



ESPECIFICACIÓN DE USO DE API

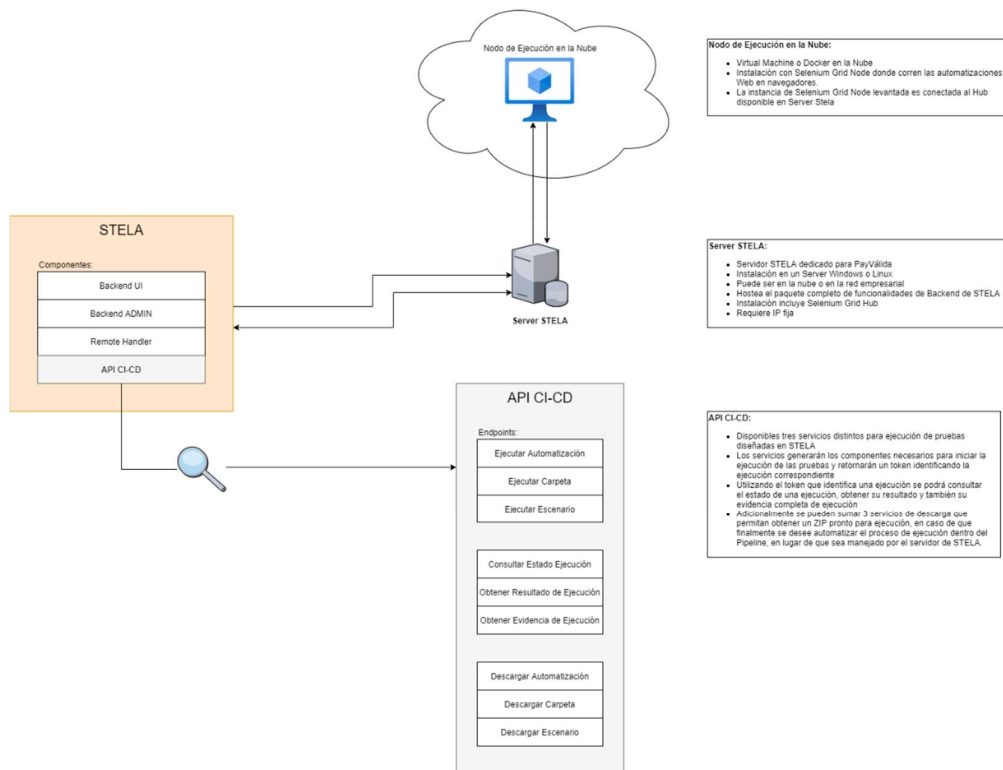
CONTINUOUS INTEGRATION | CONTINUOUS DEVELOPMENT API



INTRODUCCIÓN

Este documento lista las funcionalidades que brinda la API de STELA para integración continua. El objetivo de esta API es dar herramientas al usuario para poder solicitar la ejecución remota de pruebas creadas en STELA pero directamente desde una API, sin necesidad de hacerlo desde la UI, facilitando así la integración de la ejecución de pruebas desarrolladas en STELA a procesos de integración continua.

ARQUITECTURA



CONSIDERACIONES PREVIAS

Algunas consideraciones a tener en cuenta para el uso de la API:

- Los componentes utilizados por la API para ejecutar pruebas automatizadas de forma remota son los mismos que se utilizan para la funcionalidad de ejecución remota de STELA, disponible en la UI.

Sugerimos siempre validar el funcionamiento de una prueba utilizando ejecución remota primero, antes de utilizar la API para su ejecución, ya que es más sencilla la detección de problemas de esta forma.

- De momento la ejecución de pruebas de forma remota está **limitado a pruebas sobre tecnologías Web** que utilicen los navegadores **Mozilla Firefox** o **Google Chrome**.

De momento la interacción con archivos externos a STELA (por ejemplo, leer datos de un xls), así como la interacción directa con pantalla por medio de reconocimiento por imagen o especificación de coordenadas, no son soportadas por las funciones de ejecución de pruebas automatizadas.

- Los tokens de autenticación generados se asocian a un proyecto y solo permitirán interactuar con elementos de dicho proyecto.
- Los identificadores (IDs) de automatizaciones, carpetas y escenarios, necesarios para solicitar sus ejecuciones, de momento pueden ser obtenidos inspeccionando las interacciones de red de STELA UI en el navegador o consultando a nivel de base de datos (solo disponible para aquellas instalaciones on premise).

A futuro está programado contar con una sección a nivel del Panel de Administración de STELA que permita obtener de forma sencilla la información de los identificadores de cada uno de los elementos sin necesidad de tener que consultar tráfico / base de datos.

ENDPOINTS

DESCARGA

DESCARGAR AUTOMATIZACIÓN

A partir de un token de autorización e indicando el ID de una Automatización, se crean los archivos de STELA para la ejecución de dicha Automatización y los mismos son enviados dentro de un archivo comprimido (ZIP) para que el usuario los descargue.

URL : https://demo.stela.ai/STELAUI_API_CICD/download/automation

Tipo : GET

Consume : Parámetro (token, id) y JSON

Vía parámetro se debe enviar la información de:

- **token :** Token de autorización para utilizar la API
- **id :** id de la Automatización

Enviar un JSON como body del servicio será necesario solo si se quieren agregar datos a la descarga para ser utilizados durante la ejecución. Estos datos se pasan dentro del campo data como se ve a continuación.

```
{
  "data" : {
    "dato-string" : "John",
    "dato-num" : 32,
    "dato-boolean" : true
  }
}
```

El elemento **data** debe ser del formato **parámetro : valor**, cumpliendo las reglas de sintaxis del formato JSON. Las automatizaciones podrán ubicar los valores a través de variables en la categoría CICD-Load y con el nombre del parámetro enviado.

En el caso de ejemplo quedarían durante la ejecución disponibles las siguientes variables con sus correspondientes valores

```
[CICD-Load]
dato-string = John
dato-num = 32
dato-boolean = true
```

NOTAS:

- "data" soporta únicamente valores primitivos (String, Integer, Double, Boolean, etc.).
No se puede poner otro JSON o JSON Array como valor

Produce : Archivo (ZIP)

Se genera la carpeta con el proyecto STELA para la ejecución de la automatización y se comprime para que el usuario descargue el archivo .zip correspondiente.

DESCARGAR CARPETA

A partir de un token de autorización e indicando el ID de una Carpeta, se crean los archivos de STELA para la ejecución de dicha Carpeta, y los mismos son enviados dentro de un archivo comprimido (ZIP) para que el usuario los descargue.

URL : https://demo.stela.ai/STELAUI_API_CICD/download/folder

Tipo : GET

Consume : Parámetro (token, id) y JSON

Vía parámetro se debe enviar la información de:

- **token :** Token de autorización para utilizar la API
- **id :** id de la Carpeta

Enviar un JSON como body del servicio será necesario solo si se quieren agregar datos a la descarga para ser utilizados durante la ejecución. Estos datos se pasan dentro del campo data como se ve a continuación.

```
{
  "data" : {
    "dato-string" : "John",
    "dato-num" : 32,
    "dato-boolean" : true
  }
}
```

El elemento **data** debe ser del formato **parámetro : valor**, cumpliendo las reglas de sintaxis del formato JSON. Las automatizaciones podrán ubicar los valores a través de variables en la categoría CICD-Load y con el nombre del parámetro enviado.

En el caso de ejemplo quedarían durante la ejecución disponibles las siguientes variables con sus correspondientes valores

```
[CICD-Load]
dato-string = John
dato-num = 32
dato-boolean = true
```

NOTAS:

- "data" soporta únicamente valores primitivos (String, Integer, Double, Boolean, etc.). No se puede poner otro JSON o JSON Array como valor

Produce : Archivo (ZIP)

Se genera la carpeta con el proyecto STELA para la ejecución de la carpeta y se comprime para que el usuario descargue el archivo .zip correspondiente.

DOWNLOAD SCENARIO

A partir de un token de autorización e indicando el ID de un Escenario, se crean los archivos de STELA para la ejecución de dicho Escenario, y los mismos son enviados dentro de un archivo comprimido (ZIP) para que el usuario los descargue.

URL : https://demo.stela.ai/STELAUI_API_CICD/download/scenario

Tipo : GET

Consume : Parámetro (token, id)

Vía parámetro se debe enviar la información de:

- **token :** Token de autorización para utilizar la API
- **id :** id del Escenario

Enviar un JSON como body del servicio será necesario solo si se quieren agregar datos a la descarga para ser utilizados durante la ejecución. Estos datos se pasan dentro del campo data como se ve a continuación.

```
{
  "data" : {
    "dato-string" : "John",
    "dato-num" : 32,
    "dato-boolean" : true
  }
}
```

El elemento **data** debe ser del formato **parámetro : valor**, cumpliendo las reglas de sintaxis del formato JSON. Las automatizaciones podrán ubicar los valores a través de variables en la categoría CICD-Load y con el nombre del parámetro enviado.

En el caso de ejemplo quedarían durante la ejecución disponibles las siguientes variables con sus correspondientes valores

```
[CICD-Load]
dato-string = John
dato-num = 32
dato-boolean = true
```

NOTAS:

- "data" soporta únicamente valores primitivos (String, Integer, Double, Boolean, etc.). No se puede poner otro JSON o JSON Array como valor

Produce : Archivo (ZIP)

Se genera la carpeta con el proyecto STELA para la ejecución del escenario y se comprime para que el usuario descargue el archivo .zip correspondiente.

EJECUCIÓN

EJECUTAR AUTOMATIZACIÓN

A partir de un token de autorización e indicando el ID de una Automatización, se inicia la ejecución de la misma de forma remota, avisando al usuario que la ejecución comenzó correctamente.

****NOTA :** *Este servicio NO espera que la ejecución termine para responder con su resultado.*

URL : https://demo.stela.ai/STELAUI_API_CICD/execute/automation

Tipo : POST

Consume : JSON

JSON con el identificador de la Automatización a ejecutar, el token de autorización y parámetros para la automatización, en el elemento data.

```
{
  "id_Automation" : "5cec06251167f41649be44f1",
  "token" : "5ce6bf6b53430a21008b0ed5",
  "data" : {
    "dato-string" : "John",
    "dato-num" : 32,
    "dato-boolean" : true
  }
}
```

El elemento **data** es un campo opcional necesario únicamente si se desean pasar datos para la ejecución de la Automatización. Debe ser del formato **parámetro : valor**, cumpliendo las reglas de sintaxis del formato JSON. Las automatizaciones podrán ubicar los valores a través de variables en la categoría CICD-Load y con el nombre del parámetro enviado.

En el caso de ejemplo quedarían durante la ejecución disponibles las siguientes variables con sus correspondientes valores

```
[CICD-Load]
dato-string = John
dato-num = 32
dato-boolean = true
```

NOTAS:

- "data" soporta únicamente valores primitivos (String, Integer, Double, Boolean, etc.). No se puede poner otro JSON o JSON Array como valor

Produce : JSON

JSON con la información de la Ejecución iniciada

```
{  
  "success" : true,  
  "id_execution" : "60d20f999b03647f7bfad8c4",  
  "startTime" : "2021-06-22 13:28:13.572"  
}
```

EJECUTAR CARPETA

A partir de un token de autorización e indicando el ID de una Carpeta, se inicia la ejecución de la misma de forma remota, avisando al usuario que la ejecución comenzó correctamente.

*****NOTA :** Este servicio NO espera que la ejecución termine para responder con su resultado.*

URL : https://demo.stela.ai/STELAUI_API_CICD/execute/folder

Tipo : POST

Consume : JSON

JSON con el identificador de la Carpeta a ejecutar, el token de autorización y parámetros para la automatización, en el elemento data.

```
{
  "id_Folder" : "5cec06251167f41649be44f1",
  "token" : "5ce6bf6b53430a21008b0ed5",
  "data" : {
    "dato-string" : "John",
    "dato-num" : 32,
    "dato-boolean" : true
  }
}
```

El elemento **data** es un campo opcional necesario únicamente si se desean pasar datos para la ejecución de la Carpeta. Debe ser del formato **parámetro : valor**, cumpliendo las reglas de sintaxis del formato JSON. Las automatizaciones podrán ubicar los valores a través de variables en la categoría CICD-Load y con el nombre del parámetro enviado.

En el caso de ejemplo quedarían durante la ejecución disponibles las siguientes variables con sus correspondientes valores

```
[CICD-Load]
dato-string = John
dato-num = 32
dato-boolean = true
```

NOTAS:

- "data" soporta únicamente valores primitivos (String, Integer, Double, Boolean, etc.). No se puede poner otro JSON o JSON Array como valor

Produce: JSON

JSON con la información de la Ejecución iniciada

```
{  
  "success" : true,  
  "id_execution" : "60d20f999b03647f7bfad8c4",  
  "startTime" : "2021-06-22 13:28:13.572"  
}
```

EJECUTAR ESCENARIO

A partir de un token de autorización e indicando el ID de un Escenario, se inicia la ejecución del mismo de forma remota, avisando al usuario que la ejecución comenzó correctamente.

***NOTA : Este servicio NO espera que la ejecución termine para responder con su resultado.*

URL : https://demo.stela.ai/STELAUI_API_CICD/execute/scenario

Tipo : POST

Consume : JSON

JSON con el identificador del Escenario a ejecutar, el token de autorización y parámetros para la automatización, en el elemento data.

```
{
  "id_Scenario" : "5cec06251167f41649be44f1",
  "token" : "5ce6bf6b53430a21008b0ed5",
  "data" : {
    "dato-string" : "John",
    "dato-num" : 32,
    "dato-boolean" : true
  }
}
```

El elemento **data** es un campo opcional necesario únicamente si se desean pasar datos para la ejecución del Escenario. Debe ser del formato **parámetro : valor**, cumpliendo las reglas de sintaxis del formato JSON. Las automatizaciones podrán ubicar los valores a través de variables en la categoría CICD-Load y con el nombre del parámetro enviado.

En el caso de ejemplo quedarían durante la ejecución disponibles las siguientes variables con sus correspondientes valores

```
[CICD-Load]
dato-string = John
dato-num = 32
dato-boolean = true
```

NOTAS:

- "data" soporta únicamente valores primitivos (String, Integer, Double, Boolean, etc.). No se puede poner otro JSON o JSON Array como valor

Produce : JSON

JSON con la información de la Ejecución iniciada

```
{  
  "success" : true,  
  "id_execution" : "60d20f999b03647f7bfad8c4",  
  "startTime" : "2021-06-22 13:28:13.572"  
}
```

CONSULTA

OBTENER RESULTADO DE EJECUCIÓN

A partir de un token de autenticación y el identificador de una ejecución (ID) ya finalizada, devuelve el resultado de la misma (PASSED | FAILED)

URL : https://demo.stela.ai/STELAU_API_CICD/execute/result

Tipo : GET

Consume : Parámetros (exe_token, token)

Vía parámetro se debe enviar la información de:

- **exe_token** : Identificador (ID) de la ejecución a consultar.¹
- **token** : Token de autorización para utilizar la API

Produce : JSON

JSON con el resultado de ejecución solicitado

```
{
  "message" : "PASSED",
  "passed" : true,
  "result" : 2,
  "success" : true
}
```

Códigos de resultados (campo result):

1	Ejecución aún en curso
2	Ejecución finalizada sin errores (Passed)
3	Ejecución finalizada con errores (Failed)
-1	Ejecución no existe
-3	No se cuenta con acceso a los datos de la ejecución solicitada
-4	Otro tipo de errores

¹ Al enviar a ejecutar una Automatización, Carpeta o Escenario, la respuesta al servicio retorna un identificador para la ejecución generada.

OBTENER ESTADO DE EJECUCIÓN

A partir de un token de autenticación y el identificador de una ejecución (ID) retorna el estado actual de la misma, permitiendo distinguir entre ejecuciones en curso y aquellas pruebas automatizadas que ya han finalizado

URL : https://demo.stela.ai/STELAUI_API_CICD/execute/status

Tipo : GET

Consume : Parámetros (exe_token, token)

Vía parámetro se debe enviar la información de:

- **exe_token :** Identificador (ID) de la ejecución a consultar.²
- **token :** Token de autorización para utilizar la API

Produce : JSON

JSON con la información del estado actual de la ejecución.

```
{  
  "message" : "La ejecución está en progreso",  
  "result" : 1,  
  "success" : true  
}
```

Códigos de resultados (campo result):

1	Ejecución aún en curso
2	Ejecución finalizada
-1	Ejecución no existe
-2	Se produjo un error inesperado durante la ejecución
-3	No se cuenta con acceso a los datos de la ejecución solicitada
-4	Otro tipo de errores

² Al enviar a ejecutar una Automatización, Carpeta o Escenario, la respuesta al servicio retorna un identificador para la ejecución generada.

OBTENER EVIDENCIA DE EJECUCIÓN

A partir de un token de autenticación y el identificador de una ejecución (ID) ya finalizada, se toman los archivos generados como evidencia durante la ejecución y son enviados dentro de un archivo comprimido (ZIP) para que el usuario los descargue.

URL : https://demo.stela.ai/STELAUI_API_CICD/execute/evidence

Tipo : GET

Consume : Parámetros (exe_token, token)

Vía parámetro se debe enviar la información de:

- **exe_token :** Identificador (ID) de la ejecución a consultar.³
- **token :** Token de autorización para utilizar la API

Produce : Archivo (ZIP)

Se genera la carpeta con la evidencia de ejecución y se comprime para que el usuario descargue el archivo .zip correspondiente.

³ Al enviar a ejecutar una Automatización, Carpeta o Escenario, la respuesta al servicio retorna un identificador para la ejecución generada.