

Open API Spec

Version Number	Release Date
1.0	November 5th, 2013
1.1	November 11th, 2013
1.2	November 12th, 2013
1.3	November 13th, 2013
1.4	November 18th, 2013
1.5	November 19th, 2013
1.6	November 28th, 2013
1.7	December 17th, 2013
1.8	December 19th, 2013
1.9	December 27th, 2013
1.10	January 13th, 2014
1.11	March 4th, 2014
1.12	March 10th, 2014
1.13	May 13th, 2014
1.14	August 25th, 2015
1.15	December 3 rd , 2015
1.16	December 8th, 2015
1.17	July 18th, 2016
1.18	August 15th, 2017
1.19	September 10th, 2018
2.0	April 05th, 2019

Getting started

To start using Elevenia's Open API, please refer to these step:

1. Register as a seller

Before starting to utilize Elevenia's Open API resources, developers must register as a seller and provide required document (please refer to [register as an individual seller](#) or [business seller](#)) as a prerequisites. After registration submission, please check your email and click the confirmation link. Seller will have to wait for an approval as a valid seller by Elevenia (approval may take up to 1 x 24 hrs). When your account is ready, Elevenia will send you notification email.

The screenshot shows the Elevenia website's registration page for sellers. At the top, there's a navigation bar with links for 'Redeem Voucher', 'Download App', 'Login', 'Register', 'Seller', and 'Help'. Below the navigation bar is the Elevenia logo and a banner for 'JAGADIRI' with the text 'DAFTAR JADI SELLER elevenia SEKARANG LANGSUNG DAPAT PERLINDUNGAN JIWA GRATIS!'. The main heading asks 'Ingin berjualan di elevenia? Daftar sebagai Seller'. There are three registration options: 'Penjual Perorangan' (Individual Seller), 'Penjual Badan Usaha' (Business Seller), and 'Penjual Global' (Global Seller). Each option has a 'Daftar' button. At the bottom, there's a disclaimer and a footer with copyright information and a language selector.

Redeem Voucher Download App Login Register Seller Help

elevenia

elevenia x JAGADIRI
DAFTAR JADI SELLER elevenia SEKARANG
LANGSUNG DAPAT PERLINDUNGAN JIWA
GRATIS!

Ingin berjualan di elevenia?
Daftar sebagai Seller

Penjual Perorangan
Individual Seller
Penjual yang tidak terdaftar sebagai perusahaan
Daftar

Penjual Badan Usaha
Business Seller
Penjual pemegang ijin usaha
Daftar

Penjual Global
Global Seller
Penjual dari luar Indonesia
(Sellers located outside Indonesia)
Daftar (Register)

Pendaftaran tidak dipungut biaya. Setelah mendaftar sesuai prosedur, Anda dapat langsung berjualan di elevenia.
Jika Anda sudah mendaftar sebagai pembeli, Anda juga dapat menjadi penjual dengan mengubah status keanggotaan Anda di **My elevenia**.
Siapkan dokumen Anda untuk mempercepat proses verifikasi.

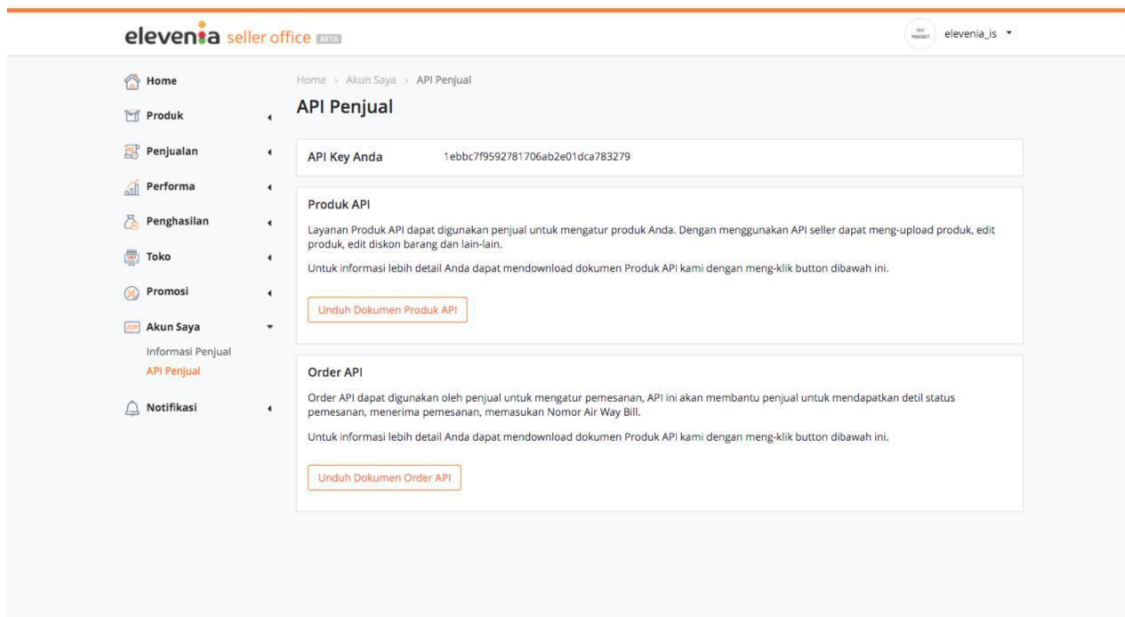
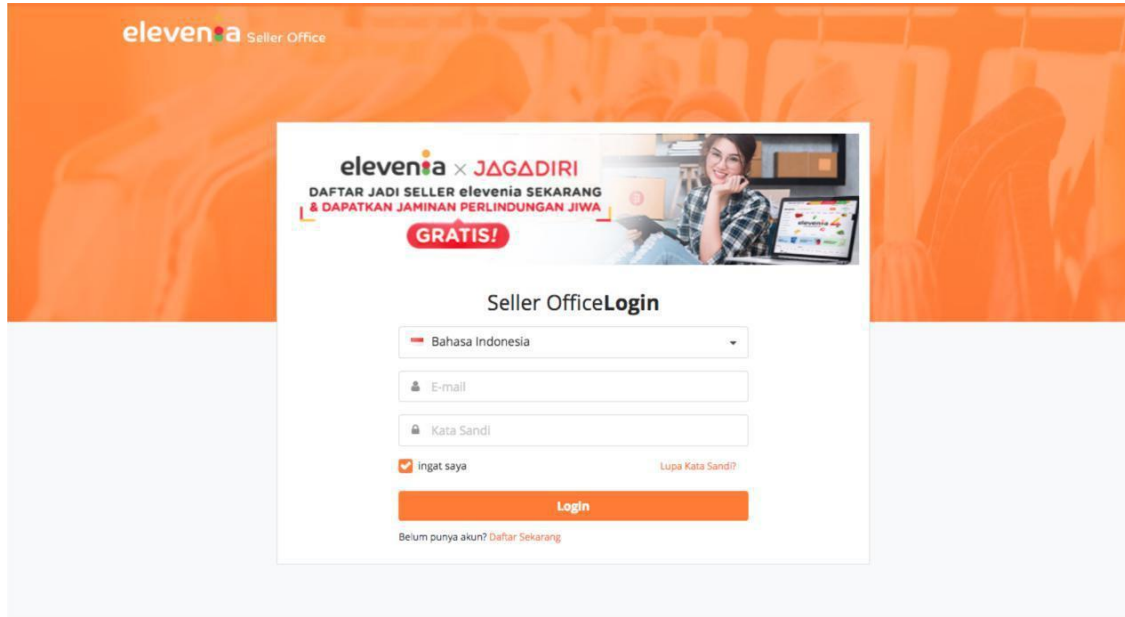
PT XL Planet sebagai pihak pengelola situs www.elevenia.co.id tidak bertanggung jawab atas kebenaran informasi dan tampilan produk yang didaftarkan oleh penjual.
Copyright © 2014-2018 PT XL Planet. All Rights Reserved.

Ganti ke versi mobile Bahasa English

2. Get API Key

Once developers have registered and approved as a seller, developers will obtain API Key. API key should be kept private by developers and will be used as authentication request to Elevenia Open API.

To get API Key, you need to log in to elevenia's seller office (<http://soffice.elevenia.co.id/login.do>). Go to Akun Saya – API Penjual menu. The API Key is located at the top of page (Please refer to image below).



3. Create Delivery Template

Before registering product, you need to define delivery template. Delivery template is an address template from where you send the product to buyer. You could define the delivery method, roles, and delivery area that is not covered by you. The delivery template is used to calculate the delivery cost.

To register delivery template, log in to seller office (please refer to step 2 on how to log in to seller office), go to “Product Information Template Management”/“Pengaturan template informasi produk” menu, then click “Delivery Template Registration”/“Registrasi Profil Pengiriman” button.

You could name the delivery template as descriptive as possible to avoid confusion. For example, you want to create a delivery service for TIKI & JNE for all province in Indonesia, you could name it “TIKI & JNE all province” or “TIKI & JNE regular service all province”.

You also have to define the address of the origin (your warehouse or store address) from the dropdown menu. The first address shown will be the address when you register your store in Elevenia for the first time. You can also add more address if you have any branch or different warehouse location from your store address.

You could also define the return address. Return address is the location to where your buyer have to return the product in case the product is not match as expected, bad quality and needs to be exchanged.

You can add up to 40 delivery template at max.

Daftar Template Informasi Pengiriman

Nama template	TIKI & JNE (10/40)	
Lokasi	Alamat Asal Pengiriman	Alamat Default Ubah/Tambah
	Alamat Pengembalian/Penukaran Produk	Kembali ke Jakarta Ubah/Tambah

[Klik di sini untuk melihat Manual Template Pengiriman](#)

Metode Pengiriman

<div> <div>elevenia express</div> <div>Regular Service</div> <div>Express Service</div> <div>Direct Delivery</div> <div>Same Day Delivery</div> <div>Ambil di Indomaret</div> </div>	
* Terapkan kondisi berikut <input type="checkbox"/> (Klik kotak di samping untuk menggunakan metode pengiriman ini) Click!	
Pengaturan Pengiriman	<div> <div>Biaya Pengiriman</div> <div> <input checked="" type="radio"/> Dibayarkan Langsung ke Kurir <input type="radio"/> Dibayarkan ke Penjual </div> </div>
	<div> <div>Pengiriman Bersyarat</div> <div> <input type="radio"/> Pick up <input type="radio"/> Drop off Pilih lokasi loker </div> </div>
	<div> <div>Pembeli diperbolehkan mengambil paket di loker</div> <div><input type="checkbox"/></div> </div>
	<div> <div>Perusahaan Pengiriman</div> <div></div> </div>

Kirim

Batal

Afterwards, you have to get the Delivery Template Number as a reference when you register / upload product using Elevenia API.

Daftar Template

Pendaftaran Template Pengiriman

	Ubah	Nomor	Nama Template	Jenis Template	Tanggal Pendaftaran
<input type="checkbox"/>	Ubah	536127	TIKI JNE	Template Pengir...	16/08/2018
<input type="checkbox"/>	Ubah	535985	Had JOB	Template Pengir...	15/08/2018
<input type="checkbox"/>	Ubah	535885	JNE SRG21602	Template Pengir...	13/08/2018
<input type="checkbox"/>	Ubah	535881	Majalaya Bandung	Template Pengir...	13/08/2018
<input type="checkbox"/>	Ubah	535354	tes tudey	Template Pengir...	06/08/2018
<input type="checkbox"/>	Ubah	535195	template dika tomcat	Template Pengir...	03/08/2018
<input type="checkbox"/>	Ubah	534207	xst232	Template Pengir...	18/07/2018
<input type="checkbox"/>	Ubah	533729	WMS	Template Pengir...	10/07/2018
<input type="checkbox"/>	Ubah	533105	jne pickup test	Template Pengir...	05/07/2018
<input type="checkbox"/>	Ubah	532329	FAJAR shipping label	Template Pengir...	12/06/2018
<input type="checkbox"/>	Ubah	532280	JNE Teluk Wondama	Template Pengir...	12/06/2018
<input type="checkbox"/>	Ubah	532164	Tes Diskon Bersyarat XRT 249	Template Pengir...	07/06/2018

Page 1 of 21 Jumlah data per halaman: 20 Hasil Pencarian : 416

When sending product using API, the xml will be wrapped in tmpItSeq tag:

```
<tmpItSeq>536127</tmpItSeq>
```

Please refer to ***Upload Product (Insert Product)*** section in this document on how to register product.

4. **Start Utilizing Elevenia API**

After step 1 and 2 completed, developers can now start utilizing Elevenia's Open API. There are two API servers, development (sandbox) for testing environment and the other is the live (production) environment.

- Testing Server: [contact us to get the testing environment]
- Production Server: <http://api.elevenia.co.id/rest/>

Authentication

Every request to Elevenia's Open API requires authentication header. Include the openapikey in the header of http request to Elevenia's api

Example:

```
curl -H "Content-type: application/xml" \  
      -H "Accept-Charset: utf-8" \  
      -H "openapikey: gty457h547rtyd456h78gy7456h758r5" \  
      \ api.elevenia.co.id
```

Authentication Error

```
<AuthMessage>  
  <resultCode>100</resultCode>  
  <resultMessage>Missing Open API Key in  
header</resultMessage> </AuthMessage>
```

Output Description

Element Name	Data Type	Description
resultCode	Numeric	Result code: 100 = Missing Open API Key in request header. 200 = Open API Key is invalid and not corresponds to any user. 300 = Seller has not been approved.
resultMessage	String	Authentication error message

API Resource Endpoints

Get Category Listing

Method Type : GET

Parameter : -

Message Type : -

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/cateservice/category>

Output description

This endpoint will return an xml of full category listing with the following xml elements.

Element name	Data Type	Description
ns2:category	-	Category(Node)
depth	Number	Depth of Tree structure. Elevenia has 3 level of category.
dispNm	String	Category Name
dispNo	Number	Category ID
parentDispNo	Number	Super ordinate Category ID

Output Example:

```
<ns2:categorys
xmlns:ns2="http://skt.tmall.business.openapi.spring.service.client.domain/"
> <ns2:category>
  <depth>1</depth>
  <dispNm>Laptop / Desktop</dispNm>
  <dispNo>1</dispNo>
  <gblDlvYn>Y</gblDlvYn>
  <parentDispNo>0</parentDispNo>
</ns2:category>
<ns2:category>
  <depth>2</depth>
  <dispNm>Notebook</dispNm>
  <dispNo>2</dispNo>
  <gblDlvYn>N</gblDlvYn>
```



```
    <parentDispNo>1</parentDispNo>
  </ns2:category>
  <ns2:category>
    <depth>3</depth>
    <dispNm>Samsung Electronics</dispNm>
    <dispNo>3</dispNo>
    <gblDlvYn>N</gblDlvYn>
    <parentDispNo>2</parentDispNo>
  </ns2:category>
  ...
  ...
  ...

  <ns2:category>
    <depth>3</depth>
    <dispNm>Atom</dispNm>
    <dispNo>7</dispNo>
    <gblDlvYn>N</gblDlvYn>
    <parentDispNo>3</parentDispNo>
  </ns2:category>
</ns2:category>
```

Get Sub-Category by Parent Category ID

Method Type : GET

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/cateservice/category/{dispCtgrNo}>

Input Description

Element name	Data Type	Mandatory	Description
dispCtgrNo	Number	Y	Parameter Category ID provided in the URL

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/catservice/category/1>

2. XML Data:

No XML Data

Output Description

An XML output of sub-category listing will be returned with the following elements.

Element name	Data Type	Description
ns2:category	-	Category(Node)
depth	Number	Depth of Tree structure. Elevenia has 3 level of category.
dispNm	String	Category Name
dispNo	Number	Category ID
parentDispNo	Number	Super ordinate Category ID

Get Category Attribute by Category ID

Method Type : GET

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/cateservice/categoryAttributes/{dispCtgrNo}>

Input Description

Element name	Data Type	Mandatory	Description
dispCtgrNo	Number	Y	Parameter Category ID provided in the URL

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/cateservice/categoryAttributes/138>

2. XML Data:

No XML Data

Output Description

An XML output of product category attributes listing will be returned with the following elements.

Element name	Data Type	Description
ns2:productCategoryAttribute	-	Product Category(Node)
prdAttrCode	Number	Category Product Attribute Code
prdAttrNm	String	Category Product Attribute Name
prdAttrNo	Number	Category Attribute ID

Output Example:

```
<?xml version="1.0" encoding="UTF-8"
standalone="yes"?> <ns2:productCtgrAttributes
xmlns:ns2="http://skt.tmall.business.openapi.spring.service.client.domain">
  <ns2:productCtgrAttribute>
    <prdAttrCd>2000005</prdAttrCd>
    <prdAttrNm>Brand</prdAttrNm>
    <prdAttrNo>161330</prdAttrNo>
  </ns2:productCtgrAttribute>
  <ns2:productCtgrAttribute>
    <prdAttrCd>2000037</prdAttrCd>
    <prdAttrNm>Material</prdAttrNm>
    <prdAttrNo>161331</prdAttrNo>
  </ns2:productCtgrAttribute>
</ns2:productCtgrAttributes>
```

Get List of Delivery Template

Method Type : GET

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/delivery/template>

Input description

No input needed

HTTP Request:

<http://api.elevenia.co.id/rest/delivery/template>

Output Description

Element name	Data Type	Description
dlvTpltNm	String	Delivery template name
dlvTpltSeq	String	Delivery template number This number will be used for registering product using xml tag: <tpltSeq>DLV NUMBER</tpltseq>

Output Example

1. Success:

```
<?xml version="1.0" encoding="UTF-8"
standalone="yes"?> <DeliveryTemplates>
  <template>
    <dlvTpltNm>TIKI & JNE (regular service)</dlvTpltNm>
    <dlvTpltSeq>231437</dlvTpltSeq>
  </template>
  <template>
    <dlvTpltNm>TIKI & JNE (express service)</dlvTpltNm>
    <dlvTpltSeq>231611</dlvTpltSeq>
  </template>
</DeliveryTemplates>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes" ?>
```

```
<ClientMessage>  
  <Message>No delivery template. Please create  
delivery template first.</Message>  
  <resultCode>500</resultCode>  
</ClientMessage>
```

Upload Product (Insert Product)

Method Type : POST

Parameter : XML

Message Type : XML/String

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/product>

Input description

Client have to send XML file of item which will be registered in Elevenia with the following elements:

Element name	Data Type	Mandatory	Description
selMnbdNckNm	String	Y	Nickname
selMthdCd	String (Number)	Y	Sales Type. '01': Ready Stock '04': Pre-Order '05': Used items
dispCtgrNo	Number	Y	Category ID
prdAttrCd	Number	Y	Category attribute code
prdAttrNm	String	Y	Category attribute name i.e: <i>brand, model, type, ISBN</i> (for books)
prdAttrNo	String	Y	Category attribute ID
prdAttrVal	String	Y	Category attribute value i.e: <i>samsung, galaxy, s3, EAN-</i> <i>123123123123</i>
prdNm	String	Y	Item Name
prdStatCd	String	Y	Item Conditions.

	(Number)		'01': New '02': Used
prdWght	Number	Y	Item weight in kilograms
dlvGrntYn	String	Y	Guarantee of delivery. The value is Y/N. If set as Y, seller is responsible to any damage happens during delivery.
minorSelCnYn	String	Y	Minors can buy (Under 17 years old). If set as N, no image is displayed to guest user and user under than 17 years old.
suplDtyfrPrdClfCd	String	Y	VAT/Tax Item Code. 01 = Barang Kena Pajak 02 = Barang Bebas Pajak * Default value is 01
prdImage01	String	Y	Main(Representative) Image URL
htmlDetail	String	Y	Item's Detailed Description (html format supported)
selTermUseYn	String	Y	Whether to use sales period (Y/N)
ProductCert certTypeCd	- String (Number)	- N	Authentication Information(Node) Type of Authentication
certKey	String	N *This parameter is mandatory IF spesific category selected	Authentication Number. Type of certification 01: BPOM 02: PIRT 03: Manual
selPrc	Number	Y	Product's price
prdSelQty	Number	Y	Set the product stocks

tmpltSeq	Number	Y	Number of Delivery information Template Number. Please refer to Getting Started > Create Delivery Template to get the Delivery number information.
asDetail	String	Y	After service information. Could be the address or contact info of the after-sales service.
rtngExchDetail	String	Y	Return/Exchange information

Input Example (Basic Product with Option):

```
<?xml version="1.0" encoding="UTF-8"?>
<Product>
  <selMnbdNckNm>test123</selMnbdNckNm>
  <selMthdCd>01</selMthdCd>
  <dispCtgrNo>361</dispCtgrNo>
  <ProductCtgrAttribute>
    <prdAttrCd>2000005</prdAttrCd>
    <prdAttrNm>Brand</prdAttrNm>
    <prdAttrNo>161753</prdAttrNo>
    <prdAttrVal>Apple</prdAttrVal>
  </ProductCtgrAttribute>
  <ProductCtgrAttribute>
    <prdAttrCd>2000009</prdAttrCd>
    <prdAttrNm>Camera Resolution</prdAttrNm>
    <prdAttrNo>161754</prdAttrNo>
    <prdAttrVal>13 MP</prdAttrVal>
  </ProductCtgrAttribute>
  <ProductCtgrAttribute>
    <prdAttrCd>2000010</prdAttrCd>
    <prdAttrNm>Capacity</prdAttrNm>
    <prdAttrNo>161755</prdAttrNo>
```

```

        <prdAttrVal>128 GB</prdAttrVal>
    </ProductCtgrAttribute>
    <ProductCtgrAttribute>
        <prdAttrCd>2000043</prdAttrCd>
        <prdAttrNm>Network</prdAttrNm>
        <prdAttrNo>161757</prdAttrNo>
        <prdAttrVal>HSDPA</prdAttrVal>
    </ProductCtgrAttribute>
    <ProductCtgrAttribute>
        <prdAttrCd>2000044</prdAttrCd>
        <prdAttrNm>Operating System</prdAttrNm>
        <prdAttrNo>161758</prdAttrNo>
        <prdAttrVal>iOS</prdAttrVal>
    </ProductCtgrAttribute>
    <ProductCtgrAttribute>
        <prdAttrCd>2000016</prdAttrCd>
        <prdAttrNm>Display Size</prdAttrNm>
        <prdAttrNo>161756</prdAttrNo>
        <prdAttrVal>5 inch</prdAttrVal>
    </ProductCtgrAttribute>
    <prdNm>iPhone 8 32 GB NEW Garansi!</prdNm>
    <prdStatCd>01</prdStatCd>
    <advrtStmt>Must Have</advrtStmt>
    <sellerPrdCd>SKU-55123413</sellerPrdCd>
    <prdWght>3</prdWght>
    <dlvGrntYn>Y</dlvGrntYn>
    <minorSelCnYn>Y</minorSelCnYn>
    <suplDtyfrPrdClfCd>01</suplDtyfrPrdClfCd>

    <prdImage01><![CDATA[https://boygeniusreport.files.wordpress.com/2017/04/iph
o ne-8-concept.jpg?quality=98&strip=all&strip=all]]></prdImage01>
    <prdImage02><![CDATA[https://9to5mac.files.wordpress.com/2017/04/iphone-8-
concept1.jpg?quality=82&strip=all&w=1000&strip=all&w=1600&h=1000]]></prdImage 02>

    <htmlDetail><![CDATA[IPHONE <strong>8</strong>]]></htmlDetail>
    <selPrc>750000</selPrc>
    <prdSelQty>10</prdSelQty>
    <asDetail>Kirim ke alamat gudang</asDetail>
    <rtngExchDetail>Dikembalikan ke
    toko</rtngExchDetail> <tmpltSeq>250758</tmpltSeq>
    <selTermUseYn>N</selTermUseYn>
    <optSelectYn>Y</optSelectYn>
    <colTitle>Size</colTitle>
    <txtColCnt>1</txtColCnt>
    <prdExposeClfCd>01</prdExposeClfCd>
    <productOptions>
        <colOptPrice>0</colOptPrice>
        <colValue0>GOLD</colValue0>
        <colCount>10</colCount>

```

```

        <optWght>0</optWght>
        <useYn>Y</useYn>
        <prdCrtfDtlsCd>02</prdCrtfDtlsCd>
        <prdCrtf>123pirt</prdCrtf>
    </productOptions>
    <productOptions>
        <colOptPrice>0</colOptPrice>
        <colValue0>BLACK</colValue0>
        <colCount>15</colCount>
        <optWght>0</optWght>
        <useYn>Y</useYn>
        <prdCrtfDtlsCd>02</prdCrtfDtlsCd>
        <prdCrtf>123pirt</prdCrtf>
    </productOptions>
</Product>

```

Input Example (Basic Product with Option):

```

<?xml version="1.0" encoding="UTF-8"?>
<Product>

    <selMnbdNckNm>Lodestar</selMnbdNckNm>
    <selMthdCd>01</selMthdCd>
    <dispCtgrNo><![CDATA[149]]></dispCtgrNo>
    o >

    <prdCrtfDtlsCd>01</prdCrtfDtlsCd>
    <prdCrtf>BPOM-VALUE</prdCrtf>

    <ProductCtgrAttributeS>
        <productCtgrAttribute>
            <prdAttrCd>2000005</prdAttrCd>
            <prdAttrNm>Brand</prdAttrNm>
            <prdAttrNo>161355</prdAttrNo>
        </productCtgrAttribute>
        <productCtgrAttribute>
            <prdAttrCd>2000051</prdAttrCd>
            <prdAttrNm>Product Type</prdAttrNm>
            <prdAttrNo>161356</prdAttrNo>
        </productCtgrAttribute>
        <productCtgrAttribute>
            <prdAttrCd>2000057</prdAttrCd>
            <prdAttrNm>Skin Type</prdAttrNm>
            <prdAttrNo>161357</prdAttrNo>
        </productCtgrAttribute>
        <productCtgrAttribute>
            <prdAttrCd>2000065</prdAttrCd>
            <prdAttrNm>Sun Protection</prdAttrNm>
            <prdAttrNo>161358</prdAttrNo>
        </productCtgrAttribute>
        <productCtgrAttribute>

```

```
<prdAttrCd>2017847</prdAttrCd>
<prdAttrNm>No.BPOM</prdAttrNm>
<prdAttrNo>167063</prdAttrNo>
</productCtgrAttribute>
<productCtgrAttribute>
  <prdAttrCd>2041808</prdAttrCd>
  <prdAttrNm>Sertifikasi</prdAttrNm>
  <prdAttrNo>173006</prdAttrNo>
</productCtgrAttribute>
</ProductCtgrAttributes>

<prdTypCd></prdTypCd>
<prdNm><![CDATA[global bpom cat mand no opt]]></prdNm>
<prdStatCd>01</prdStatCd>
<useMon></useMon>
<paidSelPrc></paidSelPrc>
<exteriorSpecialNote></exteriorSpecialNote>
<prdWght>1</prdWght>
<dlvGrntYn>N</dlvGrntYn>
<minorSelCnYn>Y</minorSelCnYn>
<suplDtyfrPrdClfCd>01</suplDtyfrPrdClfCd>

<prdImage01><![CDATA[http://img.polozum.co.kr/magicview/EssenceCream/3870167
2 72/600_600_Bimg1419713226.jpg]]></prdImage01>
  > <prdImage02><![CDATA[]]></prdImage02>
  <prdImage03><![CDATA[]]></prdImage03>
  <prdImage04><![CDATA[]]></prdImage04>
  <prdImage05><![CDATA[]]></prdImage05>
  <prdImage01Src></prdImage01Src>

<htmlDetail><![CDATA[IPHONE <strong>8</strong>]]></htmlDetail>
<advrtStmnt><![CDATA[-]]></advrtStmnt>
<orgnTypCd></orgnTypCd>
<orgnTypDtlsCd></orgnTypDtlsCd>
<orgnNmVal></orgnNmVal>
<orgnCountry></orgnCountry>
<sellerPrdCd><![CDATA[PROD3870167272]]></sellerPrdCd>
<bcktExYn>N</bcktExYn>

<reviewDispYn>Y</reviewDispYn>
<reviewOptDispYn>Y</reviewOptDispYn>
<selTermUseYn>N</selTermUseYn>
<selPrdClfCd>120:108</selPrdClfCd>
<aplBgnDy>06/26/2018</aplBgnDy>
<aplEndDy>06/26/2019</aplEndDy>
<setFpSelTermYn></setFpSelTermYn>
<selPrdClfFpCd></selPrdClfFpCd>
```

```
<wrhsPlnDy></wrhsPlnDy>
<selPrc><![CDATA[127000]]></selPrc>
<prdSelQty>100</prdSelQty>

<cuponcheck>N</cuponcheck>
<dscAmtPercnt>1</dscAmtPercnt>
<cupnDscMthdCd>01</cupnDscMthdCd>
<cupnUseLmtDyYn>N</cupnUseLmtDyYn>
<cupnIssEndDy>06/26/2019</cupnIssEndDy>

<mileageYN>N</mileageYN>
<mileageValue>1</mileageValue>
<mileageWyCd>01</mileageWyCd>
<mileageSpplLmtQty>1</mileageSpplLmtQty>
>

<pointYN>N</pointYN>
<pointValue>1</pointValue>
<spplWyCd>02</spplWyCd>
<pntSpplLmtQty>1</pntSpplLmtQty>

<optSelectYn>Y</optSelectYn>
<txtColCnt>1</txtColCnt>
<prdExposeClfCd>01</prdExposeClfCd>

<optionAllQty>100</optionAllQty>
<optionAllAddPrc><![CDATA[127000]]></optionAllAddPrc>
> <optionAllAddWght>0</optionAllAddWght>

<ProductCustOption>
  <colOptName></colOptName>
  <colOptUseYn>N</colOptUseYn>
</ProductCustOption>

<selMinLimitTypCd></selMinLimitTypCd>
<selMinLimitQty></selMinLimitQty>
<selLimitTypCd></selLimitTypCd>
<selLimitQty></selLimitQty>

<useGiftYn>N</useGiftYn>

<ProductGift>
  <giftInfo></giftInfo>
  <giftNm></giftNm>
  <aplBgnDt></aplBgnDt>
  <aplEndD></aplEndD>
</ProductGift>
<tmpltSeq>250999</tmpltSeq>
<asDetail><![CDATA[-]]></asDetail>
<rtngExchDetail><![CDATA[-]]></rtngExchDetail>
```

</Product>

Used Items

For used items, please add this lines in the XML input:

```
<prdStatCd>02</prdStatCd>
<selMthdCd>05</selMthdCd>
<!-- paidSelPrc: harga saat dibeli pertama kali -
- > <paidSelPrc>250000</paidSelPrc>
<!-- exteriorSpecialNote: kondisi barang saat ini -->
<exteriorSpecialNote>Baru digunakan 2 kali sejak pertama
kali di beli</exteriorSpecialNote>
```

Sample Code in PHP:

```
<?php
// Registrasi produk
$data = <<<_EOT_
<?xml version="1.0" encoding="euc-kr"
?> <Product>
// masukkan value-value xml produk anda sesuai dengan dokumentasi diatas
</Product>
_EOT_;
//_EOT_ tanpa spasi di
akhir $ch = curl_init();
$headers = array("Content-type: text/xml;charset=UTF-8",
"openapikey:gtty457h547rtyd456h78gy7456h758r5"); // contoh API Key, jangan
gunakan untuk testing, gunakan API key anda sendiri
curl_setopt($ch, CURLOPT_URL,
"http://dev.api.elevenia.co.id/rest/prodservices/product");
curl_setopt($ch, CURLOPT_HEADER, true); curl_setopt($ch,
CURLOPT_HTTPHEADER, $headers); curl_setopt($ch,
CURLOPT_RETURNTRANSFER, true); curl_setopt($ch,
CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
$return=curl_exec($ch);
echo "[MSG] Result -Xml
: \n"; echo $return;
?>
```

Output Description

Element name	Data Type	Description
productNo	String	Product ID
message	String	Result message

resultCode	String	Result code as follows: 200 = OK 500 = Error
------------	--------	--

Output Example:

1. Success:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
    <Message>Product uploaded successfully. Item No: 52844137</Message>
    <productNo>52844137</productNo>
    <resultCode>200</resultCode>
</ClientMessage>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
    <Message>Upload product Failed: Returns for exchange must
not forget the guidance.</ Message>
    <resultCode>500</ resultCode>
</ClientMessage>
```

Update Product

Method Type : POST

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/product>

Input Example

XML is the same format with **Insert Product API**, but use additional tag :

...

```
<prdNo>PRODUCT NO</prdNo>
```

...

Output Description

This endpoint will return product detail xml with the same format as **Upload Product** xml

View Product Details

Method Type : GET

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/product/details/{prdNo}>

Output Description

This endpoint will return product detail xml with the same format as **Upload Product** xml.

View Product Listing

Method Type : GET

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/product/listing?page={pageNumber}>

Output Description

This endpoint will return list of your product which has been uploaded into Elevenia.

```
<?xml version="1.0" encoding="UTF-8"
standalone="yes"?> <Products>
  <product>
    ... product information
  ... </product>
  <product>
    ... product information
  ... </product>
</Products>
```

Edit Product Options

Method Type : POST

Parameter : PathParam

Message Type : XML/String

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/updateProductOption/{prdNo}>

Input Description

Element name	Data Type	Mandatory	Description
prdNo	String	Y	Item ID / Product ID provided in the URL
optSelectYn	String	Y	Whether Product Number is required (Y is required)
txtColCnt	String	Y	Fixed Value 1
prdExposeClfCd	String	Y	Product Details Order. 00: Order by Number 01: Order by Name Alphabetical, Ascending (A-Z) 02: Order by Name Alphabetical, Descending (Z-A) 03: Order by Price, Desending (Low to High) 04: Order by Price, Desending (High to Low)
optionAllQty	String	Y	Batch Setting for amount of stock
optionAllAddPrc	String	Y	Batch Setting for price
optionAllAddWght	String	Y	Batch setting for additional weight
colTitle	String	Y	Option Name
colValue0	String	Y	The option value
colOptPrice	String	Y	Additional Price
colCount	String	Y	Number

useYn	String	Y	Status
optWght	String	Y	Product/Item Weight in kilograms

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodservices/UpdateProductOption/4796523>

2. XML Data:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <product>
  <optSelectYn >Y</optSelectYn>
  <txtColCnt >1</txtColCnt>
  <prdExposeClfCd >01</prdExposeClfCd>
  <colTitle >Color/Size</colTitle>
  <productOptions>
    <colOptPrice >0</colOptPrice>
    <colValue0 >Blue/XL</colValue0>
    <colCount >200</colCount>
    <optWght >40</optWght>
    <useYn>Y</useYn>
  </productOption>
  <productOptions>
    <colOptPrice >2000</colOptPrice>
    <colValue0 >Blue/XXL</colValue0>
    <colCount >300</colCount>
    <optWght >50</optWght>
    <useYn>Y</useYn>
  </productOption>
</product>
```

Output Description

If the process is succeed, all requested value will be returned as the response, else will return error message. See Example

Output Example

1. Success:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <product>
  <txtColCnt >1</txtColCnt>
  <prdExposeClfCd >01</prdExposeClfCd>
```

```
<colTitle >Color/Size/colTitle>
<productOptions>
  <colOptPrice >0</colOptPrice>
  <colValue0 >Blue/XL</colValue0>
  <colCount >200</colCount>
  <optWght >40</optWght>
  <useYn>Y</useYn>
</productOption>
<productOptions>
  <colOptPrice >2000</colOptPrice>
  <colValue0 >Blue/XXL</colValue0>
  <colCount >300</colCount>
  <optWght >50</optWght>
  <useYn>Y</useYn>
</productOption>
</product>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
  <message>Product Update
Failed</message> </ClientMessage>
```

Edit Item Price

Method Type : GET

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/product/price/{prdNo}/{selPrc}>

Input Description

Element name	Data Type	Mandatory	Description
prdNo	String	Y	Item ID / Product ID provided in URL.
selPrc	String	Y	Product Price (as a parameter in URL)

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodservices/product/price/4796523/10000>

2. XML Data:

No XML Data

Output Description

Element name	Data Type	Description
productNo	String	Product ID
message	String	Result message
resultCode	String	Result code as follows: 200 = OK 500 = Business Error -1000 = Server is under maintenance

Output Example

1. Success:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
    <productNo>4796523</productNo>
    <message>(Product Code: 4796523) prices have been
    updated.</message> <resultCode>200</resultCode>
</ClientMessage>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
    <productNo>4796523</productNo>
    <message>(Product Code: 4796523), failure to update
    the price</message> <resultCode>500</resultCode>
</ClientMessage>
```

Edit Item Discount

Method Type : POST

Parameter : XML

Message Type : XML/String

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodservices/product/priceCoupon/{prdNo}>

Input Description

Element name	Data Type	Mandatory	Description
prdNo	String	Y	Item ID / Product ID provided in the URL
selPrc	String	Y	Product Price
cuponcheck	String	Y	Whether these product can use a coupon or not
dscAmtPrcnt	String	Y	Discount value
cupnDscMthdCd	String	Y	Unit discount type 01 = Rp 02 = %
cupnUseLmtDyYn	String	Y	Promotion period (Y/N)
cupnIssEndDy	String	Y	End date of promotion period (MM/DD/YYYY)

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodservices/product/priceCoupon/4796523>

2. XML Data:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <Product>
  <selPrc>10000</selPrc>
  <cuponcheck>Y</cuponcheck>
  <dscAmtPrcnt>5</dscAmtPrcnt>
  <cupnDscMthdCd>02</cupnDscMthdCd>
```



```
<cupnUseLmtDyYn>Y</CupnUseLmtDyYn>
<cupnIssEndDy>04/30/2013</cupnIssEndDy>
</Product>
```

Output Description

Element name	Data Type	Description
productNo	String	Product ID
message	String	Result message
resultCode	String	Result code as follows: 200 = OK 500 = Business Error -1000 = Server is under maintenance

Output Example

1. Success:

```
<ClientMessage>
  <productNo>4796523</productNo>
  <message>Product prices and offers base immediately normally
have been modified. Product No.: 4796523</message>
  <resultCode>200</resultCode>
</ClientMessage>
```

2. Error:

```
<ClientMessage>
  <productNo>4796523</productNo>
  <message>(Product Code: 4796523), failure to update
the discount immediately</message>
  <resultCode>500</resultCode>
</ClientMessage>
```

Get Single ProductStockNo

Method Type : GET

Parameter : -

Message Type : -

Returned Value : XML

API Key: REQUIRED

Request URL :

<http://api.elevenia.co.id/rest/prodmarketservice/prodmarket/stck/{prdNo}>

Input Description

Element name	Data Type	Mandatory	Description
prdNo	Number	Y	Parameter Product ID provided in the URL

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodmarketservice/prodmarket/stck/1>

2. XML Data:

No XML Data

Output description

This endpoint will return an xml of full category listing with the following xml elements.

Element name	Data Type	Description
ProductStock	-	ProductStock(Node)
addPrc	Number	Option price
mixDtlOptNm	String	Detail product options
optWght	Number	Additional Weight
prdNo	Number	Product ID
prdStckNo	Number	Product Stock ID
prdStckStatCd	String	Stock Status (01: Use; 02: Sold Out)
selQty	Number	Product sale quantity
stckQty	Number	Product stock quantity
prdNm	String	Product Name
prdNo	Number	Product ID

sellerPrdCd	String	Seller's Product Code (SKU)
-------------	--------	-----------------------------

Output Example:

```
<?xml version="1.0" encoding="UTF-8"
standalone="yes"?> <ProductStocks>
  <ProductStock>
    <addPrc>0</addPrc>
    <mixDtlOptNm>Blue</mixDtlOptNm>
    <mixOptNo>1</mixOptNo>
    <optWght>0</optWght>
    <prdNo>17952</prdNo>
    <prdStckNo>1532948315</prdStckNo>
    <prdStckStatCd>01</prdStckStatCd>
    <selQty>0</selQty>
    <stckQty>20</stckQty>
  </ProductStock>
  <ProductStock>
    <addPrc>0</addPrc>
    <mixDtlOptNm>Red</mixDtlOptNm>
    <mixOptNo>1</mixOptNo>
    <optWght>0</optWght>
    <prdNo>17952</prdNo>
    <prdStckNo>1532948316</prdStckNo>
    <prdStckStatCd>01</prdStckStatCd>
    <selQty>0</selQty>
    <stckQty>10</stckQty>
  </ProductStock>
  <prdNm>[Test] Mug 4567</prdNm>
  <prdNo>17952</prdNo>
  <productComponents/>
  <sellerPrdCd>null</sellerPrdCd>
</ProductStocks>
```

Get Multiple ProductStockNo

Method Type :POST

Parameter : -

Message Type : -

Returned Value : XML

API Key: REQUIRED

Request URL :

<http://api.elevenia.co.id/rest/prodmarketservice/prodmarket/stocks>

Input Description

Element name	Data Type	Mandatory	Description
ProductStock	-	Y	Product Stock Node
prdNo	Number	Y	Product ID

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodmarketservice/prodmarket/stocks>

2. XML Data:

```
<?xml version="1.0" encoding="UTF-8"?>
<ProductStocks>
  <ProductStock>
    <prdNo>17951</prdNo>
  </ProductStock>
  <ProductStock>
    <prdNo>17952</prdNo>
  </ProductStock>
</ProductStocks>
```

Output description

This endpoint will return an xml of full category listing with the following xml elements.

Element name	Data Type	Description
ProductStock	-	ProductStock(Node)
addPrc	Number	Option price
mixDtlOptNm	String	Detail product options
optWght	Number	Additional Weight
prdNo	Number	Product ID
prdStckNo	Number	Product Stock ID
prdStckStatCd	String	Stock Status (01: Use; 02: Sold Out)

selQty	Number	Product sale quantity
stckQty	Number	Product stock quantity
prdNm	String	Product Name
prdNo	Number	Product ID
sellerPrdCd	String	Seller's Product Code (SKU)

Output Example:

```

<?xml version="1.0" encoding="UTF-8"
standalone="yes"?> <ProductStocks>
<?xml version="1.0" encoding="UTF-8"
standalone="yes"?> <ProductStockss>
  <ProductStocks>
    <ProductStock>
      <addPrc>0</addPrc>
      <mixDtlOptNm>Blue/XL</mixDtlOptNm>
      <mixOptNo>1</mixOptNo>
      <optWght>0</optWght>
      <prdNo>17951</prdNo>
      <prdStckNo>1532948313</prdStckNo>
      <prdStckStatCd>01</prdStckStatCd>
      <selQty>0</selQty>
      <stckQty>200</stckQty>
    </ProductStock>
    <ProductStock>
      <addPrc>0</addPrc>
      <mixDtlOptNm>Red/XL</mixDtlOptNm>
      <mixOptNo>1</mixOptNo>
      <optWght>0</optWght>
      <prdNo>17951</prdNo>
      <prdStckNo>1532948314</prdStckNo>
      <prdStckStatCd>01</prdStckStatCd>
      <selQty>0</selQty>
      <stckQty>100</stckQty>
    </ProductStock>
    <prdNm>[Test] Mug 4567</prdNm>
    <prdNo>17951</prdNo>
    <productComponents/>
    <sellerPrdCd>null</sellerPrdCd>
  </ProductStocks>
</ProductStocks>
  <ProductStock>
    <addPrc>0</addPrc>

```

```
<mixDtlOptNm>Blue</mixDtlOptNm>
<mixOptNo>1</mixOptNo>
<optWght>0</optWght>
<prdNo>17952</prdNo>
<prdStckNo>1532948315</prdStckNo>
<prdStckStatCd>01</prdStckStatCd>
<selQty>0</selQty>
<stckQty>20</stckQty>
</ProductStock>
<ProductStock>
  <addPrc>0</addPrc>
  <mixDtlOptNm>Red</mixDtlOptNm>
  <mixOptNo>1</mixOptNo>
  <optWght>0</optWght>
  <prdNo>17952</prdNo>
  <prdStckNo>1532948316</prdStckNo>
  <prdStckStatCd>01</prdStckStatCd>
  <selQty>0</selQty>
  <stckQty>10</stckQty>
</ProductStock>
<prdNm>[Test] Mug 4567</prdNm>
<prdNo>17952</prdNo>
<productComponents/>
<sellerPrdCd>null</sellerPrdCd>
</ProductStocks>
<message>Searched for 2.</message>
</ProductStockss>
```

Change quantity of product (Update Stock)

Method Type : PUT

Parameter : PathParam/XML

Message Type : XML/String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodmarketservices/stockqty/{prdStckNo}>

Input description

Element name	Data Type	Mandatory	Description
prdStckNo	String	Y	Stock ID, provided in the URL
prdNo	String	Y	Product/Item ID
prdStckNo	String	Y	Stock ID
stckQty	String	Y	Stock Quantity
optWght	String	Y	Product/Item Weight in kilograms

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodmarketservices/stockqty/434450294>

2. XML Data:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ProductStock>
  <prdNo>4796523</prdNo>
  <prdStckNo>434450294</prdStckNo>
  <stckQty>10</stckQty>
  <optWght>1</optWght>
</ProductStock>
```

Output Description

Element name	Data Type	Description
productNo	String	Product ID
message	String	Result message
resultCode	String	Result code as follows: 200 = OK 500 = Business Error -1000 = Server is under maintenance

Output Example

1. Success:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
  <productNo>4796523</productNo>
  <message>Options Stock No 434450294 (Product Code: 4796523), the
quantity/weight has been updated. Change Quantity/Weight 10/1</message>
  <resultCode>200</resultCode>
</ClientMessage>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
  <productNo>4796523</productNo>
  <message>Options Stock No 434450294 (Product Code: 4796523), the
update has failed</message>
  <resultCode>500</resultCode>
</ClientMessage>
```


Hide the product (Change the status of product to 105, 'Disembunyikan')

Method Type : PUT

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodstat-service/stat/stopdisplay/{prdNo}>

Input Description

Element name	Data Type	Mandatory	Description
prdNo	String	Y	Parameter Product ID provided in the URL.

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodstat-service/stat/stopdisplay/100>

2. XML Data:

No XML Data

Output description

Element name	Data Type	Description
Message	String	Result message
resultCode	String	Result code. 200 = OK 500 = Business Error -1000 = Server is under maintenance

Output Example:

1. Success:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
    <Message>Sales conditions have been corrected. [STAT:
    105]</Message> <resultCode>200</resultCode>
</ClientMessage>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
    <Message>Can not find the product information. Item No.:
    100</Message> <resultCode>500</resultCode>
</ClientMessage>
```

Re-active the hidden product (Change the status of product to 103, 'Aktif')

Method Type : PUT

Parameter : PathParam

Message Type : String(Number)

Returned Value : XML

API Key: REQUIRED

Request URL : <http://api.elevenia.co.id/rest/prodstat-service/stat/restartdisplay/{prdNo}>

Input description

Element name	Data Type	Mandatory	Description
prdNo	String	Y	Parameter Product ID provided in the URL.

Input Example

1. HTTP Request:

<http://api.elevenia.co.id/rest/prodstat-service/stat/restartdisplay/100>

2. XML Data:

No XML Data

Output Description

Element name	Data Type	Description
Message	String	Result message
resultCode	String	Result code. 200 = OK 500 = Business Error -1000 = Server is under maintenance

Output Example

1. Success:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
```

```
?> <ClientMessage>
```

```
<Message>Sales conditions have been corrected. [STAT:
103]</Message> <resultCode>200</resultCode>
</ClientMessage>
```

2. Error:

```
<?xml Version="1.0" encoding="UTF-8" standalone="Yes"
?> <ClientMessage>
  <Message>Can not find the product information. Item No.:
  100</Message> <resultCode>500</resultCode>
</ClientMessage>
```