# Saturn Studio – Supabase



Supabase es una plataforma de código abierto que proporciona una alternativa a Firebase, ofreciendo un conjunto completo de herramientas de backend, incluyendo bases de datos en tiempo real, autenticación y servicios API.

# ¿Qué es este módulo?

Este módulo permite conectar Saturn Studio con **Supabase**, una plataforma open-source alternativa a Firebase que ofrece una completa suite de herramientas backend como bases de datos en tiempo real, autenticación, almacenamiento y ejecución de funciones Postgres. Ideal para automatizaciones que requieren operaciones con datos dinámicos, embeddings o almacenamiento de archivos.

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

# Descripción de los comandos

## Get Table

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para acceder a la tabla. | credential |
| Nombre de la tabla | Nombre de la tabla que se desea obtener. | Clients |
| Ordenar por fecha de creación | Si está activado, ordena los resultados de más nuevo a más antiguo. | true |
| Asignar resultado a variable | Variable donde se almacenará el resultado. {“table”[{“id”4,”created_at””2025-03-31T164319.794747+0000˝,”name””John”,”lastname””Doe”},{“id”5,”created_at””2025-03-31T164356.786912+0000˝,”name””John”,”lastname””Doe”}]} | {var} |

Obtiene el contenido completo de una tabla específica.

## Filter Table

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para filtrar la tabla. | credential |
| Nombre de la tabla | Tabla a consultar. | Clients |
| Nombre de la columna | Columna por la que se filtrará. | Id |
| Valor | Valor por el cual se filtrará la columna. | 31233 |
| Asignar resultado a variable | Variable donde se almacenará el resultado. {“table”[{“id”450102905661143,”created_at””2025-04-03T190051.148+0000˝,”name””Wikipedia”,”lastname””web”,”embed””[-0.053218544,-0.025654549…]”}]} | {var} |

Filtra una tabla por el valor de una columna específica.

## Insert Rows

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para insertar. | credential |
| Nombre de la tabla | Tabla donde se insertarán los datos. | Clients |
| Columnas | Lista de objetos clave-valor para insertar. | [{ “Name”: “John”, “Lastname”: “Doe”, “Age”: 30 }] |
| Asignar resultado a variable | Variable donde se almacenará el resultado. {“insertedRows”[{“id”14,”created_at””2025-03-31T173647.646493+0000˝,”name””John”,”lastname””Doe”},{“id”15,”created_at””2025-03-31T173647.646493+0000˝,”name””Dave”,”lastname””Davinson”}]} | {var} |

Inserta una o más filas en la tabla especificada.

## Update Rows

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para actualizar. | credential |
| Nombre de la tabla | Tabla donde se actualizarán los datos. | Clients |
| Columna a actualizar | Nombre de la columna a modificar. | Name |
| Valor | Nuevo valor para esa columna. | John |
| Columna para filtrar | Columna que se usará para identificar los registros. | Id |
| Valor del filtro | Valor por el cual filtrar la columna. | 31233 |
| Asignar resultado a variable | Variable donde se almacenará el resultado. {“message” “Rows updated successfully”} | {var} |

Actualiza filas que cumplan una condición específica en la tabla.

## Get a Bucket

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para acceder al bucket. | credential |
| Bucket | Nombre del bucket a obtener. | Rocket |
| Asignar resultado a variable | Variable donde se almacenará el resultado. | {“bucket”{“id””Rocket”,”name””Rocket”,”owner””””,”public”true,”file_size_limit”1024,”allowed_mime_types”[“image/png”],”created_at””2025-03-31T190849.061Z”,”updated_at””2025-03-31T190849.061Z”}} |

Obtiene los detalles de un bucket de almacenamiento.

## Create a Bucket

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para crear el bucket. | credential |
| Nombre del bucket | Nombre del nuevo bucket. | Rocket Launches |
| Público | Si está activo, el bucket será público. | true |
| Límite de tamaño | Tamaño máximo de archivo en bytes. | 1024 |
| Tipos MIME permitidos | Lista separada por comas de tipos MIME permitidos. | image/jpeg, image/png |

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Asignar resultado a variable | Variable donde se almacenará el resultado. | {“bucket”{“name””RocketbotBucket”}} |

Crea un nuevo bucket de almacenamiento en Supabase.

## Upload File

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria para subir el archivo. | credential |
| Bucket | Bucket donde se subirá el archivo. | Rocket |
| Ruta del archivo | Ruta local del archivo a subir. | image.jpg |
| Asignar resultado a variable | Variable donde se almacenará el resultado. | {“file”{“path””C\Users\Desktop\saturn_studio\Rocketbot Icon.jpg”,”id””3aa9f15a-b93c-4765-92cc-e0574f71b446˝,”fullPath””AAAAAA/C/Users/Desktop/saturn_studio/Rocketbot Icon.jpg”}} |

Sube un archivo (imagen, PDF, etc.) al bucket seleccionado.

## Generate and Store Embedding

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase. | credential |
| Credencial de IA | Credencial del motor de embeddings a usar. | Name |
| Modelo de embedding | Modelo utilizado para generar embeddings. | Name |
| Nombre de la tabla | Tabla donde se almacenará el embedding generado. | Clients |
| Contenido | Contenido a partir del cual se generará el embedding. | {variable} or <!doctype html… |
| Tamaño de fragmento | Tamaño del bloque de texto a procesar. | 1024 |
| Superposición de fragmento | Superposición entre bloques de texto. | 100 |
| Asignar resultado a variable | Variable donde se almacenará el resultado. | {“table””saturn_table”,”embed_column””embed”,”processed_rows”[{“id”855046726584984 9,”text””{{“data_”[{“filename””sitefetch_1743693799824.txt”,”id””483171b10ebf1f8b5ebcc22ecaad0f2f7851f6613ca22496fb9557c0b485b34f27b388498fbe9b48efe290364503 5e38”,”size”154,”provider””sitefetch”}],”content”{“/”{“title””Sitefetch”,”url””https//sitefetch.vercel.app/”,”content””[Buy me a Ko-fi]”}}}.content./.content”,”embedding_preview”“[-0.044216092675923, -0.01996812596917152 4, 0.024202141910791397, -0.039143778383731 84, 0.07313912361860275…]”,”embedding_length”384,”created_at””2025-04-03T152336.0832”,”name””Wikipedia”,”lastname””web”}],”total_rows”1} |

Genera representaciones vectoriales (embeddings) y las almacena en una tabla de Supabase.

## Execute Postgres Function

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase para ejecutar la función. | credential |
| Nombre de la función | Nombre de la función Postgres a ejecutar. | hello_world |
| Parámetros | Parámetros a pasar a la función (en JSON). | { “embedding”: […], “match_threshold”: 0.7 } |
| Asignar resultado a variable | Variable donde se almacenará el resultado. {“result”[{“id”450102905661143,”name””Wikipedia”,”lastname””web”,”similarity”1}]} | {var} |

Ejecuta funciones personalizadas definidas en la base de datos Postgres de Supabase.

## Retrieve Documents (Vector Store)

| Parámetro | Descripción | Ejemplo |
|---|---|---|
| Credencial | Credencial de Supabase necesaria. | credential |

| Parámetro | Descripción | Ejemplo |
| --- | --- | --- |
| Credencial de IA | Motor de embeddings a utilizar. | Name |
| Modelo de embedding | Modelo para generar la consulta vectorial. | Name |
| Nombre de la función | Función Postgres usada para la búsqueda de similitud. | hello_world |
| Texto a buscar | Texto base de la búsqueda. | query… |
| Número de resultados | Cantidad de documentos similares a retornar. | 5 |
| Asignar resultado a variable | Variable donde se almacenará el resultado. {“result”[{“id”450102905661143,”name””Wikipedia”,”lastname””web”,”similarity”1}]} | {var} |

Realiza una búsqueda de similitud sobre embeddings almacenados en Supabase.

# Ejemplos de uso

- **Get Table:** Listar todos los clientes almacenados en la tabla “Clients”.
- **Insert Rows:** Agregar un nuevo usuario desde un formulario externo.
- **Generate and Store Embedding:** Procesar contenido web y guardar embeddings para búsquedas vectoriales.
- **Execute Postgres Function:** Invocar funciones personalizadas para cálculos avanzados directamente desde RPA.

# Requisitos

1. Tener una credencial válida de Supabase configurada en Saturn Studio.
2. Conocer la estructura de las tablas, columnas y funciones a utilizar.
3. Para embeddings, contar con una credencial de IA compatible y modelo activo.
4. Si se usa almacenamiento, definir correctamente permisos y límites del bucket.