Orquestador – Migración a un nuevo servidor



Esta guía describe el procedimiento técnico recomendado para trasladar la operación de un Orquestador Rocketbot existente hacia una nueva infraestructura, asegurando la continuidad de los servicios y la correcta configuración de los activos digitales.

Requisitos Previos

Antes de iniciar el proceso de traslado, es indispensable cumplir con los siguientes puntos:

- Contar con la aprobación formal del área comercial.
- Gestionar una nueva licencia de Orquestador desde el portal oficial de Rocketbot.
- Verificar el acceso administrativo tanto al servidor de origen (actual) como al de destino (nuevo).

Configuración del Nuevo Entorno

Tras la solicitud de la licencia, se habilitará el acceso al nuevo servidor:

- Se proporcionarán las credenciales de acceso para la nueva cuenta.
- La dirección de acceso estándar para el nuevo Orquestador será: https://orchestrator.myrb.io.

Ejecución de la Migración

Es fundamental comprender que la migración no es un proceso automatizado; debe realizarse de forma manual replicando la estructura del entorno anterior.

Elementos a Trasladar

Se debe asegurar la réplica de los siguientes componentes clave:

- **Procesos:** Flujos de trabajo y lógicas de negocio.
- **Instancias:** Configuración de los equipos de ejecución.
- **Formularios y Assets:** Recursos compartidos e interfaces de usuario.
- **Usuarios:** Definición de roles y esquemas de permisos.

Actualización de Robots e Integraciones

Aquellos procesos que consuman servicios del Orquestador (como los módulos Rocketbot Xperience o Assets_NOC) requieren una actualización técnica para operar con el nuevo servidor.

Acciones de Reconfiguración

- **Actualización de URL:** Reemplazar cualquier referencia a la URL antigua por la nueva dirección oficial: https://orchestrator.myrb.io.
- **Generación de Nuevas Credenciales:** Es imperativo crear y sustituir en los robots los siguientes identificadores generados en el nuevo servidor:
    - API Keys y Keys de instancias.
    - Tokens de procesos y de formularios.

- **Actualizar los robots** :
    - Reemplazar credenciales antiguas por las nuevas.
    - Validar que las conexiones funcionen correctamente.

Fase de Validación y Cierre

Para minimizar riesgos, ambos servidores permanecerán activos de manera simultánea durante la fase de transición.

1. **Pruebas de Estabilidad:** Se recomienda ejecutar pruebas completas en el nuevo entorno, validando especialmente los procesos críticos y las conexiones con sistemas externos.
2. **Cese del Servicio Anterior:** Una vez confirmada la estabilidad operativa del nuevo servidor, se debe coordinar formalmente la baja del servidor antiguo tras la validación interna de los responsables.

Recomendaciones Finales de Seguridad

- **Control de Tiempos:** Realizar las tareas de migración en franjas horarias con menor impacto operativo.
- **Resguardo de Datos:** Mantener copias de seguridad de todas las configuraciones críticas durante el proceso.
- **Trazabilidad:** Documentar rigurosamente cada cambio efectuado, especialmente en lo que respecta a nuevas claves y tokens generados.
- **Escalabilidad:** En infraestructuras de gran tamaño, se sugiere realizar el traslado de los recursos de forma escalonada por etapas.

NOC v.2025.12.03

Network Orchestrator Client (NOC)

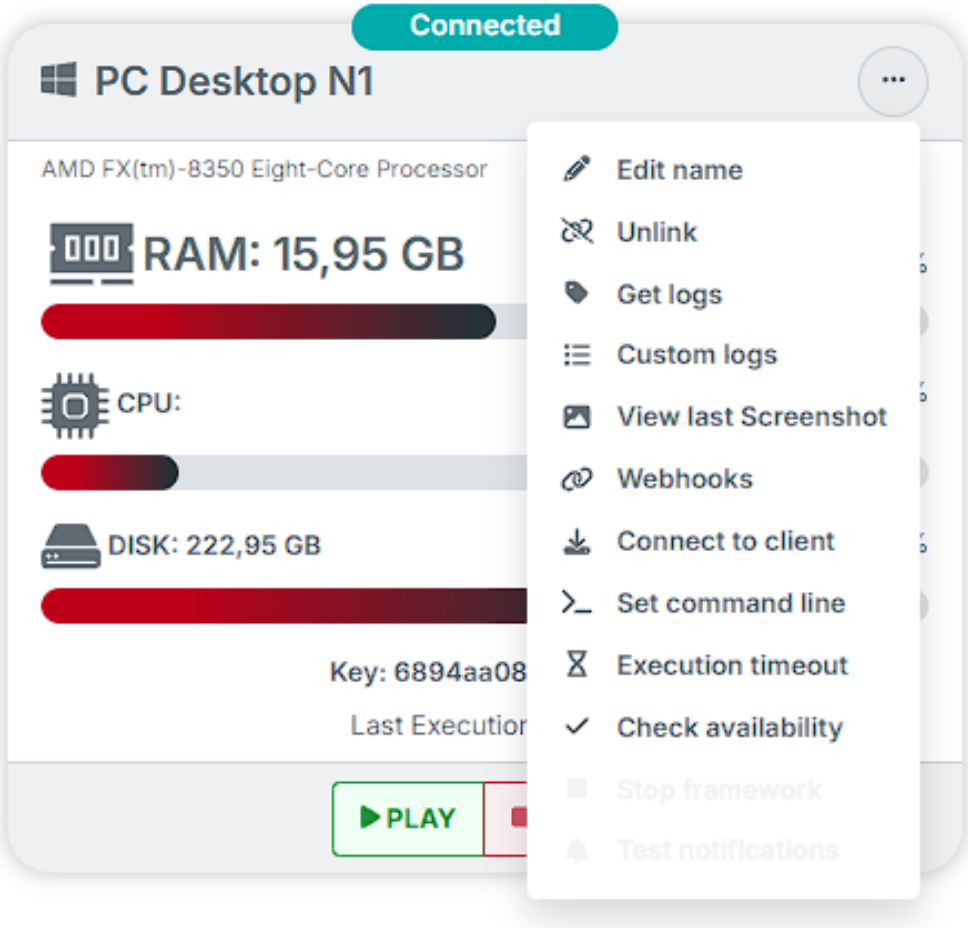
El **Network Orchestrator Client** es la herramienta encargada de vincular la capacidad de ejecución de su equipo local con el servidor del Orquestador. Esta nueva actualización permite visualizar la versión actual del **NOC** e integra herramientas avanzadas para la gestión de reconexiones, notificaciones y modos de ejecución.

Conectar instancia al cliente

De forma predeterminada, asignar un nombre a la instancia no la vincula automáticamente con el equipo. Para establecer la conexión, siga este procedimiento:

1. Haga clic en el menú de la instancia (icono de los tres puntos).
2. Seleccione el botón **[Connect to Client]**.

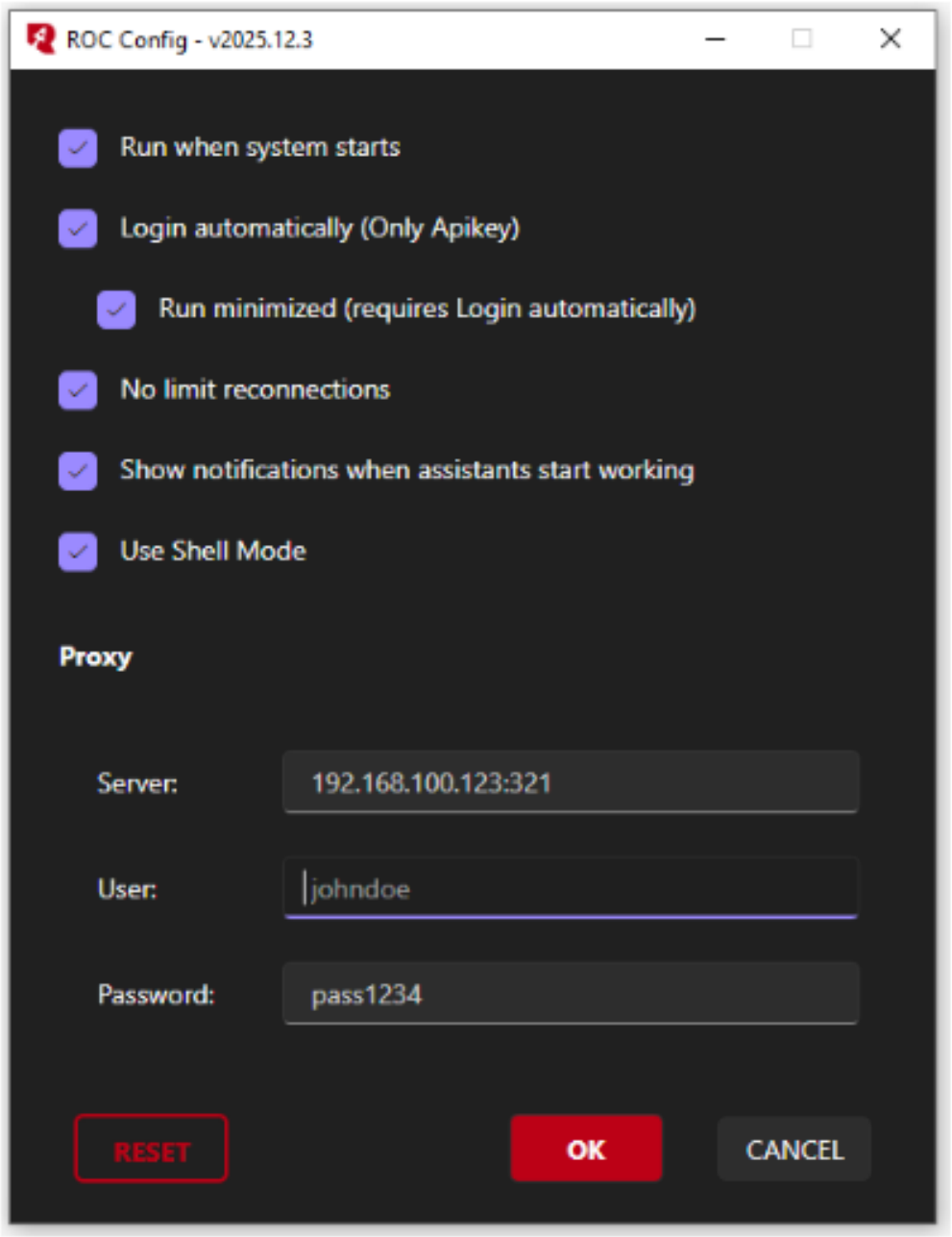
Para una guía detallada paso a paso, consulte la documentación oficial: Configuración de Instancias.



Nueva Actualización y Herramientas

La versión más reciente del cliente (***NOC***)) introduce mejoras significativas en la visibilidad y el control del entorno de ejecución:

- **Visualización de Versión:** Ahora el sistema muestra de forma directa la versión actual del **NOC** instalada en el equipo.
- **No limit reconnections:** Al habilitarse, el cliente intentará restablecer la conexión tras una caída de red de forma indefinida, sin un límite máximo de peticiones.
- **Show notifications when assistants start working:** Esta función muestra una notificación visual con el nombre y la imagen del asistente cuando la instancia ejecuta un robot configurado en modalidad “asistente”.
- **Use Shell Mode:** Permite la ejecución de robots empaquetados en formato .ZIP (consulte la documentación).



Configuración General del Cliente

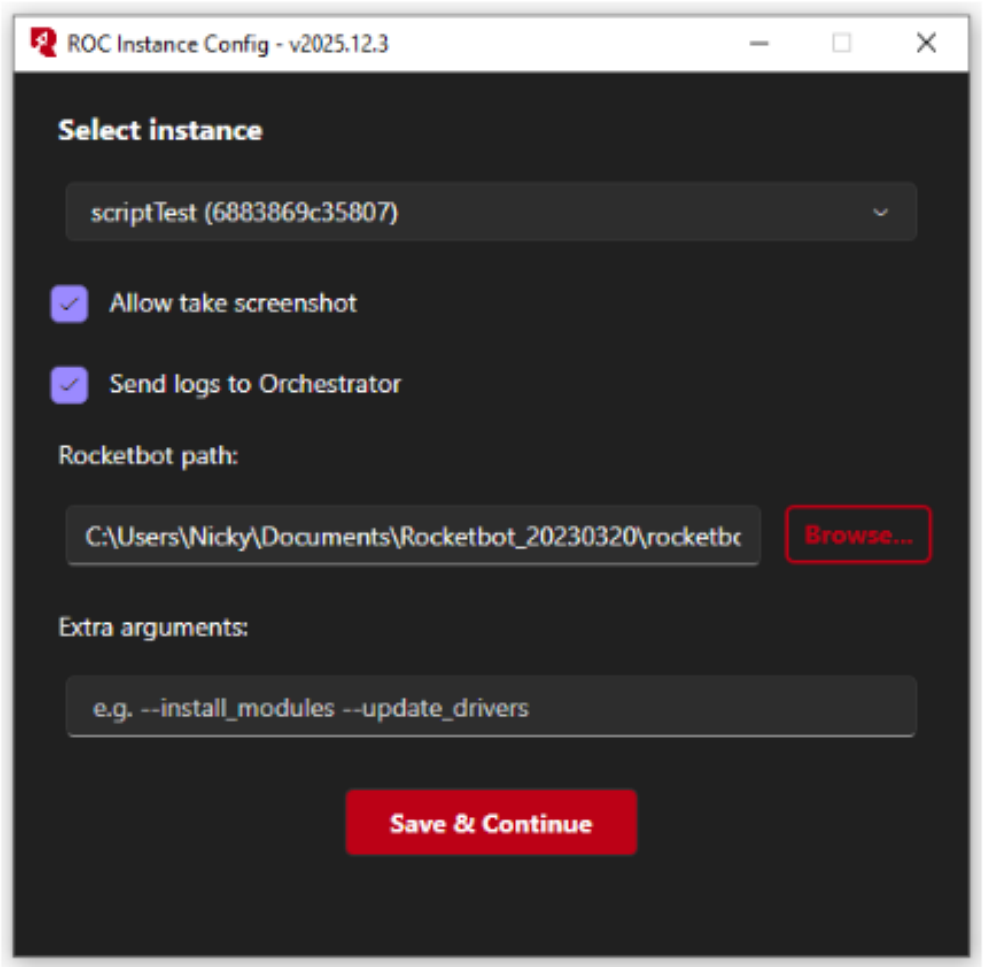
Estas opciones permiten definir el comportamiento inicial del software y las preferencias de conectividad:

Parámetro	Función
Run when Windows starts	Si está habilitado, el cliente se ejecutará automáticamente al iniciar el sistema operativo.
Login automatically (Only API KEY)	Permite iniciar sesión de forma automática al abrir el software si se utiliza una API KEY.
Run minimized	Ejecuta el cliente de forma minimizada en la bandeja del sistema.
No limit reconnections	Ante una pérdida de conexión, el NOC intentará reconectarse de forma indefinida sin límite de peticiones.

Parámetro	Función
Show notifications when assistants start working	Muestra el nombre y la imagen del asistente cuando la instancia ejecuta un robot configurado bajo esa modalidad.
Use Shell Mode	Habilita la utilización de robots en formato tipo .ZIP.
Proxy	Configuración de red para filtros de seguridad. Requiere el protocolo http, seguido de la dirección de salida y el puerto.
Reset	Restablece todos los valores a la configuración original de fábrica.

Parámetros de Instancia

Opciones específicas para la gestión de la instancia seleccionada y el rastro de ejecución:



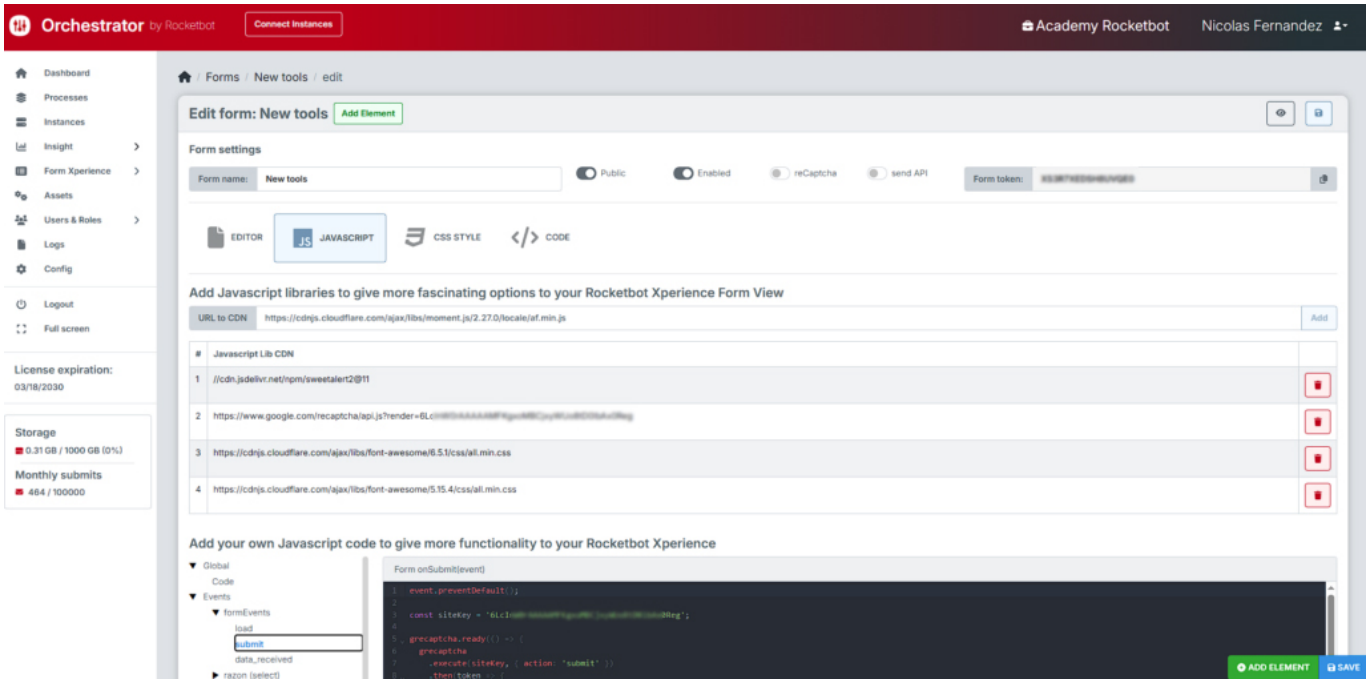
- **Select instance:** Menú desplegable para elegir la instancia que se desea vincular al equipo actual.
- **Allow take screenshot:** Permite capturar imágenes de la pantalla durante la ejecución para verificar procesos o errores visuales.(Ver **screenshot** en herramientas de instancias).
- **Send logs to Orchestrator:** Envía el registro de los comandos ejecutados por el robot hacia el Orquestador para su posterior consulta vía

getlogs.(Ver **“getlogs”** en herramientas de instancias).

- **Rocketbot path:** Ruta del directorio donde se encuentra el ejecutable de **Rocketbot Studio**.
- **Extra arguments:** Campo para ingresar comandos adicionales, como instalación de módulos o actualización de drivers.

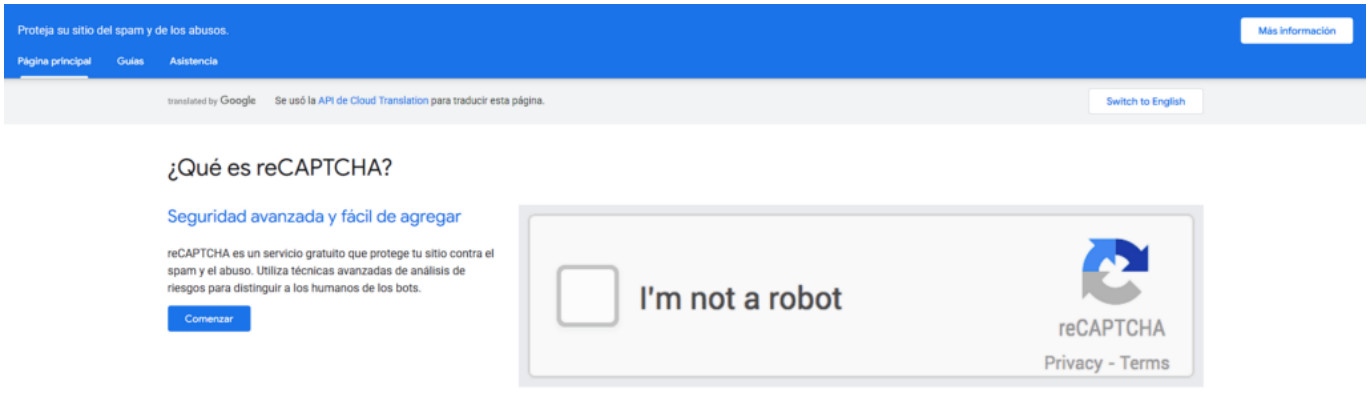
Activar Re-captcha en los formularios

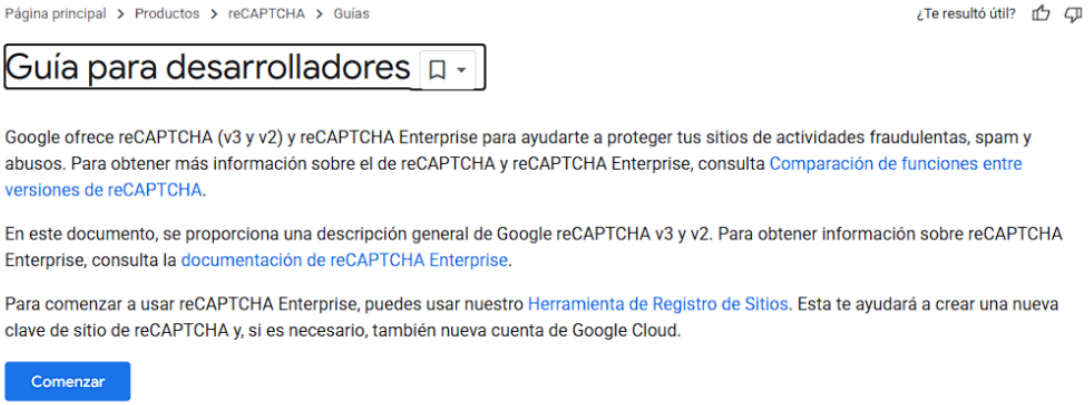
1° Ir al editor del formulario en la seccion JS.



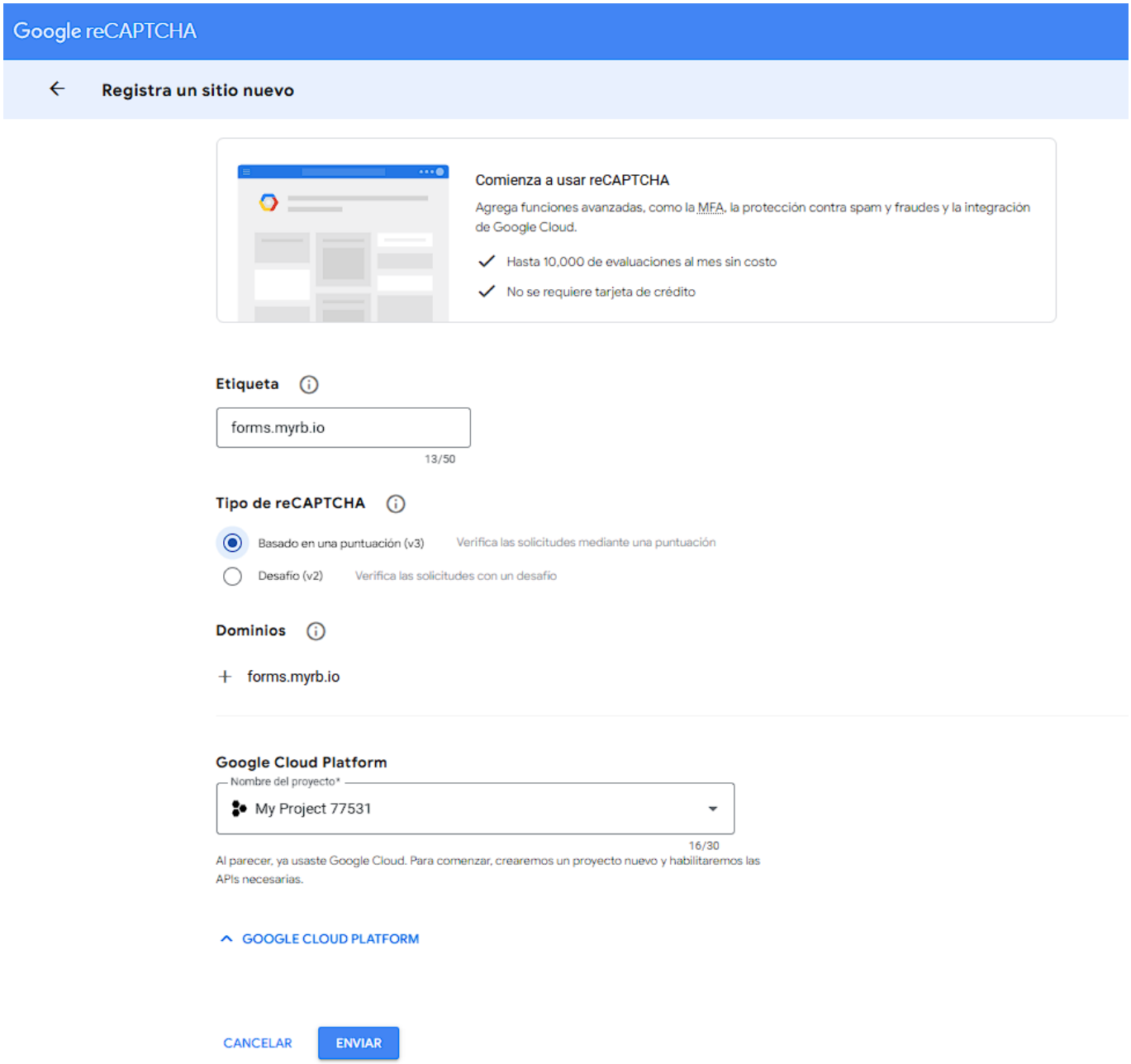
2° Crear la “clave recaptcha” en la pagina de recaptcha google(https://developers.google.com/recaptcha?hl=es-419).

3° Ir a “comenzar”, nos llevará a la guía para desarrolladores, y ahi tambien daremos en “comenzar”.





4° Registrar el formulario que quieres utilizar. (En dominio y en etiqueta coloque “forms.myrb.io”.



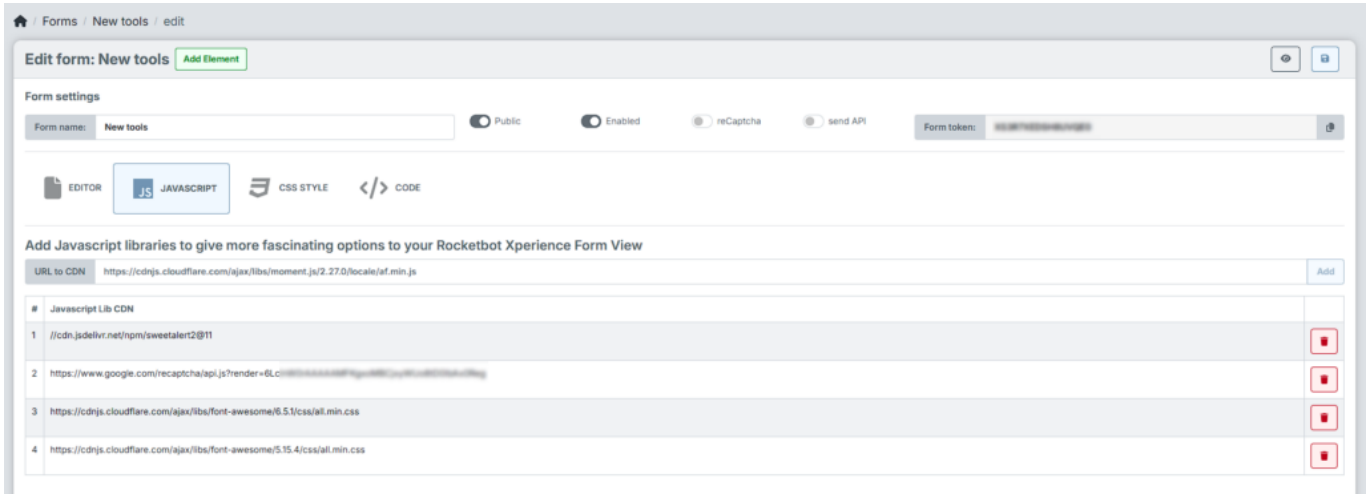
5° Configurar v3 (Verifica las solicitudes mediante una puntuación) o V2

(Verifica las solicitudes con un desafío).

Luego ir a “enviar” y nos redirigira a nuestra nueva “clave de sitio”.



6° En nuestro formumario ir a la pestaña “js” , agregar un nuevo CDN con “https://www.google.com/recaptcha/api.js?render= + nuestra clave de sitio”



7° En la terminal js -> Events/FormEvents/load tendremos que cargar el recapcha:

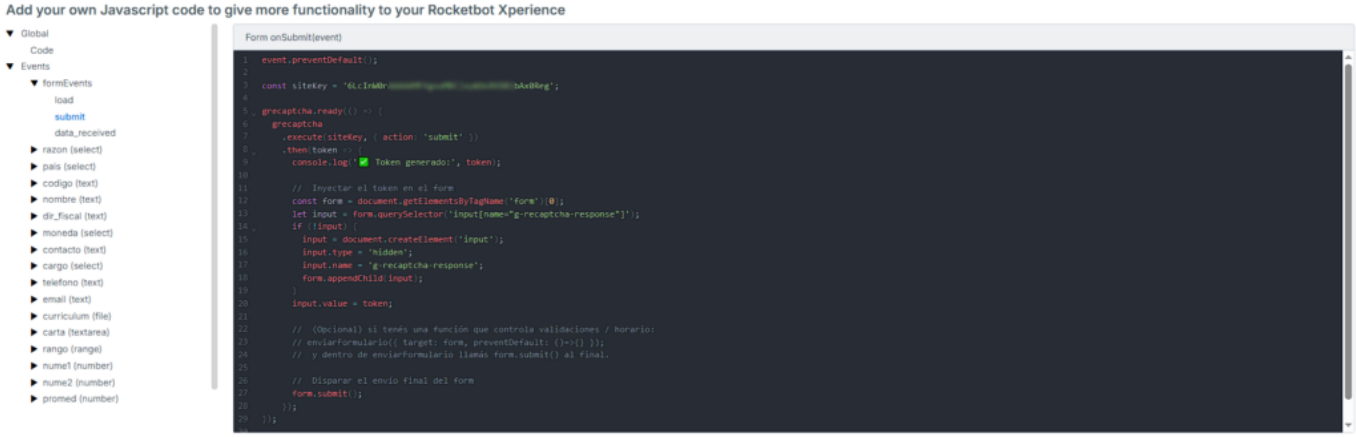
```
const siteKey = ‘nuestroCodigoSiteKey’;
// Verificamos si ya se cargó el script
if (typeof grecaptcha === ‘undefined’) {
const script = document.createElement(‘script’);
script.src = `https://www.google.com/recaptcha/api.js?render=${siteKey}`;
script.async = true;
script.defer = true;
```

```
document.head.appendChild(script);
}
```



8° En la terminal js -> Events/FormEvents/submit tendremos que ejecutar el recapcha:

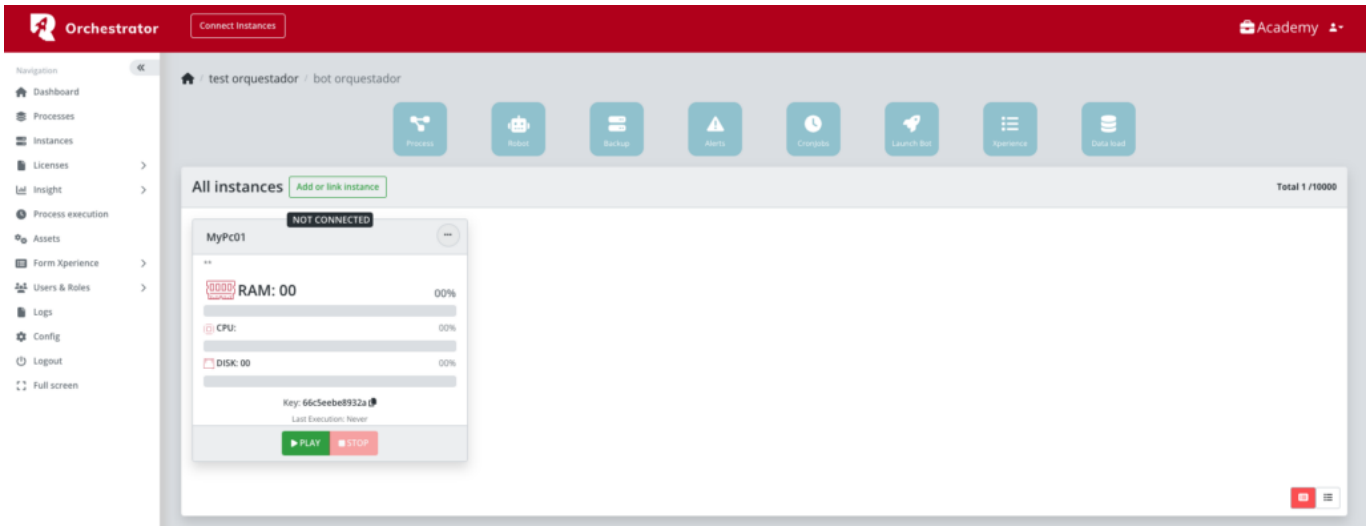
```
event.preventDefault();
const siteKey = ‘nuestroCodigoSiteKey’;
grecaptcha.ready(() => {
grecaptcha
.execute(siteKey, { action: ‘submit’ })
.then(token => {
console.log(‘ Token generado:’, token);
// Inyectar el token en el form
const form = document.getElementsByTagName(‘form’)[0];
let input = form.querySelector(‘input[name=”g-recaptcha-response”]’);
if (!input) {
input = document.createElement(‘input’);
input.type = ‘hidden’;
input.name = ‘g-recaptcha-response’;
form.appendChild(input);
}
input.value = token;
// (Opcional) si tenés una función que controla validaciones / horario:
// enviarFormulario({ target: form, preventDefault: ()=>{} });
// y dentro de enviarFormulario llamás form.submit() al final.
// Disparar el envío final del form
form.submit();
});
});
```



Asi podremos ingresar a nuestro propio recaptcha para que podamos ver la informacion de nuestra cuenta de google de los envios de informacion de nuestros formularios.

Orquestador Rocketbot: Cronjobs Advance

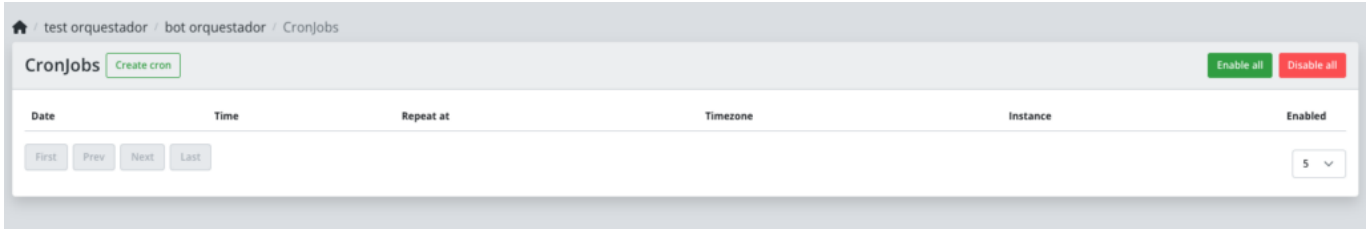
Un **Cronjob** es una tarea programada que permite ejecutar procesos de forma automática en fechas y horas específicas. Una vez que has creado un proyecto y asignado un proceso en el Orquestador, puedes configurar su periodicidad mediante esta herramienta.



Gestión de Cronjobs

Desde la interfaz de Cronjobs, tienes el control total sobre la automatización:

- **Habilitar/Deshabilitar**: Permite activar o pausar todos los cronjobs de forma masiva o individual.
- **Visualización Detallada**: Muestra la información técnica de cada tarea programada, como la instancia de ejecución y la zona horaria.
- **Creación**: Botón dedicado para configurar nuevas tareas programadas.

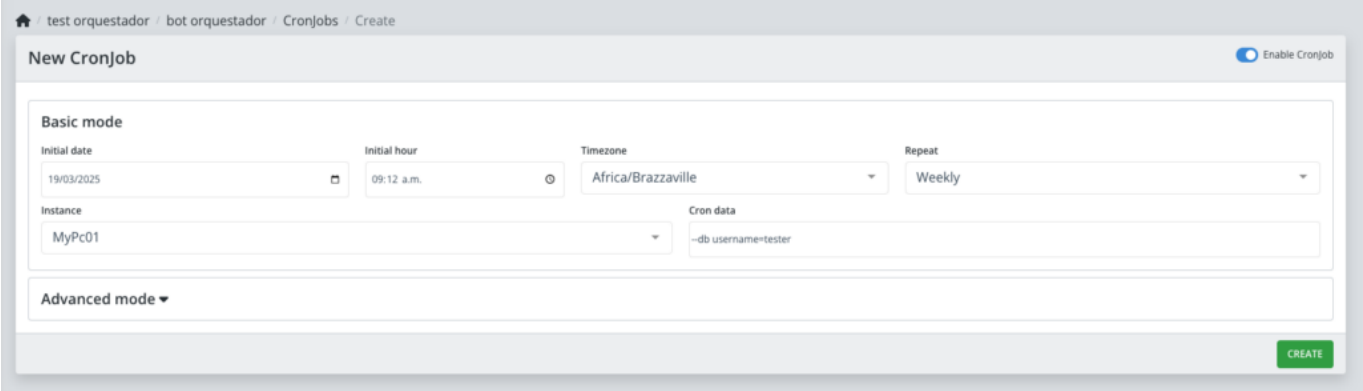


Modos de Configuración

Rocketbot ofrece dos niveles de personalización dependiendo de la complejidad de la tarea:

1. Basic Mode (Modo Básico)

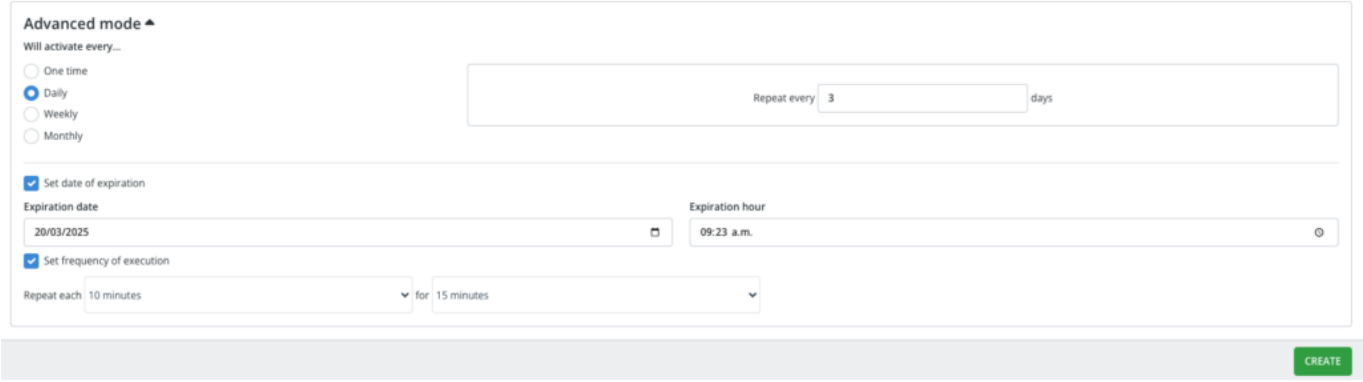
Diseñado para asignaciones rápidas y eficaces donde la periodicidad es estándar. En este modo se configuran los siguientes parámetros:



- **Initial Date**: Define el día, mes y año en que el proceso comenzará a ejecutarse.
- **Initial Hour**: Establece la hora exacta del primer arranque.
- **Timezone**: Parámetro crítico para asegurar que el robot se ejecute en el horario correcto según la región geográfica.
- **Repeat**: Permite elegir frecuencias predefinidas: diario, semanal, mensual, cada 5 o 10 minutos, o ejecución única (no repetir).
- **Instance**: Determina en qué instancia (computadora o servidor) se disparará la tarea.
- **Cron data**: En este apartado pueden ingresarse parámetros adicionales que serán enviados al robot cuando se ejecute a partir de su línea de comandos. Para obtener estos parámetros se puede usar el comando **Command line parameters** del módulo **System** nativo, o en caso de tener una versión anterior de Rocketbot, el módulo SystemAdvanced (**Get arguments**), asignando una variable para capturar la información. Command line también se une a cron data.

2. Advanced Mode (Modo Avanzado)

Este modo proporciona un control granular para escenarios donde la repetición requiere detalles específicos de tiempo.



- **Will activate every…**: Configuración personalizada para activaciones por intervalos de tiempo, días específicos de la semana o fechas mensuales.
- **This cronjob will execute one time on at**: Permite programar una ejecución única detallando con precisión absoluta la fecha y hora.
- **Set date of expiration**: Establece una fecha de “caducidad” para el cronjob, tras la cual dejará de ejecutarse automáticamente.
- **Set frequency of execution**: Control exhaustivo para definir la frecuencia de ejecución minuto a minuto.

En caso de querer mas detalles puede visitar la siguiente Documentación: [Trabajar con cronjobs]