



PROYECTO: apiRestSeguimientoEnviosk8s

DOCUMENTO: DC\_SP\_WS\_apiRestSeguimientoEnviosk8s\_v01.03.docx

SUBDIRECCION DE TECNOLOGIA Y PROYECTOS



Correos  
Express



Correos  
Express

# **Documentación para Cliente**

## **Servicio Web para el Seguimiento de**

## **Envíos k8s**

Subdirección de Tecnología y Sistemas

Este documento es propiedad de Correos Express, no pudiendo ser usado con fines distintos de aquellos para los que ha sido entregado, ni reproducido, total o parcialmente, ni transmitido o comunicado a ninguna persona sin autorización del propietario.

	PROYECTO: apiRestSeguimientoEnviosk8s	 Correos Express
	DOCUMENTO: DC_SP_WS_apiRestSeguimientoEnviosk8s_v01.03.docx	
	SUBDIRECCION DE TECNOLOGIA Y PROYECTOS	

## DATOS DE CONTROL

<b>Título</b>	DC_SP_ apiRestSeguimientoEnviosk8s_v01.03.docx		
<b>Autor</b>	Correos Express		
<b>Versión</b>	01.03	<b>Fecha Versión</b>	26/09/2022

## CONTROL DE CAMBIOS

Versión	Fecha	Autor	Resumen de Cambios
01.00	21/01/2021	Cristina Diez (Capgemini)	Migración WS a Kubernetes
01.01	16/02/2022	Cristina Diez (Capgemini)	Rebranding
01.03	26/09/2022	Ángel Álvarez (Capgemini)	Review endpoint

## LISTA DE DISTRIBUCIÓN

Nombre	Área/Departamento	Organización
Cynthia Soriano Dubo	JEQ	Altran
Javier García Arenas	PMO	Correos Express

## Tabla de Contenido

<b>1. INTRODUCCIÓN .....</b>	<b>4</b>
1.1. Objetivo .....	4
1.2. Abreviaturas, Acrónimos y Terminología.....	4
1.3. Herramientas de Referencia .....	5
<b>2. DEFINICIÓN DEL SERVICIO WEB .....</b>	<b>5</b>
2.1. Propósito.....	5
2.2. Url's de acceso .....	5
<b>3. DESCRIPCIÓN DETALLADA DEL SERVICIO WEB .....</b>	<b>6</b>
3.1. Método SeguimientoEnviosRequest.....	6
3.1.1. Parámetros de entrada.....	6
3.2. Método SeguimientoEnviosResponse.....	7
3.2.1. Parámetros de salida .....	7
3.3. Método SeguimientoEnviosFechasRequest .....	10
3.3.1. Parámetros de entrada.....	10
3.4. Método SeguimientoEnviosFechasResponse .....	10
3.4.1. Parámetros de salida .....	11
3.4.2. Códigos de error.....	11
3.4.3. Listado de códigos de error.....	11
<b>4. EJEMPLO DE CLIENTES .....</b>	<b>13</b>
4.1. PostMan .....	13

# 1. INTRODUCCIÓN

## 1.1. Objetivo

El objetivo de este Servicio Web es realizar el seguimiento de envíos.

Contará con dos opciones de búsqueda, una que permite consultar las especificaciones de un pedido en función del número de envío que tenga asignado y una segunda que permite listar el número de envíos que se han realizado entre dos intervalos de fecha de dados para un número de cliente.

## 1.2. Abreviaturas, Acrónimos y Terminología

Código	Descripción
CA	Autoridad Certificadora
CHX	Correos Express
DAO	Objetos de acceso a datos
DB / BBDD	Base de datos
MEP	Patrones de intercambio de mensajes
REST	Representational State Transfer, conjunto de principios de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web. Usado para describir cualquier interfaz entre sistemas que utilicen directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos, en cualquier formato (XML, JSON, etc) sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes, como por ejemplo SOAP. Los servicios que sólo usan POST como éste, o GET se definen como RESTlike, y si por el contrario incluyen POST, GET, PUT o DELETE se definen como RESTful.
WS	Servicio Web
WSDL	Web Services Description Language
W3C	World Wide Web Consortium
XML	eXtensible Markup Language, lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible
JSON	JavaScript Object Notation, formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que, para las máquinas, es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Estándar ECMA-262 3rd Edition – Diciembre 1999.
WSD	Web Services for Devices or Web Services on Devices , API que simplifica la programación de conexiones a servicios web en dispositivos compatibles.

### 1.3. Herramientas de Referencia

Herramienta	Código	Descripción
Entorno de prueba	PostMan	Entorno de prueba para el WS

## 2. DEFINICIÓN DEL SERVICIO WEB.

### 2.1. Propósito

El servicio web definido proporciona la posibilidad de realizar el seguimiento de envíos por intervalos de fechas o números de envío y a su vez consultar las especificaciones de cada uno de estos envíos. Este servicio web es de tipo RESTlike, con dos métodos POST a las 2 Url's de acceso mostradas en el siguiente apartado. (Cada una de las Url's nos permite consultar una funcionalidad).

La interfaz del servicio web define una serie de parámetros básicos de entrada para cada uno de los POST, en el caso del seguimiento de envíos, estos parámetros son: codigoCliente (equivalente a Código de cliente + Centro), dato e idioma, y en el caso del seguimiento de envíos por fechas, estos parámetros son: keyClifac (Código de cliente + Centro), Fecha Inicial, Fecha Final e idioma.

Los parámetros indicados para cada POST son los necesarios y obligatorios para realizar la consulta contra los sistemas de Correos Express y obtener la información de alguna de las entregas a través de los parámetros de salida.

Más adelante se definirán qué campos son obligatorios de informar para que este tipo de soluciones puedan ser cargadas correctamente en el sistema mediante los 2 POST disponibles para ello.

### 2.2. Url's de acceso

Correos Express dispone de 2 entornos de acceso al servicio de **apiRestSeguimientoEnviosk8s** para cada uno de los 2 POST de consulta disponibles:

Environment	Access URL - Seguimiento Envíos
Test	<a href="https://www.test.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnviosFechas">https://www.test.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnviosFechas</a>
Producción	<a href="https://www.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnviosFechas">https://www.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnviosFechas</a>

Environment	Access URL - Seguimiento Envíos
Test	<a href="https://www.test.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnvio">https://www.test.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnvio</a>
Producción	<a href="https://www.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnvio">https://www.cexpr.es/wspsc/apiRestSeguimientoEnviosk8s/json/seguimientoEnvio</a>

Ambos entornos permiten el acceso al recurso restringido por protocolo seguro y con autenticación mediante validación de **usuario LDAP**,

En el documento de **Acceso\_servicio\_web\_Cex** se explica todo lo relativo a la validación LDAP para poder autenticar el servicio y consumirlo desde PostMan o SoapUI.

### 3. DESCRIPCIÓN DETALLADA DEL SERVICIO WEB.

Este servicio web es de tipo RESTlike con 2 métodos POST, cada uno con su correspondiente JSON de entrada, a cualquiera de las 2 URL's de acceso mostradas en el apartado 2.2 de la página 6 de este documento.

Dicho servicio se encargará de devolver los envíos para el intervalo de fechas que se desee, mediante un POST, y a su vez con otro POST, poder consultar las especificaciones de cada envío en concreto, en base a los parámetros de entrada obligatorios para cada una de las plantillas JSON que se muestran en el punto 4 de la página 13 de este documento.

#### 3.1. Método SeguimientoEnviosRequest

Este método recoge en un JSON de entrada los 2 parámetros obligatorios para la obtención de las especificaciones de un envío en concreto y un tercer campo opcional que recoge el idioma para mostrar, en el caso de que se devuelvan y exista traducción de las descripciones o errores.

##### 3.1.1. Parámetros de entrada

###### 3.1.1.1. Elementos del tipo "Objeto"

Nombre	Descripción	Formato	Rango Valores	Obligatorio
<b>codigoCliente</b>	Este código es facilitado por Correos Express	String	NN ( Max Long: 99)	SI
<b>Dato</b>	Campos sobre los que se puede realizar la búsqueda: <ul style="list-style-type: none"> <li>- Numero Envío (numérico de entre 8 y 16 dígitos)</li> <li>- Referencia (alfanumérico de entre 8 y 20 dígitos)</li> <li>- Código de bulto (numérico de 23 dígitos)</li> <li>- Código Bulto Cliente (alfanumérico de entre 8 y 30 dígitos)</li> </ul>	String	Numero Envío: NNNNNNNNNNNNNNNNN Referencia: ANNNNNNN Código Bulto: NNNNNNNNNNNNNNNNN NNNNNNN Código Bulto Cliente: ANNNNNNN	SI

Nombre	Descripción	Formato	Rango Valores	Obligatorio
Idioma	Código Iso del país de traducción.	String	NN ( Max Long: 2)	NO

## 3.2. Método SeguimientoEnviosResponse

Este método devuelve un objeto que contiene un mensaje y un código de error junto con todos los objetos que se muestran a continuación y 2 listas de Bultos y Estados de envío.

### 3.2.1. Parámetros de salida

#### 3.2.1.1. Elementos del tipo "Objeto y lista"

Nombre	Descripción	Formato	Rango Valores	Obligatorio
error	Código del error	Integer	NNN	error
mensajeError	Mensaje del error	String	NNNNN(Long. Máx: 99999)	mensajeError
resultado	Es el estado del envío	String	N	
numEnvio	Es el número del envío	String	NNNNNNNNNNNNNNNNNN	numEnvio
ref	Referencia del envío	String	NNNNN	
refCliente	Referencia del cliente	String	NNNNNNNNN	
fecha	Fecha del envío	String	DD/MM/AA	
codRte	Código remitente	String	NNNNNNNNN	
nomRte	Nombre remitente	String	-	
nifRte	Nif remitente	String	NNNNNNNNN	
dirRte	Dirección remitente	String	-	
pobRte	Población remitente	String	-	
codPostNacRte	Código Postal Nacional del remitente	String	NNNNN	
paisSORte	País del remitente	String	-	

Nombre	Descripción	Formato	Rango Valores	Obligatorio
<b>codPostIntRte</b>	Código Postal Internacional del remitente	String	NNNNN	
<b>contacRte</b>	Contacto del remitente	String	-	
<b>telefRte</b>	Teléfono del remitente	String	NNNNNNNNN	
<b>emailRte</b>	Email del remitente	String	-	
<b>codDest</b>	Código destinatario	String	NNNNNNNNN	
<b>nomDestRte</b>	Nombre destinatario	String	-	
<b>nifDest</b>	Nif destinatario	String	NNNNNNNNN	
<b>dirDest</b>	Dirección destinatario	String	-	
<b>pobDest</b>	Población destinatario	String	-	
<b>codPostNacDest</b>	Código Postal Nacional del destinatario	String	NNNNN	<b>codPostNacDest</b>
<b>contacDest</b>	Contacto del destinatario	String	-	
<b>telefDest</b>	Teléfono del destinatario	String	NNNNNNNNN	
<b>emailDest</b>	Email del destinatario	String	-	
<b>contacOtrs</b>	Contacto OTRS	String	-	
<b>telefOtrs</b>	Teléfono OTRS	String	NNNNNNNNN	
<b>emailOtrs</b>	Email OTRS	String	-	
<b>observac</b>	Observaciones	String	-	
<b>numBultos</b>	Número de bultos	String	NNN (Long. Máx: 999)	
<b>kilos</b>	Kilos	String	NNNNNNNNN	
<b>volumen</b>	Volumen	String	NNNNNNNNN	
<b>alto</b>	Alto	String	NNNNNNNNN	
<b>largo</b>	Largo	String	NNNNNNNNN	
<b>ancho</b>	Ancho	String	NNNNNNNNN	
<b>producto</b>	Producto	String	NNN (Long. Máx: 999)	
<b>portes</b>	Portes	String	N (Long. Máx: 9)	
<b>reembolso</b>	Reembolso	String	NNNNNNNNN	
<b>entrSabado</b>	Entrega Sábado	String	-	
<b>seguro</b>	Seguro	String	-	



Nombre	Descripción	Formato	Rango Valores	Obligatorio
<b>codEstado</b>	Código de Estado	String	NN (Long. Máx: 99)	
<b>descEstado</b>	Descripción del Estado	String	-	
<b>fechaEstado</b>	Fecha del Estado	String	DDMMAAAA	
<b>horaEstado</b>	Hora del Estado	String	HHMMSS	
<b>codIncEstado</b>	Código de Incidencia de Estado	String	NN(Long. Máx: 99)	
<b>descIncEstado</b>	Descripción de la incidencia de Estado	String	-	
<b>estadoEnvios</b>	Estado de los envíos	Lista	Nos devuelve un cursor	
<b>bultoSeguimiento</b>	Seguimiento de los bultos	Lista	Nos devuelve un cursor	

### 3.2.1.2. Elementos del tipo "EstadoEnvios"

Nombre	Descripción	Tipo	Rango de valores
<b>codEstado</b>	Es el estado del envío	String	N
<b>descEstado</b>	Es la descripción del estado del envío	String	NNNNNNNNNNNNNNNNNN
<b>horaEstado</b>	Hora del cambio de estado del envío	String	NNNNN
<b>codIncEstado</b>	Es la incidencia del envío	String	N
<b>descIncEstado</b>	Es la descripción de la incidencia del envío	String	NNNNNNNNNNNNNNNNNN

### 3.2.1.3. Elementos del tipo "BultoSeguimiento"

Nombre	Descripción	Tipo	Rango de valores
<b>orden</b>	Orden del bulto	String	N
<b>codUnico</b>	Código del bulto	String	NNNNNNNNNNNNNNNNNN
<b>referencia</b>	Referencia del envío	String	NNNNN
<b>codEstado</b>	Es el estado del bulto	String	N
<b>descEstado</b>	Es la descripción del estado del bulto	String	NNNNNNNNNNNNNNNNNN
<b>fechaEstado</b>	Fecha del cambio de estado del bulto	String	NNNNNNNN

Nombre	Descripción	Tipo	Rango de valores
horaEstado	Hora del cambio de estado del bulto	String	NNNNNNN
codIncEstado	Es la incidencia del bulto	String	N
descIncEstado	Es la descripción de la incidencia del bulto	String	NNNNNNNNNNNNNNNNNN

### 3.3. Método SeguimientoEnviosFechasRequest

Este método recoge en un JSON de entrada los 3 parámetros obligatorios para la obtención del seguimiento de envíos en un intervalo determinado de fechas y uno opcional que es el idioma.

#### 3.3.1. Parámetros de entrada

##### 3.3.1.1. Elementos del tipo "objeto"

Nombre	Descripción	Tipo	Formato	Obligatorio
Keyclifac	Código de cliente + Centro	String	NNNNNNNNNN	SI
Fecha Inicial	Fecha inicial de la consulta	Date	AAAA-MM-DD	SI
Fecha Final	Fecha final de la consulta	Date	AAAA-MM-DD	SI
Idioma	Idioma de traducción	String	NN	NO

### 3.4. Método SeguimientoEnviosFechasResponse

Este método devuelve un objeto que contiene un mensaje y un código de error junto con la lista de envíos que existen para el KeyClifac y el intervalo de fechas consultado. Los envíos se muestran en un objeto con las características que se dictan a continuación.

### 3.4.1. Parámetros de salida

#### 3.4.1.1. Elementos del tipo "objeto y lista"

El POST realizado con los parámetros de entrada obligatorios nos devuelve un objeto que contiene un mensaje y un código de error junto con una lista llamada "SeguimientoEnvioFechas", la cual contiene los siguientes campos.

Nombre	Descripción	Tipo	Obligatorio
<b>error</b>	Código del error	Integer	NNNN
<b>mensajeError</b>	Mensaje del error	String	NNNNN(Long. Máx: 99999)
<b>seguimientoEnvioFecha</b>	Lista con información de los envíos	Lista	Nos devuelve un cursor

#### 3.4.1.2. Los elementos del tipo SeguimientoEnvioFecha

Nombre	Descripción	Tipo	Obligatorio
<b>numEnvio</b>	Devuelve el número del envío	String	NNNNNNNNNNNNNNNNNN
<b>referencia</b>	Devuelve la referencia del envío	String	NNNNN(Long. Máx: 99999)
<b>estadoEnvio</b>	Devuelve el estado del envío	String	NN (Long. Máx: 99)

### 3.4.2. Códigos de error.

El servicio web, pase lo que pase, siempre devolverá al usuario un código de error, tanto si todo ha salido correcto y no se han producido errores, como si se han detectado problemas.

Estos códigos de retorno irán acompañados del mensaje correspondiente para cada código.

### 3.4.3. Listado de códigos de error.

A continuación, podemos ver el listado de todos los códigos de retorno que se pueden dar en el uso de nuestro Servicio Web y que han sido detallados además en el diagrama de secuencias.

Código	Mensaje	Observaciones
0		Ok
2	ID_ENVIO DEL ENVIO QUE CONTIENE BLANCOS	El campo de datos no puede contener espacios en blanco
2	ID_ENVIO DEL ENVIO VACIO	El campo de datos no puede estar vacío
2	ID_ENVIO DEL ENVIO SUPERA EL MAXIMO DE 30 CARACTERES	El campo de datos no puede superar los 30 caracteres.
400	ERROR - REVISE EL CAMPO CLIENTE, NO PUEDE SER NULO	Los campos keyclifac o datos están vacíos
400	ERROR OBTENER_HCO_ESTADOS NO_DATA_FOUND	Error al obtener estados
401	ERROR - ALGUN CAMPO ESTA VACIO, REVISAR LOS PARÁMETROS DE ENTRADA	Revisar campos de entrada
402	ERROR EN WS - LLAMADA AL SERVICIO INCORRECTA O FALLO EN LA ESTRUCTURA	Error al recopilar valores de la base de datos
405	ERROR - REVISAR LOS PARÁMETROS DE ENTRADA (ALGUN CAMPO DE FECHA ESTA VACIO O EL FORMATO DE LA FECHA ES ERRONEO): UTILICE EL FORMATO CORRECTO [YYYY-MM-DD]	No se han introducido una o más fechas o el formato no es correcto [YYYY-MM-DD]
408	El codigoCliente debe tener una longitud de 5 dígitos	El código de cliente debe tener al menos 5 dígitos
409	NO EXISTE EL NÚMERO DE ENVÍO PARA EL CÓDIGO CLIENTE:	El keyclifac no coincide con los datos proporcionados
500	ERROR OBTENER_LISTA_BULTOS NO_DATA_FOUND	No se pudo obtener la lista de paquetes
999	ERROR GENERAL	Un error general ha ocurrido

## 4. EJEMPLO DE CLIENTES

Para cualquier cliente que usemos, sea el que sea, siempre deberemos validarnos mediante LDAP y usar una de las siguientes plantillas JSON en función del servicio que queramos consultar para lanzar nuestro POST a una de las 2 URL's indicadas:

Estructura JSON – Ejemplo Request para seguimientoEnvio
<pre>{   "codigoCliente": "xxx",   "dato": "6283000000258558",   "idioma": "PT" }</pre>

Estructura JSON – Ejemplo Request para seguimientoEnvio con Fechas
<pre>{   "fechaInicial": "2020-09-25",   "fechaFinal": "2021-01-13",   "keyCliFac": "555599999",   "idioma": "PT" }</pre>

Los clientes con los que podamos consumir nuestro WS son **Postman** o **SoapUI**, y a continuación se explica cómo configurar el Postman.

### 4.1. PostMan

Resumamos los pasos realizados hasta el momento:

- Paso 1: abrimos el Postman y creamos un nuevo proyecto.
- Paso 2: Abrimos el Postman, lo ponemos en opción **POST** y pinchamos en **Authorization**, a continuación, desplegamos la lista y seleccionamos **Basic Auth** y escribimos nuestros datos de usuario y contraseña.

Params Authorization Headers (10) Body Pre-request Script Tests Settings

TYPE  
Basic Auth

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

Username: usuario\_facilitado\_por\_cex

Password: \*\*\*\*\*

☐ Show Password

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend you to use the 'Show Password' checkbox.

3.- Vamos a Headers, pinchamos sobre **key** y escribimos **Content-Type**, después pinchamos en **value** y escribimos **application/json**.

Params	Authorization	Headers (10)	Body	Pre-request Script	Tests	Settings
<input checked="" type="checkbox"/>	Authorization					Basic dXNlcj93c190ZXN0OjY4YmIX
<input checked="" type="checkbox"/>	Cache-Control					no-cache
<input checked="" type="checkbox"/>	Postman-Token					<calculated when request is sent>
<input checked="" type="checkbox"/>	Content-Type					application/json
<input checked="" type="checkbox"/>	Content-Length					<calculated when request is sent>
<input checked="" type="checkbox"/>	Host					<calculated when request is sent>
<input checked="" type="checkbox"/>	User-Agent					PostmanRuntime/7.26.8
<input checked="" type="checkbox"/>	Accept					*/*
<input checked="" type="checkbox"/>	Accept-Encoding					gzip, deflate, br
<input checked="" type="checkbox"/>	Connection					keep-alive

4.- A continuación, vamos a la etiqueta **Body** y pinchamos sobre la tercera opción que es **raw** y colocamos el desplegable de la derecha en **JSON (application/json)**

5.- Por último, copiamos o escribimos una de nuestras 2 plantillas JSON de entrada en función de la consulta deseada y escribimos arriba la URL correspondiente a la plantilla del servicio que queremos consultar.

Recordamos las 2 URL's existentes para cada uno de los servicios a consultar:

**Seguimiento de Envíos PRE:**

<https://www.test.cexpr.es/wsps/apiRestSeguimientoEnviosk8s/json/seguimientoEnvio>

**Seguimiento de Envíos por Fechas PRE:**

<https://www.test.cexpr.es/wsps/apiRestSeguimientoEnviosk8s/json/seguimientoEnviosFechas>

(Siempre realizaremos las consultas en las URLS de PRE y mediante **POST**).

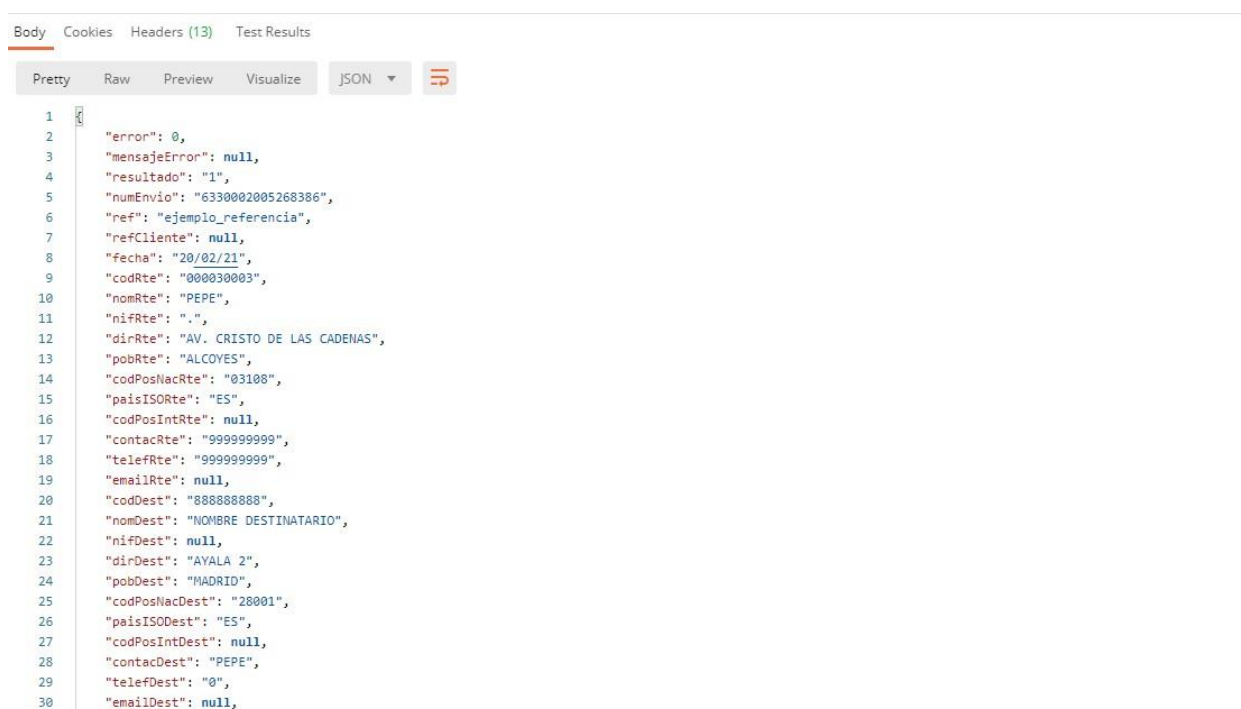
A continuación, se muestra un envío de ejemplo generado para la documentación en el entorno de preproducción.

### Seguimiento de envíos por número de envío:

Completamos el campo del codigoCliente, así como el del número de envío y el idioma en el caso de querer mostrar los mensajes de error traducidos. Por defecto, si el campo idioma va nulo se devuelve el mensaje en español.



Una vez que pulsamos en SEND recibimos el código de error 0, que indica que todo ha salido bien, y junto con él recibiremos el resultado de nuestra consulta.



```

31     "contacOtrs": null,
32     "telefOtrs": null,
33     "emailOtrs": null,
34     "observac": "OBSERVACIONES DE FUERA ",
35     "numBultos": " 1",
36     "kilos": " 1.670",
37     "volumen": " .010",
38     "alto": null,
39     "largo": null,
40     "ancho": null,
41     "producto": " 63",
42     "portes": "P",
43     "reembolso": null,
44     "entrSabado": null,
45     "seguro": null,
46     "codEstado": "1",
47     "descEstado": "SEM RECEPÇÃO",
48     "fechaEstado": "09022021",
49     "horaEstado": "100240",
50     "codIncEstado": null,
51     "descIncEstado": null,

52     "bultoSeguimiento": [
53     {
54         "orden": "01",
55         "codUnico": "63300020052683801280012",
56         "referencia": null,
57         "codEstado": "1",
58         "descEstado": "SEM RECEPÇÃO",
59         "fechaEstado": "09022021",
60         "horaEstado": "100240",
61         "codIncEstado": null,
62         "descIncEstado": null
63     }
64 ],
65     "estadoEnvios": [
66     {
67         "codEstado": "1",
68         "descEstado": "SEM RECEPÇÃO",
69         "fechaEstado": "09022021",
70         "horaEstado": "100240",
71         "codIncEstado": null,
72         "descIncEstado": null
73     }
74 ]
75 }

```

En el caso de no encontrar resultados o enviar campos vacíos, el webservice devolverá un error que se podrá obtener también en el idioma que hayamos seleccionado si existe registro en base de datos. De este modo, podemos obtener, por ejemplo:


● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON ▼

```

1 {
2   "codigoCliente": "55555555",
3   "dato": "6330002005268386",
4   "idioma": "PT"
5 }
6

```

Body Cookies Headers (13) Test Results

Pretty Raw Preview Visualize
JSON ▼


```

1 {
2   "error": 409,
3   "mensajeError": "NÃO HÁ NÚMERO DE ENVIO PARA O CÓDIGO DO CLIENTE",
4   "resultado": "0",
5   "resultado": "0"
6 }

```



### Búsquedas por fechas:

Es muy importante que el formato de las fechas vaya con guiones y no con barras, espacios, etc.

**El formato a utilizar para las fechas debe ser [AAAA-MM-DD]**

Para buscar envíos por fechas y KeyClifac, introducimos el JSON con los datos de entrada y el webservice nos devuelve el JSON con unos datos de salida similares a los que se muestran en el siguiente ejemplo:

Params Authorization Headers (10) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded **raw** binary GraphQL JSON ▼

```

1 {
2   "fechaInicial": "2020-09-29",
3   "fechaFinal": "2020-09-30",
4   "keyClifac": "555555555",
5   "idioma": "PT"
6 }
```

Body Cookies Headers (13) Test Results

Pretty Raw Preview Visualize JSON ▼

```

1 {
2   "error": 0,
3   "mensajeError": null,
4   "seguimientoEnvioFecha": [
5     {
6       "numEnvio": "418300000251624",
7       "referencia": null,
8       "estadoEnvio": "1"
9     },
10  ],
11 }
```

Del mismo modo, podríamos obtener errores devueltos en la búsqueda, como, por ejemplo:

Body Cookies Headers (13) Test Results

Pretty Raw Preview Visualize JSON ▼

```

1 {
2   "error": 402,
3   "mensajeError": "ERROR IN DB - NO DATA FOUND",
4   "seguimientoEnvioFecha": []
5 }
```