

Introducción

Los webhooks le permiten enviar datos en tiempo real a cualquier otra plataforma que esté utilizando. Éstos se disparan al ocurrir un evento determinado, por ejemplo, al finalizar la inscripción de un alumno.

Debéra elegir a qué eventos quiere subscribirse para enviar los datos a a la URL que especifique. De esta manera, sabremos cuándo debemos enviar los datos y a dónde.

Para usar este módulo deberá tener el siguiente permiso:

8700 - Permite al usuario la completa gestión de los webhooks (agregar, quitar y ver)

IMPORTANTE: El cliente es dueño de sus datos y se responsabiliza del tratamiento y divulgación de los mismos al procesar las solicitudes.

IMPORTANTE: Para comprender y hacer uso de los webhooks es necesario tener un conocimiento intermedio de desarrollo y manejo de solicitudes HTTP.

Requerimientos

Algunas cosas a tener en cuenta al configurar y trabajar con webhooks:

Las URL pueden ser HTTP o HTTPS. En caso de usar HTTPS debe asegurarse que su servidor tenga configurado e instalado un certificado SSL válido.

Su servidor debe ser capaz de procesar la solicitud en menos de 10 segundos. Si no recibimos una respuesta, el webhook expirará y la conexión se cerrará.

Para confirmar que se recibe un webhook, su terminal debe devolver uno de los siguientes códigos de estado HTTP 2xx:

- 200 OK
- 201 Created

- 202 Accepted
- 204 No Content

El **cuerpo de la respuesta** debe contener el JSON: {"status": "ok"} para que el webhook no se vuelva a enviar nuevamente.

IMPORTANTE: Su servidor debe devolver {"status": "ok"} al procesar la solicitud. Sin esta respuesta, se intentará procesar el webhook aún cuando el código de estado HTTP haya sido un 2xx

IMPORTANTE: En la plataforma de TEST los webhooks no se envían automáticamente. Deberá lanzarlos de forma manual. Para más información: [Envío manual](#).

Autenticación

Para verificar que el envío de datos proviene de Quinttos, puede agregar su propio código interno de seguridad al webhook. Esto le permitirá aceptar solo las solicitudes que contengan el código interno de seguridad especificado.

El código interno de seguridad se envía como un encabezado en la solicitud HTTP con el nombre **X-Secret** al invocar la URL del webhook. Un ejemplo de un encabezado es el siguiente:

```
Content-Type: application/json
X-Webhook-Event: enrollment_added
X-Webhook-ID: dd7145e0-0fc4-439e-a890-a10b1fce0ceb
X-Request-ID: 23ecbdd3-c6dd-479e-a72b-9f68811a9f63
X-Secret: dsdiq32i4dma1
```

Política de reintento

Si la solicitud del webhook falla debido a un problema con la red o el servidor, se volverá a intentar en un total de 5 veces.

Solicitudes que se vuelven a intentar

Se vuelven a enviar los webhook en los siguientes casos:

- 2xx (si el cuerpo de la respuesta no tiene `{"status": "ok"}`)
- 429 (demasiadas solicitudes)
- 5xx
- Tiempo de espera (no se recibió respuesta en 10 segundos)

Características

Todos los webhooks tienen las siguientes características:

- Todos los webhooks proporcionan datos en formato JSON
- Todos los webhooks se envían mediante el método HTTP POST
- Todos los webhooks poseen un ID
- Todos los intentos de envío ti
- Cada envío contiene los siguientes campos:
 - event - tipo de evento y la fecha y hora cuando ocurrió

Webhook headers

Cada solicitud tendrá los siguientes encabezados:

- Content-Type: información sobre el tipo de contenido. Ej: application/json
- X-Webhook-Event: tipo de evento. Igual que el campo event que se envía con los datos.
- X-Webhook-ID: UUID de webhook (único por evento, igual para la primera solicitud y reintentos)
- X-Request-ID - UUID de solicitud (único por solicitud)
- X-Secret: código interno de seguridad para estar seguros que el envío de datos proviene desde Quinttos

Revision #4

Created 6 February 2024 13:48:37 by Implementaciones Academias

Updated 6 February 2024 20:47:17 by Soporte Quinttos