

Studio Rocketbot – Variables internas

Rocketbot Studio cuenta con variables nativas que puedes utilizar para acceder a información rápidamente. Estas variables pueden ser utilizadas desde cualquier comando y no necesitas crearla en la sección de variables o generar una acción para generarlas.

Variable	Descripción
<code>%rocketbot_last_status%</code>	dict. Almacena el estado del último comando ejecutado. {'status': 'True', 'message': 'setvar ', 'img': "", 'vars': [], 'ifs': [], 'extra': [], 'time': '0.004066944122314453'}
<code>%log_path%</code>	str. Almacena la ruta del archivo log de la ejecución en curso
<code>%base_path%</code>	str. Almacena la ruta de la carpeta de rocketbot
<code>%date%</code>	str. Almacena la fecha y hora actual. El formato es YYYY-MM-DD HH:MM:SS.ms
<code>%day%</code>	str. Retorna el día al momento de usar la variable
<code>%month%</code>	str. Retorna el mes al momento de usar la variable
<code>%year%</code>	int. Retorna el año al momento de usar la variable
<code>%hour%</code>	int. Retorna la hora al momento de usar la variable
<code>%minute%</code>	int. Retorna el minuto al momento de usar la variable
<code>%second%</code>	int. Retorna el segundo al momento de usar la variable
<code>%milisecond%</code>	str. Almacena la parte de los milisegundos de la hora al momento de usar la variable
<code>%machine%</code>	str. Almacena el nombre de la máquina donde se está ejecutando el proceso
<code>%tab%</code>	str. Contiene un tab
<code>%newline%</code>	str. Contiene un salto de línea o nueva línea
<code>%enter%</code>	str. Contiene un salto de línea o nueva línea
<code>%osname%</code>	str. Almacena el nombre del sistema operativo
<code>%username%</code>	str. Almacena el nombre de usuario de la computadora

Variable	Descripción
<code>%db_path%</code>	str. Almacena la ruta de la base de datos
<code>%project_path%</code>	str. Almacena la ruta de la carpeta de la base de datos. Para versiones superior a la 2024, corresponde a la carpeta del proyecto
<code>%robot_name%</code>	str. Contiene el nombre del robot
<code>%production%</code>	bool. Retorna True si la ejecución es en producción. En caso contrario, retorna False

[Studio Rocketbot – Iniciar un proyecto](#)

Este documento explica como crear un proyecto en Studio que pueda ser ejecutado en cualquier ambiente que cuente con una licencia de Rocketbot Studio.

Nuevo proyecto

Para crear un nuevo proyecto, desde Studio ve a Inicio y haz click en el botón Nuevo Proyecto y verás el siguiente modal.

New project ✕

Create new project

Create a new project to start working with Rocketbot

Main Robot

Author

Description

Select Folder

File path

i











Save Cancel

En el modal de nuevo proyecto, completa los siguientes campos:

- Robot principal: Especifica un nombre para tu robot principal y tu proyecto. Este nombre debe ser único para cada proyecto, ya que se utilizará para crear una carpeta con este nombre
- Descripción: Especifica una descripción que resuma lo que quieres hacer con este proyecto.
- Author: Indica el nombre del autor del proyecto
- Seleccionar carpeta: Selecciona la ubicación en la que quieres crear el proyecto. La ubicación predeterminada donde se crean los proyectos es en %USERPROFILE%\Documents\rocketbot

Estructura de carpetas y archivos

Al completar los datos y crear el proyecto, se crearán los siguientes archivos y carpetas:

Name	Date modified	Type	Size
 robot.db	11/13/2023 10:09	ANSYS 2021 R1 .d...	48 KB
 README.md	10/2/2023 16:44	Markdown Source...	1 KB
 package.json	10/2/2023 16:44	JSON Source File	1 KB
 nombre_proyecto.bat	10/2/2023 16:44	Windows Batch File	1 KB
 screenshot	10/2/2023 16:44	File folder	
 robots	10/2/2023 16:50	File folder	
 resources	10/2/2023 16:44	File folder	
 modules	10/2/2023 16:44	File folder	
 logs	11/13/2023 10:06	File folder	
 .git	11/13/2023 10:09	File folder	

- robot.db: Base de datos SQLite creada por defecto donde se almacenará tu robot. En esta base de datos contendrá todos los subrobots que sean creados y podrás llamarlos desde el comando [Ejecuta otro script Rocketbot](#)
- README.md: Archivo markdown con el nombre del proyecto y la descripción. Puedes editarlo para agregar una documentación acerca de como ejecutar tu robot
- package.json: Contiene información acerca de tu robot. Este archivo contiene datos como: nombre, descripción, versión, autor, licencia, nombre del robot principal.

□ Puedes incluir además las versiones de los módulos que utilice tu robot.

```
{
  "name": "Rocket",
  "description": "",
  "version": "1.0.0",
  "main": "run_main",
  "author": "",
  "license": "MIT",
  "modules": [
    {"name": "AdvancedExcel", "version": "13.2.1"}
  ]
}
```

}

- Archivo batch: Este archivo contiene la instrucción de consola para ejecutar tu robot. Está asociado al ambiente y rutas donde se creó el proyecto.

```
cd C:\rocketbot
rocketbot.exe -start=Rocket -
db="%USERPROFILE%\Documents\rocketbot\Rocket\robot.db"
```

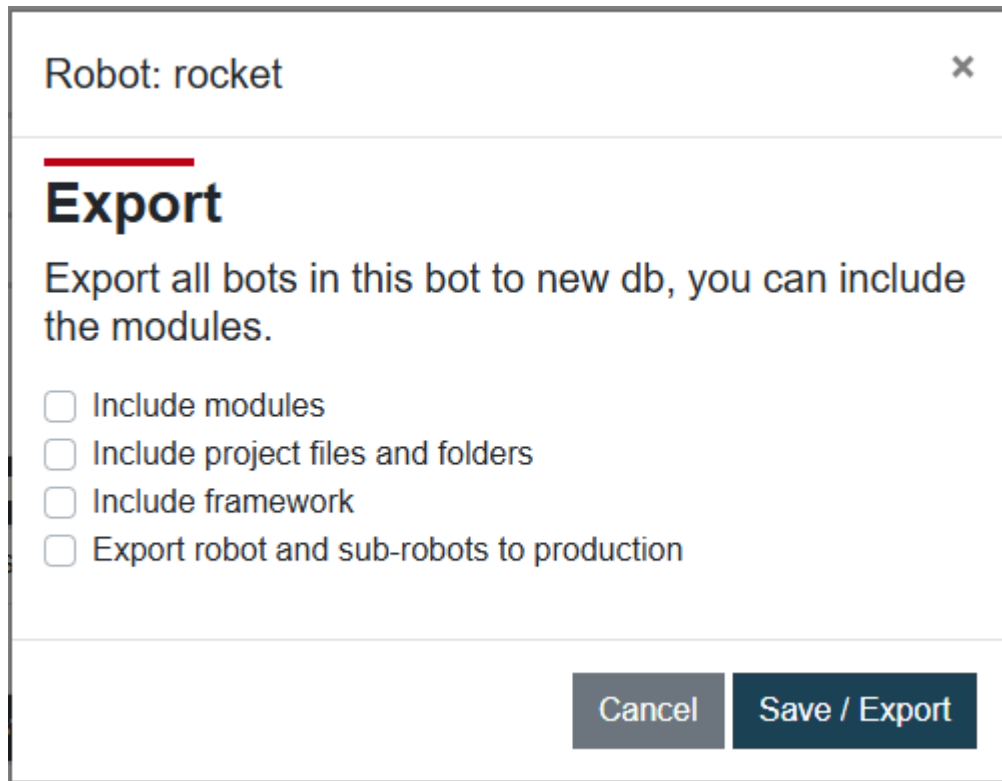
- screenshot: Carpeta donde podrás almacenar las capturas de pantallas que realiza tu robot
- robots: Carpeta con las base de datos de robots como servicio que utiliza tu robot. En esta carpeta puedes agregar robots de otros proyectos para ser utilizados en tu robot
- Resources: Carpeta donde deberás alojar los archivos que tu robot necesita para trabajar y los archivos que genera.

□ Puedes agregar subcarpetas si lo necesitas, como carpeta para descargas, plantillas, archivos de configuración, etc

- modules: Carpeta con los módulos que utilizará tu robot. Esta carpeta te permite tener diferentes versiones de los módulos para diferentes proyectos.
- logs: Carpeta donde se almacenarán los logs generados por la ejecución de cada comando de tu robot. Para que los robots se almacenen en esta carpeta, debes indicarlo en la configuración de tu robot, en caso contrario, los almacenará en la carpeta de rocketbot.
- .git: Carpeta generada por git al generar un nuevo repositorio. Esto te permitirá usar desde studio los comandos de git para el control de versiones

Exportar un proyecto

Para exportar un proyecto debes ir a la pestaña Robot del ambiente de desarrollo y dar click en el botón **Exportar robots a Db**. Dependiendo de lo que necesites, deberás marcar alguno de las casillas:



The screenshot shows a dialog box titled "Robot: rocket" with a close button (X) in the top right corner. Below the title bar, the word "Export" is displayed in a large, bold font, underlined with a red line. Below this, the text reads: "Export all bots in this bot to new db, you can include the modules." There are four checkboxes listed below: "Include modules", "Include project files and folders", "Include framework", and "Export robot and sub-robots to production". At the bottom right of the dialog, there are two buttons: "Cancel" and "Save / Export".

- Sin marcar casillas: Si no se marca ninguna casilla, se creará solo una base de datos de tu robot con la última versión de cada uno de los subrobots.
- Incluir módulos: Al marcar esta casilla, se creará una carpeta comprimida con todos los módulos que utiliza tu robot
- Incluir archivos y carpetas del proyecto: Al marcar esta casilla, se creará una copia de la carpeta de tu proyecto en la ruta indicada luego de dar click en **Guardar/Exportar**
- Incluir Framework: Al marcar esta casilla, se exportará además una versión reducida de rocketbot que solo permite ejecutar los robots. Con esta versión reducida, puedes ejecutar tu robot sin permitir entrar a Studio
- Exportar robots a producción: Al marcar esta casilla, la base de datos creada se encriptará y no se podrá ver el contenido del robot desde Studio.

⚠ Si marcas la casilla Exportar robots a producción, no selecciones una base de datos existente. Al encriptar los datos, no podrás volver a ver ni editar el contenido y podría perder tu robot

Diferencia con versiones anteriores

Esta documentación está basada en las funcionalidades de Rocketbot Studio 2024.05.28 o superior. Versiones tienen diferencias en la creación de proyectos y se deben configurar manualmente para replicar a las últimas versiones

Versión 2023

La estructura de carpetas al crear un proyecto fue integrada en la versión Rocketbot Studio 2023.03.30 pero con ligeras diferencias

Carpetas y archivos

package.json	Este archivo no es generado en la versión 2023
modules	Esta carpeta se utiliza para almacenar las versiones de los módulos, pero el robot no los utiliza en la ejecución
logs	Esta carpeta era utilizada para almacenar los logs generados intencionalmente con el robot. No se almacenaban los logs generados por la actividad del robot
robots	Esta carpeta era utilizada para almacenar copias de seguridad de la base de datos original. No se podían utilizar los robots como servicios

Menú de exportar

La versión 2023 no cuenta con la opción de exportar incluyendo archivos y carpetas del proyecto ni incluir el framework

Versión 2020 e inferior

























Las versiones inferior a la 2023 no cuentan con la funcionalidad de crear proyecto. Para estas versiones, se debe crear una base de datos y luego generar la estructura de carpetas de forma manual.

Para exportar una base de datos, puedes elegir entre dos botones en la pestaña robot. Exportar robots y subrobots, y exportar robots y subrobots a producción

[Studio Rocketbot – Pestaña variables](#)

La pestaña variables en Studio te permite crear, modificar y organizar las variables del robot.

Acciones

#	Name	Data	
<input type="checkbox"/>	1 path	C:/Users/rocket/Documents/data.xlsx	    
Credentials			 
<input type="checkbox"/>	1 password	*****	    
<input type="checkbox"/>	2 user	user@rocketmail.com	    
Keys			 
<input type="checkbox"/>	1 secret	gAAAAABmWLVHajjfoh7swcQ9ouoQpOKWsNbPjzEA2-sll0BmW-... Expand (20 more)	    

Secciones

Descripción

Desactivar ()	Indica si la variable está activa o no. Si la casilla está marcada, el robot asumirá que la variable no debe ser usada
Nombre	Nombre de la variable
Datos	Contenido de la variable en el momento que se observa
Limpiar ()	Vacía o vuelve al valor por defecto de la variable
Eliminar ()	Elimina la variable de la lista de variables pero no de los comandos que la utilicen
Editar ()	Muestra el modal para editar la variable
Copiar ()	Copia al portapapeles el contenido de la variable o las variables si hay más de una seleccionada
Exponer ()	Convierte la variable a una variable expuesta o variable de entrada. Para más información, ir Expose

Crear variables

Para crear una nueva variable para tu robot, debes dar click en el botón (+) verde y completar los datos de la ventana emergente

Name

Type

Variable category

 Default value

Data

 View raw
 View JSON

Add

Cancel

Campo**Descripción**

Nombre	<p>Nombre único de la variable. Obligatorio. El nombre no puede repetirse en el mismo robot y solo puede contener caracteres alfanuméricos o guiones</p>
Tipo	<p>Permite elegir como manejar la visualización de la variable. Opcional. Las opciones disponibles son:</p> <ul style="list-style-type: none"> – General – Password: Modificará el input a tipo password para ocultar los datos – File: Modificará el input a file para poder seleccionar archivos fácilmente – Folder: Modificará el input a tipo folder para poder seleccionar carpetas – Encrypted: Encriptará los datos para que no vuelvan a ser visibles
Categoría	<p>Categoría en donde se agruparán las variables. Opcional. Si el nombre de la categoría no existe, se creará un nuevo grupo</p>
Valor Default	<p>Valor por defecto que se asignará cuando se limpien los datos. Opcional. Si la casilla no está marcada, el valor por defecto será vacío Cuando se de click en alguno de los botones para limpiar los datos, en lugar de vaciar la variable, se asignará el valor plano por defecto, si la variable es de tipo encrypted, no la encriptará</p>

Campo

Descripción

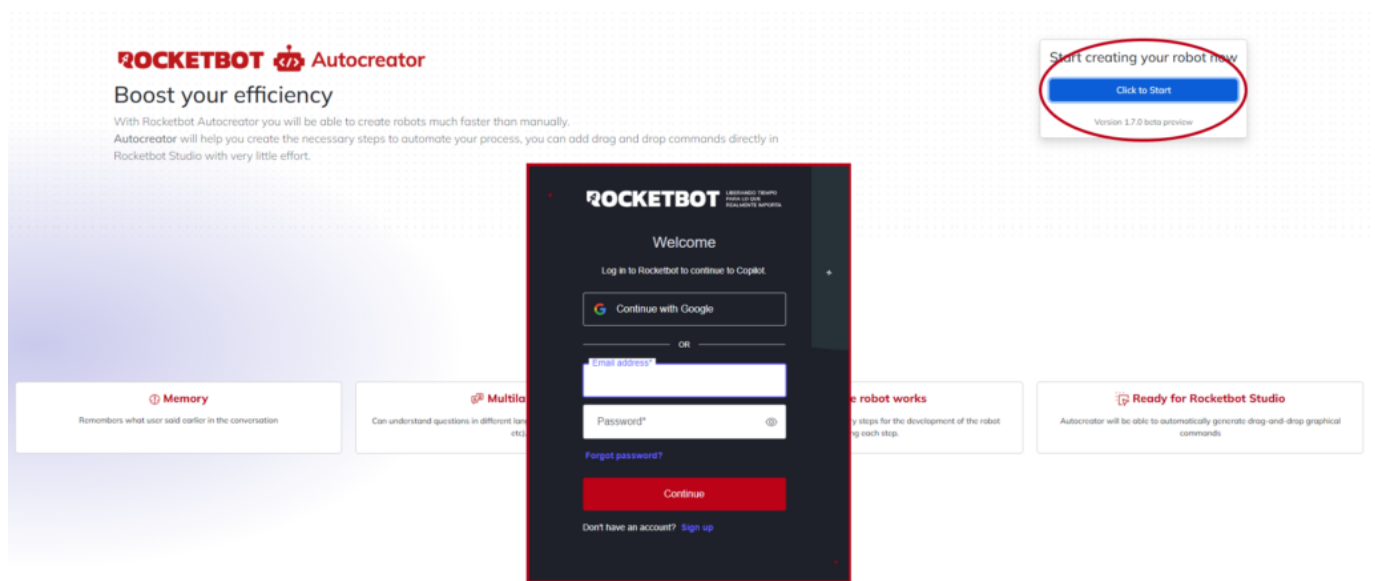
Dato	El valor que contiene la variable en el momento de crearse. Opcional. A diferencia del campo default, acá se ve el valor de la variable en un estado puntual y si algún comando la modifica, no puede volver al estado inicial.
------	--

AutoCreator

AutoCreator es un asistente de Rocketbot para la creación de Robots para la automatización de procesos y Formularios para la plataforma Rocketbot Xperience.

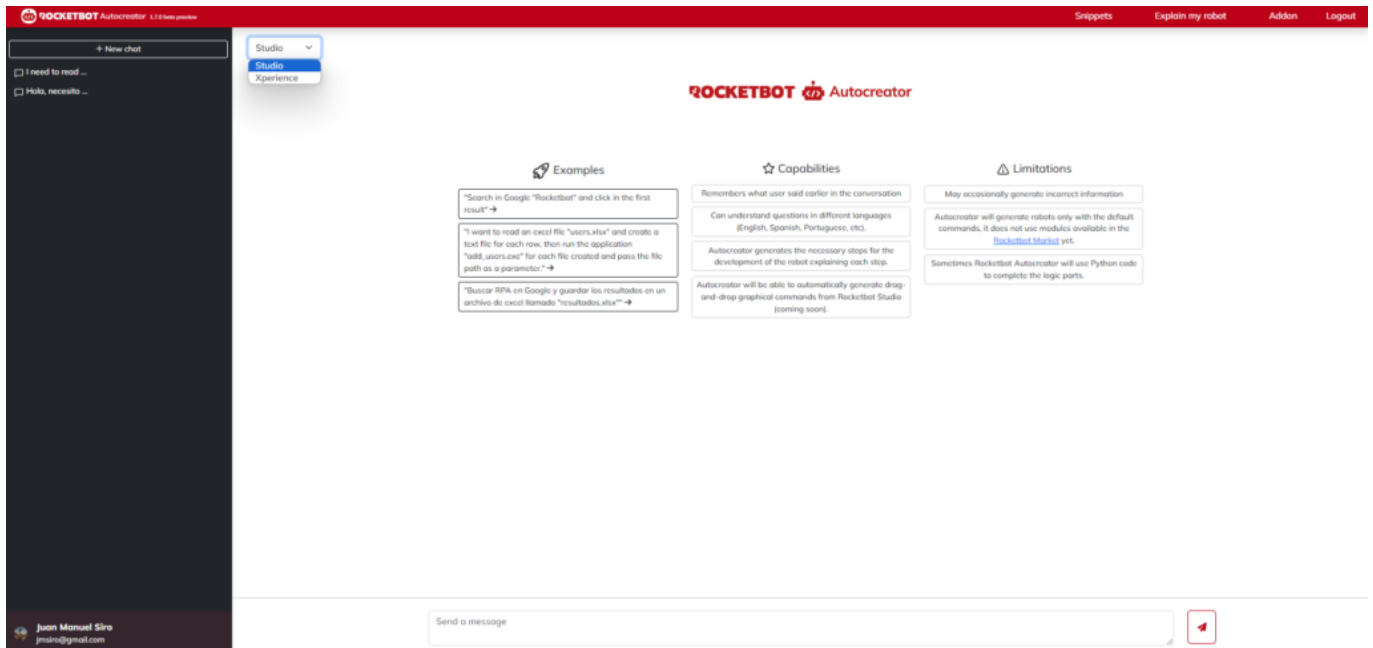
Para ingresar existen dos maneras:

- Directamente desde el link <https://copilot.rocketbot.com/#/>
- Mediante el Addon correspondiente desde Rocketbot Studio

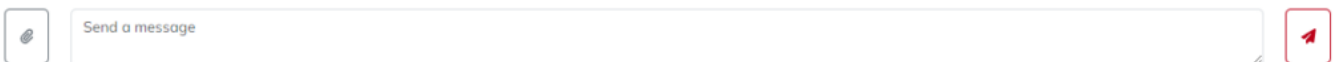


Al ingresar realizar el login desde el botón *Click to Start*.

Una vez realizado el login, será redirigido al menú de chats. Allí puede elegir la plataforma con la cual desea que la IA lo asista: **Studio** o **Xperience**.

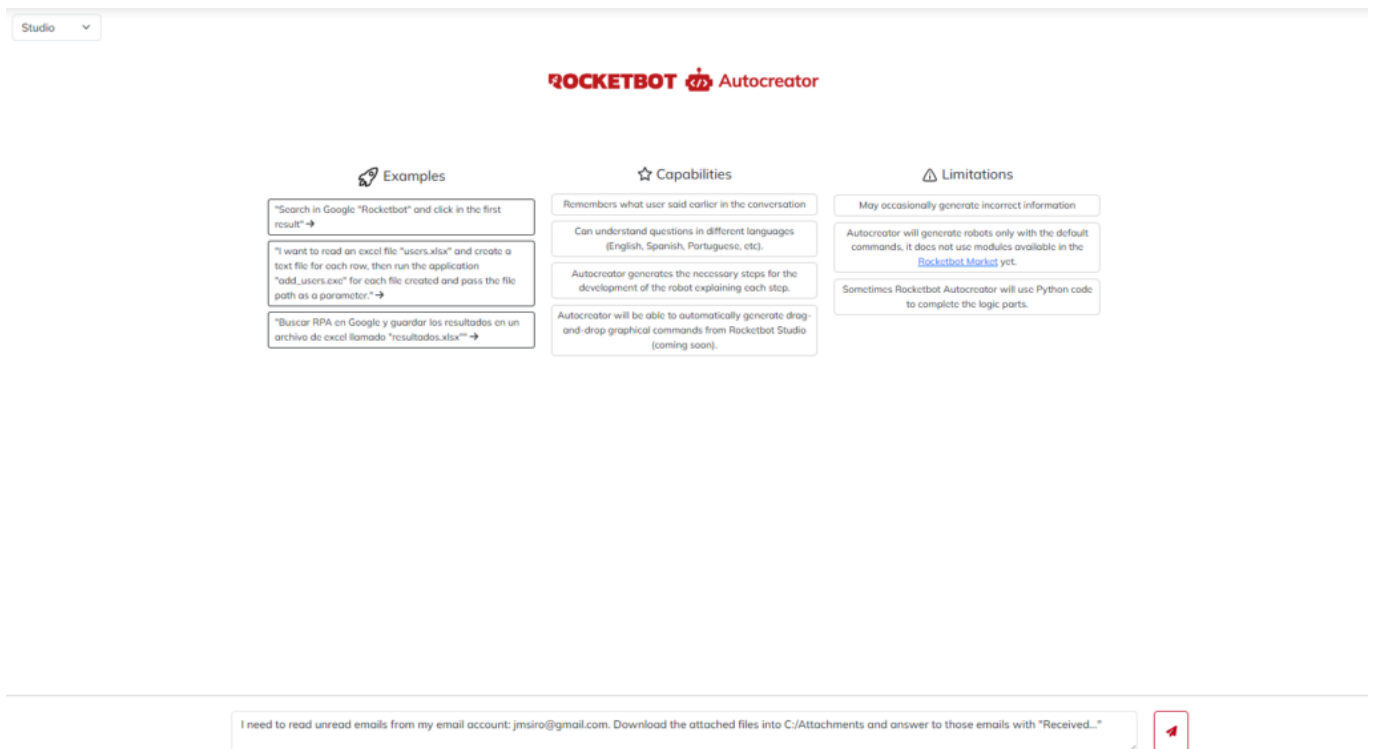


Si selecciona **Xperience**, a la izquierda del recuadro del chat, se habilitará el botón para cargar la imagen sobre la cual la IA deberá basarse para crear el formulario.



Crear un Robot – [REDACTED] :

Seleccionar **Studio** y escribir la solicitud.



Enviar la consulta y aguardar la respuesta de la IA.

Studio

ROCKETBOT Autocreator

Examples

- "Search in Google "Rocketbot" and click in the first result" →
- "I want to read an excel file "users.xlsx" and create a text file for each row, then run the application "add_users.exe" for each file created and pass the file path as a parameter." →
- "Buscar RPA en Google y guardar los resultados en un archivo de excel llamado "resultados.xlsx" →

Capabilities

- Remembers what user said earlier in the conversation
- Can understand questions in different languages (English, Spanish, Portuguese, etc).
- Autocreator generates the necessary steps for the development of the robot explaining each step.
- Autocreator will be able to automatically generate drag-and-drop graphical commands from Rocketbot Studio (coming soon).

Limitations

- May occasionally generate incorrect information
- Autocreator will generate robots only with the default commands, it does not use modules available in the [Rocketbot Market](#) yet.
- Sometimes Rocketbot Autocreator will use Python code to complete the logic parts.

WAITING FOR RESPONSE...

I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and answer to those emails with "Received..."

La IA responderá con el listado de pasos que interpreta necesarios para cumplir con la solicitud. El usuario dispondrá de 2 opciones:

- Generar el robot tomando los pasos propuestos como base.
- Solicitar a la IA que genere una nueva serie de pasos.

Studio

I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and answer to those emails with "Received..."

I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and answer to those emails with "Received..."

Proceso 📄 🗑️

Paso a paso sugerido

Entorno	Detalle
Gmail	Configurar conexión con la cuenta de correo electrónico: jmsiro@gmail.com
Gmail	Leer correos electrónicos no leídos
Gmail	Descargar los archivos adjuntos de los correos no leídos en la carpeta "C:/Attachments"
Gmail	Responder a cada correo no leído con el mensaje "Received..."


Generate bot
Regenerate steps


Regenerate

Send a message

Seleccionar la opción deseada y esperar la respuesta de la IA.


I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and ans

 I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and answer to those ema

< 1/1 >  **Proceso**

Paso a paso sugerido


Entorno	Detalle
Gmail	Configurar conexión con la cuenta de correo electrónico jmsiro@gmail.com
Gmail	Leer correos electrónicos no leídos
Gmail	Descargar los archivos adjuntos de los correos no leídos en la carpeta "C:/Attachments"
Gmail	Responder a cada correo no leído con el mensaje


 **Regenerate steps**

SENDING DATA...

La IA responderá con el código del robot generado.

I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and answer to those emails with "Received..."

 I need to read unread emails from my email account: jmsiro@gmail.com. Download the attached files into C:/Attachments and answer to those emails with "Received..."

< 1/1 >  **Proceso**

Paso a paso sugerido

Entorno	Detalle
Gmail	Configurar conexión con la cuenta de correo electrónico jmsiro@gmail.com
Gmail	Leer correos electrónicos no leídos
Gmail	Descargar los archivos adjuntos de los correos no leídos en la carpeta "C:/Attachments"
Gmail	Responder a cada correo no leído con el mensaje "Received..."


Here is the Rocketbot code to read unread emails from the specified email account, download the attachments to a local folder, and respond to those emails with a message:

Paso	Descripción
1	Configure email account for IMAP and SMTP.
2	Get all unread email IDs.
3	Check if there are unread emails.
4	Iterate over each unread email.
5	Read the email and download attachments.
6	Send a reply with "Received..."
7	Save status of email reply.

Code Structure Generate JSON Install addon and start from RocketbotStudio

```
robot_name: email_reader_responder
description: Read unread emails, download attachments, and respond with "Received..."

vars = [
  "email_user": "jmsiro@gmail.com",
  "email_password": "*****", # Replace with the actual password
  "attachments_path": "C:/Attachments",
  "unread_emails": [],
  "email_id": "",
  "reply_message": "Received...",
  "send_status": false
]
```



El usuario podrá descargar el robot en formato JSON o agregarlo directamente a Rocketbot Studio.

```
#robot_name: email_reader_responder
#description: Read unread emails, download attachments, and respond with "Received..."

vars = {
  'email_user': 'jmsiro@gmail.com',
  'email_password': '*****', # Replace with the actual password
  'attachments_path': 'C:/Attachments',
  'unread_emails': [],
  'email_id': '',
  'reply_message': 'Received...',
  'send_status': False
}

# Configure email account for IMAP and SMTP
email.configemail(smtp='smtp.gmail.com', smtp_port=587, imap='imap.gmail.com', imap_port=993, ssl=True, imap_ssl=True, user-vars['email_user'], password-vars['email_password'])

# Get all unread email IDs
vars['unread_emails'] = email.getunreademail(command='UNSEEN')

# Check if there are unread emails
if len(vars['unread_emails']) > 0:

  # Iterate over each unread email
  for vars['email_id'] in vars['unread_emails']:

    # Read the email and download attachments
    email.reademail(id=vars['email_id'], file=vars['attachments_path'])

    # Send a reply with "Received..."
    vars['send_status'] = email.sendemail(to=vars['email_user'], subject='Re: ', msg=vars['reply_message'], file='')

    # Check if email was sent successfully and write status
    if vars['send_status']:
      # Status can be logged or used as needed
      pass
    else:
      # Handle failed email send, if necessary
      pass

else:
  # If no unread emails, log or perform other actions as needed
  pass
```

Para enviarlo directo a Rocketbot Studio, deberá abrir AutoCreator desde el addon correspondiente.

Con **Create bot** la IA responderá la solicitud de manera independiente a los comandos que se posean en el robot abierto en Rocketbot Studio.

Con **Add commands** la IA responderá la solicitud teniendo en cuenta los comandos que ya posea el robot desde donde se llame AutoCreator.



Rocketbot AutoCreator has opened
in a new tab.



Create your bot with AutoCreator.

Create bot

Add commands to your bot.

Add commands

```
Code Structure Generate JSON Create Bot

#robot_name: email_reader_responder
#description: Read unread emails, download attachments, and respond with "Received..."

vars - {
  'email_user': 'jmsiro@gmail.com',
  'email_password': '*****', # Replace with the actual password
  'attachments_path': 'C:/Attachments',
  'unread_emails': [],
  'email_id': '',
  'reply_message': 'Received...',
  'send_status': false
}

# Configure email account for IMAP and SMTP
email.configemail(setp='smtp.gmail.com', setp_port=587, imap='imap.gmail.com', imap_port=993, ssl=True, imap_ssl=True, user=vars['email_user'], password=vars['email_password'])

# Get all unread email IDs
vars['unread_emails'] = email.getunreademail(command='UNSEEN')

# Check if there are unread emails
if len(vars['unread_emails']) > 0:

  # Iterate over each unread email
  for vars['email_id'] in vars['unread_emails']:

    # Read the email and download attachments
    email.reademail(id=vars['email_id'], file=vars['attachments_path'])

    # Send a reply with "Received..."
    vars['send_status'] = email.sendemail(to=vars['email_user'], subject='Re: ', msg=vars['reply_message'], file='')

    # Check if email was sent successfully and write status
    if vars['send_status']:
      # Status can be logged or used as needed
      pass
    else:
      # Handle failed email send, if necessary
      pass

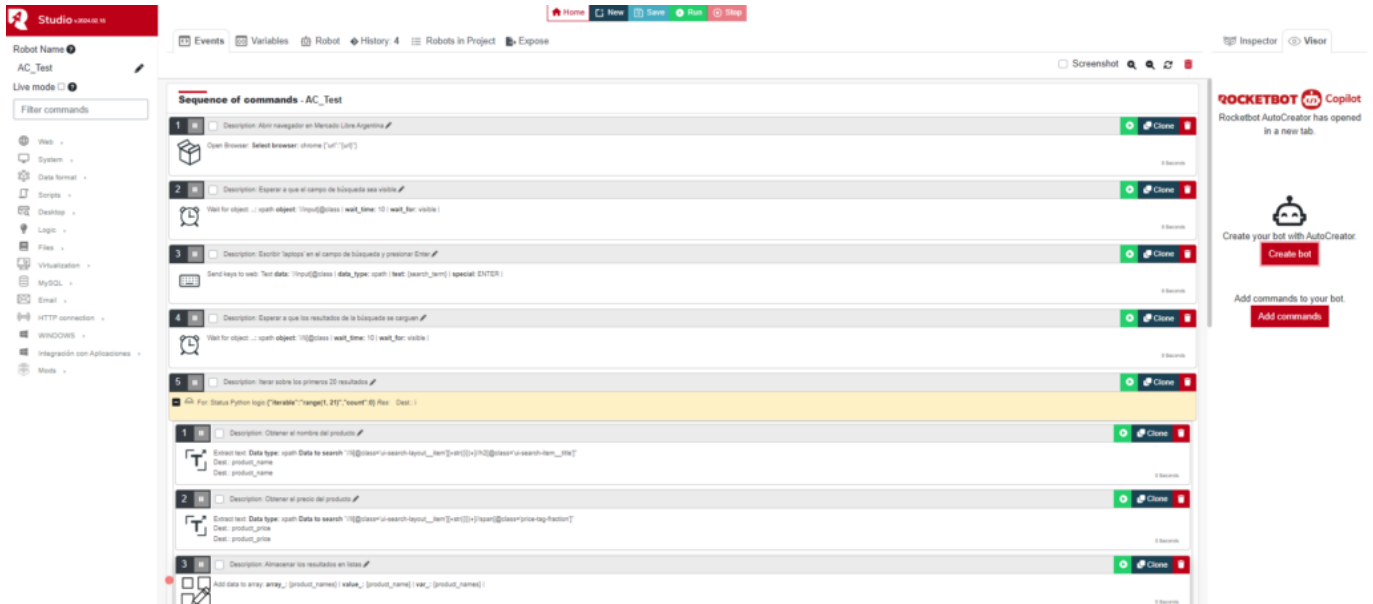
else:
  # If no unread emails, log or perform other actions as needed
  pass
```

Presionar **Create Bot** para agregar los comandos en el robot abierto en Studio. Una vez completada la operación, un Alerta aparecerá en el navegador indicando que los comandos han sido agregados.

Code added to Rocketbot Studio

Aceptar


Al volver a Rocketbot Studio se encontrará con los comandos y las variables generados por la IA creadas



Crear un Formulario Xperience – Paso a paso:

Seleccionar **Xperience**, presionar el botón con el clip para seleccionar la imagen y enviar la solicitud.

Xperience ▾

ROCKETBOT  Autocreator

Examples

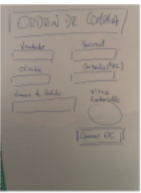
- "Search in Google "Rocketbot" and click in the first result" →
- "I want to read an excel file "users.xlsx" and create a text file for each row, then run the application "add_users.exe" for each file created and pass the file path as a parameter." →
- "Buscar RPA en Google y guardar los resultados en un archivo de excel llamado "resultados.xlsx" →

Capabilities

- Remembers what user said earlier in the conversation
- Can understand questions in different languages (English, Spanish, Portuguese, etc).
- Autocreator generates the necessary steps for the development of the robot explaining each step.
- Autocreator will be able to automatically generate drag-and-drop graphical commands from Rocketbot Studio (coming soon).

Limitations


- May occasionally generate incorrect information
- Autocreator will generate robots only with the default commands, it does not use modules available in the [Rocketbot Market](#) yet.
- Sometimes Rocketbot Autocreator will use Python code to complete the logic parts.



 Could you create a Xperience form from the image I'm sending you? 

Enviar la solicitud y aguardar la respuesta de la IA.

Xperience ▾

ROCKETBOT  Autocreator

Examples

- "Search in Google "Rocketbot" and click in the first result" →
- "I want to read an excel file "users.xlsx" and create a text file for each row, then run the application "add_users.exe" for each file created and pass the file path as a parameter." →
- "Buscar RPA en Google y guardar los resultados en un archivo de excel llamado "resultados.xlsx" →

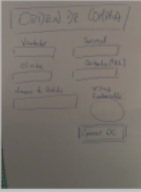
Capabilities

- Remembers what user said earlier in the conversation
- Can understand questions in different languages (English, Spanish, Portuguese, etc).
- Autocreator generates the necessary steps for the development of the robot explaining each step.
- Autocreator will be able to automatically generate drag-and-drop graphical commands from Rocketbot Studio (coming soon).

Limitations

- May occasionally generate incorrect information
- Autocreator will generate robots only with the default commands, it does not use modules available in the [Rocketbot Market](#) yet.
- Sometimes Rocketbot Autocreator will use Python code to complete the logic parts.

AI IS WORKING...



La IA responderá con el código del formulario generado.

☐ Could you create a Xperience form from the image I'm sending you?

👤 Could you create a Xperience form from the image I'm sending you?

```
< 1 / 1 > Code Download JSON
```

```
{
  "form": [
    {
      "type": "title",
      "id": "id_title",
      "container_class": "col-md-12",
      "label": "ORDEN DE COMPRA",
      "class": "text-center"
    },
    {
      "type": "text",
      "id": "id_vendedor",
      "container_class": "col-md-6",
      "label": "Vendedor"
    },
    {
      "type": "text",
      "id": "id_sucursal",
      "container_class": "col-md-6",
      "label": "Sucursal"
    }
  ]
}
```

Descargar **JSON** y crear un nuevo formulario en el **Orquestador** desde archivo.


Create a new Form

Create an amazing form to collect data easily.

Form name:
testAutoCreator

Start from scratch
 Import form from file
 Based on other form (coming soon)

form.json



Crear formulario y publicar.

ORDEN DE COMPRA

Vendedor

Sucursal

Cliente

Contacto (TEL)

Número de Pedido

Firma Responsable

Submit Settings


Submit button text:

Redirect to after submit:

Como recibir información desde un robot en formularios Xperience

La funcionalidad principal de [Xperience](#) es enviar los datos ingresados por el usuario y enviarlos al robot para que utilice esa información, pero algunas veces necesitamos que el robot sea quien envíe información de vuelta al formulario para mostrar información al usuario.

En el formulario

1. Crear un formulario o ir a un formulario existente
2. Habilitar la opción **Send API**
3. En el menú Javascript, agregar la siguiente librería → [xperience.js](#)

4. En la consola de Javascript agregar lo siguiente

```
rocket.on('start', (xperience, data) => {  
  console.log('DATA: ', data) // esta línea te permite ver el resultado en la  
  consola del formulario  
  //Acá el código inicial, lo que envía el robot la primera vez  
})
```

```
rocket.on('change', (xperience, res) => {  
  if (res.success) {
```

```
    console.log('RES: ',res) // esta línea te permite ver el resultado en la
    consola del formulario
    //Acá la información que estará enviando el robot posteriormente
  }
});
```

En el robot

1. Crear una variable llamada `xperience`. Esta variable la llenará el comando **Get Form queue data** del módulo [Xperience](#)

2. Utilizar el comando **Return Message to Xperience** para enviar un mensaje de vuelta al formulario

□ Puedes enviar cualquier tipo de dato permitido por Rocketbot y este se enviará como un string.

Es importante hacer un `JSON.parse` desde el formulario al recibir los datos.

```
result = JSON.parse(res.data.replaceAll('"', ''));
```

3. Utilizar el comando **Send File to Xperience** para enviar un archivo y alojarlo en Xperience

□ Puedes obtener el archivo mediante postman con el siguiente endpoint de lectura:

```
GET: /api/form/extraFile/{xperience}/{token}
```

Siendo `{xperience}` la variable generada en el punto nro 1, y `{token}` el resultado del comando

Send File to Xperience

Ejemplo

El ejemplo te permite mostrar una ventana de alerta con los datos retornados desde el robot

□ Para que te funcione el código, tienes que agregar esta url a los CDN:

```
//cdn.jsdelivr.net/npm/sweetalert2@11
```

```
rocket.on('change', (xperience, res) => {
```

```
if (res.success) {
    console.log('RES: ',res) // esta línea te permite ver el resultado en la
    consola del formulario
    if(res.data){
        swal.fire({
            title: 'Estos son los datos enviados desde el robot ' + res.data,
            icon: 'info',
        });
    }
}
}else{
    swal.fire({
        title: 'Ocurrió un error al solicitar datos',
        icon: 'error'
    });
}
});
```

[Rocketbot Addon: Remote Desktop Helper](#)

Cree un archivo .rdp para conectarse a un Escritorio remoto automáticamente

Un problema común, que suele presentarse al enviar a producción un bot en el servidor de tu cliente, es que el mismo por políticas de IT de la empresa no permite mantener la sesión al desconectarse. Aun cuando se utilice realice la desconexión vía TSCON (Documentación: [RPA en Servidores Virtuales – Rocketbot Docs](#)).

Como solución a este inconveniente y para que no sea necesario intervenir manualmente en la apertura del ambiente para que el/los bot/os programados puedan ejecutarse correctamente, se puede crear un nuevo usuario en el mismo ambiente (que no este sujeto a las políticas del ambiente productivo) que solo cumpla la función de ejecutar un archivo .rdp que se conecte al ambiente donde Rocketbot debe ejecutarse de manera programada. Al abrir una sesión mediante RDP se posibilita la ejecución de los bots con normalidad.

Para facilitar la creación de este archivo .rdp Rocketbot creo la herramienta RDP Helper.

Antes de utilizar la herramienta:

Inicie sesión en el escritorio remoto con un usuario con privilegios administrativos. Abra una línea de comando y escriba *control userpasswords2*

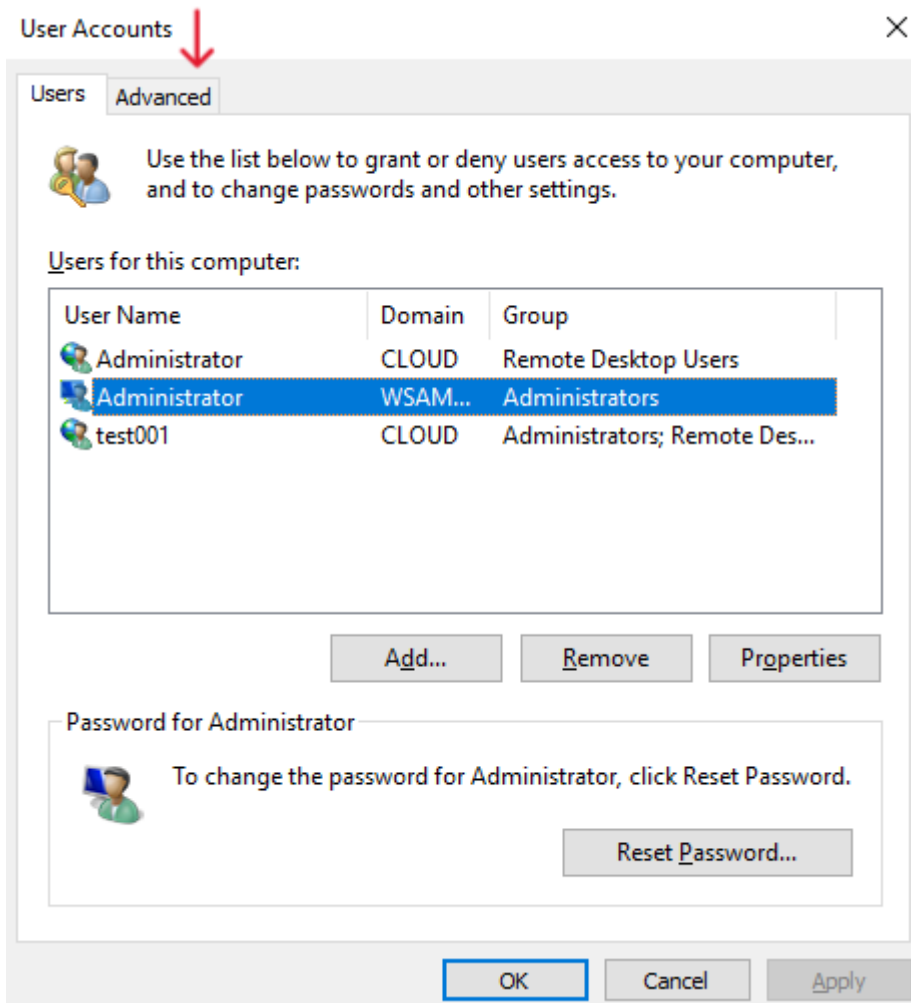
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

D:\Users\testqa>control userpasswords2

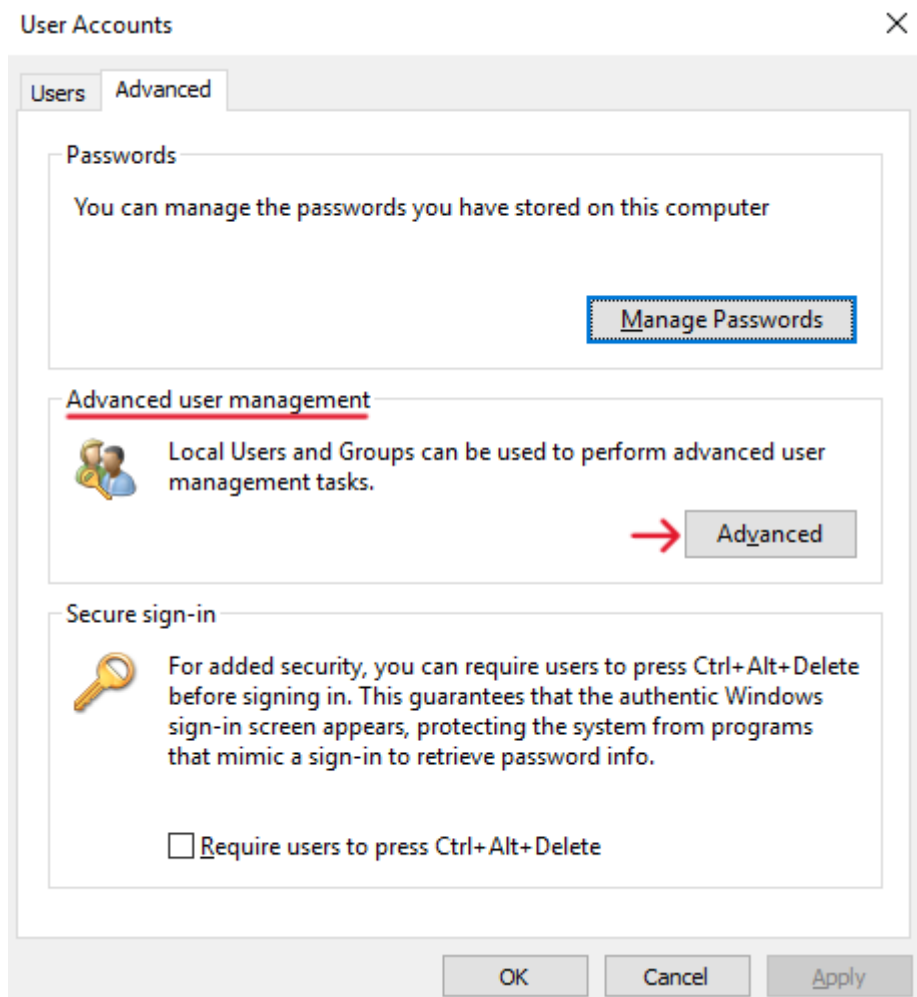
D:\Users\testqa>
```

Ventana de *Símbolo del sistema* con comando para abrir *Cuentas de usuario*

Abra la pestaña **Avanzado** y en **Administración avanzada de usuarios** haga clic en el botón **Avanzado**.

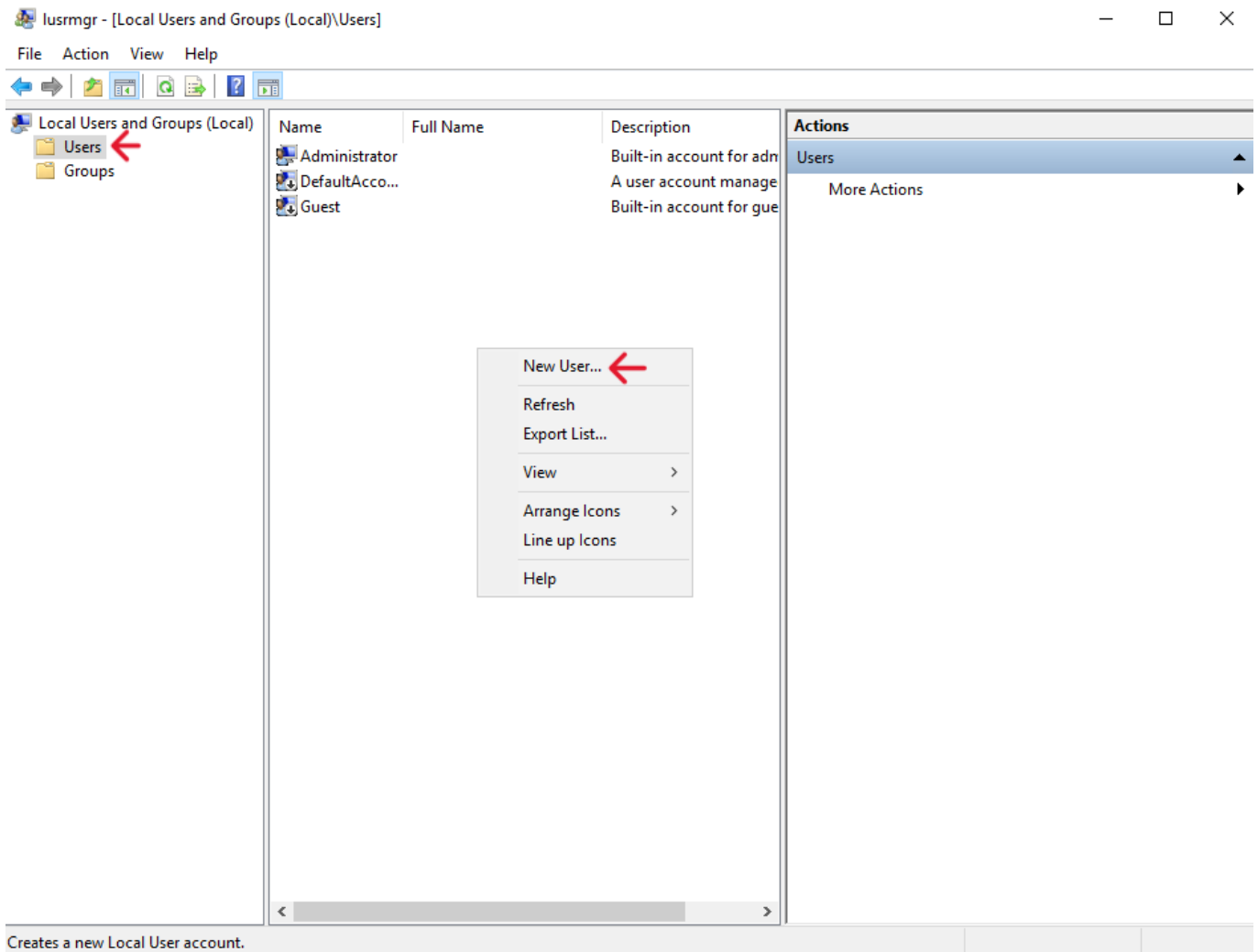


Ventana *Cuentas de usuario*



Ventana *Cuentas de usuario* – Pestaña *Avanzado*

Seleccione **Usuarios**, haga clic derecho en el panel central y seleccione **Nuevo usuario**.



Creates a new Local User account.

Ventana *Usuarios y grupos locales*

Complete el formulario con el *nombre de usuario* y la *contraseña* y marque la opción **La contraseña nunca caduca**.

User name: user

Full name:

Description: RDP runner

Password: ●●●●

Confirm password: ●●●●

User must change password at next logon

User cannot change password

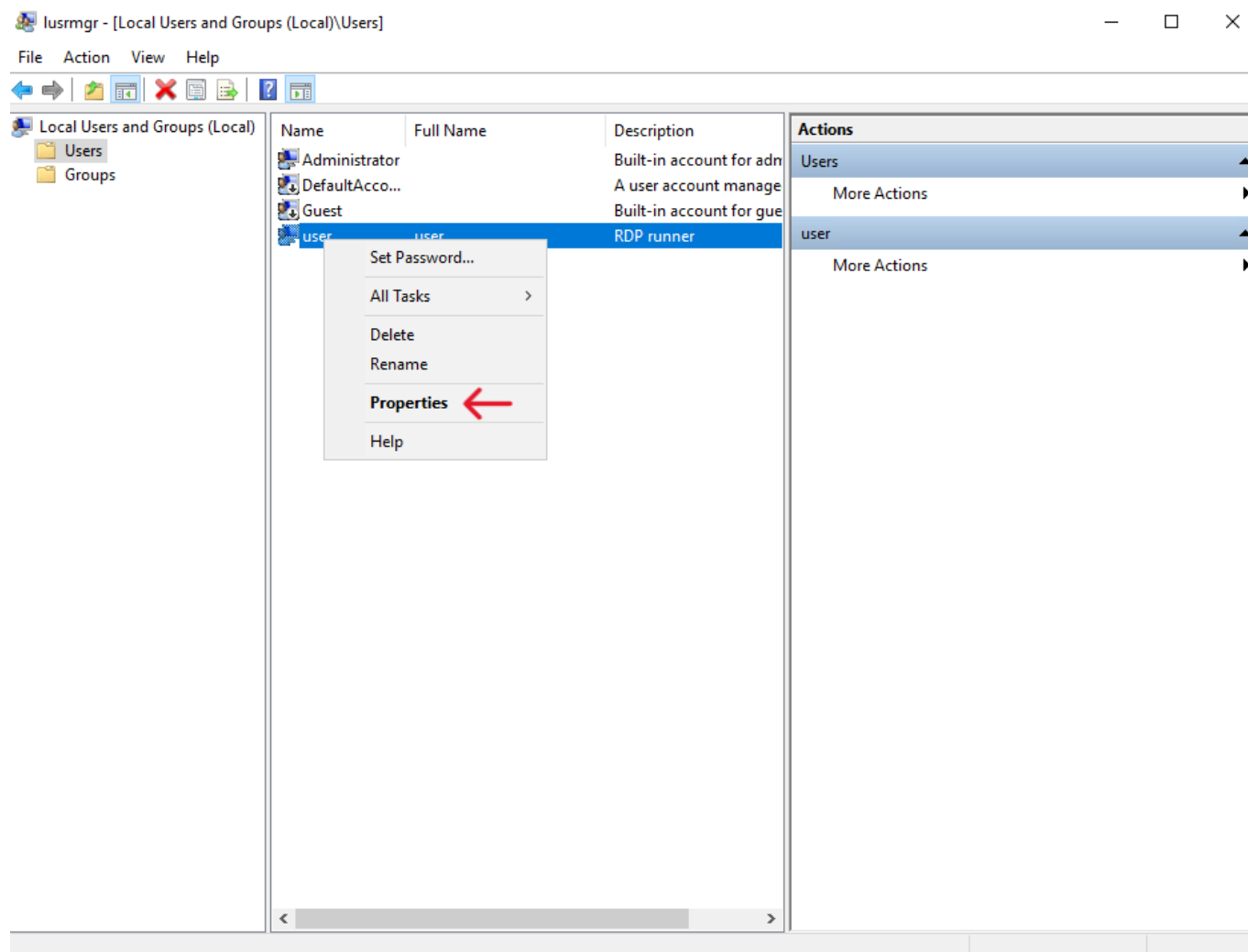
Password never expires

Account is disabled

Create

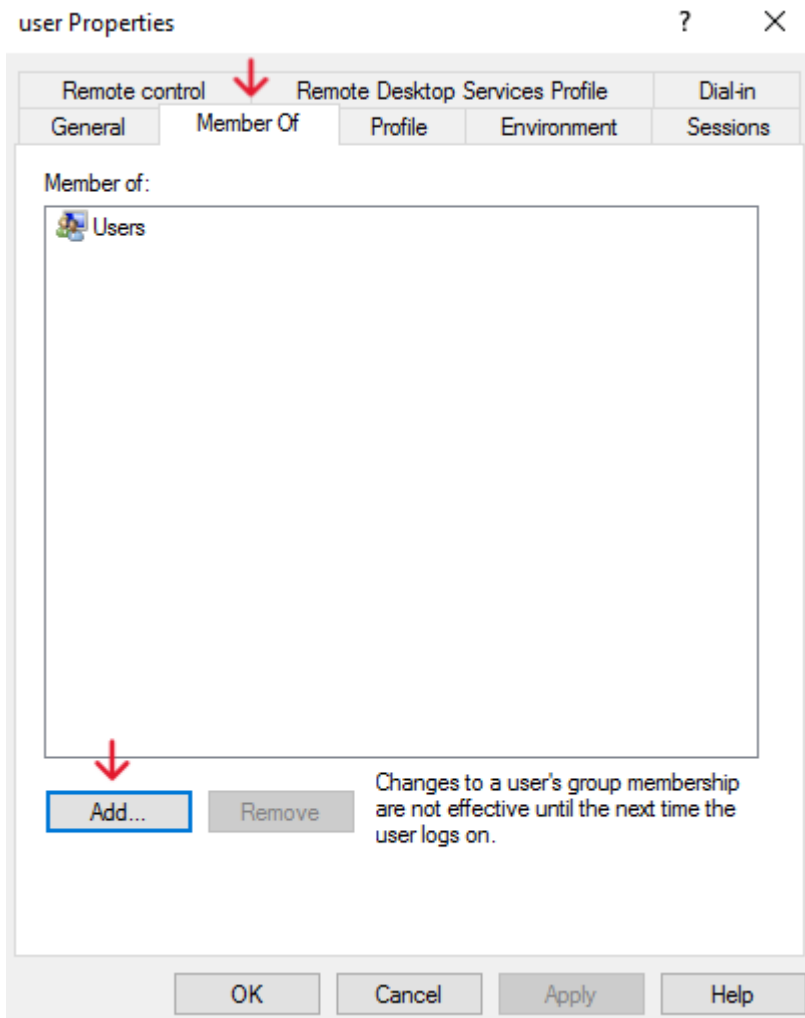
Ventana *Nuevo Usuario*

Cierre esa ventana y regrese a Cuentas de usuario, abra la pestaña **Usuarios**, seleccione el usuario creado y haga clic en el botón **Propiedades**.

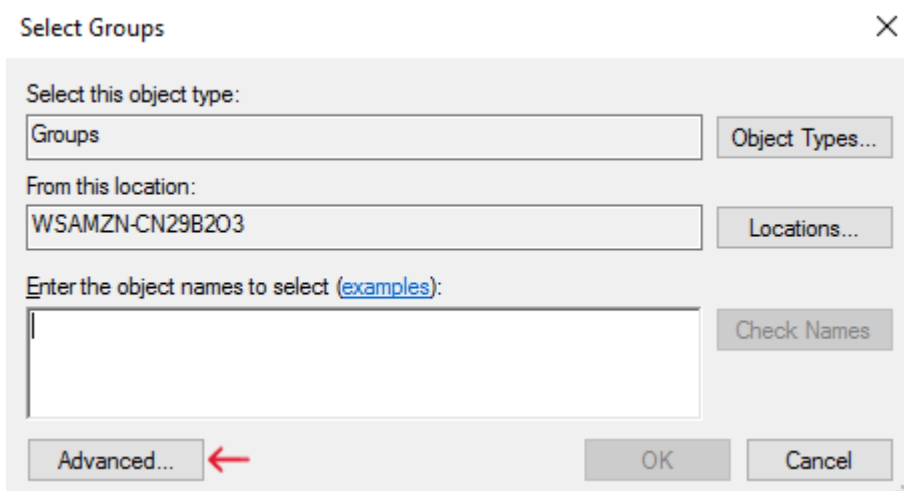


Ventana Usuarios y grupos locales – Menú desplegable de usuario

En la ventana Propiedades de usuario, abra la pestaña **Miembro de**, haga clic en **Agregar**, luego haga clic en **Avanzado**.

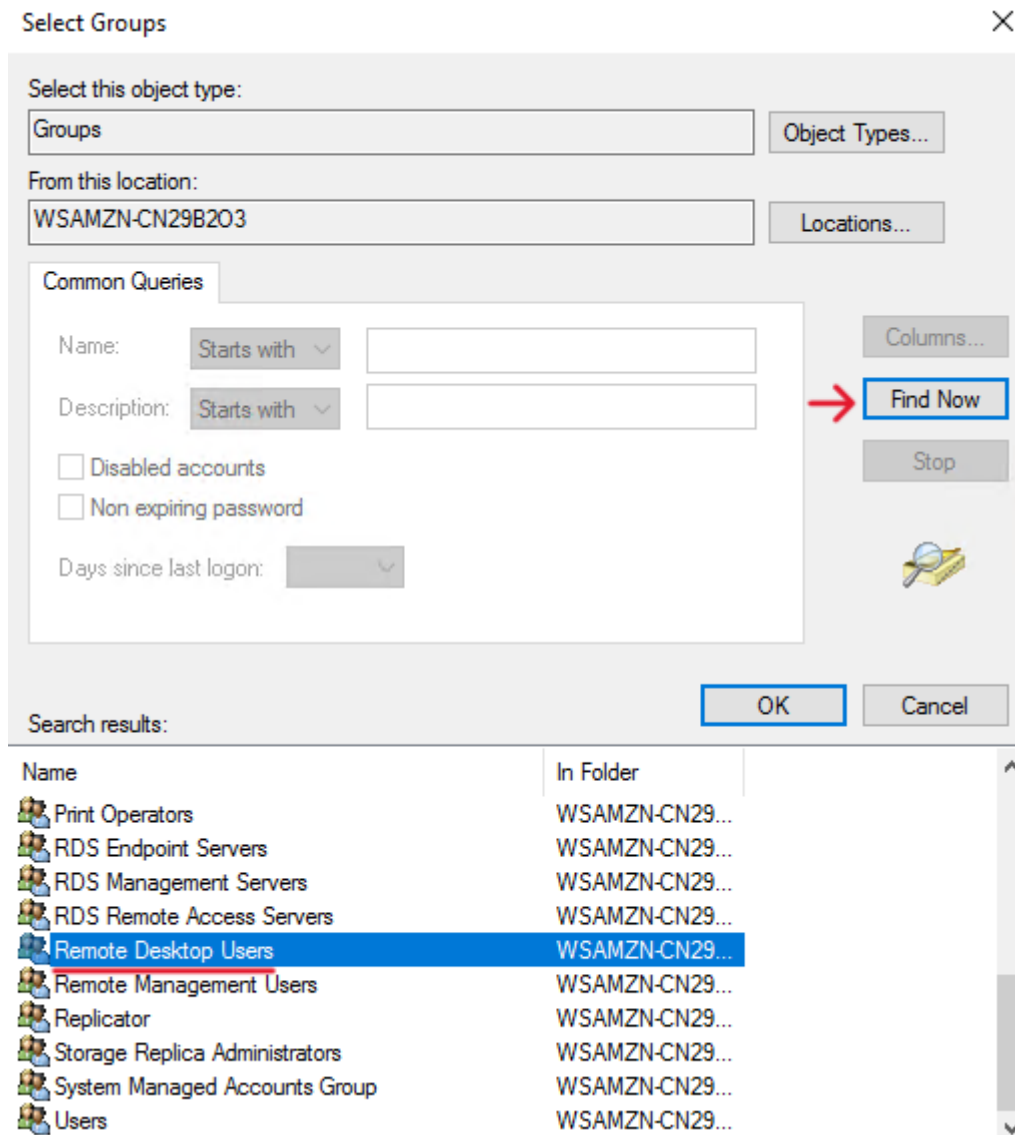


Ventana *Propiedades de <usuario>*



Ventana *Seleccionar Grupos*

Haga clic en **Buscar ahora** y seleccione **Usuario de escritorio remoto**.



Ventana *Seleccionar Grupos (Avanzado)*

Finalmente aplique los cambios.

⚠ **Importante:**



Como configurar la herramienta:

1. Conéctese a la cuenta y abra Rocketbot Studio (No es necesaria una licencia productiva, con una de desarrollador es suficiente).
2. Abra el complemento Asistente de Escritorio remoto y complete el formulario con los datos necesarios para realizar una Conexión a Escritorio remoto sobre la cuenta que posee el Rocketbot productivo. Una vez completado con lo solicitado:
 - IP del ambiente
 - Usuario y Contraseña
 - Carpeta donde guardar el archivo a crear
 - Nombre del archivos **.rdp** que se creara
 - Modo (Opcional, por defecto será Pantalla Completa, pero puede

seleccionarse modo Ventana)

- Resolución con la cual abrir el escritorio remoto (800×600; 1024×768; 1920×1080; etc.).

Nota: Debe ser la resolución con la cual se desarrollo el bot.

3. Dar guardar para crear el archivo **.rdp**.
4. Cree una tarea automatizada para ejecutar el archivo generado así se abrirá el escritorio remoto para ejecutar los bots en el horario programado sin intervención humana.
5. Desconéctese de la cuenta creada para ejecutar el archivo **.rdp** usando TSCON.bat ([Documentación](#))

□ [REDACTED]:

[REDACTED]

l [REDACTED]

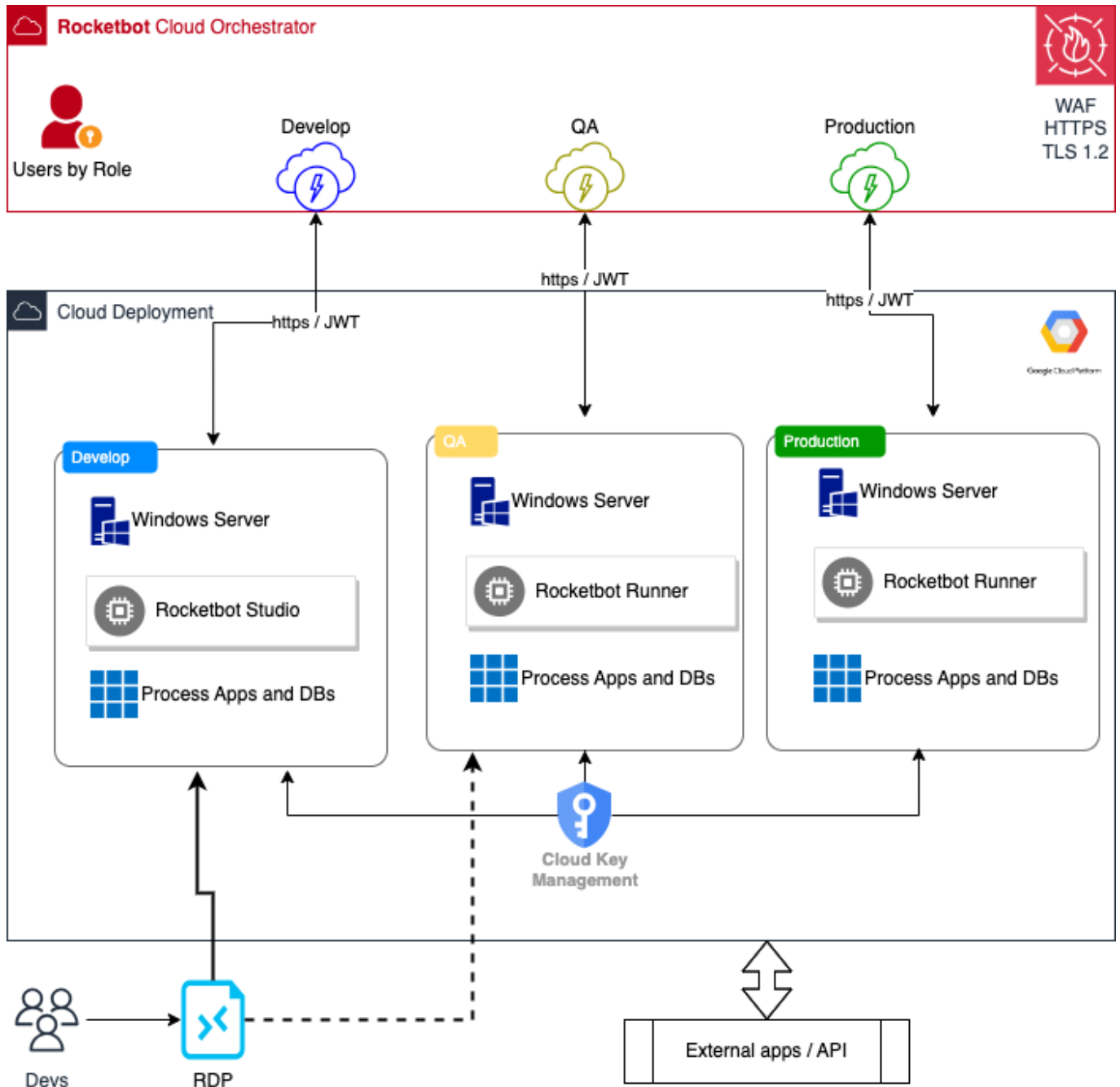
[Despliegue de Rocketbot en GCP](#)



Google Cloud Platform

En esta guía, te explicaremos los pasos para crear una máquina virtual (VM) en Google Cloud Platform (GCP) y montar Rocketbot Studio para ejecutar robots.

Arquitectura recomendada.



Crear y administrar VM de Windows Server

Compute Engine proporciona [imágenes públicas](#) con Windows Server que puedes usar para crear instancias.

Antes de comenzar

1. En la página del selector de proyectos de la consola de Google Cloud, selecciona o [crea un proyecto de Google Cloud](#).

Nota: Si no planeas conservar los recursos que creaste durante este procedimiento, crea un proyecto en lugar de seleccionar uno existente. Cuando termines, puedes borrar el proyecto y quitar todos los recursos asociados con él.

[Ir al selector de proyectos](#)

2. [Asegúrate de que la facturación esté habilitada para tu proyecto de Google Cloud.](#)
3. Habilita la API de Compute Engine. [Habilitar la API de Compute Engine](#)
4. Si no usas Windows en tu máquina local, instala un cliente de RDP, como el [Escritorio remoto de Chrome](#).

Crea una instancia de VM de Windows Server

En la consola de Google Cloud, ve a la página **Crear una instancia**.

[Ir a Crear una instancia](#)

Nombre *
instance-4

ADMINISTRAR ETIQUETAS DE INSTANCIA Y ETIQUETAS DE RECURSO

Región *
us-central1 (Iowa)

La región es permanente

Zona *
us-central1-a

La zona es permanente

Configuración de la máquina



Try the new C3A machine series, optimized for price-performance and sustainable.

TRY NOW

De uso general

Optimizado para procesamiento

NUEVO

Con optimización de memoria

GPU

Tipos de máquinas para cargas de trabajo comunes, optimizados en función del costo y la flexibilidad

Series	Descripción	vCPUs	Memory	Plataforma
<input type="radio"/> C3	Rendimiento alto y coherente	4 - 176	8 - 1,408 GB	Intel Sapphire Rapids
<input type="radio"/> C3D	Rendimiento alto y constante	4 - 360	8 - 2,880 GB	AMD Genoa
<input checked="" type="radio"/> E2	Procesamiento diario de bajo costo	0.25 - 32	1 - 128 GB	Según la disponibilidad
<input type="radio"/> N2	Precio y rendimiento equilibrados	2 - 128	2 - 864 GB	Intel Cascade y Ice Lake
<input type="radio"/> N2D	Precio y rendimiento equilibrados	2 - 224	2 - 896 GB	AMD EPYC
<input type="radio"/> T2A	Cargas de trabajo de escalamiento horizontal	1 - 48	4 - 192 GB	Altra de Ampere basada e
<input type="radio"/> T2D	Cargas de trabajo de escalamiento horizontal	1 - 60	4 - 240 GB	AMD EPYC Milan
<input type="radio"/> N1	Precio y rendimiento equilibrados	0.25 - 96	0.6 - 624 GB	Intel Skylake

En la sección **Disco de arranque**, haz clic en **Cambiar** para comenzar a configurar el disco de arranque.

En la pestaña **Public images**, selecciona **Windows Server** en la lista **Sistema operativo**.

Elige **Windows Server 2019 Datacenter** en la lista **Versión**.

Disco de arranque

Selecciona una imagen o instantánea para crear un disco de arranque o adjuntar un disco existente. ¿No encuentras lo que buscas? Explora cientos de soluciones de VM en [Marketplace](#)

IMÁGENES PÚBLICAS

IMÁGENES PERSONALIZADAS

INSTANTÁNEAS

Sistema operativo

Windows Server

Versión *

Windows Server 2019 Datacenter

x86-64, Server with Desktop Experience, x64 built on 20231011

Tipo de disco de arranque *

Disco persistente equilibrado

COMPARAR TIPOS DE DISCOS

Tamaño (GB) *

50

Aprovisiona de 50 a 65536 GB

Haz clic en **Seleccionar**.

En la sección **Firewall**, selecciona **Permitir tráfico HTTP**.

Para crear la VM, haz clic en **Crear**.

Espera un momento hasta que se inicie la instancia de VM. Una vez que la instancia esté lista, se mostrará en la página **Instancias de VM** con un ícono de estado verde.

Conéctate a las VM de Windows mediante

RDP

WINDOWS

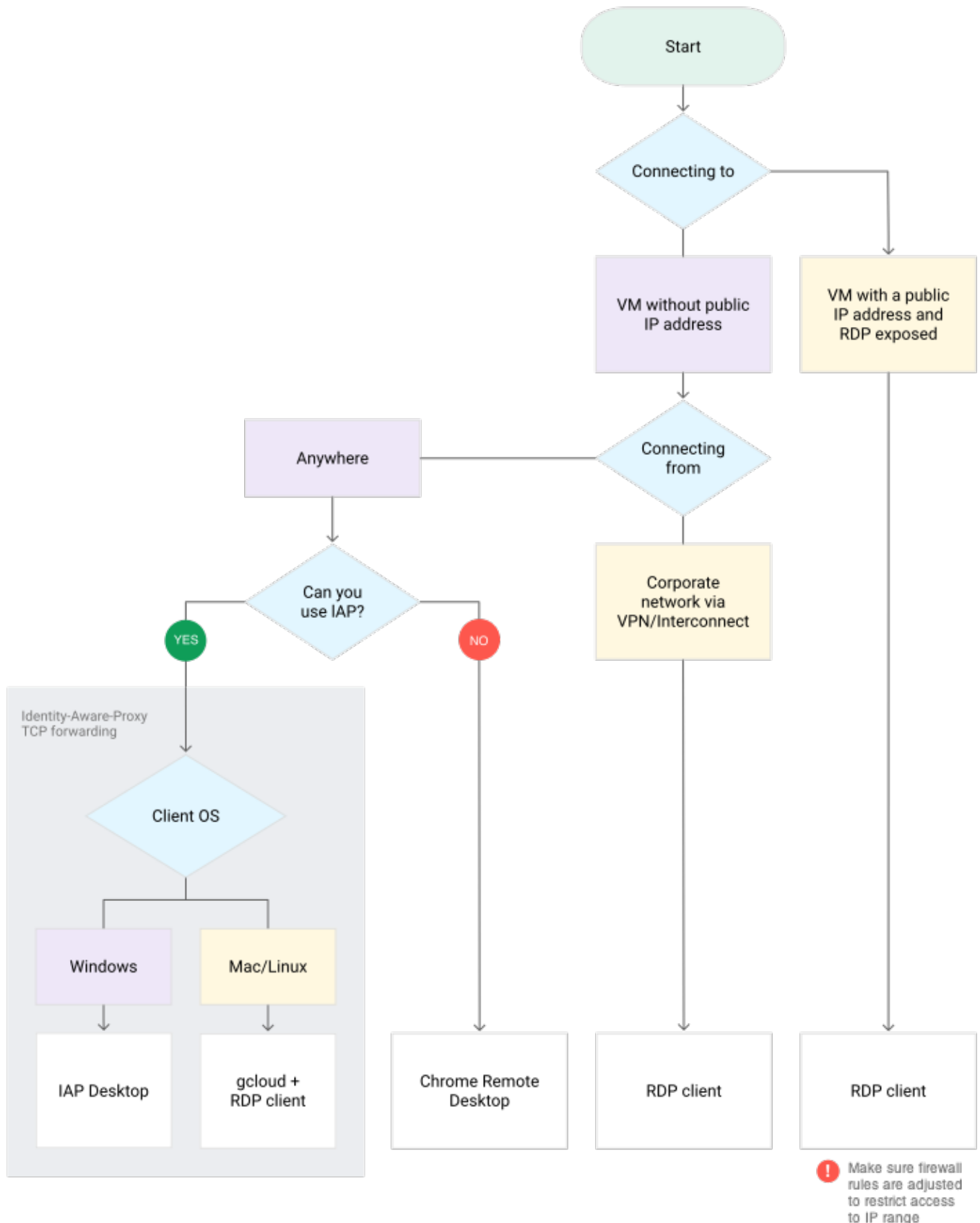
En este documento, se describe cómo conectarse a instancias de máquina virtual (VM) de Windows mediante RDP. Para conocer otras formas de conectarse a las VM de Windows, consulta [Conéctate a las VM de Windows mediante PowerShell](#) y [Conéctate a las VM de Windows mediante SSH](#).

Antes de comenzar

- Si deseas usar los ejemplos de línea de comandos de esta guía, haz lo siguiente:
 1. Instala [Google Cloud CLI](#) o actualízala a la última versión.
 2. [Configura una región y una zona predeterminadas](#).
- Asegúrate de que la VM permita el acceso a través del protocolo de escritorio remoto (RDP). De forma predeterminada, Compute Engine crea reglas de firewall que permiten el acceso RDP en el puerto TCP 3389. Para verificar que estas reglas de firewall existan, visita la [página de reglas de firewall en la consola de Google Cloud](#) y busca las reglas de firewall que permitan conexiones tcp:3389.

Conéctate a VM de Windows mediante RDP

Compute Engine admite varias formas de conectarte a tus instancias de Windows.



La mejor manera de conectarse al escritorio remoto de una instancia de Windows depende de varios factores:

- Si te conectas desde cualquier lugar de la Internet pública (Conectar desde > **Cualquier** de la ilustración anterior), es mejor habilitar la

redirección de [TCP de Identity-Aware Proxy](#) para tu proyecto. Luego, usa IAP Desktop (en Windows) o la CLI de Google Cloud junto con un cliente RDP. Si deseas obtener más información, consulta [Clientes de escritorio remoto de Microsoft](#) para conectarte a la instancia de Windows. Si no puedes usar el reenvío de TCP de Identity-Aware Proxy, usa el [Escritorio remoto de Chrome](#).

- Si la instancia de VM tiene una dirección IP pública y las reglas de firewall permiten el acceso de RDP, usa un cliente de RDP. Si deseas obtener más información, consulta [Clientes de escritorio remoto de Microsoft](#) para conectarte a la instancia de Windows.
- Si la instancia de VM no tiene una IP pública y te conectas mediante el siguiente comando: [Cloud VPN o Cloud Interconnect](#), puedes conectarte a la dirección IP privada de la VM mediante un cliente RDP. Para obtener más información, consulta [Clientes de escritorio remoto de Microsoft](#).

Si tienes dificultades para conectarte mediante RDP, consulta [Solución de problemas de RDP](#). Si no puedes conectarte a una instancia de Windows mediante el escritorio remoto, consulta [Conéctate a la SAC de Windows](#) en este documento.

Para conectarte al escritorio remoto de una instancia de Windows, usa uno de los siguientes procedimientos.

Puedes usar la app de Conexión a Escritorio remoto de Microsoft que forma parte de Windows para conectarte a instancias de Windows.

Antes de conectarte mediante la app de Conexión a Escritorio remoto de Microsoft, asegúrate de que se cumpla uno de los siguientes requisitos previos:

- Tu instancia de VM tiene una dirección IP pública, y tus [reglas de firewall](#) permiten el tráfico de entrada de TCP desde la dirección IP pública de tu cliente hacia la instancia mediante el puerto 3389.
- Tu red local está conectada a tu VPC mediante [Cloud VPN](#) o [Cloud Interconnect](#), y las [reglas de firewall](#) permiten el tráfico de entrada de TCP de la dirección IP privada de tu cliente a la instancia mediante el puerto 3389.

Para conectarte con el Escritorio remoto de Microsoft Windows, sigue estos pasos:

1. [Crea una cuenta de Windows y una contraseña](#), si aún no tienes una.

2. Para conectarte mediante Internet, usa la dirección IP *externa*. Para conectarte mediante Cloud VPN o Cloud Interconnect, usa la dirección IP *interna*.

Para identificar las direcciones IP internas y externas de tu instancia de Windows, completa uno de los siguientes pasos:

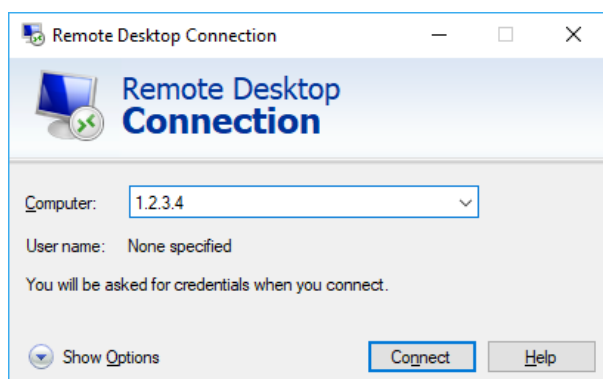
- En la consola de Google Cloud, ve a la página **Instancias de VM**.

[Ir a la página Instancias de VM](#)

- Con la CLI de gcloud, ejecuta `gcloud compute instances list`:

```
gcloud compute instances list
```

3. Abre la Conexión a Escritorio remoto de Microsoft Windows en tu máquina con Windows. Puedes encontrar el archivo ejecutable en `%systemroot%\system32\mstsc.exe`



4. En el cuadro **Computer**, ingresa la dirección IP.

Si configuraste la instancia con el fin de que use un número de puerto diferente para RDP, agrega ese número después de la dirección IP, por ejemplo: `1.2.3.4:3389`.

5. Haz clic en **Conectar** (Connect).

6. Ingresa tu nombre de usuario y contraseña, y haz clic en **Aceptar**.

Si olvidaste la contraseña, puedes [restablecerla](#).

Fuente:

<https://cloud.google.com/compute/docs/instances/connecting-to-windows?hl=es-419#remote-desktop-connection-app>

Instalar Rocketbot en Windows

1. Para instalar Rocketbot dirijase a <https://rocketbot.com/es/rocketbot-studio-rpa/>
2. Elija descargar y seleccione la version de Rocketbot para Windows.
3. Descargue y descomprima el ZIP en un carpeta dentro de la maquina donde ejecutará Rocketbot. por ejemplo: C:\\rocketbot

	Windows 10/11/Server 2016+	
	v2023.06.29 353 MB – ZIP	
6a6bd	MD5: 85c0d4d7847656a76186288e999c6201	
	DESCARGAR	
	Windows 7/8/10/11/Server 2012+	

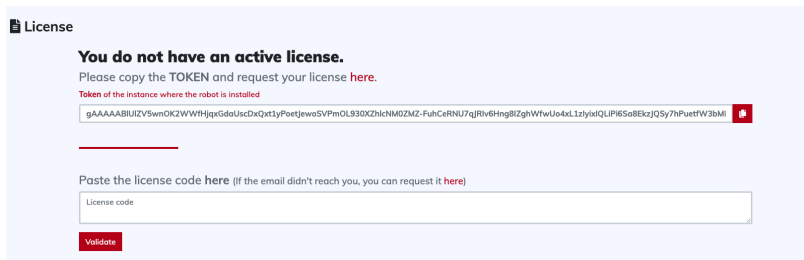
4. Ingrese a la carpeta de Rocketbot y ejecute la aplicacion **rocketbot.exe**



5. Se abra el navegador por defecto en el puerto 5000 (configurable) y le pedirá una licencia.



6. Copie el Token y complete los datos en la pagina de licencias



7. Recibirá en su correo la licencia. Pegue el código en Rocketbot Studio y comience a programar robots.

Consideraciones y TIPS

Rocketbot Studio/Runner requiere una sesion activa y desktop activo.

Mantener interfaz al desconectarse del servidor

Para evitar estos problemas, los servidores windows cuentan con un comando llamado **tscon.exe** que permite tomar la sesión que tienes abierta en el servidor y moverla a la consola, de esta manera tu robot tendrá una interfaz donde trabajar luego de desconectarte del servidor.

En las pestañas *Desconectarme y ejecutar una tarea* o *Desconectarme y ejecutar ROC* de la aplicación **RDP-Rocketbot** puedes crear un archivo bat con el comando **tscon**. Este archivo debes ejecutarlo como administrador cada vez que quieras desconectarte del servidor (o ejecutar tu robot, según la opción elegida) y te desconectará del servidor y mantendrá la interfaz gráfica.

Cuidado! Al no abrir el escritorio remoto desde una pantalla física, la resolución puede cambiar y ser menor. Revisa la sección tres para ver como trabajar con esto.

El bat puedes editarlo y revisar que es lo que contiene, pero resumimos lo importante acá. El bat tiene un script de powershell que obtiene el id de la sesión que estas usando al entrar al rdp y la utiliza en el comando **tscon** para que este la abra en la sesión de la consola.

```
@powershell -NoProfile -ExecutionPolicy unrestricted -Command  
"$sessionid=((quser $env:USERNAME | select -Skip 1) -split '\\s+')[2]; tscon  
$sessionid /dest:console" 2> UnlockErrors.log
```

Si necesitas que el bat ejecute tu robot al desconectarte, verás algo así

```
@powershell -NoProfile -ExecutionPolicy unrestricted -Command  
"$sessionid=((quser $env:USERNAME | select -Skip 1) -split '\\s+')[2]; tscon  
$sessionid /dest:console" 2> UnlockErrors.log
```

```
timeout 10  
cd c:\\rocketbot  
rocketbot.exe -start=nombre_robot
```

Mantener la interfaz al minimizar el escritorio remoto

Para poder minimizar el escritorio remoto y no afectar el funcionamiento del robot, debes modificar algunos registros en la máquina que abre el rdp, estos registros son:

- Para el usuario actual:
 - HKEY_CURRENT_USER\\Software\\Microsoft\\Terminal Server Client (32 bit)
 - HKEY_CURRENT_USER\\Software\\Wow6432Node\\Microsoft\\Terminal Server Client (64 bit)
- Para todos los usuarios:
 - HKEY_LOCAL_MACHINE\\Software\\Microsoft\\Terminal Server Client (32 bit)
 - HKEY_LOCAL_MACHINE\\Software\\Wow6432Node\\Microsoft\\Terminal Server Client (64 bit)

En la pestaña *Conectarme a Escritorio Remoto* de la aplicación RDP-Rocketbot, puedes cambiar los registros con un solo botón. Solo recuerda ejecutar la aplicación como administrador.

Cambio de resolución al desconectarte del servidor

Si ya solucionaste los problemas anteriores, pero tu **robot** falla y/o aún no reconoce las imágenes, es probable que tengas un problema con la **resolución**. Ya mencionamos en la primera parte de este post que al desconectarte el servidor pierde la interfaz gráfica y es necesario utilizar **tscon**. Esto

permite que el robot pueda sacar capturas, pero como el servidor ya no cuenta con tu pantalla para ajustar una resolución, toma sus propias resoluciones por defecto, que en la mayoría de los casos es de 1024x768.

Si tu robot trabaja con **virtualización**, este cambio de resolución puede afectarlo, por lo cual es importante que, antes de comenzar a desarrollar tu robot, revises las resoluciones que permite tu servidor al desconectarte.

En la pestaña *Ver las resoluciones de pantalla* de la aplicación RDP-Rocketbot, puedes ver las resoluciones que permite el servidor en la tabla *Resoluciones de pantalla desde la consola de TSCON*.

Esas resoluciones son las resoluciones que puedes utilizar, pero no quiere decir que puedes comenzar a construir el robot. Debes revisar la resolución que tiene el servidor al desconectarte y si es posible cambia a la resolución que necesitas. Para cambiar la **resolución de pantalla**, puedes utilizar el módulo [Windows](#) y también puedes utilizar el comando *Sacar captura de pantalla* de la sección Desktop para validar si la resolución es la correcta o no.

Esto lo puedes realizar con **RDP-Helper**, acá encontrarás los pasos necesarios [Pasos Resolución](#)

Si al salir de un escritorio remoto no es posible modificar la resolución de forma manual y/o usando el módulo de Rocketbot, puedes modificar realizar la siguiente modificación:

Buscar en el regedit todos los registros que se llamen **DefaultSettings_XResolution** y **DefaultSettings_YResolution** y cambiar la resolución a la que necesitas

Prevenir el bloqueo de pantalla

Para evitar que el servidor bloquee la pantalla debes, en primer lugar, consultar con el equipo de TI si es posible modificar las configuraciones necesarias para que esto no ocurra, ya que dependiendo de las políticas de seguridad de la empresa, podría no ser posible modificar esta configuración.

En la sección *Configuraciones* de la aplicación RDP-Rocketbot, puedes modificar los registros necesarios para deshabilitar el bloqueo de pantalla. El registro que se modificará es `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows`

Cambiando Políticas [Link](#)

For Windows 7/Windows 10/Windows Server 2012 and above:

- Press + and type **gpedit.msc**

Windows+R

- Navigate the tree view on the left to **Computer**

Configuration/Administrative Templates/Windows Components/Remote Desktop Services/Remote Desktop Session Host/Session Time Limits

- Adjust the four settings as desired (I believe you want to adjust ***Set time limit for disconnected sessions*** and ***Terminate session when time limits are reached***)

Habilitar Pantallas Virtuales

□ Si experimentas problemas con la resolución de la VM, consulta la siguiente [documentación al respecto](#).

En algunos casos, la VM solo permitía una resolución muy baja por defecto. Siguiendo la documentación indicada y asegurándonos de **apagar la VM antes de activar la casilla de configuración**, logramos establecer una resolución adecuada para el proceso requerido.

Conectar con Orquestador

Para conectar con el Orquestador de Rocketbot debe contar con una cuenta habilitada de Rocketbot Orchestrator Center. En la VM de GCP tiene que tener habilitado el acceso a las web ***.myrb.io** y ***.rocketbot.com** esto permitirá al cliente de orquestador NOC conectarse con las url de api de orquestador.

INSTALACIÓN Y CONFIGURACIÓN CLIENTE ORQUESTADOR

1. Descargar cliente desde el orquestador: Haga click en el boton del menu superior del orquestod "ROC Client" para descargar el ZIP con el cliente.

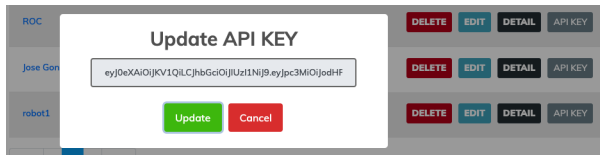


2. Descomprima noc.zip en una carpeta donde tenga permisos de escritura, no se recomienda que sea en la carpeta personal o desktop. Recomendacion :
C:\\rocketbot\\noc

Nombre	Fecha de modificación	Tipo	Tamaño
certifi	11-05-2020 12:47	Carpeta de archivos	
psutil	11-05-2020 12:47	Carpeta de archivos	
robots	25-11-2020 14:49	Carpeta de archivos	
_bz2	21-04-2020 16:22	Python Extension ...	72 KB
_ctypes	21-04-2020 16:22	Python Extension ...	105 KB
_elementtree	21-04-2020 16:22	Python Extension ...	166 KB
_hashlib	21-04-2020 16:22	Python Extension ...	1.154 KB
_lzma	21-04-2020 16:22	Python Extension ...	181 KB
_socket	21-04-2020 16:22	Python Extension ...	62 KB
_ssl	21-04-2020 16:22	Python Extension ...	1.513 KB
base_library	14-05-2020 11:45	Archivo WinRAR Z...	753 KB
noc	14-05-2020 11:45	Aplicación	2.001 KB
noc.exe.manifest	14-05-2020 11:45	Archivo MANIFEST	2 KB
noc	10-02-2021 9:12	Opciones de confi...	1 KB
package	24-04-2020 15:23	Archivo JSON	4 KB
pyexpat	21-04-2020 16:22	Python Extension ...	163 KB
python36.dll	21-04-2020 16:22	Extensión de la ap...	3.374 KB
README.md	24-04-2020 15:10	Archivo MD	1 KB
select	21-04-2020 16:22	Python Extension ...	23 KB
unicodedata	21-04-2020 16:22	Python Extension ...	877 KB
VCRUNTIME140.dll	21-04-2020 16:22	Extensión de la ap...	82 KB

3. Configurar archivo noc.ini con:

a. Credenciales de usuario con un ROL con permisos limitados



b. Key de instancia: Ingrese al proceso dentro del proyecto, una vez adicionado una instancia le mostrará una Key

Key: 601d[redacted]3507  copy

Last Execution: 6 months ago

c. Url de orquestador, ej: <https://roc.myrb.io/s4> (sin / final) d. Ruta de instalación de Rocketbot, incluyendo ejecutable. por ejemplo

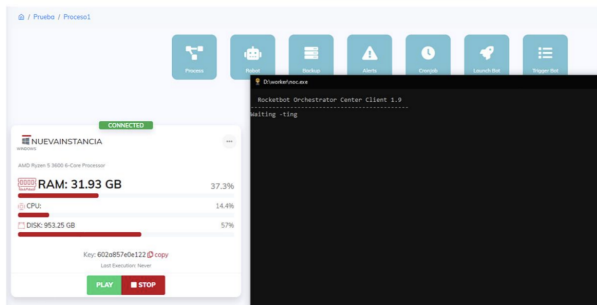
c:\\rocketbot

```

1  [USER]
2  # user access
3  user =
4  password =
5  #or API Key
6  apikey = eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHFi
7  # Instance ID
8  key = xxxxxx
9
10
11 [NOC]
12 # you Url Orchestrator
13 # https://devnoc.myrb.io
14 # https://roc.myrb.io/s1
15 # https://roc.myrb.io/s2
16 # https://roc.myrb.io/s4
17 # https://roc.myrb.io/dev
18 server = https://roc.myrb.io/dev You, 6 months ago * chore(noc.ini): update
19 proxy =
20 #Send logs to Orchestrator?
21 logs = true
22 [ROCKETBOT]
23 # Rocketbot bin
24 path =/Users/lucianodavidcuello/projects/rocketbot/dist/rocketbot/rocketbot
25 # No debug on Rocketbot?
26 nodebug = false
27
28 #Screenshot available ?
29 screenshot = true
30 #Command line options
31 comandline = --update-drivers
32

```

4. Ejecute noc.exe y el cliente de orquestador mostrara que la instancia está conectada.



Como actualizar a Rocketbot 2023

Las versiones anteriores a [Rocketbot 2023.03.20](#) no cuentan con actualizaciones automáticas, por lo que se necesita actualizar manualmente a la última versión de rocketbot. En este post explicaremos los pasos que debes realizar para actualizar Rocketbot

[Ver updates](#)

Migración de archivos

Para mantener la licencia y los robots, tienes que copiar desde tu versión anterior a la versión 2023 los siguientes archivos

- `license.json`: Este archivo contiene tu licencia
- `robot.db`: Este archivo contiene la base de datos de robots que viene por defecto en Rocketbot

Actualización de módulos

La versión 2023.03.20 realiza una actualización de la versión de python que utiliza. Esto puede generar conflictos con algunos módulos de las versiones anteriores, por tanto, no recomendamos copiar la carpeta `modules` de versiones anteriores, sino instalar las últimas versiones disponibles en el [marketplace](#)

Para actualizar o instalar los módulos que necesitan tus robots, puedes ir a la pestaña Robot de tu robot y al final de la página, instalar o actualizar los módulos correspondientes



Sección de módulos en la pestaña Robot

También puedes buscarlo desde el addon Install Mods o desde nuestro [Marketplace](#)

Migración de Robots

La última versión, además de actualizar el Python interno de 3.6.8 de 32bits a [Python 3.10.4 de 64 bits](#) también corrige el funcionamiento de alguno de los comandos nativos. Esto puede ocasionar que robots desarrollados en versiones anteriores, no funcionen correctamente en la última versión. Ya sea por correcciones para bugs de comandos o por librerías externas de Python instaladas en Rocketbot.

Para migrar los robots recomendamos lo siguiente:

- Para robots en producción, no actualizar a la nueva versión a menos que sea necesario. Puedes mantener las dos versiones de Rocketbot funcionando en la misma máquina
- Actualiza el Python de tu máquina a [Python 3.10 de 64 bits](#) e instala las librerías que utilizan tus robots en la carpeta “modules/libs” de la nueva versión
- Actualiza a la última versión los archivos que indica el updater
- Si quieres migrar un robot productivo, realiza pruebas con la nueva versión antes de actualizar

Url utilizadas por el Updater

Las actualizaciones automáticas son revisadas y descargadas directamente desde el repositorio de Github de Rocketbot. Si tienes restricciones a URLs públicas, te dejamos un listado de las urls que necesita rocketbot para actualizar

- Consulta de nuevas versiones:
<https://raw.githubusercontent.com/rocketbot-cl/rocketbot/main/package.json>
- Descarga de archivos:
<https://github.com/rocketbot-cl/rocketbot/raw/main/updates/>
- Actualización de drivers de Chrome:
<https://chromedriver.storage.googleapis.com/>
- Actualización de drivers de Firefox:
<https://github.com/mozilla/geckodriver/releases>

[RPA on Virtual Server](#)

Depending on the process you wish to automate and how you intend to run the **robots**, your **server** will need to meet specific requirements for the robot to operate well.

Processes can be divided into two types: those that require a **graphical interface** (robots that take screenshots or use the keyboard and mouse) and those that do not (robots that emulate a terminal, automate xlsx files, use an API, etc.). It is critical to consider this when developing a robot because most servers **lose the graphical interface** when disconnected, causing robots to fail.

Screen locking and **minimizing the remote desktop** can also have an impact on your robot's performance. The operating system should never lock the screen on a machine running an automated process (**RPA**), as the robot will be **unable** to take **screenshots** or move the mouse to the indicated places. When you minimize the rdp, Windows may pause some processes or remove the interface, so you must configure your machine to prevent this from happening.

In **Rocketbot**, we have developed an application that allows you to easily configure your server, creating the necessary.bat files to run your robot and/or modifying the **registers** to prevent screen problems. You can download this application here: [RDP-Rocketbot](#). Remember to save it to the server and/or machine as needed.

Maintain interface when server disconnects.

To avoid these issues, Windows servers have a command called **tscon.exe** that allows you to take an open session on the server and move it to the console, giving your robot an interface to work with after disconnecting from the server.

The tscon command can be used to create a bat file in the RDP-Rocketbot application's *Disconnect and run a task* or *Disconnect and run ROC tabs*. This file must be run as administrator each time you want to disconnect from the server (or run your robot, depending on the option selected), and it will disconnect you from the server while maintaining the graphical interface.

Beware! If the remote desktop is not opened from a physical screen, the resolution may change and become lower. Refer to section three for alternatives.

You can edit the bat and examine its contents, but we have summarized the crucial information here. The bat contains a powershell script that retrieves the session id of the session you are using when logging into the rdp and uses it in the tscon command to open the console session.

```
@powershell -NoProfile -ExecutionPolicy unrestricted -Command  
"$sessionid=((quser $env:USERNAME | select -Skip 1) -split '\s+')[2]; tscon  
$sessionid /dest:console" 2> UnlockErrors.log
```

If you need the bat to run your robot after disconnecting, you will see something like this:

```
@powershell -NoProfile -ExecutionPolicy unrestricted -Command  
"$sessionid=((quser $env:USERNAME | select -Skip 1) -split '\s+')[2]; tscon  
$sessionid /dest:console" 2> UnlockErrors.log
```

```
timeout 10  
cd c:\rocketbot  
rocketbot.exe -start=nombre_robot
```

Maintain interface when minimizing remote desktop

To minimize the remote desktop without interfering with the robot's operation, modify the following registers in the machine that opens the rdp:

- For the current user:
 - HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client (32 bit)
 - HKEY_CURRENT_USER\Software\Wow6432Node\Microsoft\Terminal Server Client (64 bit)
- For all users:
 - HKEY_LOCAL_MACHINE\Software\Microsoft\Terminal Server Client (32 bit)
 - HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Terminal Server Client (64 bit)

You can change the logs with a single click in the RDP-Rocketbot application's Connect to Remote Desktop tab. Just keep in mind that you must run the application as an administrator.

Change in resolution after disconnecting from the server

If you have already resolved the previous issues, but your **robot** still fails and/or does not recognize the images, you most likely have a **resolution** issue. We already mentioned in the first section of this post that when you disconnect the server, the graphical interface is lost and you must use **tscon**. This enables the robot to take screenshots, but because the server no longer has access to your screen to set a resolution, it uses its own default resolution, which is 1024×768 in most cases.

If your robot uses **virtualization**, this resolution change may have an impact, so it is critical that you check the resolutions that your server allows when you disconnect before beginning to develop your robot.

The resolutions allowed by the server in the Screen resolutions table from the TSCON console can be seen in the *View screen resolutions* tab of the [RDP-Rocketbot](#) application.

Those are the resolutions you can use, but that does not mean you can begin building the robot. When you disconnect, you should check the resolution of the server and, if possible, change it to the resolution you require. You can use the [Windows](#) module to change the **screen resolution**, and you can also use

the Take screenshot command in the Desktop section to see if the resolution is correct.

A sample robot can be downloaded here [□](#)

[test_resolucionDownload](#)

If when exiting the remote desktop it is not possible to change the resolution manually and/or using the Rocketbot module, you can try the following:

Change the resolution in regedit by searching for all registries named **DefaultSettings XResolution** and **DefaultSettings YResolution**.

Prevent screen lock

To prevent the server from blocking the screen, you should first consult with the IT team to see if it is possible to modify the necessary settings to prevent this from happening, as this may not be possible depending on the company's security policies.

In the Settings section of the RDP-Rocketbot application, you can modify the registers needed to disable the screen lock. The registry to be modified is *HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows*

Changing Policies [Link](#)

Automatización de Aplicaciones de Escritorio

Las aplicaciones de escritorio no son tan estandarizadas como los son las aplicaciones web, mientras que para aplicaciones web utilizamos en su mayoría comandos que **automatizan** el navegador, para aplicaciones de escritorio tenemos diferentes métodos según el tipo de aplicación. En esta entrada te mostraremos las **consideraciones** y **métodos** que debes conocer antes de comenzar a desarrollar un robot para automatizar **aplicaciones de escritorio**.

Procesos que pueden prescindir de una aplicación

Existen algunos procesos que si bien utilizamos una aplicación para realizarlos, existen métodos que permiten a los robots hacerlo de forma más eficiente. Recuerda que automatizar no es solo hacer robots que repliquen el comportamiento humano frente a la computadora.

En este tipo de aplicaciones se encuentran aquellas actividades que se pueden realizar por terminal o scripts bash, aplicaciones que tienen su versión web y que cuentan con una API o que trabajan bajo algún protocolo. Estos procesos pueden ser:

- Automatización de correos electrónicos por conexión imap, smtp, pop.
- Manipulación de archivos como mover, renombrar, leer, crear, etc.
- Consultas a base de datos
- Aplicaciones con versión web como [Slack](#), [Google Drive](#), [Telegram](#), [One Drive](#)
- Conexiones [ssh](#) o [telnet](#)
- Aplicaciones [simuladas en una terminal](#)

No necesitas tener instalada la aplicación para automatizar estos procesos, solo debes descargar el módulo correspondiente en nuestro [Marketplace](#). Estos procesos en su mayoría permiten ejecuciones en paralelo.

Aplicaciones con su propio sistema de scripting

Algunas empresas ya conocen el poder de la automatización y nos permiten interactuar con sus aplicaciones a través de su propio sistema de scripting. De esta manera, podemos controlar la aplicación sin necesidad de utilizar el mouse o teclado y utilizar los métodos internos que tiene cada aplicación para automatizar nuestro proceso.

Para automatizar estas aplicaciones es necesario tener instalada la aplicación en la misma máquina donde se ejecutará tu robot. Aplicaciones de este tipo son: [SAP](#), [SAPBI](#), [Excel](#), [Word](#), [Outlook](#) y puedes encontrar módulos para automatizarlos en [Marketplace](#). Si conoces alguna aplicación que necesitas automatizar y esta tiene sistema de scripting, comunícate con nosotros y desarrollamos el componente.

Dependiendo de la aplicación, puedes ejecutar más robots al mismo tiempo. Revisa si la aplicación permite abrir múltiples instancias.

Aplicaciones Java o Windows Forms

Aunque algunas aplicaciones no tengan su propio sistema de scripting, si pueden ser automatizados de esta forma a través de las dll de Java o Windows. Automatizar una aplicación por scripting es similar a automatizar una página web, solo necesitas obtener información del objeto que quieres controlar, como el título, la clase, el id, el tipo de objeto, etc.

Esto no puede obtenerse directamente de la aplicación, pero puedes usar alguno de nuestros grabadores como el [DesktopRecorder](#) o [JavaRecorder](#). Con esta herramienta puedes grabar las acciones que quieres realizar en la aplicación para luego replicar lo grabado en un robot. No es sencillo determinar que aplicaciones permiten automatizarlo por scripting. La mejor recomendación es descargar los grabadores y revisar si detectan tu aplicación.

La aplicación debe estar instalada en la misma máquina donde se ejecutará el robot y al hacer uso del mouse y/o teclado, no se puede trabajar con robots en paralelo que automaticen el mismo proceso

Otras aplicaciones

Si tu aplicación no se puede automatizar de las formas mencionadas anteriormente, puedes automatizar la aplicación por virtualización y/o comandos de teclado. Para esto, es necesario tener algunas consideraciones:

- Si el robot se ejecutará en un servidor, revisa la entrada [RPA en servidores virtuales](#)
- Si la aplicación permite utilizar combinaciones de tecla, utiliza el teclado por sobre virtualización. Puedes conocer los shortcut de la aplicación al ver una letra subrayada bajo la palabra donde quieres hacer click. Para replicar en Rocketbot, revisa la siguiente entrada [Enviar Tecla](#)
- Si se debe extraer información, revisa si puedes usar el [portapapeles](#) para copiar el contenido.
- Como se sacarán capturas de pantalla, se utilizará el mouse o el teclado. No se puede automatizar la misma aplicación en paralelo