

Open API Spec

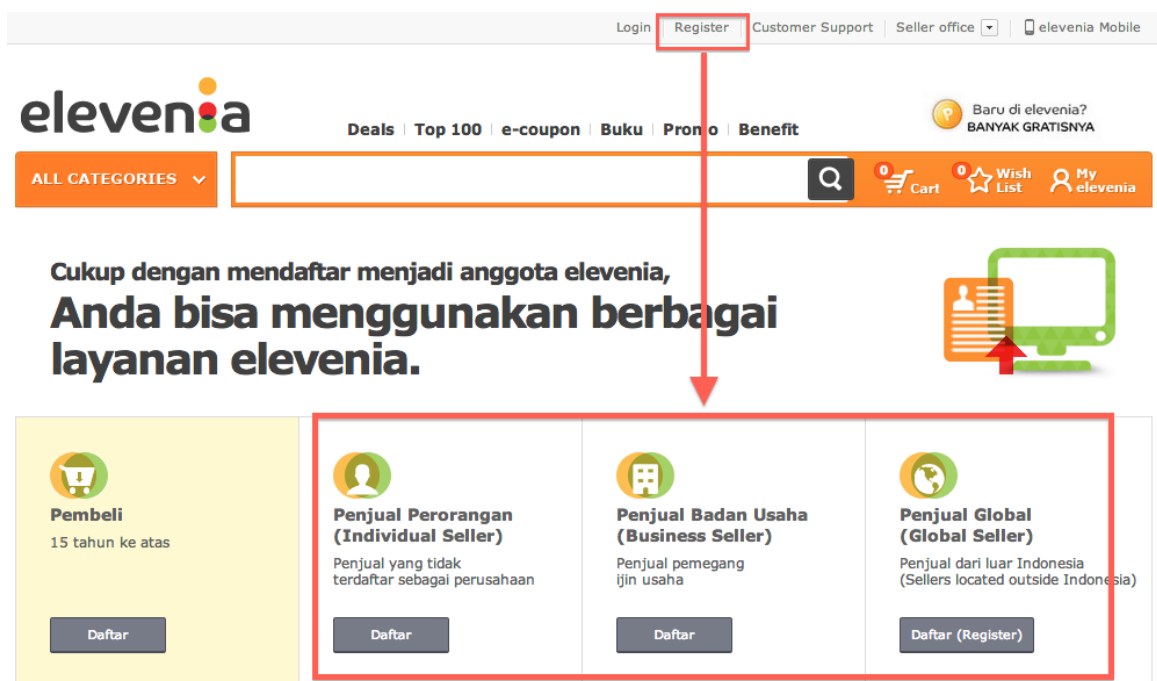
Version Number	Release Date
1.0	November 5th, 2013
1.1	November 11th, 2013
1.2	November 12th, 2013
1.3	November 13th, 2013
1.4	November 18th, 2013
1.5	November 19th, 2013
1.6	November 28th, 2013
1.7	December 17th, 2013
1.8	December 19th, 2013
1.9	December 27th, 2013
1.10	January 13th, 2014
1.11	March 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 18 th , 2016
1.18	August 15, 2017

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.




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 "Seller Information"/"Pengaturan Informasi Penjual" menu. The API Key is located at the bottom of page (Please refer to image below).

elevenia Seller Office Login



Pilih Bahasa Bahasa Indonesia

E-mail

Kata Sandi

[Daftar](#) [Lupa Kata Sandi](#)

LOGIN

elevenia SELLER OFFICE
abe@elevenia.co.id | [Pengaturan Toko Penjual](#) | [Sign Out](#)

Pengaturan produk	Pengaturan pesanan	Pengaturan Settlement	Pengaturan promosi	Informasi Anggota
Main Pengaturan informasi penjual				Pengaturan informasi penjual
<p>Refresh Buka di Tautan Baru</p> <p>* Alamat Pengembalian/Penukaran Jln. Bandung No E/1, Karangpawitan Karawang Kab. Karawang JAWA BARAT Atur</p> <p>* Cara Penerimaan Pembayaran Transfer bank Nama Bank PT. BA</p> <p>Nama Pemilik Rekening Abraham R.L. Tobing No. Rekening bank 1090705969</p> <p>Nama Cabang Karawang</p>				<p>Pengaturan Peningkat Penjual</p> <p>Pengaturan Saldo Tunai Penjual</p> <p>Pengaturan poin penjual</p> <p>Riwayat Voucher Penjual</p>
<p>Informasi Toko Penjual</p> <p>Nama Toko Penjual abexlp Atur Seller store Alamat Seller store http://www.elevenia.co.id/miniMall/MiniMallAction/getMiniMallHome.do?storeId=25963508</p>				
<p>Daftar layanan Open API</p> <p>Layanan API penjual Pemintaan layanan API Unduh API Spesifikasi Key yang digunakan 48ae1344ee58903e56</p>				
<p>Lanjut Batal</p>				

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.

Below is the user interface of registering delivery template

The screenshot shows the Elevenia Seller Office interface. The left sidebar contains a menu with options: Product Registration, Pengaturan informasi produk, Pengaturan informasi penjual, and Pengaturan template informasi produk (highlighted with a red circle). The main content area is titled 'Template Informasi' and includes a breadcrumb trail: Home > Pengaturan produk > Template Informasi. Below the title is a 'Petunjuk Pengguna' section with instructions. The main form area has fields for 'Jenis Template' (set to 'Profil Pengiriman'), 'Nama Template', 'Tanggal Registrasi', and 'Periode'. There are 'Q Cari' and 'Reset' buttons. Below the form is a 'Daftar Template' section with a sub-header 'Registrasi Profil Pengiriman' and a table listing existing templates.

	Edit	Nomor	Nama Template	Jenis Template	Tanggal Registrasi
<input type="checkbox"/>	Edit	232116	TIKI doang all service	Profil Pengiriman	2013/11/27
<input type="checkbox"/>	Edit	231810	TIKI & JNE Regular (Test)	Profil Pengiriman	2013/11/20
<input type="checkbox"/>	Edit	231611	test_template	Profil Pengiriman	2013/11/15
<input type="checkbox"/>	Edit	231437	Template02	Profil Pengiriman	2013/11/12
<input type="checkbox"/>	Edit	231333	Template01	Profil Pengiriman	2013/11/11

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.

Daftarkan template informasi pengiriman

Nama template	TIKI & JNE All service (22/40)	
Lokasi	Alamat Asal Pengiriman Produk	Silahkan pilih salah satu alamat disini, atau tambahkan alamat baru Ubah/Tambah
	Alamat Pengembalian/Penukaran Produk	Silahkan pilih salah satu alamat disini, atau tambahkan alamat baru Ubah/Tambah
Atur Biaya Pengiriman	Cara Pengiriman	<input checked="" type="radio"/> Kurir <input type="radio"/> Pengiriman Langsung Click!
	Pilih Biaya Pengiriman	<input checked="" type="radio"/> Pembayaran Lunas <input type="radio"/> Gratis
		<input type="checkbox"/> Pengiriman gratis bersyarat
	Perusahaan Pengiriman	<input checked="" type="checkbox"/> JNE Express <input checked="" type="checkbox"/> JNE Regular <input checked="" type="checkbox"/> TIKI Express <input checked="" type="checkbox"/> TIKI Regular
	Area Pengiriman yang Tidak Dilayani	<p>* Anda dapat memilih wilayah pengiriman yang tidak terjangkau dengan mengklik tombol "Pilih Kota"</p> <p>Sekarang total 0 buah kota yang terdaftar sebagai daerah yang tidak terjangkau pengiriman.</p> <p>Pilih kota</p>

daftar

Batal

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

Daftar Template			
Registrasi Profil Pengiriman			
	Edit	Nomor	Nama Template
<input type="checkbox"/>	Edit	232116	TIKI doang all service
<input type="checkbox"/>	Edit	231810	TIKI & JNE Regular (Test)
<input type="checkbox"/>	Edit	231611	test template
<input type="checkbox"/>	Edit	231437	Te
<input type="checkbox"/>	Edit	231333	Template

DELIVERY TEMPLATE
NUMBER

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

```
<tmplSeq>231810</tmplSeq>
```

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 Y/N	Max Length	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 Y/N	Max Length	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 Attribute Code
prdAttrNm	String	Category 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
dlvTmplNm	String	Delivery template name
dlvTmplSeq	String	Delivery template number

		This number will be used for registering product using xml tag: <code><tmplSeq>DLV_NUMBER</tmplSeq></code>
--	--	---

Output Example

1. Success:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DeliveryTemplates>
  <template>
    <dlvTplNm>TIKI & JNE (regular service)</dlvTplNm>
    <dlvTplSeq>231437</dlvTplSeq>
  </template>
  <template>
    <dlvTplNm>TIKI & JNE (express service)</dlvTplNm>
    <dlvTplSeq>231611</dlvTplSeq>
  </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 Y/N	Max Length	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
prdCrtfDtIsCd	Number	N		certification product type 01 : BPOM 02 : PIRT 03 : Manual
prdCrtf	String	N		Product certification number for certain category product
ProductCtgrAttribute	-	-	-	Category attribute (Parent Node)
prdAttrCd	Number	Y		Category attribute code
prdAttrNm	String	Y		Category attribute name i.e: <i>brand, model, type, ISBN (for books)</i>
prdAttrNo	String	Y		Category attribute ID
prdAttrVal	String	Y		Category attribute value i.e: <i>samsung, galaxy, s3, EAN-123123123123</i>
prdTypCd	Number	N	?	Service Type
prdNm	String	Y	?	Item Name
prdStatCd	String (Number)	Y	?	Item Conditions. '01': New '02': Used
useMon	Number	N	?	Period of use (If the product is used item)

paidSelPrc	Number	N	?	Sales Price at purchased time (for used item only)
exteriorSpecialNote	String	N	?	The current items conditions e.g: scratches (for used items only)
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 prdImage02 prdImage03 prdImage04 prdImage05 prdImage01Src	String String String String String	Y N N N N N	?	Main(Representative) Image URL Additional Image 1 URL Additional Image 2 URL Additional Image 3 URL Additional Image 4 URL Byte code of Image
htmlDetail	String	Y	?	Item's Detailed Description (html format supported)
advrtStmt	String	N	?	Item's advertisement information. e.g: <i>"Hot item's of this week"</i>
orgnTypCd orgnTypDtlsCd orgnNmVal orgnCountry	String (Number) String - String	N - N N	?	Code of Country of Origin 코드 Regional Code of Country of Origin Name of Country of Origin Country of Origin

sellerPrdCd	String	N	?	Seller's Item/Product Code (SKU)
bcktExYn	String	N	?	Restriction of storing in Cart. If set to Y, the purchase of this product should be paid immediately, can't be combined with other items.
ProductCert certTypeCd certKey	- String (Number) String	- N N	?	Authentication Information(Node) Type of Authentication Authentication Number. (ex: BPPOM number)
reviewDispYn	String	N	?	Whether display the Item Comments/Review or not
reviewOptDispYn	String	N	?	Enable/Disable the Review/Comment in the item's page
selTermUseYn	String	Y	?	Whether to use sales period (Y/N)
selPrdClfCd	String	N	?	Sales Period Code. 3:101 = 3 days 5:102 = 5 days 7:103 = 7 days 15:104 = 15 days 30:105 = 30 days 60:106 = 60 days 90:107 = 90 days 120:108 = 120 days 0:100 = direct input, This should be set to 'Y' for pre-order item, and set the pre-order period with aplBgnDy and aplEndDy
aplBgnDy	String	N	?	Sales period (or Pre-Order) Start Date (MM/DD/YYYY)

aplEndDy	String	N	?	Sales period (or Pre-Order) End Date (MM/DD/YYYY)
setFpSelTermYn	String	N	?	Set Fixed Price Sales Period
selPrdClfFpCd	String	N	?	Sales Period Code. Set Fixed Price Sales Period – include this if selTermUseYn = 'Y'
wrhsPlnDy	String	N	?	Due date of Stock
selPrc	Number	Y	?	Product's price
prdSelQty	Number	Y	?	Set the product stocks
cuponcheck	String	N	?	Enable/Disable discount program
dscAmtPercnt	String (Number)	N	?	Discount value
cupnDscMthdCd	String (Number)	N	?	Discount unit code 01 = Rp (in rupiah) 02 = % (in percentage)
cupnUseLmtDyYn	String	N	?	Limit the discount program (in days)
cupnLssEndDy	String	N	?	Discount program's end date
mileageYN	String	N	?	Enable/Disable token giveaway
mileageValue	String	N	?	Token value to give to buyer after purchasing item
mileageWyCd	String	N	?	Token unit code 01 = Rp (in rupiah) 02 = % (in percentage)
mileageSpplLmtQty	String	N	?	Limit the token giveaway after reaching an amount of treshold (by number of purchase). Buyer won't get anymore token after a certain number of purchase

pointYN	String	N	?	Enable/Disable point giveaway
pointValue	String	N	?	Point value to give to buyer after purchasing item
spplWyCd	String	N	?	Point unit code 02 = Rp (in rupiah) 01 = % (in percentage)
pntSpplLmtQty	String	N	?	Limit the point giveaway after reaching an amount of treshold (by number of purchase). Buyer won't get anymore point after a certain number of purchase
optSelectYn	String	N	?	Whether it is multi-choice or not
txtColCnt	String (Number)	N	?	Fixed Data Value 1
prdExposeClfCd	String (Number)	N	?	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 (Number)	N	?	Batch setting for quantity
optionAllAddPrc	String (Number)	N	?	Batch setting for price
optionAllAddWght	String (Number)	N	?	Batch setting for additional weight
colTitle	String	N	?	Option Name
ProductOptions	-	-	?	Item Option (Node)

colValue0 colCount useYn	String Number String	N N N		Option Data Type Option of Amount of Stock Used or New product (Y/N)
ProductCustOption colOptName colOptUseYn	- String String	- N N	?	Writable Option by Buyer(Node) Writable Name Option by Buyer Whether use Option or not
ProductComponent addPrdGrpNm compPrdNm sellerAddPrdCd addCompPrc compPrdQty compPrdVatCd addUseYn addPrdWght	- String String String Number Number String (Number) String Number	- N N N N N N N N	?	Additional setting for Item (Node) Additional Item Name Additional Item Data Type Additional Item Number of Sellers Additional setting for Price Amount of Sales VAT Status Additional Weight setting
selMinLimitTypCd	String (Number)	N	?	Minimum purchase quantity code (see code reference)
selMinLimitQty	Number	N	?	Amount of minimum purchase quantity
selLimitTypCd	String (Number)	N	?	Maximum purchase amount code (see code reference)
selLimitQty	Number	N	?	Amount of maximum purchase quantity
useGiftYn	String	N	?	Enable/Disable free gift
ProductGift giftInfo giftNm aplBgnDt aplEndDt	- String String String String	- N N N N	?	Free Gift information(Node) Free Gift information Free Gift Name Start Date of Free Gift program End Date of Free Gift program
tmpltSeq	Number		?	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):

```
<?xml Version="1.0" encoding="UTF-8"?>
<Product>
  <selMthdCd>01</selMthdCd>
  <dispCtgrNo>53</dispCtgrNo>
  <ProductCtgrAttribute>
    <prdAttrCd>2000042</prdAttrCd>
    <prdAttrNm>Neck Style</prdAttrNm>
    <prdAttrNo>161075</prdAttrNo>
    <prdAttrVal>Neck style Value</prdAttrVal>
  </ProductCtgrAttribute>
  <ProductCtgrAttribute>
    <prdAttrCd>2000017</prdAttrCd>
    <prdAttrNm>Fabric</prdAttrNm>
    <prdAttrNo>161074</prdAttrNo>
    <prdAttrVal>Fabric Value</prdAttrVal>
  </ProductCtgrAttribute>
  <prdNm>Test Product - Do Not Buy This Items</prdNm>
  <advrtStmt>Hot Items Of The Month !!</advrtStmt>
  <orgnTypCd>03</orgnTypCd>
  <orgnNmVal>Other: China</orgnNmVal>
  <sellerPrdCd>SKU-00001</sellerPrdCd>
  <suplDtyfrPrdClfCd>01</suplDtyfrPrdClfCd>
  <prdStatCd>01</prdStatCd>
  <minorSelCnYn>Y</minorSelCnYn>
  <prdImage01>
    <![CDATA[http://soffice.11st.co.kr/img/layout/logo.gif]]>
  </prdImage01>
  <htmlDetail>
    <![CDATA[<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Suspendisse pulvinar vestibulum diam elementum tempus.</p>]]>
  </htmlDetail>
  <selTermUseYn>N</selTermUseYn>
  <selPrc>50000</selPrc>
```

```

<prdSelQty>150</prdSelQty>
<dlvCnAreaCd>01</dlvCnAreaCd>
<spplWyCd>01</spplWyCd>
<dlvCstInstBasiCd>03</dlvCstInstBasiCd>
<dlvCst1>2500</dlvCst1>
<PrdFrDlvBasiAmt>50000</PrdFrDlvBasiAmt>
<dlvCstPayTypCd>01</dlvCstPayTypCd>
<addrSeqOut>4</addrSeqOut>
<addrSeqIn>8</addrSeqIn>
<rtngdDlvCst>2500</rtngdDlvCst>
<exchDlvCst>6000</exchDlvCst>
<asDetail>Sorry. No after sales service for this product.</asDetail>
<rtngExchDetail>Exchange Returns Information 070-7400-3719</rtngExchDetail>
<company>Testing Manufacturer</company>
<brand>Testing Brand</brand>
<modelNm>Testing Model</modelNm>
<prcCmpExpYn>Y</prcCmpExpYn>
</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 Y/N	Max Length	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
productOption colValue0 colOptPrice colCount useYn optWght	- String String String String	Y Y Y Y	?	Item Option (Node) The option value Additional Price Number Status Additional Weight

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 Y/N	Max Length	Description
prdNo	String	Y	?	Item ID / Product ID provided in URL.
selPrc	String		?	Sales 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	Mandato ry Y/N	Max Length	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

dscAmtPercnt	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>
  <dscAmtPercnt>5</dscAmtPercnt>
  <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 Y/N	Max Length	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 addPrc mixDtlOptNm optWght prdNo prdStckNo prdStckStatCd selQty stckQty	- Number String Number Number Number String Number Number	ProductStock(Node) Option price Detail product options Additional Weight Product ID Product Stock ID Stock Status (01: Use; 02: Sold Out) Product sale quantity 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 Y/N	Max Length	Description
ProductStock prdNo	- Number	Y Y	?	Product Stock Node 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 addPrc mixDtlOptNm optWght prdNo prdStckNo prdStckStatCd selQty stckQty	- Number String Number Number Number String Number Number	ProductStock(Node) Option price Detail product options Additional Weight Product ID Product Stock ID Stock Status (01: Use; 02: Sold Out) Product sale quantity 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>
  </ProductStocks>
</ProductStockss>

```



```

    <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/prodservices/stockqty/{prdStckNo}>

Input description

Element name	Data Type	Mandatory Y/N	Max Length	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/prodservices/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 Y/N	Max Length	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/prodstatervice/stat/restartdisplay/{prdNo}>

Input description

Element name	Data Type	Mandatory Y/N	Max Length	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>
```