

# **HOSPITAL CLÍNIC DE BARCELONA**

PROYECTO:  
TRANSFORMACIÓN TECNOLÓGICA DE LA GESTIÓN DE  
MATERIALES

Etiquetado RFID e Integración  
de Proveedores

HCB

*ABRIL DE 2026*

# 1. Índice

<b>1.</b>	<b>ÍNDICE</b> .....	<b>2</b>
<b>2.</b>	<b>RESUMEN EJECUTIVO DEL DOCUMENTO</b> .....	<b>2-1</b>
<b>3.</b>	<b>CODIFICACIÓN RFID</b> .....	<b>3-2</b>
3.1.	ALCANCE.....	3-2
3.2.	PROCESO DE ETIQUETAJE.....	3-2
3.2.1.	<i>Formato físico de la etiqueta RFID:</i> .....	3-2
3.2.2.	<i>Codificación de la etiqueta RFID:</i> .....	3-3
<b>4.</b>	<b>CONSIDERACIONES PARA LA INTEGRACIÓN LOGÍSTICA CON EL HCB</b> .....	<b>4-4</b>
4.1.	URL TEST Y PROD .....	4-4
<b>5.</b>	<b>API</b> .....	<b>5-5</b>
5.1.	AUTENTICACIÓN POR JWT .....	5-6
5.2.	CONSULTA PARA CONFIRMAR QUE EL SERVICIO ESTÁ OPERATIVO.....	5-7
5.3.	ASOCIAR UN TAG RFID A UN ARTÍCULO EN EL PROVEEDOR.....	5-7
5.3.1.	<i>Consideraciones y errores posibles</i> .....	5-10
5.4.	EL PROVEEDOR PUEDE CONSULTAR LOS CONSUMOS.....	5-11
5.5.	EL PROVEEDOR PUEDE CONSULTAR LOS STOCKS .....	5-14
5.6.	EL PROVEEDOR PUEDE CONSULTAR TAGS CONCRETOS.....	5-16

## 2. Resumen ejecutivo del documento

Este manual técnico está dirigido a los proveedores de material médico del Hospital Clínic de Barcelona (HCB), tanto de material fungible como de prótesis, que deseen implementar el etiquetado RFID en origen. El objetivo es facilitar la integración logística y la trazabilidad de los productos mediante tecnología RFID, optimizando la gestión y el control de inventarios.

El documento describe:

- Los requisitos mínimos para que los proveedores puedan etiquetar sus materiales con RFID.
- Los procedimientos para el registro y consulta de etiquetas a través de una API o una plataforma web específica.
- Las normas de codificación aceptadas (SGTIN+), con ejemplos detallados de estructura de datos y formatos.
- Los endpoints y métodos de autenticación necesarios para interactuar con el sistema del HCB, incluyendo el uso de JWT y ApiKey.
- Los posibles errores y consideraciones técnicas en el proceso de integración.
- Los procedimientos para consultar consumos, stocks y estados de los tags RFID asociados a los artículos suministrados.

El manual enfatiza la importancia de coordinar el alta de los EAN/GTIN en el sistema del HCB antes de iniciar el proceso y proporciona contactos de soporte tanto del hospital como del proveedor tecnológico. Dado que los EAN deben de estar previamente dados de ALTA en el sistema de HCB, se debe coordinar con su contacto para registrar estos mismos en caso de no estar registrados. Contactar con [altaean@clinic.cat](mailto:altaean@clinic.cat).

## 3. Codificación RFID

### 3.1. Alcance

Este documento pretende dar toda la información necesaria para que cualquier proveedor de material fungible y/o prótesis, en modalidad de compra directa, o bien, de pago por uso (realizando depósitos y/o consignas), pueda formar parte de la cadena logística más eficaz y eficiente del HCB mediante la tecnología de trazabilidad RFID.

Se definirán, por lo tanto:

- 1) Los requisitos mínimos para que el proveedor pueda etiquetar con sus medios sus materiales.
- 2) El intercambio de información necesaria en el ámbito de sistemas para reducir manualidades integrando el sistema del proveedor y del HCB por la gestión logística de este documento pretende dar toda la información necesaria para que cualquier proveedor de material fungible y/o prótesis, en modalidad de compra directa, o bien, de pago por uso (realizando depósitos y/o consignas), pueda formar parte de la cadena logística más eficaz y eficiente del HCB mediante la tecnología de trazabilidad RFID.

### 3.2. Proceso de etiquetaje

En este apartado definiremos el resultado esperado del etiquetado por parte del proveedor, en ningún caso entraremos a definir la tipología de software ni hardware que deberá utilizar para realizar la codificación del tag, pero sí los mínimos necesarios para garantizar la integración y la lectura de los tags RFID con la infraestructura del HCB teniendo en cuenta los estándares internacionales.

#### 3.2.1. Formato físico de la etiqueta RFID:

Requisitos para la validación:

- Debe operar en UHF de forma pasiva con Frecuencia de funcionamiento 860 - 960 MHz y con distancia mínima de lectura  $\geq 3$  metros
- Debe operar bajo las Normas Internacionales } EPC Class 1 Gen 2 / ISO/IEC 18000-63.
- Tamaño de 97 mm x 15 mm o similares.
- Para productos que se dificulte la lectura de etiquetas planas, la utilización de etiquetas tipo banderola o equivalentes que se puedan leer con dispositivos RFID con fiabilidad.
- Etiquetas operativas durante todo el ciclo logístico. - No deben desprenderse durante transporte o manipulación
- Pre-codificados con datos del producto según EPC Tag Data Standard que define la EPC™, y también especificados los contenidos de la memoria de los Tags RFID Gen 2. La pre-codificación debe contener información de la unidad a consumir según el estándar internacional GS1 con el tipo de codificación SGTIN+.
- El HCB realizará pruebas de validación de lectura previas al suministro recurrente.

### 3.2.2. Codificación de la etiqueta RFID:

Por parte del HCB se aceptará un solo tipo de codificación SGTIN+ basada en los estándares internacionales GS1 de codificaciones con el objetivo de seguir la guía de interoperabilidad entre RFID y códigos de barras: [https://www.gs1.org/docs/barcodes/RFID\\_Barcode\\_Interoperability\\_Guidelines.pdf](https://www.gs1.org/docs/barcodes/RFID_Barcode_Interoperability_Guidelines.pdf)

- 1) SGTIN :Norma para chips con un mínimo de 256 bits de EPC. Esto es un ejemplo del formato a seguir, para poder ver el estándar SGTIN+ actualizado lo puede encontrar en la información de la web: <https://ref.gs1.org/standards/tds/>

	Header	+AIDC toggle bit	Filter Value	GTIN														Encoding Indicator	Length Indicator	GTIN Serial Number SN	Expiration Date					Lot				
				+AIDC Data Header_1	+AIDC Data_1 (YYMMDD)	+AIDC Data Header_2	Encoding Indicator	Length Indicator	+AIDC Data_2 (Lot Number)																					
Raw Data	(SGTIN+)	(0=FALSE / 1=TRUE)	1=POS Item	9	5	2	1	4	3	2	8	5	1	7	3	6	4	(integer encoding)	(char number of SN)	12345	(1	7)	22	05	31	(1	0)	(integer encoding)	(char number of SN)	47958709
Bit Input	11110111	1	001	10000001	010001	001001	001001	001001	001001	001001	001001	001001	001001	001001	001001	001001	000	00101	0001100000111001	0001	0111	00101101	0111	0001	0000	000	01000	010110110111100101010110101		
Bit qty	8	1	3	4	4	4	4	4	4	4	4	4	4	4	4	4	3	5	17	4	4	7	4	5	4	4	3	5	27	
Hex Output	F7	9	9	5	2	1	4	3	2	8	5	1	7	3	6	4	05	3039	17	2CBF	10	08	2DBCAB5							

EPC Output	F7995214328517364053039172CBF10082DBCA B5
------------	---

Con esta codificación estándar se podrán trazar internacionalmente cada uno de los materiales etiquetados mediante RFID. La información logística para integrar con el HCB, como el albarán, se enviará vía Web Service (ver punto 4: Consideraciones para la integración logística con el HCB).

## 4. Consideraciones para la Integración logística con el HCB

### 4.1. URL TEST y PROD

TEST:

API <https://test-proveedorshcb.lamarfid.com:18014>

Ejemplo para un endpoint: <https://test-proveedorshcb.lamarfid.com:18014/api/ping>

Web <https://test-proveedorshcb.lamarfid.com>

PROD: <https://proveedorshcb.lamarfid.com:18010>

API <https://proveedorshcb.lamarfid.com:18010>

Ejemplo para un endpoint: <https://proveedorshcb.lamarfid.com:18010/api/ping>

Web <https://proveedorshcb.lamarfid.com>

Vigilar los puertos distintos de PROD y TEST.

## 5. API

Todos los requests son HTTP POST/GET JSON, con Content-Type = “application/json”.

En algunos sistemas, puede ser que un endpoint <http://example/> funcione y otro <http://example> no funcione o también a la inversa (ver barra final). Atención también si la URL es case-sensitive o case-insensitive.

Si se llama a un endpoint inexistente, la petición recibe HTTP 404 Not Found.

Si se llama un endpoint con el método equivocado (GET en vez de POST por ejemplo), recibe un HTTP 405 Method not allowed.

Si se llama un endpoint proporcionando en el body un JSON mal formado, recibe un HTTP 400 Bad Request.

Otros posibles errores al realizar la petición, se recibe una respuesta HTTP No-OK.

Si la petición llega a la API correctamente, se recibirá siempre una respuesta HTTP 200 OK con un objeto JSON de respuesta. Este objeto puede representar una operación correcta o incorrecta.

- Si es correcta, el objeto contiene el campo StatusCode = 200
- Si es incorrecta, el objeto contiene el campo StatusCode <> 200.

El campo Message contiene el texto del error. El campo IDError el código del error.

El usuario de la API debe tener:

- Acceso a los endpoints que se encuentran en servidor ubicado dentro de red privada del Clínic.
- Token de acceso a la API. Este token fijo (key) es único para cada proveedor y se les proporciona por otro canal (no hay proceso de login para obtener el token)

## 5.1. Autenticación por JWT

Actualmente se autentica el mensaje usando un campo Token en el JSON del body de la request que contiene un secreto compartido entre el proveedor y el middleware.

Este método queda desfasado y se sustituye por el siguiente método:

- 1) El proveedor accede al servicio (por HTTPS siempre) de autenticación, en un endpoint:

Endpoint: HTTP POST
<a href="https://{hostname}/api/LoginProveedor">https://{hostname}/api/LoginProveedor</a>
Request:
<pre>{   "grant_type": "client_credentials",   "client_id": "", // id de cliente, proporcionado a Campo ApiKey   "client_secret": "", // secreto de cliente, proporcionado a Campo ApiSecret   "client_source": "" // en blanco }</pre>
Response: HTTP 200 OK
<pre>{   "access_token": "\${TOKEN_JWT}",   "expires_in": segundos,   "message": "OK",   "status_code": 200,   "token_type": "Bearer" }</pre>

La privacidad de la comunicación de las credenciales queda protegida por el TLS de la llamada HTTPS.

El token JWT proporcionado (de caducidad indeterminada), debe añadirse a todas las llamadas a servicio (excepto /PING) para autenticarse y poder ejecutarse esa operación.

El token se pasará en un header Authorization de tipo Bearer, por ejemplo:

“Authorization: Bearer **TOKEN**”

Quedan por concretar el significado de los campos opcionales del token JWT. Por el momento, el propio token JWT autentica la petición realizada al servicio.

Las credenciales (usuario y contraseña) del proveedor no son editables por el proveedor y se les enviarán por un canal aparte.

## 5.2. Consulta para confirmar que el servicio está operativo

Endpoint: HTTP GET
<a href="https://{hostname}/api/Ping">https://{hostname}/api/Ping</a>
Request:
(nada)
Response: HTTP 200 OK
<pre>{   "ID": "@Clinic_SVC_2: Ping 2.7.1214.0",   "Message": "2023-04-20 10:41:59",   "StatusCode": 200 }</pre>

Se puede ejecutar en un navegador web.

## 5.3. Asociar un tag RFID a un artículo en el proveedor

Esta llamada API permite a los proveedores registrar al sistema Middleware qué tags RFID van con cada artículo. Se registra con los códigos de artículo del Clínic. Así, en el momento que los tags sean leídos por una antena del Clínic, en ese instante ya puede aplicarse la trazabilidad o el consumo de los tags.

El proveedor debe tener a priori:

- Tener los EAN13/DUN14 (GTIN) ya registrados al ERP del Clínic. Si no son conocidos, se rechazará la operación.
- El código de pedido, número de línea del pedido (opcionales) y el código del albarán (obligatorio).

- Saber el tipo de artículo o de asociación que quiere registrar: Normal, Prótesis, Fungible, ...
- En esta versión 1 de la API:
  - Se suben las asociaciones 1 por 1 (el array ListaAsociaciones contiene solo 1 elemento).
  - El array de Clave-Valor CamposExtra, no se utiliza (el array está vacío).
  - El campo TipoAsociacion debe ser siempre “Normal”.
  - Solo se permite registrar material Fungible o Prótesis.
- Campos opcionales:
  - NumSerie (obligatorio para artículos de tipo Prótesis, “N/A” en los otros casos)
  - NumPedido
  - PosicionPedido
- Campos obligatorios:
  - FechaCaducidadString
  - Lote
  - Albaran
  - GTIN (EAN13 o DUN14)
  - EPC
  - TipoArticulo

- TipoAsociacion (debe valer "Normal")

Endpoint: HTTP POST
<a href="https://{hostname}/api/GuardarAsociacionesProveedor">https://{hostname}/api/GuardarAsociacionesProveedor</a>
Request:
<pre>{   "Version": 1, // Siempre 1   "ApiKey": "API_KEY", // Pasamos de nuevo nuestro api key por este campo   "ListaAsociaciones": [ // Array con 1 solo elemento     {       "GTIN": "EAN13 o DUN14 del artículo, debe existir previamente en BD",       "EPC": "ABCD0000111122223333", // EPC a asociar, en hexadecimal       "FechaCaducidadString": "2023-01-01 00:00:00",       "Lote": "valor lote",       "NumSerie": "valor num. Serie",       "NumPedido": "valor pedido", // string(10)       "PosicionPedido": 123,       "Albaran": "valor albarán", // string(16)       "TipoArticulo": "Fungible",       "TipoAsociacion": "Normal",       "CamposExtra": [         {           "Clave": "clave",           "Valor": "valor"         }       ]     }   ] }</pre>
Response OK: HTTP 200 OK
<pre>{   "StatusCode": 200, // Siempre 200   "Message": "Texto informativo o en blanco",   "IDError": "OK", // OK o en blanco   "Data": "" // en blanco }</pre>

} Response ERROR: HTTP 200 OK
{ "StatusCode": 400, // Distinto de 200 "Message": "Mensaje de error", "IDError": "Código del error", "Data": "" // en blanco }

<b>TipoArticulo:</b>
Fungible à Material Fungible
Protesis à Prótesis

<b>TipoAsociacion:</b>
Consigna à Consigna (No usar)
Normal à Normal (Usar siempre este)

Nota: Si un EAN está registrado como Fungible en el Middleware, se cambiará el tipo indicado en la API por la de ese tipo.

### 5.3.1. Consideraciones y errores posibles

Causas y respuestas de errores posibles:

- E101: Debe definir Version 1
- E102: Solo se permite mandar 1 asociación por llamada en Versión 1
- E103: Debe definir EPC
- E104: Debe definir GTIN (EAN13, DUN14, ...)
- E105: Debe definir FechaCaducidadString

- E106: No se puede parsear FechaCaducidadString (ISO8601 yyyy-MM-ddTHH:mm:ss.fffzzz o bien yyyy-MM-dd HH:mm:ss). Se acepta yyyy-MM-ddT00:00:00.000Z para indicar una fecha sin hora.
- E107: Debe definir Lote
- E108: Debe definir NumSerie
- E109: Debe definir Albaran
- E110: EPC duplicado, ya asociado
- E111: No se permite asociar un GTIN (EAN13/DUN14) desconocido en el sistema Middleware
- E112: Tipo de asociación incorrecto (Consigna, Normal, ...)
- E113: Tipo de artículo incorrecto (Fungible, Protesis, ...)
- E200: Token inválido
- E300: Otros errores

Los strings demasiado largos se cortarán (máximo típico de 50 caracteres).

El sistema desconoce la categoría de los artículos: si son prótesis o no, ...

## 5.4. El proveedor puede consultar los consumos

Este endpoint permite consultar los consumos de tags realizados en un rango de fechas, y para unos filtros concretos (por EPC, por GTIN, por Ubicación, por Almacén, ...). Los filtros se aplicarán de la manera más exacta posible, si son contradictorios el resultado será una lista vacía. El resultado de una consulta puede pedirse troceado para no bloquear el sistema: Por ejemplo, en bloques de 1000 (de 0 a 999, de 1000 a 1999, de 2000 a 2999, ...). Los parámetros de conexión necesarios (nombre ubicaciones, código de proveedor, ...) se proporcionarán a su debido tiempo.

La respuesta contiene los datos tal como están registrados en la base de datos del sistema, o en blanco si no se disponen.

Para consultar los consumos, debe indicarse "C" en el campo Operación.

Un proveedor solo podrá consultar sus propios datos de consumo o stock. Esto se asegura con el uso del token secreto, único para cada proveedor.

Endpoint: HTTP POST (Consulta consumos)
<a href="https://{hostname}/api/ConsultaTagsProveedor">https://{hostname}/api/ConsultaTagsProveedor</a>
Request:
<pre>{   "Version": 1, // Siempre 1   "Filtros": { // Se puede dejar vacío     "EPC": "1234", // EPC a consultar, o en blanco para todos     "GTIN": "1234", // DUN14 o EAN13 a consultar, o en blanco para todos     "Almacen": "nombre", // Código de almacén, o en blanco para todos     "Ubicacion": "nombre" // Código de ubicación, o en blanco para todos   },   "Operacion": "C", // C para consumidos, S para stocks   "FechaHoraConsumoInicio": "2023-10-30 00:00:00", // en blanco o inicio rango   fechas de consumo, formato ISO   "FechaHoraConsumoFinal": "2024-10-30 00:00:00", // en blanco o final rango   fechas de consumo, formato ISO   "Take": 1000, // Número máximo de registros a retornar   "Skip": 0 // Índice del primer registro a retornar }</pre>
Response OK: HTTP 200 OK
<pre>{   "NumTags": 123, // Cantidad de tags en respuesta   "Tags": [     {       "EPC": "1234", // EPC encontrado en la BD       "GTIN": "1234", // DUN14 o EAN13 registrado para ese tag       "FechaHoraCaducidad": "2024-10-30 00:00:00", // En formato ISO o en blanco si       se desconoce       "Lote": "abc", // Lote o en blanco si se desconoce       "NumSerie": "abc", // Número de serie o en blanco si se desconoce       "CodProveedorSAP": "abc", // Código del proveedor       "CodMaterialSAP": "1234", // Código asignado a ese artículo por el hospital     }   ] }</pre>

<pre> "RefProveedor": "abc", // Referencia asignada a ese artículo, "Consigna": "X", // "X" si está considerado material en consigna, en blanco si no "Consumido": "X", // "X" si consumido, "" si en stock "FechaHoraConsumo": "2024-10-30 00:00:00", // En formato ISO o en blanco si no consumido "Almacen": "abc", // Almacén (o instituto) visto el tag por última vez "Ubicacion": "abc", // Si Stock, ubicación actual del tag, si consumo, ubicación de consumo "Estado": "abc", // Campo Estado actual del tag. No se informa, se espera en blanco. "Extra": [ { "Clave": "abc", // Para futuras ampliaciones "Valor": "abc", // Para futuras ampliaciones }, ... ] }, ... ] "StatusCode": 200, // Siempre 200 "Message": "Texto informativo o en blanco", "IDError": "OK", // OK o en blanco "Data": "" // en blanco } </pre>
Response ERROR: HTTP 200 OK
<pre> { "StatusCode": 400, // Distinto de 200 "Message": "Mensaje de error", "IDError": "Código del error", "Data": "" // en blanco } </pre>

## 5.5. El proveedor puede consultar los stocks

Mismo endpoint i mismos parámetros que el punto anterior 4.4 de consulta de consumos, pero indicando la operación “S” de consulta de stocks. Para stocks, no hace falta indicar el período de consumo.

Como en el caso anterior, un proveedor solo podrá consultar sus propios datos de stock.

Endpoint: HTTP POST (Consulta stocks)
<a href="https://{hostname}/api/ConsultaTagsProveedor">https://{hostname}/api/ConsultaTagsProveedor</a>
Request:
<pre>{   "Version": 1, // Siempre 1   "Filtros": {     "EPC": "1234", // EPC a consultar, o en blanco para todos     "GTIN": "1234", // DUN14 o EAN13 a consultar, o en blanco para todos     "Almacen": "nombre", // Código de almacén, o en blanco para todos     "Ubicacion": "nombre" // Código de ubicación, o en blanco para todos   },   "Operacion": "S", // C para consumidos, S para stocks   "FechaHoraConsumoInicio": "", // en blanco para stocks   "FechaHoraConsumoFinal": "", // en blanco para stocks   "Take": 1000, // Número máximo de registros a retornar   "Skip": 0 // Índice del primer registro a retornar }</pre>
Response OK: HTTP 200 OK
<pre>{   "NumTags": 123, // Cantidad de tags en respuesta   "Tags": [     {       "EPC": "1234", // EPC encontrado en la BD       "GTIN": "1234", // DUN14 o EAN13 registrado para ese tag       "FechaHoraCaducidad": "2024-10-30 00:00:00", // En formato ISO o en blanco si se desconoce       "Lote": "abc", // Lote o en blanco si se desconoce       "NumSerie": "abc", // Número de serie o en blanco si se desconoce     }   ] }</pre>

```
"CodProveedorSAP": "abc", // Código del proveedor
"CodMaterialSAP": "1234", // Código asignado a ese artículo por el hospital
"RefProveedor": "abc", // Referencia asignada a ese artículo,
"Consigna": "X", // "X" si está considerado material en consigna, en blanco si no
"Consumido": "", // "X" si consumido, "" si en stock
"FechaHoraConsumo": "", // En blanco si en stock
"Almacen": "abc", // Almacén (o instituto) visto el tag por última vez
"Ubicacion": "abc", // Si Stock, ubicación actual del tag, si consumo, ubicación de
consumo
"Estado": "abc", // Campo Estado actual del tag. No se informa, se espera en
blanco.
"Extra": [
{
"Clave": "abc", // Para futuras ampliaciones
"Valor": "abc" // Para futuras ampliaciones
}, ...
]
}
"StatusCode": 200, // Siempre 200
"Message": "Texto informativo o en blanco",
"IDError": "OK", // OK o en blanco
"Data": "" // en blanco
}
```

Response ERROR: HTTP 200 OK

```
{
"StatusCode": 400, // Distinto de 200
"Message": "Mensaje de error",
"IDError": "Código del error",
"Data": "" // en blanco
}
```

## 5.6. El proveedor puede consultar tags concretos

Indicando una lista de tags, podemos consultar su ficha y estado actual.  
Como en el caso anterior, un proveedor solo podrá consultar sus propios datos.

Endpoint: HTTP POST (Consulta tags)
<a href="https://{hostname}/api/ConsultaTagsProveedorPorTag">https://{hostname}/api/ConsultaTagsProveedorPorTag</a>
Request:
<pre>{   "Version": 1, // Siempre 1   "Tags": [ // Lista de EPC a consultar     "123456",     "78910",     ...   ] }</pre>
Response OK: HTTP 200 OK
<pre>{   "NumTags": 123, // Cantidad de tags en respuesta   "Tags": [     {       "EPC": "1234", // EPC encontrado en la BD       "GTIN": "1234", // DUN14 o EAN13 registrado para ese tag       "FechaHoraCaducidad": "2024-10-30 00:00:00", // En formato ISO o en blanco si se desconoce       "Lote": "abc", // Lote o en blanco si se desconoce       "NumSerie": "abc", // Número de serie o en blanco si se desconoce       "CodProveedorSAP": "abc", // Código del proveedor       "CodMaterialSAP": "1234", // Código asignado a ese artículo por el hospital       "RefProveedor": "abc", // Referencia asignada a ese artículo,       "Consigna": "X", // "X" si está considerado material en consigna, en blanco si no       "Consumido": "", // "X" si consumido, "" si en stock       "FechaHoraConsumo": "", // En blanco si en stock       "Almacen": "abc", // Almacén (o instituto) visto el tag por última vez     }   ] }</pre>

```

"Ubicacion": "abc", // Si Stock, ubicación actual del tag, si consumo, ubicación de
consumo
"Estado": "abc", // Campo Estado actual del tag. No se informa, se espera en
blanco.

"Extra": [
{
"Clave": "abc", // Para futuras ampliaciones
"Valor": "abc" // Para futuras ampliaciones
}, ...
]
}, ...
]
"StatusCode": 200, // Siempre 200
"Message": "Texto informativo o en blanco",
"IDError": "OK", // OK o en blanco
"Data": "" // en blanco
}

```

Response ERROR: HTTP 200 OK

```

{
"StatusCode": 400, // Distinto de 200
"Message": "Mensaje de error",
"IDError": "Código del error",
"Data": "" // en blanco
}

```