

# Documentación Orquestador

## Índice

- **SEGURIDAD:**

- [Infraestructura.](#)
- [Autenticación.](#)
- [Conexión.](#)

- **LOGIN:**

- Obtener credenciales de acceso entregadas por Rocketbot o por su administrador.
- [Ingresar a Rocketbot Orquestador.](#)
- [Inicio de sesión incorrecto.](#)
- [Restaurar Contraseña.](#)
- [Bloqueo de sesión por intentos fallidos.](#)

- **CONFIGURACION DE PERFIL:**

- [Mi perfil.](#)
  - [Change Password.](#)
  - [Edit profile.](#)
  
- [Storage y envíos disponibles.](#)
- [Company Name.](#)
- Mi licencia.
- [Autenticación de 2 Pasos.](#)
- [Expiración User ApiKey.](#)

- **DASHBOARD:**

- [Crear proyecto.](#)
- [Crear proceso.](#)

- **PROCESS:**

- [Procesos.](#)
- [Interfaz.](#)
  - [Process.](#)
  - [Robot.](#)

- [Backup.](#)
- [Alerts.](#)
- [Cronjobs.](#)
- [Launch Bot.](#)
- [Xperience.](#)
- [Data Load.](#)

- [Procesos Privados.](#)

- **INSTANCES:**

- [Crear una instancia.](#)
- [Lista.](#)
  - [Connect to client.](#)
    - [Herramientas de una Instancia dentro del proceso.](#)
  
- [Detalles.](#)
- [Eliminar.](#)

- **Licenses:**

- **INSIGHT:**

- [Detalles.](#)
- [Schedules for instance.](#)
- [Execution chart.](#)

- **PROCESS EXECUTION:**

- [Ver ejecuciones en tiempo real de nuestros procesos.](#)

- **ASSETS:**

- [Lista de Assets.](#)
- [Crear Assets.](#)
- [Editar Assets.](#)
- [Eliminar Assets.](#)
- [¿Cómo pasar el valor del Asset a mi robot?](#)

- **FORMS XPERIENCE:**

- [Listado de formularios.](#)
  - [Crear un formulario.](#)
  - [Editar un formulario.](#)
  - [Eliminar un formulario.](#)
  
- [Configurar un formulario xperience.](#)
  - [Propiedades del formulario.](#)
  - [Propiedades del completado del formulario.](#)
  - [Pestaña de Editor.](#)
  - [Pestaña JS.](#)
    - [Funciones y eventos.](#)
  
  - [Pestaña CSS.](#)
  - [Pestaña del código.](#)
  
- [Utilizar xperience con un robot.](#)
- [APIs Xperience.](#)

- **QUEUES:**

- [Lista de Queues.](#)
- [¿Que son las Queues y como se utilizan?](#)

- **USER FORMS:**

- [Lista de User forms.](#)
- [Crear User form.](#)
- [Editar user form.](#)

- **USERS & ROLES:**

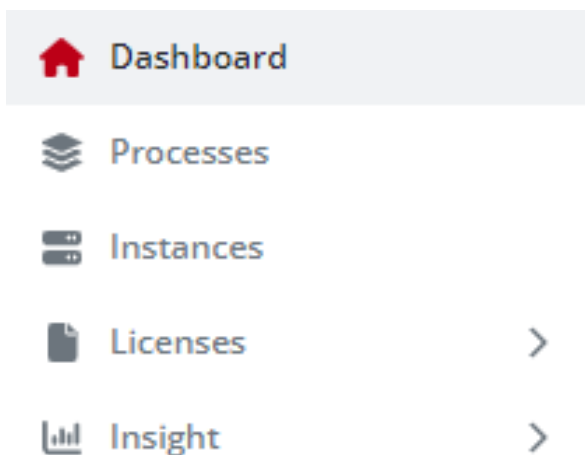
- [Lista de usuarios.](#)
  - [Crear usuario.](#)
  - [Eliminar usuario.](#)
  - [Generacion de Api Key.](#)
  - [Deshabilitar/habilitar usuario.](#)
  - [Editar/asignar Rol.](#)

- [Lista de roles.](#)
    - [Crear rol.](#)
    - [Eliminar rol.](#)
  
  - LOGS:
    - [Logs de usuarios.](#)
    - [Exportar registro.](#)
  
  - CONFIG:
    - [Zona horaria.](#)
    - [Deshabilitar usuario por inactividad.](#)
- 

## [Orquestador Rocketbot : Insight](#)

### ¿Dónde lo encuentro?

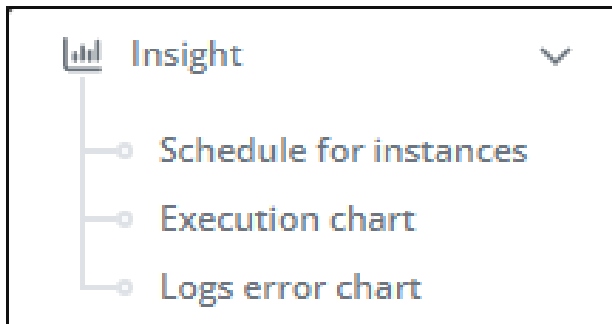
En el orquestador, en el menú lateral, encontraremos el apartado llamado **Insight**.



# ¿Para qué sirve?

En esta sección podremos visualizar varios gráficos que nos proporcionarán información sobre nuestros robots.

## Opciones de visualización



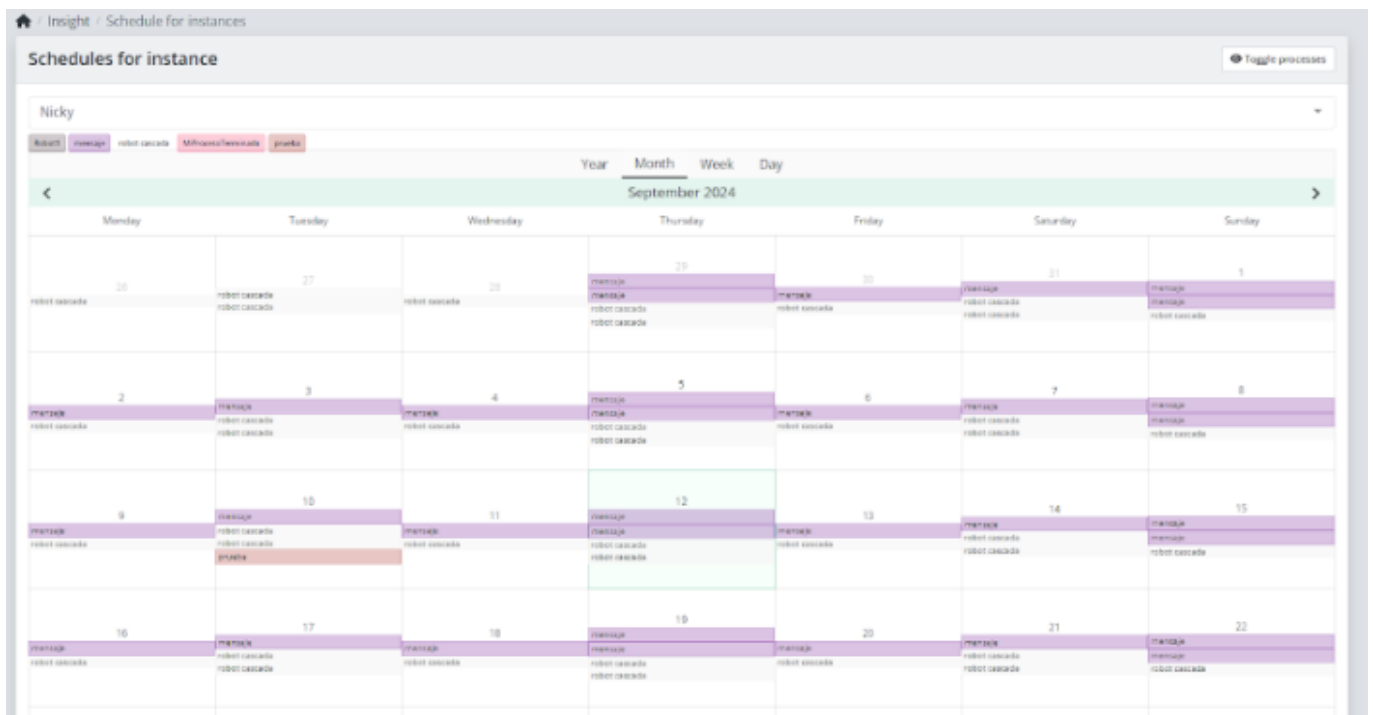
### Schedules for instance:

En esta pestaña tendremos un almanaque con un rango de fechas que definamos nosotros mismos (podremos seleccionar una vista anual, mensual, semanal o diaria).

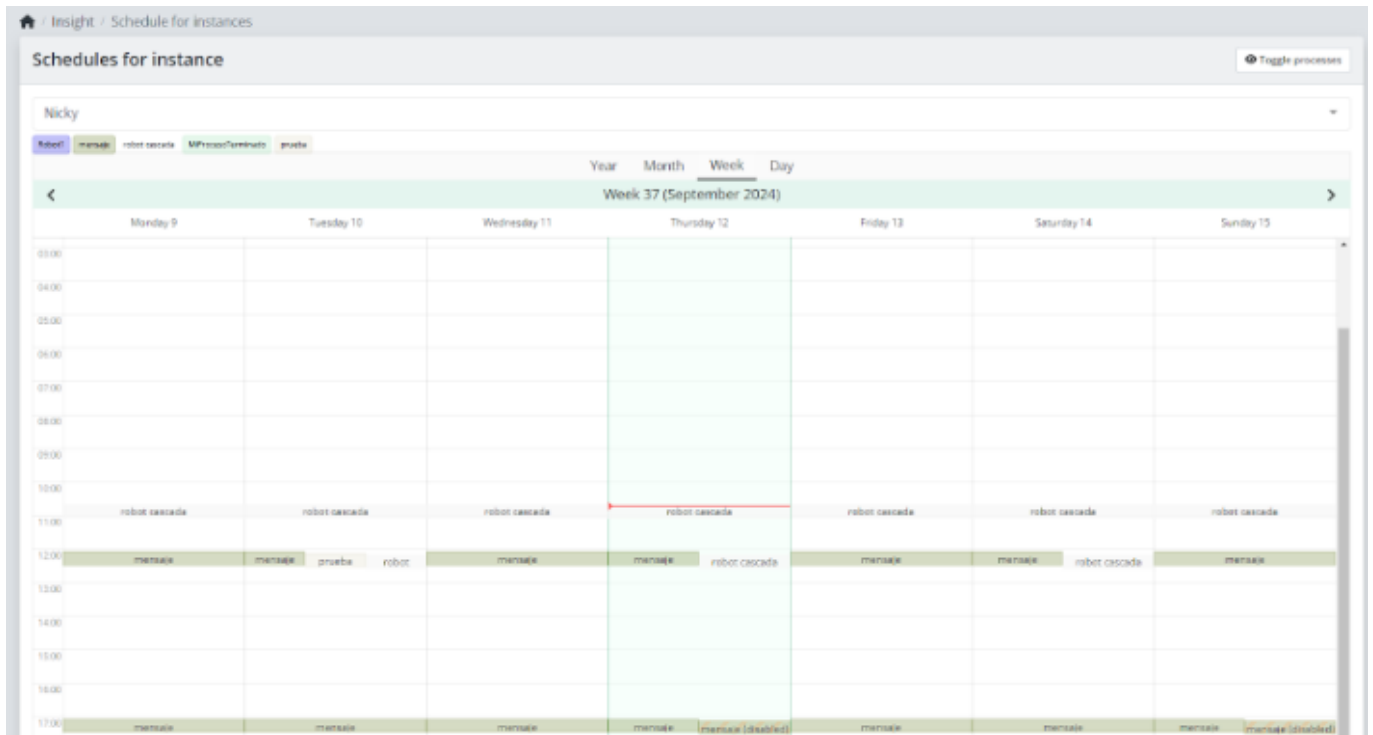
En él, podremos visualizar las ejecuciones de robots programadas mediante **cronjobs** dentro del rango que hemos determinado.

### Ejemplo de visualización

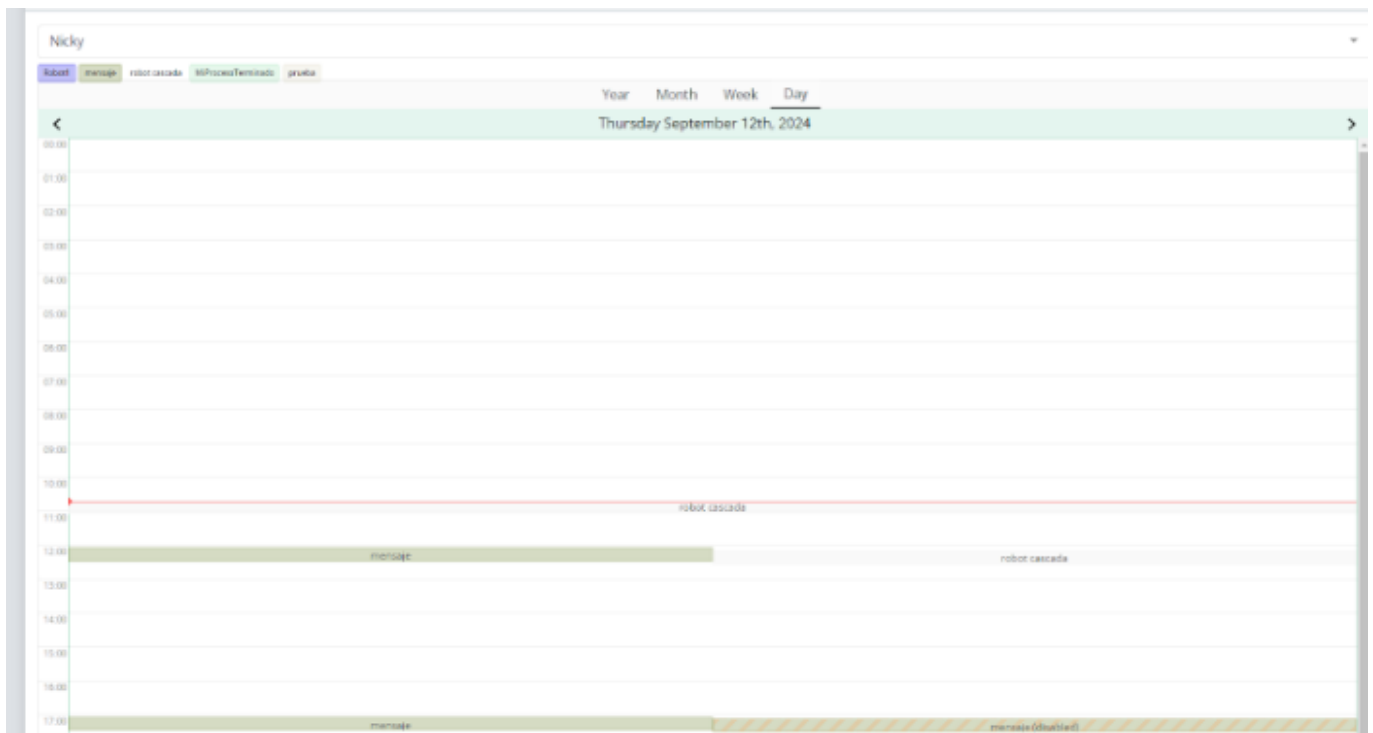
Por Mes



Por Semana



Por Día



También podremos ver el detalle de cualquier robot haciendo clic en él, lo que abrirá un modal con la información detallada.

## Event info

Status

Enabled

Start time

12:05

Timezone

America/Argentina/Buenos\_Aires

Repeat

Every 1 days

Start time according to timezone

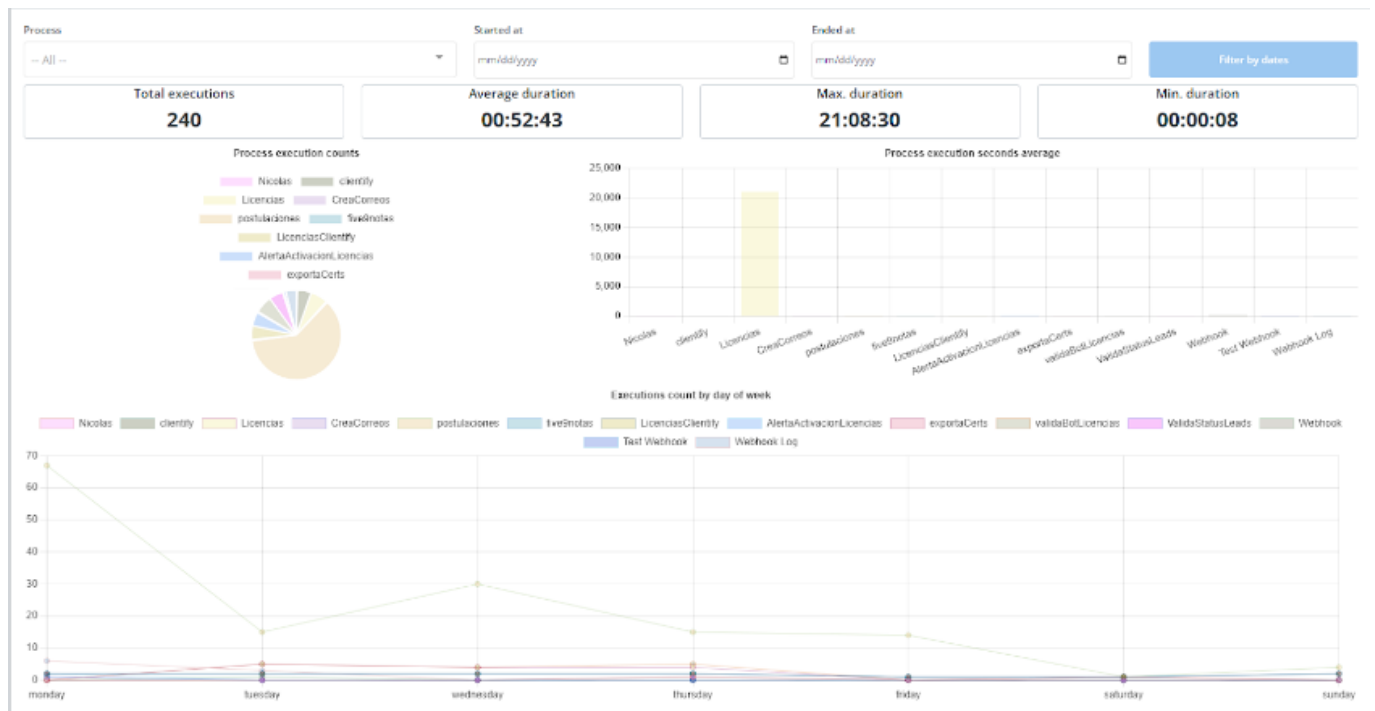
2024-09-12T12:05:43-03:00

Close

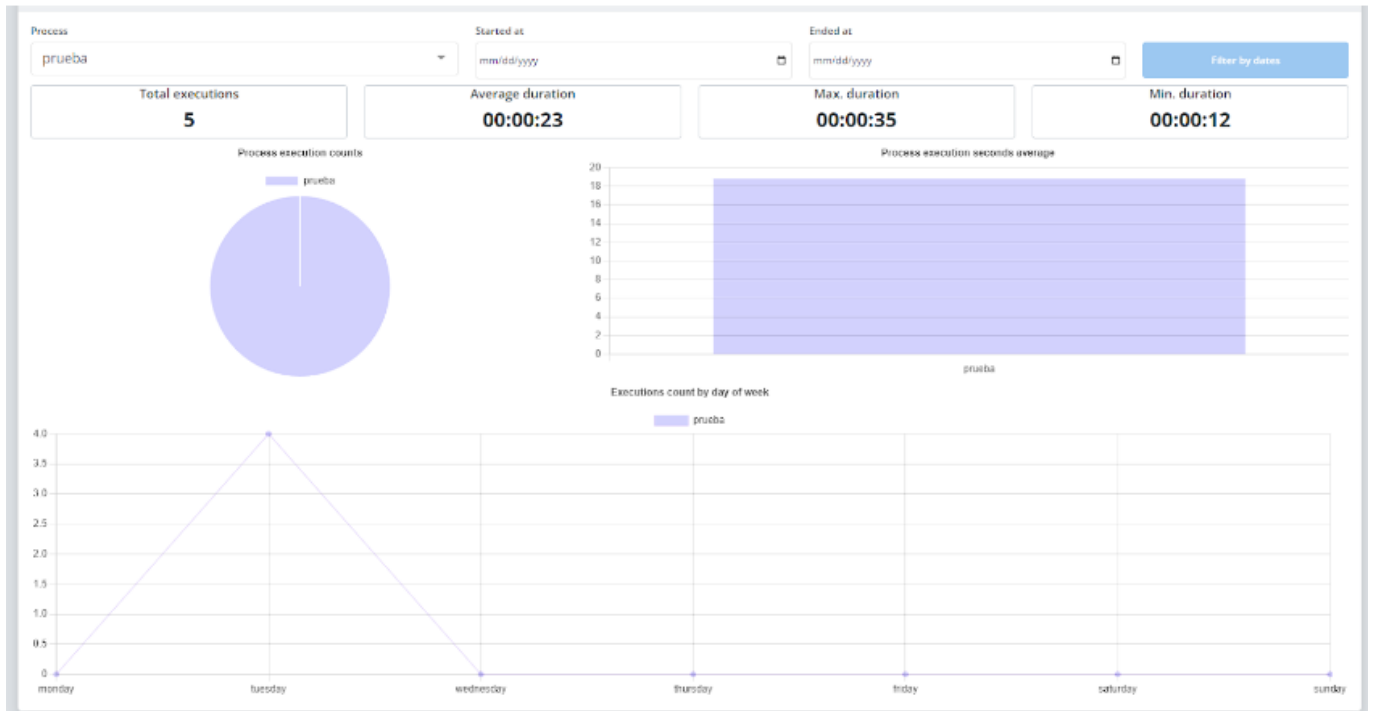
## Execution chart:

En esta pestaña tendremos varios gráficos para obtener información sobre la ejecución de nuestros robots.

Podremos ver el total de ejecuciones, el promedio de duración de estas, la duración más extensa de un robot y la duración más corta de los robots.



Al seleccionar un proceso, podremos ver con más detalle todas las ejecuciones de sus robots.



## Orquestador Rocketbot : Procesos Privados

Este sistema permite convertir procesos públicos en privados y viceversa.

### Diferencias entre procesos públicos y privados:

- En un **proceso público**, cualquier usuario registrado en la cuenta del cliente (es decir, cualquier usuario que figure en la tabla "users") puede ingresar.
- En un **proceso privado**, solo pueden acceder los usuarios que estén incluidos en la lista de "miembros" específica para ese proceso.

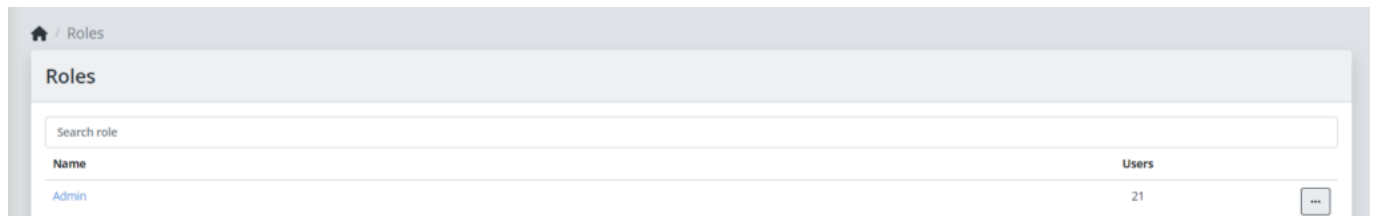
### Pasos para privatizar un proceso:

#### Verificar permisos necesarios

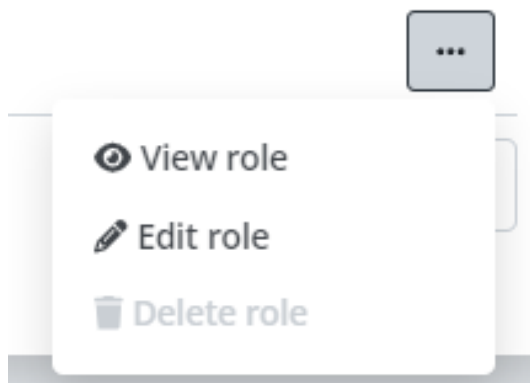
Antes de privatizar un proceso, es esencial contar con los permisos adecuados, incluyendo la capacidad de agregar "miembros".

Para esto, dirígete a la lista de users y roles , busca el rol asignado a tu

usuario ( para saber qué rol tienes, revisa la tabla )



En el menú lateral (tres puntos), se mostrarán las opciones para visualizar los permisos del rol. También podrás editar o eliminar el rol, siempre que tengas los permisos adecuados para realizar estas acciones.



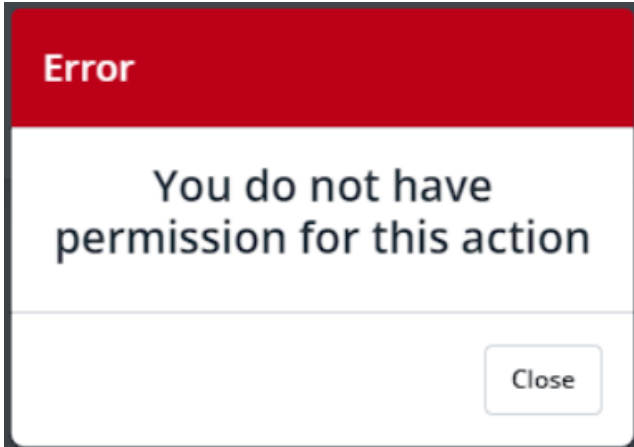
Al ingresar a la vista de permisos, podrás verificar si tienes habilitadas las siguientes acciones.

Admin								
<b>General</b>								
	Projects	Processes	Users	Roles	Instances	Assets	Forms	Configs
View	✓	✓	✓	✓	✓	✓	✓	-
Add	✓	✓	✓	✓	✓	✓	✓	-
Delete	✓	✓	✓	✓	✓	✓	✓	-
Modify	✓	✓	✓	✓	✓	✓	✓	✓
<b>Processes</b>								
Set CRON	Change robot	Link instances	Unlink instances	Run individual instances	Run all instances	View process execution	Modify members of proces	
✓	✓	✓	✓	✓	✓	✓	✓	✓
<b>Configs</b>								
View queues	Modify queues	View user forms	Modify user Forms	Change company name	View logs			
✓	✓	✓	✓	✓	✓			
<b>Users</b>								
Generate API key	Disable user							
✓	✓							

Asegúrate de que tu rol tenga habilitado el permiso "Modify members of process".

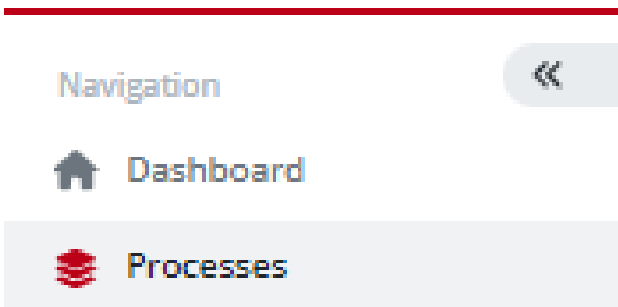
## Modify members of proces

Si no cuentas con este permiso, aparecerá el siguiente mensaje de advertencia:



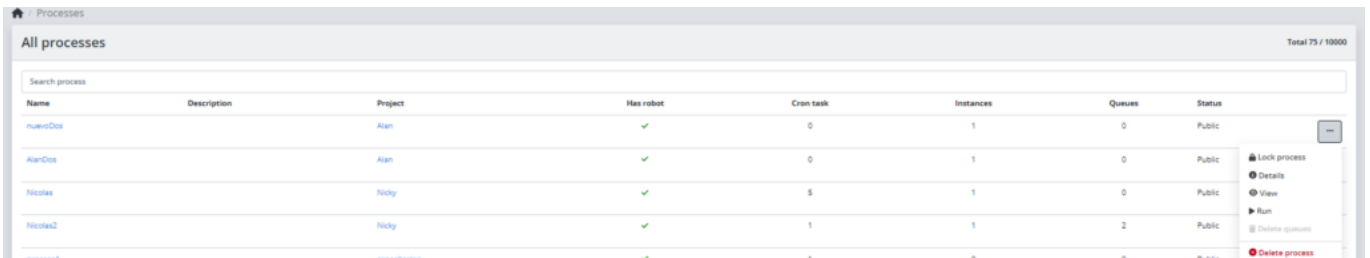
## Seleccionar el proceso

Una vez confirmados los permisos, dirígete a la lista de procesos o navega hasta el proceso del proyecto que deseas privatizar.



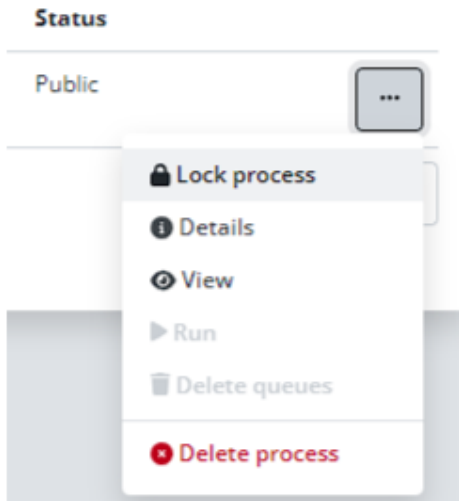
## Privatizar el proceso

Identifica el proceso que deseas privatizar, accede a su menú de herramientas (...), y selecciona la opción **“Lock process”**.



A screenshot of the "Processes" page in a web application. The page title is "All processes" and it shows a table with 7 columns: Name, Description, Project, Has robot, Cron task, Instances, Queues, and Status. There are 5 rows of process data. A dropdown menu is open for the first row, showing options: Lock process, Details, View, Run, Delete queues, and Delete process.

Name	Description	Project	Has robot	Cron task	Instances	Queues	Status
name00a		Alan	✓	0	1	0	Public
Alan00a		Alan	✓	0	1	0	Public
Nicky		Nicky	✓	5	1	0	Public
Nicky2		Nicky	✓	1	1	2	Public
proces1		capacitacion	✓	1	0	0	Public



Esta acción cambiará el estado del proceso a privado.

---

## Status

---

Private

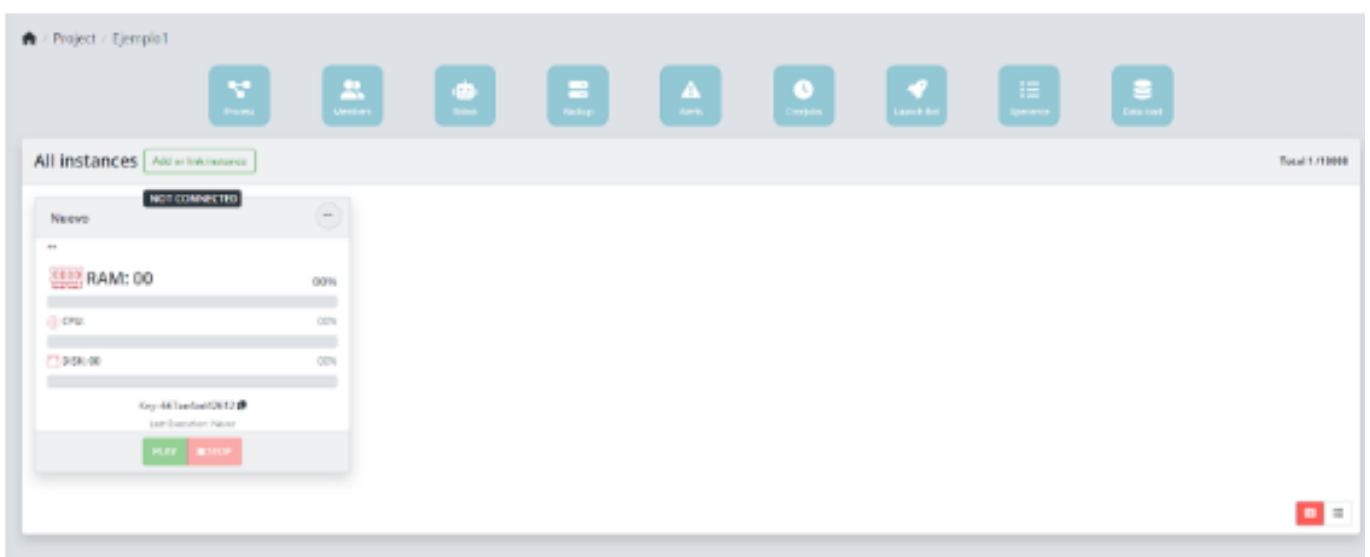
## Convertirse en administrador del proceso

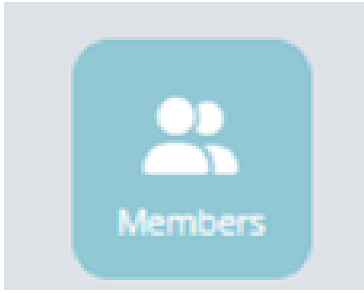
Al privatizar un proceso, automáticamente te conviertes en el “organizador” o “administrador” del mismo.

Esto te otorga acceso total al proceso y la capacidad de agregar miembros.

## Agregar miembros al proceso

Una vez dentro del proceso, aparecerá una nueva herramienta llamada “Members”.





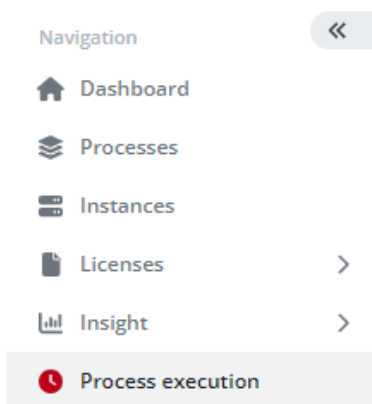
Al hacer clic en ella, se abrirá un modal con la lista completa de usuarios. Aquí podrás seleccionar a los usuarios que deseas agregar como miembros del proceso.

---

## [Orquestador Rocketbot : Process Execution](#)

### ¿Dónde lo encuentro?

Lo puedes ubicar en el menú lateral, con un ícono de reloj.



### Función

En esta sección podremos observar las ejecuciones en tiempo real de nuestros procesos.

Execution

### Process Execution Time

Started at: 10/09/2024 | Ended at: 17/09/2024 | [Filter by dates](#)

Search process execution:  | Process: --All-- | Instance: --All--

Process	Instance	Robot	Started at	Ended at	Duration
Nicolas	Nicolas	mensaje2	2024-09-05 22:05:48	2024-09-05 22:06:33	00:00:45
clientify	AMAZONBOTS	Clientify_Master	2024-09-01 06:00:14	2024-09-01 06:01:21	00:01:07
clientify	AMAZONBOTS	Clientify_Master	2024-09-02 06:00:06	2024-09-02 06:01:15	00:01:09
clientify	AMAZONBOTS	Clientify_Master	2024-09-03 06:00:08	2024-09-03 06:01:24	00:01:16
clientify	AMAZONBOTS	Clientify_Master	2024-09-04 06:00:07	2024-09-04 06:01:23	00:01:16
clientify	AMAZONBOTS	Clientify_Master	2024-09-05 06:00:13	2024-09-05 06:01:21	00:01:08
clientify	AMAZONBOTS	Clientify_Master	2024-09-06 06:00:14	2024-09-06 06:01:25	00:01:11
clientify	AMAZONBOTS	Clientify_Master	2024-09-07 06:00:15	2024-09-07 06:01:23	00:01:08
clientify	AMAZONBOTS	Clientify_Master	2024-09-08 06:00:10	2024-09-08 06:01:09	00:00:59
clientify	AMAZONBOTS	Clientify_Master	2024-09-09 06:00:09	2024-09-09 06:01:12	00:01:03

First | Prev | 1 | 2 | 3 | 4 | ... | Next | Last | 10 | [xlsx](#)

Se nos visualizarán los siguientes ítems:

1. Nombre del proceso.
2. Nombre de la instancia.
3. Nombre del robot.
4. Fecha y hora de inicio del robot.
5. Fecha y hora de final del robot.
6. Duración que tuvo el robot.

## Filtro de Búsqueda

De ser necesaria una búsqueda, tendremos a disposición los filtros de búsqueda, con los cuales contamos con

Process Execution Time

Started at: 10/09/2024 | Ended at: 17/09/2024 | [Filter by dates](#)

Search process execution:  | Process: --All-- | Instance: --All--

- Started at: Filtro de inicio de fecha de ejecución de robot.
- Ended at: Filtro de fin de fecha de ejecución de robot.
- Filter by dates: Botón para iniciar la búsqueda de robots que coincidan entre la o las fechas configuradas.
- Search process execution: Búsqueda general.
- Process: Filtrar por nombre de proceso.
- Instance: Filtrar por nombre de instancia.
- Boton "XLSX": Imprimir resultados en un Excel.

---

# Orquestador Rocketbot: APIs Xperience

## ¿Qué es una API?

Una API (Interfaz de Programación de Aplicaciones) es un conjunto de reglas, protocolos y herramientas que permiten a diferentes sistemas de software comunicarse e interactuar entre sí, de manera estructurada. Las direcciones URL a través de las cuales se envían solicitudes y se recibe información de las APIs se denominan **endpoints**.

## Requisitos Previos

- Acceso al Orquestador
- El usuario del Orquestador debe tener una [api Key generada](#).

## Endpoints disponibles

### Listar Proyectos

Devuelve un listado de los proyectos cargados en el Orquestador, con sus respectivos procesos y robots asignados.

Método URL	Parámetros
POST <a href="https://www.myrb.io/dev/api/project/list">https://www.myrb.io/dev/api/project/list</a>	<b>Authorization:</b> tipo Bearer Token; es el Token de acceso válido para autenticar al usuario (API Key asociada al perfil en el orquestador)

Avanzado

### Ejemplos de requests

- **cURL**

```
curl --location --request POST 'https://www.myrb.io/dev/api/project/list' \
--header 'Authorization: Bearer <token>'
```

- **Python**

```
import requests
```

```
url = "https://www.myrb.io/dev/api/project/list"

payload = {}
headers = {
  'Authorization': 'Bearer <token>'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

### Ejemplo de response

```
{
  "success": true,
  "data": [
    {
      "name": "test orquestador",
      "token": "JR0EYGNSEXEMTQWUP",
      "created_at": "2024-08-15 08:42:16",
      "id": 69,
      "description": null,
      "process_count": 1,
      "process": [
        {
          "id": 168,
          "project_id": 69,
          "token": "1KZQIJKTTR75CGWF",
          "name": "bot orquestador",
          "last_robot": {
            "id": 143,
            "name": "bot orquestador",
            "user_id": 67,
            "client_id": 3,
            "cron": null,
            "version": null,
            "version_rocketbot": null,
            "robot":
"myrb_clients/dev/project/69/process/168/robot/rocket.bot",
            "md5": "7af04cc5dbbea2e6ff1f27c183ef4772",
            "comment": null,
            "type": 1,
            "status": 0,
            "process_id": 168,
            "start_bot": "bot_orquestador",
            "created_at": "2024-08-21 09:37:50",
            "updated_at": "2024-08-21 09:37:50"
          },
          "backup": null,
          "trigger_process": null,
          "client": null,
```

```

    "robots": [
      {
        "id": 143,
        "name": "bot orquestador",
        "user_id": 67,
        "client_id": 3,
        "cron": null,
        "version": null,
        "version_rocketbot": null,
        "robot":
"myrb_clients/dev/project/69/process/168/robot/rocket.bot",
        "md5": "7af04cc5dbbea2e6ff1f27c183ef4772",
        "comment": null,
        "type": 1,
        "status": 0,
        "process_id": 168,
        "start_bot": "bot_orquestador",
        "created_at": "2024-08-21 09:37:50",
        "updated_at": "2024-08-21 09:37:50"
      }
    ]
  }
}

```

## Subir robot a un proceso existente

Permite asociar un robot a un proceso ya existente en el Orquestador.

Método URL

Formato

Parámetros

**Authorization:**  
tipo Bearer Token;  
es el Token de  
acceso válido para  
autenticar al  
usuario (API Key  
asociada al perfil  
en el  
orquestador).  
**name:** Nombre del  
proceso al que se  
asignará el robot.  
**startbot:** Nombre  
del robot como  
aparece en Studio.  
**process\_id:** Token  
del proceso. Hace  
referencia al  
**token** del proceso,  
se puede obtener  
desde el ROC ó  
utilizando el  
endpoint  
**/api/project/list.**  
**project\_id:**  
Identificador del  
proyecto al que  
pertenece el  
proceso. Debe  
obtenerse  
previamente  
utilizando el  
endpoint  
**/api/project/list.**  
**file:** robot.bd

POST <https://www.myrb.io/dev/api/process/robot/update/data> multipart/form-data

Avanzado

## Ejemplos de requests

- **cURL**

```
curl --location 'https://www.myrb.io/dev/api/process/robot/update/data' \  
--header 'Authorization: Bearer <token>' \  
--form 'name="process1"' \  
--form 'startbot="navegador"' \  
--form 'process_id="NDQGBJBSE4LDRSVA"' \  
--form 'project_id="56"' \  
--form 'file=@"/C:/Users/navegador.db"'
```

- **Python**

```
import requests
```

```
url = "https://www.myrb.io/dev/api/process/robot/update/data"
```

```
payload = {'name': 'process1',  
'startbot': 'navegador',  
'process_id': 'NDQGBJBSE4LDRSVA',
```

```
'project_id': '56'}
files=[
('file', ('navegador.db', open('/C:/Users/navegador.db', 'rb'), 'application/octe
t-stream'))
]
headers = {
  'Authorization': 'Bearer <token>'
}
```

```
response = requests.request("POST", url, headers=headers, data=payload,
files=files)
```

```
print(response.text)
```

### Ejemplo de response

```
{
  "success": true,
  "data": {
    "robot": "myrb_clients/dev/project/56/process/152/robot/rocket.bot",
    "md5": "484871810fe6e7ed78b4bacb53a408b3",
    "start_bot": "navegador",
    "status": 0,
    "type": 1,
    "name": "process1",
    "user_id": 66,
    "process_id": 152,
    "client_id": 3,
    "updated_at": "2024-09-18 16:58:14",
    "created_at": "2024-09-18 16:58:14",
    "id": 145,
    "process": {
      "id": 152,
      "name": "process1",
      "token": "NDQGBJBSE4LDRSVA",
      "robot_id": 131,
      "config": null,
      "client_id": 3,
      "data": null,
      "status": 0,
      "minutes": null,
      "sent_email": null,
      "email": null,
      "trigger": null,
      "created_at": "2024-08-05 17:14:14",
      "updated_at": "2024-08-05 13:14:14",
      "project_id": 56,
      "form_id": null,
      "private": 0,
      "last_robot": {
        "id": 145,
```

```
    "name": "process1",
    "user_id": 66,
    "client_id": 3,
    "cron": null,
    "version": null,
    "version_rocketbot": null,
    "robot":
"myrb_clients/dev/project/56/process/152/robot/rocket.bot",
    "md5": "484871810fe6e7ed78b4bacb53a408b3",
    "comment": null,
    "type": 1,
    "status": 0,
    "process_id": 152,
    "start_bot": "navegador",
    "created_at": "2024-09-18 16:58:14",
    "updated_at": "2024-09-18 16:58:14"
},
"backup": null,
"trigger_process": null,
"client": {
    "id": 3,
    "name": "Academy",
    "owner": null,
    "parent": null,
    "type": 2,
    "status": 0,
    "created_at": "2022-09-05 11:07:44",
    "updated_at": "2023-01-03 11:04:21"
},
"robots": [
    {
        "id": 131,
        "name": "process1",
        "user_id": 66,
        "client_id": 3,
        "cron": null,
        "version": null,
        "version_rocketbot": null,
        "robot":
"myrb_clients/dev/project/56/process/152/robot/rocket.bot",
        "md5": "5594faf198e491c7868cf600204f39d2",
        "comment": null,
        "type": 1,
        "status": 0,
        "process_id": 152,
        "start_bot": "saludo",
        "created_at": "2024-08-05 17:14:14",
        "updated_at": "2024-08-06 10:38:46"
    },
    {
        "id": 145,
```



```
--data-urlencode 'process_id=CEDLU6FK79BOLPUM'
```

- **Python**

```
import requests

url = "https://www.myrb.io/dev/api/process/robots/start"

payload = 'process_id=CEDLU6FK79BOLPUM'
headers = {
    'Authorization': 'Bearer <token>',
    'Content-Type': 'application/x-www-form-urlencoded'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

### **Ejemplo de response**

```
{
  "success": true
}
```

### **Obtener todas las queues**

Devuelve un listado de las queues de todos los [formularios de Xperience](#) existentes en el Orquestador, pudiendo además filtrar por fecha de inicio y fin de las mismas.

Método URL	Formato	Parámetros
------------	---------	------------

**Authorization:**  
tipo Bearer  
Token; es el  
Token de  
acceso válido  
para  
autenticar al  
usuario (API  
Key asociada  
al perfil en  
el  
orquestador).  
**from:** Fecha de  
inicio de la  
búsqueda  
(YYYY-MM-DD).  
**to:** Fecha de  
fin de la  
búsqueda  
(YYYY-MM-DD)

POST <https://www.myrb.io/dev/api/formData/all> application/json

Avanzado

## Ejemplos de request

- cURL

```
curl --location 'https://www.myrb.io/dev/api/formData/all' \  
--header 'Authorization: Bearer <token>' \  
--header 'Content-Type: application/json' \  
--data '{  
  "from": "2024-06-9",  
  "to": "2024-07-16"  
'
```

- Python

```
import requests  
import json  
  
url = "https://www.myrb.io/dev/api/formData/all"  
  
payload = json.dumps({  
  "from": "2024-06-9",  
  "to": "2024-07-16"  
})  
headers = {  
  'Authorization': 'Bearer <token>',  
  'Content-Type': 'application/json'  
}
```

```
response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

### Ejemplo de response

```
{
  "success": true,
  "data": [
    {
      "id": 8315,
      "form_name": "five9notas",
      "form_token": "6HNZNSYVG4W2MCS0",
      "created_at": "2024-07-10 11:04:55",
      "processed_at": "2024-07-10 11:06:26",
      "status": 1,
      "user_form_email": null,
      "locked_user": "Bot Integracion",
      "locked": 0,
      "form_id": 25
    },
    {
      "id": 8316,
      "form_name": "Clientify",
      "form_token": "F9J65RTNWWPMTYD8",
      "created_at": "2024-07-11 10:25:35",
      "processed_at": "2024-07-11 10:27:13",
      "status": 1,
      "user_form_email": null,
      "locked_user": "Robot Interno",
      "locked": 0,
      "form_id": 14
    }
  ]
}
```

### Añadir datos a queue

Completa un formulario de Xperience y coloca los datos en cola para ser procesados (Debe tener activa la opción "send API" en el formulario).

Método URL

Formato

Parámetros

**Authorization:**  
tipo Bearer  
Token; es el  
Token de  
acceso válido  
para  
autenticar al  
usuario (API  
Key asociada  
al perfil en  
el  
orquestador).  
**Dato1:**  
**Dato2:**  
**(Otros Datos):**  
Se pueden  
incluir campos  
adicionales  
según el  
diseño del  
formulario.

**https://www.myrb.io/dev/api/formData/addQueue/:token**  
POST **token:** Representa el token de formulario creado en multipart/form-data el  
ROC Xperience.

Avanzado

## Ejemplo de request

- **cURL**

```
curl --location 'https://www.myrb.io/dev/api/formData/addQueue/:token' \  
--header 'Authorization: Bearer <token>' \  
--form 'email="prueba@prueba.com"' \  
--form 'nombre="prueba1"'
```

- **Python**

```
import requests  
  
url = "https://www.myrb.io/dev/api/formData/addQueue/:token"  
  
payload = {'email': 'prueba@prueba.com',  
          'nombre': 'prueba1'}  
files=[  
  
]  
headers = {  
    'Authorization': 'Bearer <token>'  
}  
  
response = requests.request("POST", url, headers=headers, data=payload,  
                             files=files)  
  
print(response.text)
```

## Ejemplo de response

```
{
  "success": true,
  "data": {
    "xperience":
"eyJpdjI6Inpid0RRVlV0bEczTW9DM1Z5SEFrV1E9PSIsInZhbHVlIjo1OVJCZHhHTDBwQ0p1Y2ps
VU9WZTYxbE5paFwvRTQ4cWlKSfZ3UUxGcVNUQXh4UTBx0FEwc01Ba0pvV1kwQklRdXByTE1JU2tWM
TMxZkx5Qk41SFNTNmdRPT0iLCJtYWMiOiI5YjFiNjAyMTRiODUwMjk3YjdjZGEwZjFmMjE4MjhiNG
I5MzA5MTViNmVhMTE4OTI0NmExNTgwYWE4YWJlYWNjIn0="
  }
}
```

## Obtener todos los queues id

Devuelve un listado con los IDs de todas las queues con estado "Not Processed" de un formulario de Xperience existente.

Método URL

Parámetro

**Authorization:**

tipo Bearer Token; es el Token de acceso válido para autenticar al usuario (API Key asociada al perfil en el orquestador).

POST **https://www.myrb.io/dev/api/formData/get/:token:**  
**:token:** Representa el token de formulario creado en ROC Xperience.

Avanzado

### Ejemplos de request

- **cURL**

```
curl --location --request POST
'https://www.myrb.io/dev/api/formData/get/:token' \
--header 'Authorization: Bearer <token>'
```

- **Python**

```
import requests

url = "https://www.myrb.io/dev/api/formData/get/:token"

payload = {}
headers = {
  'Authorization': 'Bearer <token>'
}
```

```
response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

### Ejemplo de response

```
{
  "success": true,
  "data": [
    {
      "id": 4734,
      "form_name": null,
      "user_form_email": null,
      "locked_user": null,
      "form_token": null,
      "form": null,
      "user_form": null,
      "locker": null
    },
    {
      "id": 4735,
      "form_name": null,
      "user_form_email": null,
      "locked_user": null,
      "form_token": null,
      "form": null,
      "user_form": null,
      "locker": null
    }
  ]
}
```

### Cambiar de estado una queue

Permite modificar el estado de un queue.

Método URL

Formato

Parámetro

**Authorization:**  
tipo Bearer  
Token; es el  
Token de  
acceso válido  
para  
autenticar al  
usuario (API  
Key asociada  
al perfil en  
el  
orquestador).  
**status:** Un  
número entero  
que indica el  
nuevo estado  
de la cola. 0:  
No procesado  
1: Procesado  
**lock:** Un nú 0:  
No bloqueada  
1: Bloqueada

POST **https://www.myrb.io/dev/api/formData/setStatus/:id:**  
:id: Representa el ID de la cola que se desea  
modificar. Este ID debe obtenerse previamente  
utilizando el endpoint **/formData/get/:token.**  
multipart/form-data

Avanzado

## Ejemplos de request

- **cURL**

```
curl --location 'https://www.myrb.io/dev/api/formData/setStatus/:id' \  
--header 'Authorization: Bearer <token>' \  
--form 'status="1"' \  
--form 'locked="0"'
```

- **Python**

```
import requests  
  
url = "https://www.myrb.io/dev/api/formData/setStatus/:id"  
  
payload = {'status': '1'}  
files=[  
  
]  
headers = {  
    'Authorization': 'Bearer <token>'  
}  
  
response = requests.request("POST", url, headers=headers, data=payload,  
files=files)  
  
print(response.text)
```

## Ejemplo de response

```
{
  "success": true
}
```

## Obtener datos de un formulario

Permite recuperar los datos de un formulario específico que se encuentra dentro de una queue determinada. Es útil para procesar los datos de un formulario una vez que ha sido añadido a la queue.

Método URL

Parámetros

	<b>https://www.myrb.io/dev/api/formData/getQueue/:id/:token</b>	<b>Authorization:</b>
POST	<b>:id:</b> Representa el ID de la cola. Este ID debe obtenerse previamente utilizando el endpoint <b>/formData/get/:token</b> .	tipo Bearer
	<b>:token:</b> Representa el token de formulario creado en ROC Xperience.	Token; es el
		Token de
		acceso válido
		para
		autenticar al
		usuario (API
		Key asociada
		al perfil en
		el
		orquestador).

Avanzado

## Ejemplos de request

- **cURL**

```
curl --location --request POST
'https://www.myrb.io/dev/api/formData/getQueue/:id/:token' \
--header 'Authorization: Bearer <token>'
```

- **Python**

```
import requests

url = "https://www.myrb.io/dev/api/formData/getQueue/:id/:token"

payload = {}
headers = {
  'Authorization': 'Bearer <token>'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

## Ejemplo de response

```
{
  "success": true,
  "data": {
    "id": 8728,
    "data": "{\"Dato1\":\"prueba\",\"Dato2\":\"prueba1\"}",
    "user_form_email": null,
    "xperience":
"eyJpdii6InNxMXhRekhVUUJrWVptcGpz0UM1aXc9PSIsInZhbHVlIjoiY1NoMGtpcW5udk5vTmZx
VlVYZFg1V29CUGU4SDB1UVUwWuXfuYVNmb2FVNkE2OE5MMnRwelorWGxtMU9VZ2pjTGJnSUNLaE5cL
1ArSkpPdk1BendRYVlvTlMyU0QwclN0E2VENmSDEyZTJRPSIsIm1hYyI6ImYyYjc2ZmM30TAwMW
QzNDVlMmViNmM1NTIxNjc0OTc0Y2IwOTA2Yjg4YTMwODNlODUwNzU1NWQ4MmM2ZGRmMjcifQ=="
  }
}
```

## Obtener Asset

Se utiliza para solicitar información de un asset almacenado en el orquestador. La búsqueda se realiza filtrando por nombre e instancia a la que pertenece.

Método URL	Formato	Parámetros
POST	<a href="https://www.myrb.io/dev/api/assets/get">https://www.myrb.io/dev/api/assets/get</a>	<b>Authorization:</b> tipo Bearer Token; es el Token de acceso válido para autenticar al usuario (API Key asociada al perfil en el orquestador). <b>name:</b> El nombre del asset. <b>instance:</b> Key de la instancia, tal como se define en el archivo noc.ini.

Avanzado

## Ejemplos de request

- cURL

```
curl --location 'https://www.myrb.io/dev/api/assets/get' \
--header 'Authorization: Bearer <token>' \
--data-urlencode 'name=toNotify' \
```

```
--data-urlencode 'instance=66b1418f93213'
```

- **Python**

```
import requests

url = "https://www.myrb.io/dev/api/assets/get"

payload = 'name=toNotify&instance=66b1418f93213'
headers = {
    'Authorization': 'Bearer <token>',
    'Content-Type': 'application/x-www-form-urlencoded'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

### **Ejemplo de response**

```
{
  "success": true,
  "data": {
    "id": 31,
    "name": "toNotify",
    "type": "password",
    "value": "prueba@prueba.com",
    "client_id": 3,
    "process_id": 0,
    "instance_id": 0,
    "created_at": "2023-03-21 17:12:35",
    "updated_at": "2023-10-25 12:21:49"
  }
}
```

---

## **[Rocketbot Xperience – Funciones y eventos](#)**

*nota: este video será subido a youtube, con una intro y subtítulos*

---

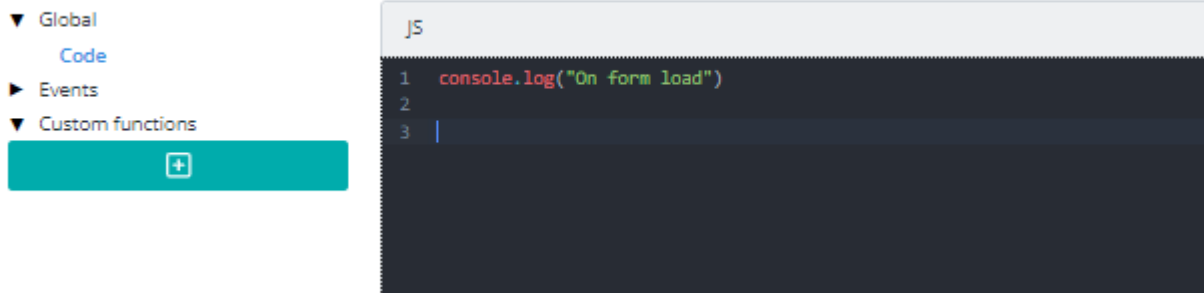
En cada form de Xperience, se podrán crear funciones y asignarlas a eventos

de input y de form.

## Código global

Se encuentra en la pestaña JAVASCRIPT. Código que afecta a todo el form y se ejecuta cuando se carga. Es retroactivo con el código Javascript de versiones de Orquestador anteriores.

Add your own Javascript code to give more functionality to your Rocketbot Xperience



## Eventos

Evento	Descripción	Parámetros
<i>load</i>	Se ejecuta cuando se carga el formulario.	evento
<i>submit</i>	Cuando se completa y se envía la información del formulario.	evento
<i>data_received</i>	Cuando se recibe información extra mediante api.	xperience, data

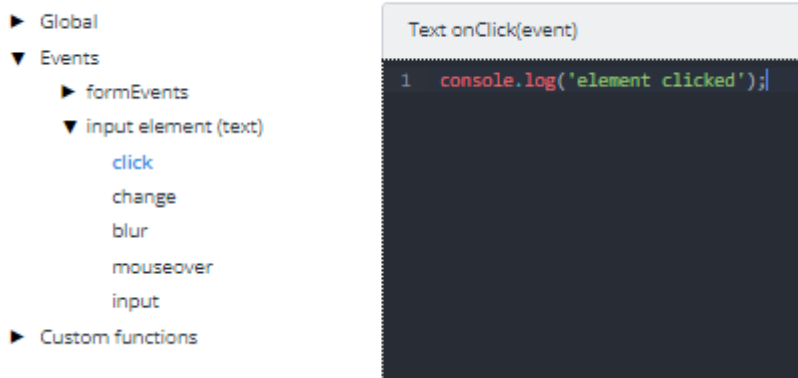
## Eventos de elemento

Sólo los elementos de form del grupo de Inputs y los elementos Select tendrán asociados eventos.

Evento	Descripción	Parámetros
<i>On click</i>	Se ejecuta cuando se clickea el elemento	evento
<i>On change</i>	Se ejecuta cuando el valor del input cambia y pierde el foco	evento
<i>On blur</i>	Se ejecuta cuando un elemento previamente enfocado pierde el foco	evento
<i>On mouse over</i>	Se ejecuta cuando se pasa el mouse por encima del elemento	evento
<i>On input</i>	Se ejecuta inmediatamente después de que el valor del input cambia	evento

En la pestaña JAVASCRIPT aparecerán en la lista todos los elementos creado que posean eventos. Se le podrá asignar código personalizado a cada evento de cada elemento.

## Add your own Javascript code to give more functionality to



## Funciones custom

El usuario podrá crear sus propias funciones, código reutilizable. Se encuentra en la pestaña JAVASCRIPT, bajo "custom functions".

The dialog box "Create custom function" contains the following elements:

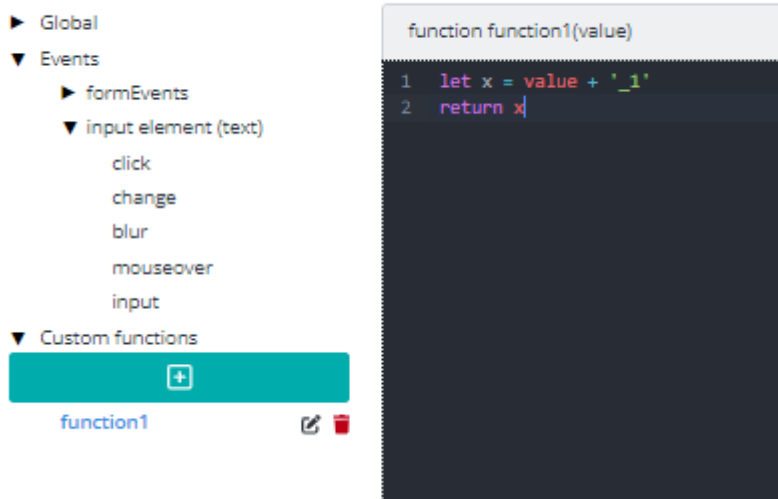
- Function name:
- Starts with a letter, \_ or \$ ✗
- Contains letters, numbers, \_ or \$ ✗
- Do not contain empty spaces ✗
- Do not use reserved words ✓
- Do not repeat function name ✓
- Function parameters:
- Enter a parameter and press enter a comma or a whitespace to create it
- Buttons: Cancel, Create

Las funciones requieren de un nombre.

- Tiene que comenzar con una letra, \_ o \$. Puede contener números, pero no como primer carácter.
- No tiene que tener espacios vacíos entre caracteres.
- No puede tener de nombre [palabras reservadas](#).
- No puede haber dos funciones con el mismo nombre.

No es obligatorio, pero cada función puede tener uno o más parámetros. Cada parámetro debe cumplir las mismas condiciones que el nombre de función (empezar en letra, \_, \$; sin espacios vacíos, sin [palabras reservadas](#), no puede haber dos parámetros repetidos en la misma función).

## Add your own Javascript code to give more functionality



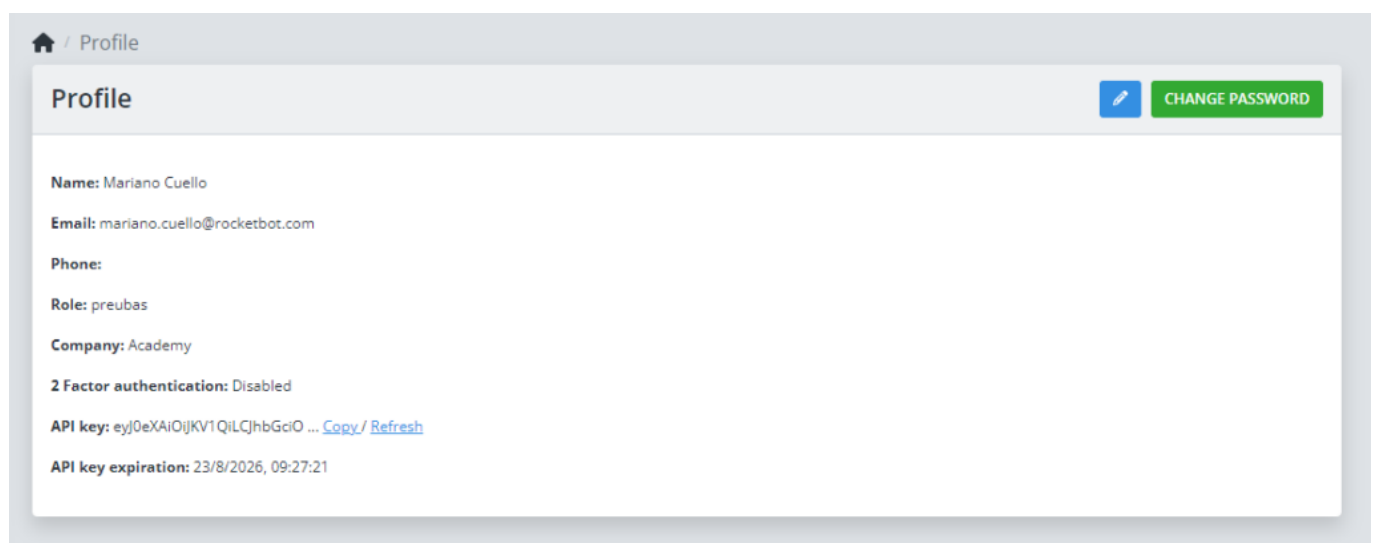
Se puede editar el código que pertenecerá exclusivamente a la función.

■ Tener cuenta que el código de las funciones no se ejecutará a menos que sean llamadas en el código Global, en los eventos de form, o en los eventos de algún elemento.

---

## [Orquestador Rocketbot : Autenticacion de 2 Pasos](#)

Para la autenticacion de 2 pasos se debe acceder al perfil y en el menú , debera dirigirse al boton de edit profile .



Si requiere informacion sobre el perfil puede leer el siguiente documento : [Orquestador Rocketbot : Mi Perfil](#)

Una vez en el menu del Edit Profile debe hacer lo siguiente

### Edit profile

Name

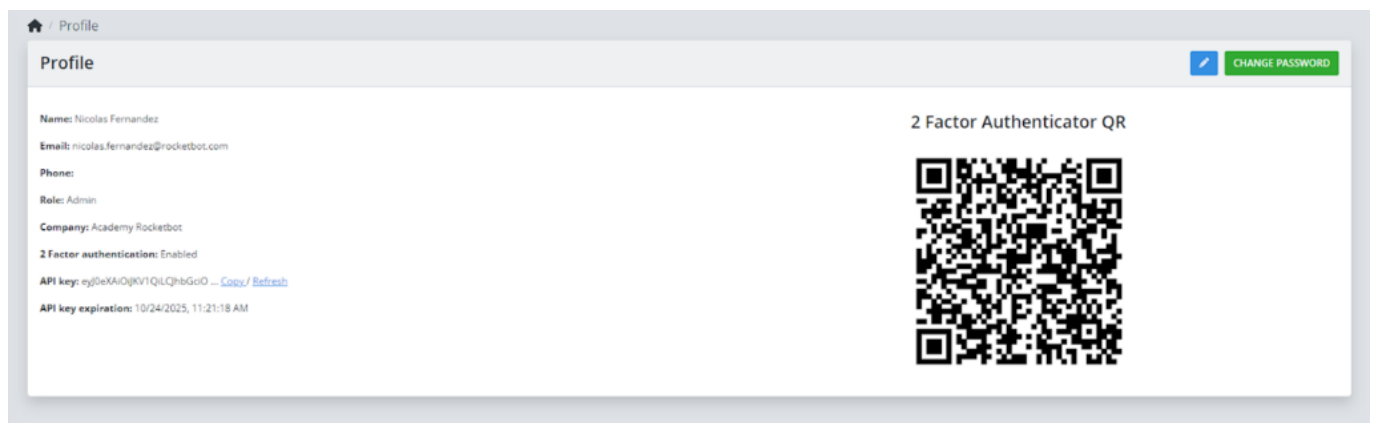
Phone

2 Factor authentication

---

## Autenticación en dos pasos

Una vez que se seleccione la opción "2 Factor Authentication" aparecerá el siguiente código QR sobre el perfil



Luego deberá ingresar desde su celular a la aplicación google authenticator, seleccionar la opción escanear código QR y escanear el código QR que aparece en su profile.

Una vez realizados los pasos anteriormente mencionados verá que cuando vuelva a iniciar sesion se le pedirá un código como se puede ver en la siguiente imagen

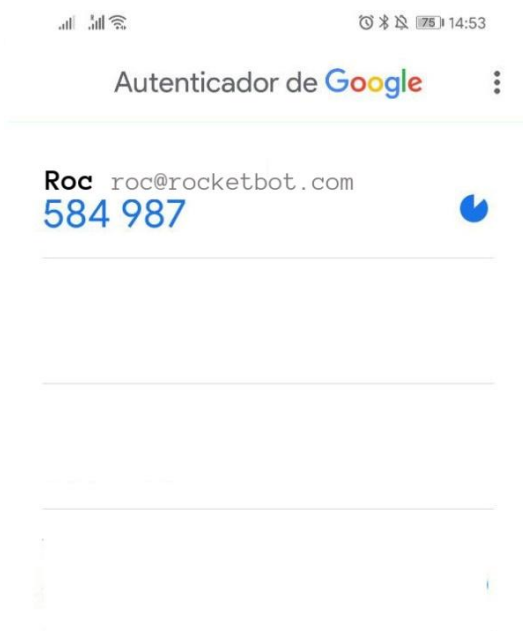
# ROCKETBOT

## Orchestrator

Code verification

Verificate

Ahora ingrese a google authenticator



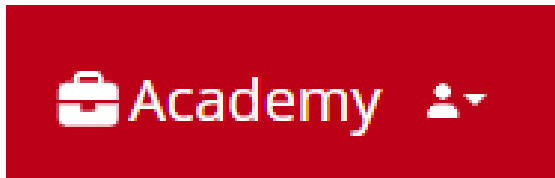
Obtenga el código de verificación e ingréselo en Rocketbot Orchestrator para iniciar sesión con doble autenticación.

---

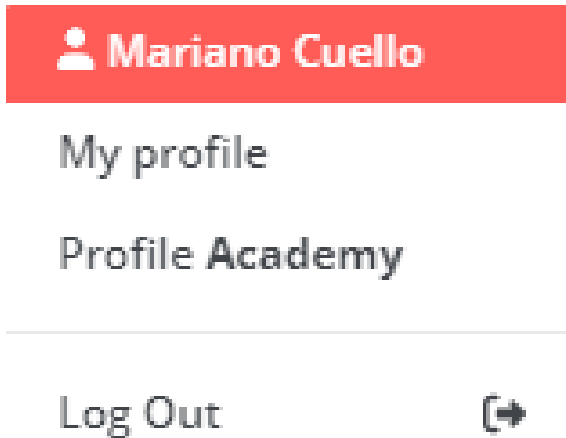
# Orquestador Rocketbot : Mi Perfil

## ¿Dónde encuentro mi perfil?

En el Orquestador, en el apartado que se encuentra en el borde superior derecho, encontrará el botón “Mi perfil”. Al lado de este botón se mostrará la empresa asociada.



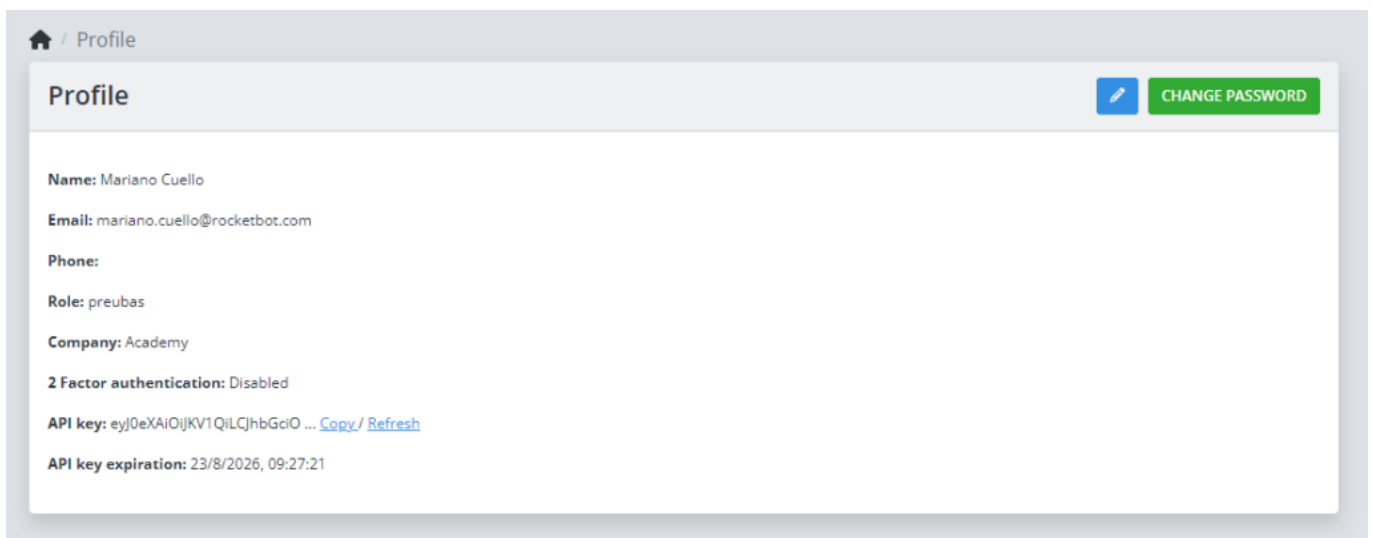
Al hacer clic en “Mi perfil”, se desplegará el siguiente menú:



Opciones disponibles:

## My profile

Aquí podrá observar los datos del perfil seleccionado.

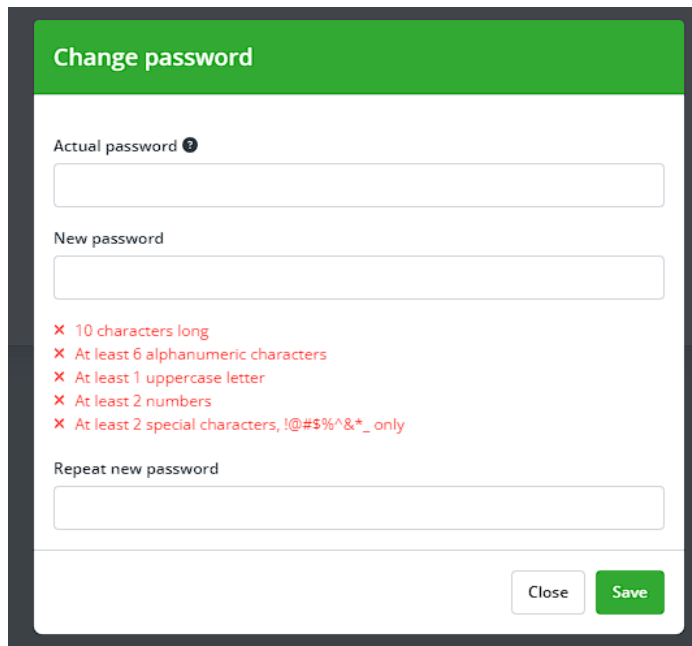


En el perfil podrá hacer lo siguiente:

## Change Password

Para cambiar la contraseña, dirígete a “Change Password”. Se abrirá un modal donde deberá introducir la contraseña actual, luego la nueva y repetir la nueva contraseña.

La contraseña debe cumplir con las condiciones impuestas; cuando se cumplen, la condición cambia de color rojo a verde, indicando que se está cumpliendo con dicha condición.



The image shows a modal window titled "Change password" with a green header. It contains three input fields: "Actual password" (with a help icon), "New password", and "Repeat new password". Below the "New password" field, there are five red error messages, each preceded by a red 'X' icon:

- × 10 characters long
- × At least 6 alphanumeric characters
- × At least 1 uppercase letter
- × At least 2 numbers
- × At least 2 special characters, !@#%&\*\_ only

At the bottom right of the modal, there are two buttons: "Close" and "Save".

## Edit profile

Aquí puede editar la información, activar o desactivar la autenticación en 2 pasos, ingresar un número telefónico, y, si lo desea, cambiar el nombre.

Si desea saber más sobre la autenticación en 2 pasos, puede consultar el siguiente documento : [Orquestador Rocketbot : Autenticacion 2 pasos](#)

## Edit profile

Name

Mariano Cuello

Phone

2 Factor authentication

Disabled

Close

Save

## Company Name

Al hacer clic en el nombre de la empresa, aparecerá una ventana con información sobre el tipo de cuenta que tiene. Junto al nombre hay un botón para modificar el nombre de la empresa, siempre y cuando tenga los permisos necesarios para hacerlo.

También se visualizarán las limitaciones de la cuenta, que incluyen un número limitado de usuarios, proyectos, procesos, instancias y formularios. Además, se mostrará un contador de los assets utilizados (los cuales no están limitados).

## Company info

 **Academy**



License expiration: 04/07/2026


Users

 29/100


Projects

 21/21

Processes

 80/100

Instances

 21/100


Forms

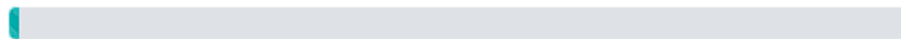
 44/100

Assets

 13

Storage

 0.98 GB / 150 GB (1%)



Clear all files

Clear processed files

Monthly submits

 200 / 15000

Close

## Log Out

Para salir de la sesión, seleccione "Log Out".

## Informacion extra

Si necesita información sobre cómo acceder al Orquestador, puede leer el siguiente documento : [Acceder al Orquestador Rocketbot .](#)

---

# Orquestador Rocketbot: Bloqueo de sesión por intentos fallidos

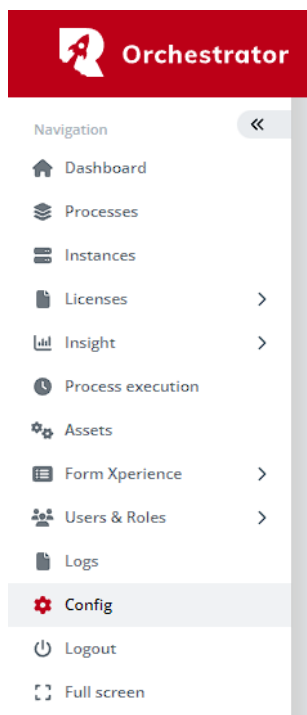
## Función

Sirve para bloquear temporalmente el inicio de sesión de un usuario en caso de que se ingresen usuario o contraseña incorrectos después de 5 intentos. Durante el período de bloqueo, aunque se ingrese la contraseña correcta, no se permitirá iniciar sesión hasta que finalice el bloqueo temporal

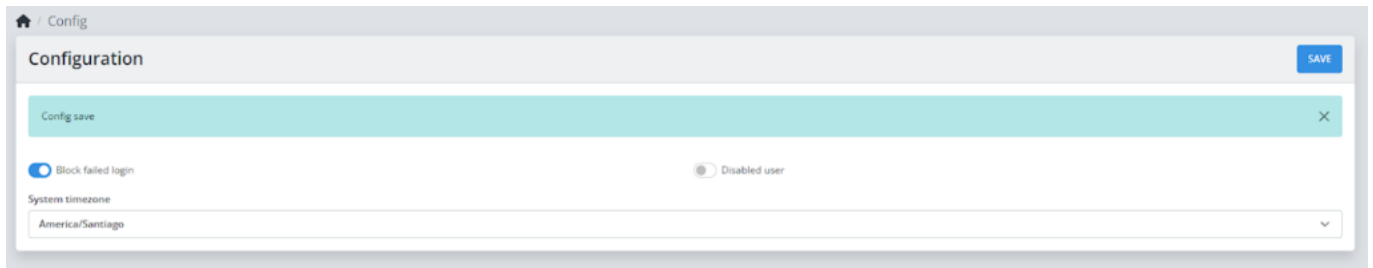
El tiempo de bloqueo es directamente proporcional a la cantidad de intentos fallidos de logeo que se tienen. Empieza con un (1) minuto y aumenta.

## Modo de activación

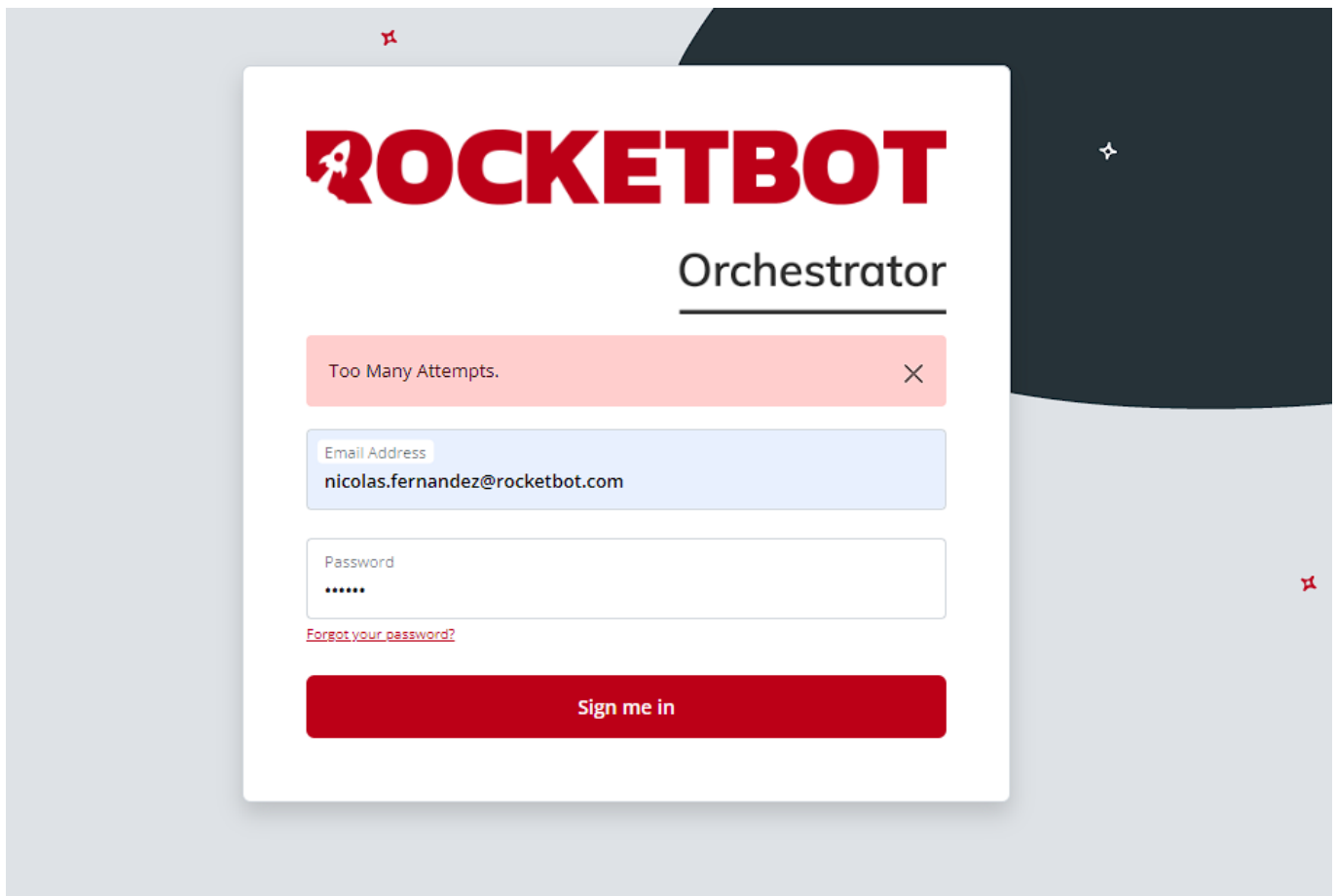
1. Ir a la pestaña “config”:



2. Activar el “Block failed login” Y guardar con el botón “save”.



3. Mientras el “Block failed login” este activado, cada vez que se ingrese 5 veces erróneamente un usuario o contraseña. Se bloqueara temporalmente mostrando el siguiente mensaje:



4. Durante el período de bloqueo por más que se ingrese la contraseña correcta tampoco lo dejará iniciar sesión, hasta que pase el bloqueo temporal.

# Orquestador Rocketbot : Ingresar al Orquestador

## Ingresar a Rocketbot Orquestador

- Ingrese las credenciales de acceso proporcionadas por Rocketbot o por su administrador.
- Introduzca el usuario y la contraseña proporcionados por Rocketbot.
- Haga clic en "Login".

**ROCKETBOT**  
Orchestrator

Email Address

Password

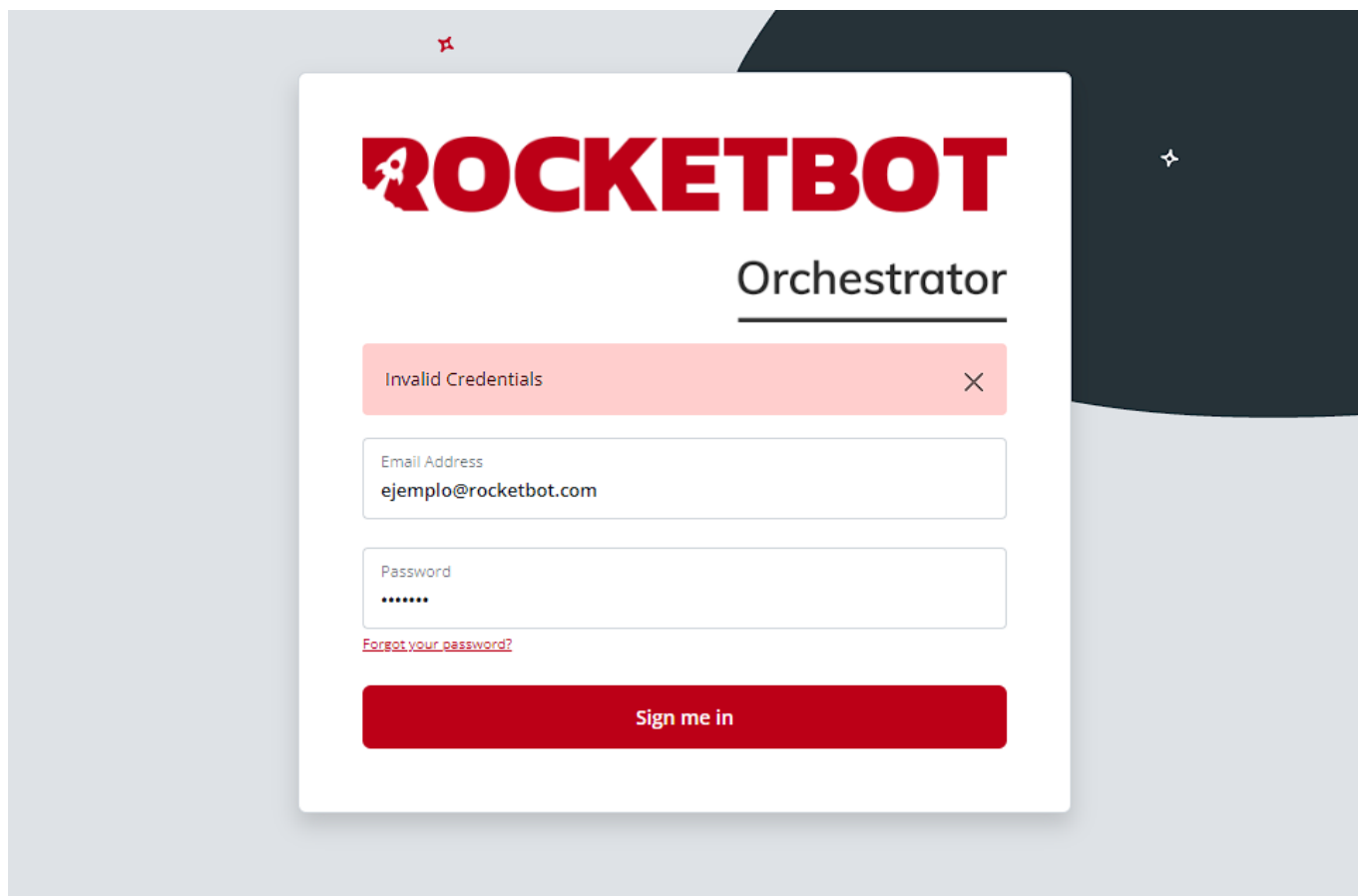
[Forgot your password?](#)

Sign me in

## Inicio de sesión incorrecto

Si se ingresan credenciales incorrectas, será notificado a través de un mensaje que le informará que las credenciales ingresadas no coinciden con ningún dato en nuestros registros.

En caso de que se intente acceder repetidamente sin éxito y esté activado el modo de bloqueo de sesión ([Ver](#)), solo se permitirán 5 intentos. Luego, deberá esperar 1 minuto para poder reintentarlo. Después, el tiempo de espera aumentará con cada intento adicional. Lo recomendable es RESTAURAR LA CONTRASEÑA.



The image shows a login form for 'ROCKETBOT Orchestrator'. At the top left of the form is the Rocketbot logo, a red rocket icon. Below the logo, the word 'ROCKETBOT' is written in large, bold, red capital letters, and 'Orchestrator' is written in a smaller, black font below it. A red error message box with a close button (X) is displayed, containing the text 'Invalid Credentials'. Below the error message are two input fields: 'Email Address' with the value 'ejemplo@rocketbot.com' and 'Password' with a masked password '\*\*\*\*\*'. A red link labeled 'Forgot your password?' is located below the password field. At the bottom of the form is a large red button with the text 'Sign me in'.

## Restaurar Contraseña

Si tiene problemas para recordar su contraseña, tiene la opción "Forgot your password?" que lo redireccionará a la siguiente vista:

 **ROCKETBOT**

## Orchestrator

---

**Email**

RESET PASSWORD

En este formulario, debe ingresar el email asociado a su usuario y hacer clic en "Reset password". Luego, dirígete a la casilla de correo electrónico que ingresó anteriormente, donde recibirá un correo para restablecer la contraseña. Debe seguir los pasos que aparecen en el correo para obtener su nueva contraseña.

En caso de necesitar información sobre su perfil, puede acceder al siguiente documento: [Orquestador Rocketbot : Mi Perfil](#)

Si desea obtener información sobre el bloqueo del perfil por intentos fallidos en el inicio de sesión, puede acceder al siguiente documento: [Bloqueo del perfil por intentos fallidos](#)