resultado de la creación.	{var}

Obtener Campos Personalizados de Proyecto (Get Project Custom Fields)

Extrae los campos personalizados configurados para un proyecto en particular.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para la consulta.	credential
Workspace	Espacio de trabajo desde donde se obtendrán los campos.	My Workspace
Project	Proyecto del cual se extraerán los campos personalizados.	Marketing Project
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: [{“name”:”Priority”, “id”:”12034567890″,”type”:”enum”, “options”:[{“id”:”111″,”name”:”High”}]}]	{var}

Obtener Tarea (Get Task)

Recupera los detalles completos de una tarea específica mediante su ID.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para obtener los detalles.	credential
Task ID	ID único de la tarea a consultar.	12034567890
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“id”:”12034567890″,”name”:”Nombre de la tarea”,”notes”:”descripción”, “due_on”:”2023-12-31″, “assignee”:”John Doe”, “assignee_id”:”1122334455″, “project_id”:”9988776655″}	{var}

Eliminar Tarea (Delete Task)

Borra de forma permanente una tarea de la plataforma.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para eliminar la tarea.	credential
Task ID	ID de la tarea que se desea borrar.	12034567890
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“success”:true, “task_id”:”12034567890″, “message”:”Tarea eliminada con éxito”}	{var}

Listar Usuarios (List Users)

Obtiene el listado de todos los usuarios vinculados a un espacio de trabajo.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para listar los usuarios.	credential
Workspace	Espacio de trabajo desde el cual se listarán los usuarios.	My Workspace
Project	Proyecto desde el cual se listarán los usuarios (opcional).	Marketing Project
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“id”:”1122334455″,”name”:”John Doe”}	{var}

Actualizar Tarea (Update Task)

Modifica los datos de una tarea ya existente.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para la actualización.	credential
Task ID	ID único de la tarea a modificar.	1212531686562092
Task Name	El nuevo nombre para la tarea.	Nuevo nombre de tarea
Workspace	Espacio de trabajo donde se ubica la tarea.	My Workspace
Project	Proyecto donde se encuentra la tarea.	Marketing Project
Assignee	Nuevo usuario asignado a la tarea.	nombre_usuario
Description	Nueva descripción para la tarea.	Nueva descripción
Due Date	Nueva fecha de vencimiento (Formato DD-MM-YYYY).	31-12-2026
Custom Fields	Campos personalizados en formato JSON.	mi_valor
Mark as Completed	Marca la tarea como completada automáticamente.	
Assign result to variable	Variable donde se almacenará el resultado de la actualización.	{var}

Cargar Archivo Adjunto a Tarea (Upload Attachment to Task)

Sube un archivo local y lo adjunta a una tarea específica en Asana.

Parámetro	Descripción	Ejemplo
Credential	Credencial de Asana necesaria para cargar el archivo.	credential
Task ID	ID de la tarea a la cual se le subirá el adjunto.	12034567890
Workspace	Espacio de trabajo donde se ubica la tarea.	My Workspace
Project	Proyecto donde se encuentra la tarea.	Marketing Project
Local File	Archivo local que se desea cargar como adjunto.	seleccionar archivos

Parámetro	Descripción	Ejemplo
Assign result to variable	Variable donde se almacenará el resultado. Ejemplo: {“id”:”1212461679463197″, “resource_type”: “attachment”, “name”: “Nombre del archivo”}	{var}