

Saturn Studio/Rocketbot – File Storage

La sección **File Storage** (almacenamiento de archivos) permite gestionar archivos para el desarrollo de robots dentro de **Saturn Studio**.

Es importante considerar la capacidad máxima de almacenamiento disponible. Dentro del entorno de desarrollo, es necesario dirigirse al módulo **File**, donde se podrá configurar su uso según las necesidades del robot.

Es posible hacer uso de los archivos para diversas funciones, por ejemplo, enviar correos electrónicos (**módulo Gmail**) o enviar mensajes con imágenes al asistente de ChatGPT (**módulo OpenAI**), entre otras opciones.

Se pueden realizar las siguientes acciones:

- Subir archivos.
- Visualizar la cantidad de archivos almacenados.
- Visualizar el almacenamiento utilizado y el almacenamiento máximo disponible.
- Visualizar la información del archivo, incluyendo su nombre, proveedor (usuario), tamaño y fecha de creación.
- Eliminar archivos de manera individual o en su totalidad.
- Consultar las tareas asignadas a cada archivo.

File Storage

Here you can view all the files stored in the system. You can delete files that are no longer needed. Can config your robot to delete files automatically after stopping the robot.

Upload a file

Seleccionar archivo

1
Files

16.94 KIB / **100** MIB
Used storage / Max storage

File Name	Provider	Size	Created	Actions
ticket test.jpg	user	16.94 KIB	in a few seconds	✖ ✔

Files (Desarrollo de Bots)

Para utilizar los archivos previamente cargados en el sistema, es necesario acceder al módulo **Files** y seleccionar el comando **Get Files** (obtener archivos).




En la configuración, se podrá filtrar la búsqueda de archivos por nombre.

Se puede asignar el resultado de la ejecución a una variable, para confirmar la existencia del archivo. Si el archivo existe, la variable asignada contendrá su **ID**, nombre, espacio ocupado en almacenamiento y el proveedor correspondiente.


Get files ✕

Add a description 


 Get files uploaded to the system

{ x }

 Test Command

Filter by name 

ticket test

Assign result to variable 

file

Cancel

Ok

Robot: 1 ✕

Add

Name:

file

Type :



Disable

Group:

Data: [RAW view](#) [JSON view](#) [Table view](#)

```
[{"id":"775e63b67073754a3aa9337b6304e867b87a4b824cb8f851ea7ddc7e7c6989ee","filename":"ticket test.jpg","size":17348,"provider":"user"}]
```

Edit

Cancel

Ejemplo de Desarrollo

Una vez que el archivo ha sido subido correctamente a **File Storage**, durante el desarrollo del bot se deberá acceder al módulo **Files** y seleccionar **Get Files**.

File Storage

Here you can view all the files stored in the system.
You can delete files that are no longer needed. Can config your robot to delete files automatically after stopping the robot.

1
Files

16.94 KIB / 100 MIB
Used storage / Max storage

Upload a file

File Name	Provider	Size	Created	Actions
ticket test.jpg	user	16.94 KIB	in a few seconds	✖ 🔄

- Email IMAP and SMTP
- **Files**
- Get files
- Gemini
- Gmail

```
graph LR
    Start((Start)) --- GetFiles((Get files))
    GetFiles --- Command["{"module_name": "files", "m..."}"]
```

En este proceso:

1. Se establecerá el nombre del archivo como criterio de búsqueda.
2. Se asignará una variable que verificará su existencia.

Get files
✖

Add a description

? Get files uploaded to the system { x } 🔧 Test Command

Filter by name ?

Assign result to variable ?

Cancel
Ok

Para evaluar la presencia del archivo, se utilizará un comando **IF**, que analizará la lógica basada en la variable asignada. Si la variable confirma la existencia del archivo, el flujo de ejecución continuará.

IF
✕

Add a description

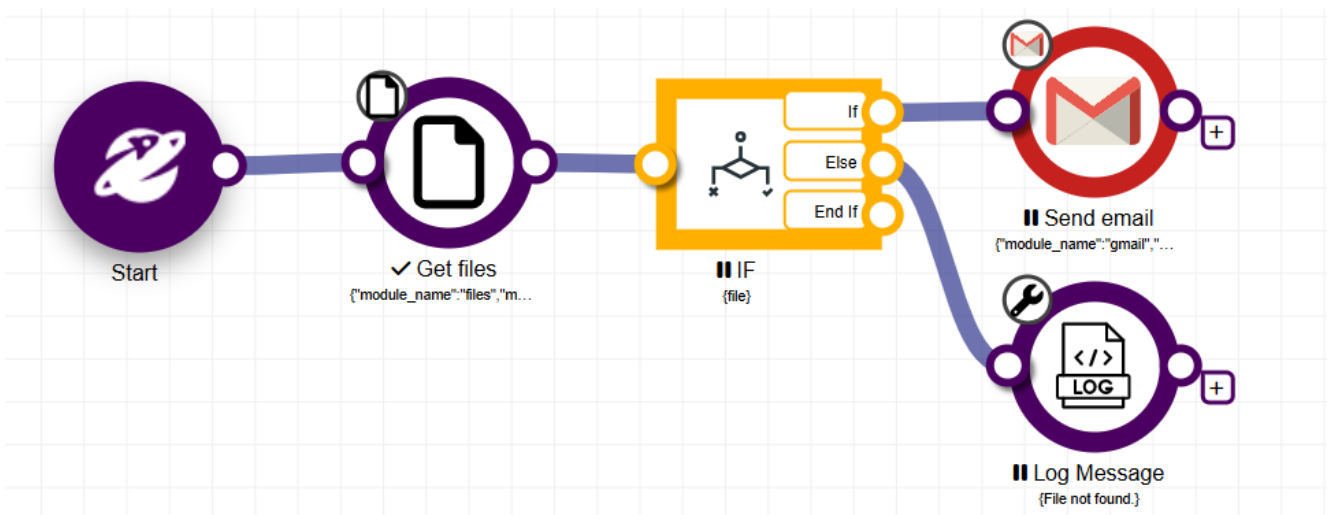
? Evaluate a condition with JS logic {x} Test Command

Javascript Logic

```
{file}
```















Cancel
Ok

La función final de este bot será enviar un correo electrónico con el archivo adjunto. Para ello, se empleará el módulo **Gmail**, seleccionando la opción **Send Email**.



Una vez completados los campos requeridos, en la sección **Attached Files** (archivos adjuntos), se podrán visualizar los archivos contenidos en la variable. Si la carga del archivo es correcta, se procederá a asignar una variable que confirme el envío exitoso del correo.

Message ?

B *I* U        Normal    Sans Serif    

Hello

You can use variables between {} to customize the message for example: Hello {name}

Attached files ?

file 

Files in variable

- ticket test.jpg - USER

Assign result to variable ?

send_file 

Cancel

Ok

Si todo el proceso se ejecuta correctamente, el archivo adjunto será enviado a la dirección de correo electrónico designada.