

Subir archivos con extensión .zip al robot

¿Cómo subir los archivos?

En el modal de creación/edición de proceso, además de poderse subir archivos con extensión .db, se podrán subir archivos con extensión .zip.




Para subir archivos con extensión .zip debemos dirigirnos al modal de creación/edición de robot.

Add process

Process name	Start robot name	Upload DB
<input type="text" value="Robot zip"/>	<input type="text" value="Robot zip Name"/>	<input type="button" value="Seleccionar archivo"/> <input type="text" value="Ningún archivo seleccionado"/>

Process name must be at least 3 characters long, no special characters. Only one whitespace between characters.

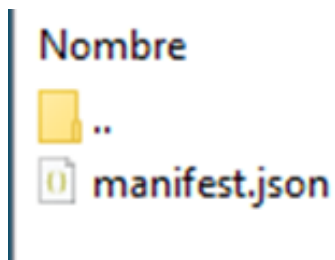


Add a process and load a robot to run in one or more instances You can load a robot by exporting to DB production or a DB project

Donde se debera ingresar el "start robot name" el cual debe ser el nombre con el que queremos llamar al nuevo zip subido. Y luego en "upload DB" debemos subir el archivo en formato .zip

El zip que se suba debe tener un archivo llamado manifest.json con el

siguiente contenido:



- {
- "name": "nombre del archivo",
- "version": "versión del archivo",
- "description": "Descripción del archivo",
- "main": "Método que se utilizara y corresponde al comando principal"
- "arguments": "parámetros que se utilizaran"
- }

Deben asegurar que lo que van a ejecutar esté instalado en la instancia, de lo contrario en los logs de la ejecución en el orquestador aparecerá un error.

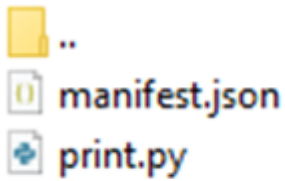
Ejemplos:

1

Ip config

- {
- "name": "script name",
- "version": "1.0",
- "description": "script description",
- "main": "ipconfig"
- "arguments": ["/all"]
- }

Y dará como resultado:



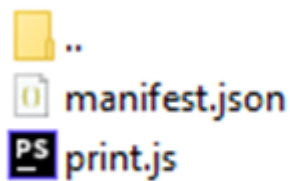
Y dará como resultado:

```
testing orchestrator noc zip
-----
{"cpu": "13.2%", "memory": "39.5% (6.31 GB \ / 15.95 GB)"}
-----
```

3

Node

- {
- "name": "python print",
- "version": "1.0",
- "description": "Prints a message in the console",
- "main": "node",
- "arguments": ["print.js"]
- }



Y dará como resultado:

```
Hello, this is a message from Node.js!
-----
{"cpu": "15.4%", "memory": "39.8% (6.35 GB \ / 15.95 GB)"}
-----
```

4

Ping a google

- {
- "name": "ping google",
- "version": "1.0",
- "description": "Sends a ping to Google",
- "main": "ping",
- "arguments": ["google.com"]
- }

Y dará como resultado:

```
Haciendo ping a google.com [142.250.79.110] con 32 bytes de datos:
Respuesta desde 142.250.79.110 bytes=32 tiempo=12ms TTL=116
Respuesta desde 142.250.79.110 bytes=32 tiempo=12ms TTL=116
Respuesta desde 142.250.79.110 bytes=32 tiempo=15ms TTL=116
Respuesta desde 142.250.79.110 bytes=32 tiempo=11ms TTL=116

Estadísticas de ping para 142.250.79.110:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
-----
{"cpu": "19.1%", "memory": "42.5% (6.77 GB \ / 15.95 GB)"}
-----
Mínimo = 11ms, Máximo = 15ms, Media = 12ms
-----
{"cpu": "19.1%", "memory": "42.5% (6.77 GB \ / 15.95 GB)"}
-----
```