



Sistemas Electrónicos del Paraguay S.A.
www.sepsa.com.py

API – Especificación de Servicio REST Proyecto Recepción Ciega

Histórico de Revisiones

Fecha	Versión	Descripción del motivo de revisión	Autor
12/AGO/2019	0.0.1	Creación del documento.	N/D

Índice

Resumen del documento	4
Preliminares	4
Punto de Acceso	4
Formato de Intercambio de Datos	4
Tipos de Respuestas y Formato	4
Códigos de Respuesta	5
Endpoints	5
Sistema	5
Ping	6
Documentos	6
Importar Orden de Compra.....	6
Consultar Notas de Remisión	7
Consultar Recepciones Aprobadas	9

Resumen del documento

Este documento presenta los métodos expuestos en forma de servicio REST de manera que clientes externos puedan consumir el servicio general de recepción ciega de insumos. Específicamente, estos métodos expuestos servirán como base para que el servidor y los diferentes clientes puedan establecer una comunicación directa y homogénea.

Se describe en las secciones siguientes:

- Métodos expuestos de forma pública.

Preliminares

En esta sección se hace una introducción breve a diferentes conceptos y decisiones tomadas dentro del diseño del API.

Punto de Acceso

Se define una URL base que indicará un camino homogéneo para consumir los diferentes endpoints que se presentarán en las secciones posteriores. Esta URL base contara con el siguiente formato:

[esquema]://[autoridad]/[contexto]/rest/api/[versión]/[endpoint]

Dónde:

- **Esquema:** típicamente HTTP o HTTPS. Idealmente **HTTPS**.
- **Autoridad:** Dirección IP o nombre de dominio correspondiente al servidor al cual se hace referencia, especificando opcionalmente un número de puerto.
- **Contexto:** Nombre del espacio de aplicaciones en el cual se accede al servicio.
- **Versión:** Identificador de versión del API que se desea consumir. Ejemplo: v1, v1.1, v2, etc.
- **Endpoint:** Path independiente del método que se desea consumir.

Ejemplo:

http://www.sepsa.com.py/recepcion/rest/api/v1/sistema/ping

Formato de Intercambio de Datos

Se establece como formato de intercambio y serialización de datos al formato JSON. El cual es considerado un estándar *de-facto* para la representación y transporte de datos entre clientes y servidores de un servicio REST.

Al utilizar JSON, se espera que los clientes envíen cadenas de texto conforme a las especificaciones de cada endpoint del API y a su vez el servidor retornará la correspondiente respuesta según indique.

Tipos de Respuestas y Formato

El servidor responderá siempre una respuesta JSON con un cuerpo base genérico el cual se verá modificado por los diferentes endpoints definidos. Básicamente, el cuerpo de una respuesta del servidor estará compuesto por los siguientes elementos:

- **Status:** Estado de la respuesta desde la perspectiva del servidor que indica el resultado de la operación, agregando otros posibles datos adicionales.
- **Payload:** Datos relativos a la respuesta específica de la operación efectuada, que interesa al cliente que consume el endpoint.

El campo status corresponderá a datos específicos que indican el estado final de la operación. En caso de una operación exitosa, el status presentará los siguientes subcampos:

- **Success:** Con un valor igual a **true**. Indicando que la operación fue exitosa.
- **Timestamp:** Numero entero que representa el tiempo en milisegundos (UTC) en que la respuesta fue generada por el servidor, relativo al tiempo local del servidor.

Ejemplo:

```
{
  status: {
    timestamp: [numérico],
    success: true
  },
  payload: [json]
}
```

En caso de que la operación tuviera algún error, el campo status de la respuesta incluirá un subcampo adicional llamada **cause** la cual indicará en forma de texto la causa por la cual la operación no pudo ser efectuada de manera correcta.

Ejemplo:

```
{
  status: {
    timestamp: [numérico],
    success: false,
    cause: [texto]
  },
  payload: [json]
}
```

Códigos de Respuesta

En un caso ideal, debería ser suficiente para el consumidor del servicio verificar el campo status de la respuesta recibida desde el servidor para saber si la operación efectuada tuvo éxito. Sin embargo, existen algunas situaciones donde además de esta forma genérica de respuesta es necesaria una indicación más propia del protocolo subyacente de transporte (HTTP o HTTPS) para indicar el estado final de una operación. Para eso se acuerda la utilización de cuatro códigos numéricos para determinar semánticamente el estado de una operación. Las cuales son:

- **HTTP 200 – OK:** Enviado por defecto para peticiones que fueron procesadas correctamente por el servidor. Esto no necesariamente significa que el subcampo **success** del campo **status** sea verdadero.
- **HTTP 400 – BAD REQUEST:** Enviado cuando los datos de entrada de la operación no están en el formato correcto o son inválidos.
- **HTTP 403 – FORBIDDEN:** Enviado cuando no se tiene autorización para realizar una operación dada. Generalmente esto se soluciona iniciando una nueva sesión en el sistema.

Endpoints

En esta sección se definen los diferentes endpoints o métodos que se hacen públicos para ser consumidos por los diferentes clientes.

Sistema

En esta sección se definen las operaciones relacionadas al estado general del sistema.

Ping

Permite conocer si el sistema está activo.

Path	/sistema/ping
Método	GET
Parámetros en URL	Ninguno
Parámetros en Cabecera	Ninguno
Parámetros en Cuerpo	Ninguno
Respuesta - Éxito	<p>Código: 200</p> <ul style="list-style-type: none"> Una cadena de texto simple indicando que el sistema está en línea. <p>Ejemplo:</p> <pre>{ "pong" }</pre>
Respuesta - Error	Ninguna
Observaciones	Ninguna

Documentos

En esta sección se definen las operaciones relacionadas a la consulta y manipulación de documentos transmitidos.

Importar Orden de Compra

Permite el alta de una nueva orden de compra.

Path	/oc/importar
Método	POST
Parámetros en URL	Ninguno
Parámetros en Cabecera	Ninguno
Parámetros en Cuerpo	<p>Datos concretos de la orden de compra a importar:</p> <ul style="list-style-type: none"> Datos de Orden de Compra: número, fecha de emisión. Datos del Proveedor: RUC, razón social. Datos de entrega: condiciones especiales de entrega, fecha límite para plazo de entrega, lugar de entrega. Datos de carácter legal: fecha de contrato, número de contrato, número de resolución, número de licitación. Datos de insumos solicitados: cantidad, código GTIN de producto, código interno de producto, marca, origen.

	<p>Ejemplo:</p> <pre> { "nro": [texto], "fecha": [fecha], "proveedor": { "razon_social": [texto], "ruc": [texto] }, "entrega": { "condiciones": [texto], "fecha": [fecha], "lugar": [texto], }, "legal": { "fecha_contrato": [fecha], "nro_contrato": [texto], "nro_licitacion": [texto], "nro_resolucion": [texto] }, "items": [{ "cantidad": [numérico], "producto": { "codigo": [texto], "gtin": [texto], "descripcion": [texto], "marca": [texto], "precio": [numérico], "procedencia": [texto] } }] } </pre>
Respuesta - Éxito	Código 200: <ul style="list-style-type: none"> Respuesta base de éxito con campo <code>success=true</code>.
Respuesta - Error	Código 400: <ul style="list-style-type: none"> Si el cuerpo de la petición presenta datos incorrectos o incompletos.
Observaciones	Ninguna

Consultar Notas de Remisión

Permite consultar y obtener las notas de remisión finales relacionadas a una orden de compra cuyos avisos de despacho (proformas) fueron confirmadas y aprobadas.

Path	/oc/notasRemision
Método	GET
Parámetros en URL	<ul style="list-style-type: none"> oc: Número de orden de compra.

	<p>Ejemplo:</p> <pre>oc=[texto]</pre>
Parámetros en Cabecera	Ninguno
Parámetros en Cuerpo	Ninguno
Respuesta - Éxito	<p>Código 200: Array de objetos JSON que describen las notas de remisión para una orden de compra. Por cada ítem en la nota de remisión se detalla:</p> <ul style="list-style-type: none"> • Datos de Recepción: número, fecha de emisión de la nota de remisión, número de orden de compra relacionada. • Datos de productos enviados por el proveedor: datos propios de producto, cantidad enviada, presentación, lote, vencimiento, datos logísticos relacionados al producto, observaciones adicionales. <p>Ejemplo:</p> <pre>{ "notas_remision": [{ "nro": [texto], "fecha": [fecha], "oc": [texto], "items": [{ "producto": { "codigo": [texto], "gtin": [texto], "descripcion": [texto], "marca": [texto], "precio": [numérico], "procedencia": [texto] }, "cantidad": [numerico], "presentacion": [texto], "lote": [texto], "vencimiento": [fecha], "logistica": { "unidad": [texto], "ancho": [numerico], "alto": [numerico], "profundo": [numerico], "peso": [numerico], } }, { "observaciones": [texto] }] }] }</pre>

Respuesta - Error	Ninguna
Observaciones	Ninguna

Consultar Recepciones Efectivas

Permite consultar y obtener las recepciones aprobadas relacionadas a una orden de compra en particular.

Path	/oc/recepciones
Método	GET
Parámetros en URL	<ul style="list-style-type: none"> oc: Número de orden de compra. <p>Ejemplo:</p> <pre>oc=[texto]</pre>
Parámetros en Cabecera	Ninguno
Parámetros en Cuerpo	Ninguno
Respuesta - Éxito	<p>Código 200: Array de objetos JSON que describen las recepciones aprobadas para una orden de compra. Por cada ítem de recepción se detalla:</p> <ul style="list-style-type: none"> Datos de Recepción: número, fecha de recepción, número de nota de remisión relacionada. Datos de productos recepcionados: datos propios de producto, lote, vencimiento, observaciones adicionales. <p>Ejemplo:</p> <pre>{ "recepciones": [{ "nro": [texto], "fecha": [fecha], "nr": [texto], "items": [{ "producto": { "codigo": [texto], "gtin": [texto], "descripcion": [texto], "marca": [texto], "precio": [numérico], "procedencia": [texto] }, "cantidad": [numérico], "lote": [texto],</pre>

	<pre> "vencimiento": [fecha], "observaciones": [texto] }] } </pre>
Respuesta - Error	Ninguna
Observaciones	Ninguna