

وزارة التجارة
Ministry of Commerce



Ministry of Commerce

Products Availability & Prices Observation
Initiative
10/23/2025

v1.9

Contents

Change Record	3
Parameter value.....	3
1. Login	4
2. Get Region List	6
3. Get Unit List	10
4. Get Code Type List	13
5. Get City List	16
6. Get Product Code.....	19
7. Post Add Product Retail Stage	22
8. Post Update Product Retail Stage.....	25
9. Get Production Product Code	27
10. Post Add Product Production Stage.....	31
11. Post Update Product Production Stage	34
12. Get Export Product Code	36
13. Post Add Product Export Stage	40
14. Post Update Product Export Stage.....	43
15. Get Inventory Product Code	45
16. Post Add Product Inventory Stage	49
17. Post Update Product Inventory Stage.....	52
18. Get import Product Code.....	55
19. Post Add Product Import Stage.....	58
20. Post Update Product Import Stage	61
21. Supply-Chain Stages Definition	63
22. FAQ.....	65

Change Record

Date	Author	Version	Change Reference
22 Jan 2023	MC Team	1.0	Initial
24 Jan 2023	MC Team	1.1	Add Import, Export, Inventory, Production
13 April 2023	MC Team	1.2	Changes for Types and Description
22 May 2023	MC Team	1.3	Add the replacement for BaseURL
2023-06-10	MC Team	1.4	Add Clarify
2023-12-20	MC Team	1.5	Update Example for Get Products List
2024-07-10	MC Team	1.6	Add FAQ
2025-08-27	MC Team	1.7	Add a Definition for the Supply chain
2025-09-04	MC Team	1.8	Add FAQ
2025-10-23	MC Team	1.8	Add Transaction (Id) to Table

Parameter value

To test the APIs, you can change the {{BaseUrl}} in any following step with the following:

- 1- Testing environment: <https://vendorapistg.mc.gov.sa/api/v1.0/>
- 2- Production environment: <https://pqpoapi.mc.gov.sa/api/v1.0/>

1. Login

1.1. Service Attributes

Service Property	Description
Service Name	Login
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	none

1.2. Interface description

This API can be used to Post the Vendor Account in Price Monitoring based on the base parameters defined below.

1.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}VendorAccount/Login

1.4. Request Body

Property	Type	Required	Description
apiKey	String	Yes	Api key
apiSecret	string	Yes	Api Secret

1.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Content-Type	Yes	text/json	

1.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Token	String	

1.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

****Note:** The description of the HTTP codes can be found in the Appendix section of this document.

1.8. Request and Response

1.8.1 Request Sample

```
{
  "apiKey": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "apiSecret": "string"
}
```

1.8.2 Success Response Sample

```
{
  "confirm": true,
  "message": null,
  "modelStateErrors": [],
  "success": true,
  "value": {
    "token": "eyJhbGciOiJIUzI1NilsInR5cCl6IkpxVCJ9.eyJodHRwOi8vc2NoZW1hcy5taWNyb3NvZnQuY29tL3dzLzlwMDgvMDYvaWRlbnRpdHkvY2xhaW1zL3JvbGUIoJTkXBwbGlccilsImh0dHA6Ly9zY2hlbWFzLnhtbHNvYXAub3JnL3dzLzlwMDUvMDUvaWRlbnRpdHkvY2xhaW1zL25hbWVpZGVudGi maWVyljoiZWQ2ZDY0MmYtNTBIZC00ZTI4LWjkZTUtMDNINGM4OGY5Mzg3liwiaHR0cDovL3NjaGVtYXMueG1sc29hcC5vcmcvd3MvMjAwNS8wNS9pZGVudGi0eS9jbGFpbXMvbmFtZSI6InN1cHBsaWVylwianRpIjoiYzlkNDI5NTUtNGFkNS00YzFmLTlmOTYtNDM3YTdiNTA0NzA4liwiQXBpS2V5joiNGNhYmYwNWltYmY0NC00ODhhLWI2YzUtMDA5MDcxNzU4ZmY3liwibmJmljoxNjczMzY4NjMzLCJieHAIoJE2NzM4MDA2MzMslmzcyI6Imh0dHBzOi8vbG9jYWxob3N0OjQ0Mzc4liwiYXVkljoiaHR0cHM6Ly9sb2NhbGhvc3Q6NDQzNzgifQ.fnC6aan4nu3SgRnauAzFnfV_54ixmQI4vZAkudoALs",
    "refreshToken": "eyJhbGciOiJIUzI1NilsInR5cCl6IkpxVCJ9.eyJodHRwOi8vc2NoZW1hcy5taWNyb3NvZnQuY29tL3dzLzlwMDgvMDYvaWRlbnRpdHkvY2xhaW1zL3JvbGUIoJTkXBwbGlccilsImh0dHA6Ly9zY2hlbWFzLnhtbHNvYXAub3JnL3dzLzlwMDUvMDUvaWRlbnRpdHkvY2xhaW1zL25hbWVpZGVudGi maWVyljoiZWQ2ZDY0MmYtNTBIZC00ZTI4LWjkZTUtMDNiNGM4OGY5Mzg3liwiaHR0cDovL3NjaGVtYXMueG1sc29hcC5vcmcvd3MvMjAwNS8wNS9pZGVudGi0eS9jbGFpbXMvbmFtZSI6InN1cHBsaWVylwianRpIjoiYzlkNDI5NTUtNGFkNS00YzFmLTlmOTYtNDM3YTdiNTA0NzA4liwiQXBpS2V5joiNGNhYmYwNWltYmY0NC00ODhhLWI2YzUtMDA5MDcxNzU4ZmY3liwibmJmljoxNjczMzY4NjMzLCJleHAIoJE2ODExNDQ2MzMslmzcyI6Imh0dHBzOi8vbG9jYWxob3N0OjQ0Mzc4liwiYXVkljoiaHR0cHM6Ly9sb2NhbGhvc3Q6NDQzNzgifQ._3I1fz8plc98KyAEIh6S1miQZ4QRdD57p6fYDI3Tjm8"
  },
  "correlationId": null
}
```

1.8.3 Error Response Sample

```
{
  "confirm": false,
  "message": "string",
  "modelStateErrors": [],
  "success": false,
  "value": {
  },
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

2. Get Region List

2.1. Service Attributes

Service Property	Description
Service Name	GetRegionList
Version	1.0
Service Owner	PriceMonitoring



Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

2.2. Interface description

This API can be used to Get the Regions list from MC.

2.3. Service Endpoint (Base URL)

Environment	URL
Staging	{BaseUrl}/api/v1/Lookup/GetRegionList

2.4. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

2.5. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object (List)</i>		
RegionID	int	Unique Reference ID
NameAr	string	Arabic Name For Region API response occurred while Updating the product.
NameEn	string	English Name For Region API response occurred while Updating the product.

****Note:** Response Object properties will be available only if the **Response Code is '0'** in the Response Status object.

2.6. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

****Note:** The description of the HTTP codes can be found in the Appendix section of this document.

2.7. Request and Response



2.8.1 Request Sample

```
{  
  "name": "GetRegionList",  
  "request": {  
    "method": "GET",  
    "header": [],  
    "url": {  
      "raw": "{{BaseUrl}}Lookup/GetRegionList",  
      "host": [  
        "{{BaseUrl}}Lookup"  
      ],  
      "path": [  
        "GetRegionList"  
      ]  
    },  
    "response": []  
  }  
}
```

2.8.2 Success Response Sample

```
{  
  "nameEn": "",  
  "nameAr": "الحدود الشمالية",  
  "id": 1  
}
```

2.8.3 Error Response Sample

```
{"confirm": false,  
 "message": "string",  
 "modelStateErrors": [],  
 "success": false,  
 "value": {},  
 "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"}  
}
```

3. Get Unit List

3.1. Service Attributes

Service Property	Description
Service Name	GetUnitList
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer token

3.2. Interface description

This API can be used to Get the Unit in Price Monitoring based on the base parameters defined below.

3.3. Service Endpoint (Base URL)

Environment	URL
Staging	 {{BaseUrl}}Lookup/GetUnitList

3.4. Request Body

Property	Type	Required	Description
Id	Int	Yes	unit Id
NameEN	string	Yes	Unit Name (English)
NameAR	string	Yes	Unit Name (Arabic)

3.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
token	Yes	text/json	

3.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
UnitID	int	Unique Reference ID
NameAr	string	Arabic Name For Unit API response occurred while Updating the product.
NameEn	string	English Name For Unit API response occurred while Updating the product.

****Note:** Response Object properties will be available only if the **Response Code is '0'** in the Response Status object.

3.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

****Note:** The description of the HTTP codes can be found in the Appendix section of this document.

3.8. Request and Response

3.8.1 Request Sample

```
{  
  "name": "GetUnitList",  
  "request": {  
    "method": "GET",  
    "header": [],  
    "url": {  
      "raw": "{{BaseUrl}}Lookup/GetUnitList",  
      "host": [  
        "{{BaseUrl}}Lookup"  
      ],  
      "path": [  
        "GetUnitList"  
      ]  
    }  
  }  
}
```

3.8.2 Success Response Sample

```
{  
  "nameEn": "Milli",  
  "nameAr": "ملي",  
  "id": 79  
}
```

3.8.3 Error Response Sample

```
{ "confirm": false,  
  "message": "string",  
  "modelStateErrors": [],  
  "success": false,  
  "value": {},  
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"  
}
```

4. Get Code Type List

4.1. Service Attributes

Service Property	Description
Service Name	GetCodeTypeList
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

4.2. Interface description

This API can be used to Get the Code Type List in Price Monitoring based on the base parameters defined below.

4.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}Lookup/GetCodeTypeList

4.4. Request Body

Property	Type	Required	Description
Id	Int	Yes	Code Type Id
NameEN	string	Yes	CodeType (English)
NameAR	string	Yes	CodeType (Arabic)

4.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

4.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
CodeType ID	int	Unique Reference ID
NameAr	string	Arabic Name For Code Type API response occurred while Updating the product.
NameEn	string	English Name For Code Type API response occurred while Updating the product.

****Note:** Response Object properties will be available only if the **Response Code is '0'** in the Response Status object.

4.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

****Note:** The description of the HTTP codes can be found in the Appendix section of this document.



4.8. Request and Response

4.8.1 Request Sample

```
{  
  "name": "GetCodeTypeList",  
  "request": {  
    "method": "GET",  
    "header": [],  
    "url": {  
      "raw": "{{BaseUrl}}Lookup/GetCodeTypeList",  
      "host": [  
        "{{BaseUrl}}Lookup"  
      ],  
      "path": [  
        "GetCodeTypeList"  
      ]  
    },  
    "response": []  
  }  
}
```

4.8.2 Success Response Sample

```
{  
  "nameEn": "Facility code",  
  "nameAr": "كود المنشأة",  
  "id": 1  
}
```

4.8.3 Error Response Sample

```
{ "confirm": false,  
  "message": "string",  
  "modelStateErrors": [],  
  "success": false,  
  "value": {}  
,  
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"  
}
```

5. Get City List

5.1. Service Attributes

Service Property	Description
Service Name	GetCityList
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

5.2. Interface description

This API can be used to Get the City in Price Monitoring based on the base parameters defined below.

5.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}Lookup/GetCityList

5.4. Request Body

Property	Type	Required	Description
Id	Int	Yes	City Id
NameEN	string	Yes	City Name (English)
NameAR	string	Yes	City Name (Arabic)
RegionId	Int	Yes	Region Id

5.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

5.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
CityID	int	Unique Reference ID
NameAr	string	Arabic Name For City API response occurred while Updating the product.
NameEn	string	English Name For City API response occurred while Updating the product.
RegionId	int	Reference ID get from City API response occurred while Updating the product.

****Note:** Response Object properties will be available only if the **Response Code is '0'** in the Response Status object.

5.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

****Note:** The description of the HTTP codes can be found in the Appendix section of this document.

5.8. Request and Response

5.8.1 Request Sample



```
{  
  "name": "GetCityList",  
  "request": {  
    "method": "GET",  
    "header": [],  
    "url": {  
      "raw": "{{BaseUrl}}Lookup/GetCityList",  
      "host": [  
        "{{BaseUrl}}Lookup"  
      ],  
      "path": [  
        "GetCityList"  
      ]  
    }  
  }  
}
```

5.8.2 Success Response Sample

```
{  
  "regionId": 6,  
  "nameEn": "",  
  "nameAr": "الرياض",  
  "id": 28  
}
```

5.8.3 Error Response Sample

```
{ "confirm": false,  
  "message": "string",  
  "modelStateErrors": [],  
  "success": false,  
  "value": {},  
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"  
}
```

6. Get Product Code

6.1. Service Attributes

Service Property	Description
Service Name	GetProductCode
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

6.2. Interface description

This API can be used to Get the Vendor Product Code with units support in Price Monitoring based on the base parameters defined below.

6.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}VendorProduct/GetProductCode

6.4. Request Body

Property	Type	Required	Description

6.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

6.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	

Description	Success	
Media Type	text/json	
<i>Response Object</i>		
ProductId	int	Unique Reference ID
ProductNameEN	string	English Name For Product
ProductNameAR	string	Arabic Name For Product
ProductCodes		
productCode	String	Product Code
productCodeTypeId	Int	Product Code Type Reference Id
supportUnits	List	List of units that can be used with this product.

6.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

6.8. Request and Response

6.8.1 Request Sample

```
{
  "name": "GetProductCode",
  "request": {
    "method": "GET",
    "header": [],
    "url": {
      "raw": "{{BaseUrl}}VendorProduct/GetProductCode",
      "host": [
        "{{BaseUrl}}VendorProduct"
      ],
      "path": [
        "GetProductCode"
      ]
    },
    "response": []
  }
}
```

6.8.2 Success Response Sample

```
{
  "confirm": true,
  "message": "string",
```



```

"modelStateErrors": [ {
    "name": "string",
    "value": "string" }],
"success": true,
"value": {
    {
        "productId": "862e25be-1e2c-4585-9280-0059194076b2",
        "productNameAr": "كيلو دجاج محلي طازج فقيه",
        "productNameEn": "",
        "productCodes": [
            {
                "productCode": "ITFC03",
                "productCodeTypeId": 2
            }],
        "supportUnits": [
            {
                "unitId": 80,
                "nameAr": "لتر",
                "nameEn": "Litre"
            },
            {
                "unitId": 85,
                "nameAr": "حبة",
                "nameEn": "Piece"
            },
        ],
    }]
}], "correlationId": "string" }

```

6.8.3 Error Response Sample

```
{
    "confirm": false,
    "message": "string",
    "modelStateErrors": [],
    "success": false,
    "value": {
    },
    "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

7. Post Add Product Retail Stage

7.1. Service Attributes

Service Property	Description
Service Name	Post Add Product Retail Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

7.2. Interface description

This API will be used to add the product details to the ministry system for the first time only.

7.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}VendorProduct/AddProductRetailStage

7.4. Request Body

Property	Type	Required	Description
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
availableQuantity	int	Yes	Available quantity is for vendor's external use only and is meant for direct sale to end customers.
soldQuantity	int	Yes	Sold Quantity



itemWeight	int	Yes	Item weight
itemPrice	int	Yes	Item price
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

7.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

7.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	

7.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

7.8. Request and Response

7.8.1 Request Sample

```
{
  "productId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "productCodeTypId": 0,
  "regionId": 0,
  "cityId": 0,
  "unitId": 0,
  "productCode": "string",
  "availableQuantity": 0,
  "soldQuantity": 0,
  "itemWeight": 0,
  "itemPrice": 0,
  "addDate": "2023-01-18T14:22:20.246Z",
  "UpdateDate": "2023-01-18T14:22:20.246Z"
}
```



}

7.8.2 Success Response Sample

```
{  
    "confirm": false,  
    "message": null,  
    "modelStateErrors": [],  
    "success": true,  
    "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",  
    "correlationId": null  
}
```

7.8.3 Error Response Sample

```
{  
    "confirm": false,  
    "message": "city is incorrect",  
    "modelStateErrors": [],  
    "success": false,  
    "value": "",  
    "correlationId": "2abb80c9-800d-408e-906f-b8fe69e2a57e" }
```

8. Post Update Product Retail Stage

8.1. Service Attributes

Service Property	Description
Service Name	Post update Product Retail Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

8.2. Interface description

This API can be used to Post the Vendor Account in Price Monitoring based on the base parameters defined below.

8.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}VendorProduct/UpdateProductRetailStage

8.4. Request Body

Property	Type	Required	Description
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
id	GUID	Yes	Transaction Id, it was generated by Added process, it's using only with same (ProductId, RegionId, CityId)

availableQuantity	int	Yes	Available quantity is for vendor's external use only and is meant for direct sale to end customers.
soldQuantity	int	Yes	Sold Quantity
itemWeight	int	Yes	Item weight
itemPrice	int	Yes	Item price
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

8.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

8.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	Id

8.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

8.8. Request and Response

8.8.1 Request Sample

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "productId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "productCodeTypId": 0,
  "regionId": 0,
  "cityId": 0,
```



```
{
  "unitId": 0,
  "productCode": "string",
  "availableQuantity": 0,
  "soldQuantity": 0,
  "itemWeight": 0,
  "itemPrice": 0,
  "addDate": "2023-01-18T14:22:20.246Z",
  "UpdateDate": "2023-01-18T14:22:20.246Z",
}
```

8.8.2 Success Response Sample

```
{
  "confirm": false,
  "message": null,
  "modelStateErrors": [],
  "success": true,
  "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",
  "correlationId": null
}
```

8.8.3 Error Response Sample

```
{
  "confirm": false,
  "message": "string",
  "modelStateErrors": [],
  "success": false,
  "value": {
  },
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

9. Get Production Product Code

9.1. Service Attributes

Service Property	Description
Service Name	GetProductionProductCode
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS

Request Authentication	Bearer Token
------------------------	--------------

9.2. Interface description

This API can be used to Get the production Product Code in Price Monitoring based on the base parameters defined below.

9.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}ProductionPrice/GetProductionProductCode

9.4. Request Body

Property	Type	Required	Description
----------	------	----------	-------------

9.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

9.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
ProductId	int	Unique Reference ID
ProductNameEN	string	English Name For Product
ProductNameAR	string	Arabic Name For Product
<i>ProductCodes</i>		
productCode	String	Product Code
productCodeTypeId	Int	Product Code Type Reference Id
supportUnits	List	List of units that can be used with this product.



9.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

9.8. Request and Response

9.8.1 Request Sample

```
{
  "name": "GetProductionProductCode",
  "request": {
    "method": "GET",
    "header": [],
    "url": {
      "raw": "{{BaseUrl}}ProductionPrice/
GetProductionProductCode",
      "host": [
        "{{BaseUrl}} ProductionPrice "
      ],
      "path": [
        " GetProductionProductCode "
      ]
    },
    "response": []
  }
}
```

9.8.2 Success Response Sample

```
{
  "confirm": true,
  "message": "string",
  "modelStateErrors": [ {
    "name": "string",
    "value": "string" }],
  "success": true,
  "value": {
    "product": {
      "productId": "862e25be-1e2c-4585-9280-0059194076b2",
      "productNameAr": "كيلو 1 دجاج محلي طازج فقيه",
      "productNameEn": "",
      "productCodes": [
        {
          "productCode": "ITFC03",
          "productCodeTypId": 2
        }],
      "supportUnits": [
        {
          "unitId": 80,
          "nameAr": "لتر",
          "nameEn": "Litre"
        },
        {
          "unitId": 85,
          "nameAr": "حبة",
          "nameEn": "Piece"
        }
      ]
    }
  }
}
```



```
        "nameEn": "Piece"
    },
}
},
"1",
"correlationId": "string" }
```

9.8.3 Error Response Sample

```
{ "confirm": false,
  "message": "string",
  "modelStateErrors": [],
  "success": false,
  "value": {
  },
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

10. Post Add Product Production Stage

10.1. Service Attributes

Service Property	Description
Service Name	Post Add Product Production Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

10.2. Interface description

This API can be used to Post the Product Production Product price in Price Monitoring based on the base parameters defined below.

10.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}ProductionPrice/AddProductProductionStage

10.4. Request Body

Property	Type	Required	Description
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
availableQuantity	int	Yes	Available Quantity
soldQuantity	int	Yes	Sold Quantity
itemWeight	int	Yes	Item weight

itemPrice	int	Yes	Item price
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

10.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

10.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	

10.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

10.8. Request and Response

10.8.1 Request Sample

```
{
  "productId": "8478a853-0866-4db8-9723-00f3b05810b7",
  "productCodeTypeId": 2,
  "regionId": 6,
  "cityId": 28,
  "unitId": 83,
  "productCode": "ITFC03",
  "availableQuantity": 2500,
  "soldQuantity": 6000,
  "itemWeight": 1,
  "itemPrice": 19.5,
```



```
{
  "addDate": "2023-01-23",
  "updateDate": "2023-01-23"
}
```

10.8.2 Success Response Sample

```
{
  "confirm": false,
  "message": null,
  "modelStateErrors": [],
  "success": true,
  "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",
  "correlationId": null
}
```

10.8.3 Error Response Sample

```
{
  "confirm": false,
  "message": "city is incorrect",
  "modelStateErrors": [],
  "success": false,
  "value": "",
  "correlationId": "2abb80c9-800d-408e-906f-b8fe69e2a57e" }
```

11. Post Update Product Production Stage

11.1. Service Attributes

Service Property	Description
Service Name	Post update Product Production Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

11.2. Interface description

This API can be used to Post the Production product in Price Monitoring based on the base parameters defined below.

11.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}ProductionPrice/UpdateProductProductionStage

11.4. Request Body

Property	Type	Required	Description
id	String	Yes	
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3



productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
availableQuantity	int	Yes	Available Quantity
soldQuantity	int	Yes	Sold Quantity
itemWeight	int	Yes	Item weight
itemPrice	int	Yes	Item price
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

11.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

11.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	Id

11.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

11.8. Request and Response

11.8.1 Request Sample

```
{
  "id": "5eb92b9b-f93a-46eb-9d3b-6e77b56efbbd",
  "productId": "862e25be-1e2c-4585-9280-0059194076b2",
  "productCodeTypeId": 2,
  "regionId": 6,
  "cityId": 28,
  "unitId": 83,
  "productCode": "ITFC03",
```

```
{
  "availableQuantity": 2000,
  "soldQuantity": 6000,
  "itemWeight": 1,
  "itemPrice": 25,
  "addDate": "2023-01-23",
  "updateDate": "2023-01-23"
}
```

11.8.2 Success Response Sample

```
{
  "confirm": false,
  "message": null,
  "modelStateErrors": [],
  "success": true,
  "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",
  "correlationId": null
}
```

11.8.3 Error Response Sample

```
{
  "confirm": false,
  "message": "string",
  "modelStateErrors": [],
  "success": false,
  "value": {
  },
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

12. Get Export Product Code

12.1. Service Attributes

Service Property	Description
Service Name	GetExportProductCode
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS

Request Authentication	Bearer Token
------------------------	--------------

12.2. Interface description

This API can be used to Get the Export Product Code in Price Monitoring based on the base parameters defined below.

12.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}ExportProduct/GetExportProductCode

12.4. Request Body

Property	Type	Required	Description
----------	------	----------	-------------

12.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

12.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
ProductId	int	Unique Reference ID
ProductNameEN	string	English Name For Product
ProductNameAR	string	Arabic Name For Product
<i>ProductCodes</i>		
productCode	String	Product Code
productCodeTypeId	Int	Product Code Type Reference Id
supportUnits	List	List of units that can be used with this product.

12.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

12.8. Request and Response

12.8.1 Request Sample

```
{
  "name": "GetExportProductCode",
  "request": {
    "method": "GET",
    "header": [],
    "url": {
      "raw": "{{BaseUrl}}ExportProduct/GetExportProductCode",
      "host": [
        "{{BaseUrl}} ExportProduct "
      ],
      "path": [
        "GetExportProductCode"
      ]
    },
    "response": []
  }
}
```

12.8.2 Success Response Sample

```
{
  "confirm": true,
  "message": "string",
  "modelStateErrors": [ {
    "name": "string",
    "value": "string" }],
  "success": true,
  "value": {
    {
      "productId": "862e25be-1e2c-4585-9280-0059194076b2",
      "productNameAr": "كيلو دجاج محلي طازج فقيه",
      "productNameEn": "",
      "productCodes": [
        {
          "productCode": "ITFC03",
          "productCodeTypeId": 2
        }],
      "supportUnits": [
        {
          "unitId": 80,
          "nameAr": "لتر",
          "nameEn": "Litre"
        },
        {
          "nameAr": "kg",
          "nameEn": "Kg"
        }
      ]
    }
  }
}
```



```
"unitId": 85,  
"nameAr": "جبة",  
"nameEn": "Piece"  
},  
}  
}],  
"correlationId": "string" }
```

12.8.3 Error Response Sample

```
{"confirm": false,  
"message": "string",  
"modelStateErrors": [],  
"success": false,  
"value": {},  
"correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"}
```

13. Post Add Product Export Stage

13.1. Service Attributes

Service Property	Description
Service Name	Post Add Product Export Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

13.2. Interface description

This API can be used to Post the Product Export price in Price Monitoring based on the base parameters defined below.

13.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}ExportProduct/AddProductExportStage

13.4. Request Body

Property	Type	Required	Description
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
exitPort	String	Yes	The exit port
exportedQuantity	Number	Yes	Exported Quantity



exportedPrice	Number	Yes	Exported price
exportedUnit	String	Yes	Exported unit
exportedCountry	String	Yes	Exported country
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

13.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

13.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	

13.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

13.8. Request and Response

13.8.1 Request Sample

```
{
  "productId": "abc22647-6185-4104-b846-0118b2964eb2",
  "regionId": 6,
  "cityId": 28,
  "unitId": 83,
  "productCodeTypeId": 2,
  "productCode": "ITFC03",
  "exitPort": "Riyadh Exit",
  "exportedQuantity": 2500,
  "exportedPrice": 16.5,
  "exportedUnit": "Kilo",
```



```
{
  "exportedCountry": "USA",
  "addDate": "2023-01-23",
  "updateDate": "2023-01-23"
}
```

13.8.2 Success Response Sample

```
{
  "confirm": false,
  "message": null,
  "modelStateErrors": [],
  "success": true,
  "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",
  "correlationId": null
}
```

13.8.3 Error Response Sample

```
{
  "confirm": false,
  "message": "city is incorrect",
  "modelStateErrors": [],
  "success": false,
  "value": "",
  "correlationId": "2abb80c9-800d-408e-906f-b8fe69e2a57e" }
```

14. Post Update Product Export Stage

14.1. Service Attributes

Service Property	Description
Service Name	Post update Product Export Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

14.2. Interface description

This API can be used to Post the Export product in Price Monitoring based on the base parameters defined below.

14.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}ExportProduct/UpdateProductExportStage

14.4. Request Body

Property	Type	Required	Description
id	String	Yes	The reference number from the add step
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
exitPort	String	Yes	The exit port

exportedQuantity	Number	Yes	Exported Quantity
exportedPrice	Number	Yes	Exported price
exportedUnit	String	Yes	Exported unit
exportedCountry	String	Yes	Exported country
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

14.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

14.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	Id

14.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

14.8. Request and Response

14.8.1 Request Sample

```
{
  "id": "71db0977-6c85-4bef-8b90-fd456454c194",
  "productId": "862e25be-1e2c-4585-9280-0059194076b2",
  "regionId": 6,
  "cityId": 28,
  "unitId": 83,
  "productCodeTypId": 2,
```



```
{
    "productCode": "ITFC03",
    "exitPort": "Riyadh Exit",
    "exportedQuantity": 2500,
    "exportedPrice": 16.5,
    "exportedUnit": "Kilo",
    "exportedCountry": "USA",
    "addDate": "2023-01-23",
    "updateDate": "2023-01-23"
}
```

14.8.2 Success Response Sample

```
{
    "confirm": false,
    "message": null,
    "modelStateErrors": [],
    "success": true,
    "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",
    "correlationId": null
}
```

14.8.3 Error Response Sample

```
{
    "confirm": false,
    "message": "string",
    "modelStateErrors": [],
    "success": false,
    "value": {
    },
    "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

15. Get Inventory Product Code

15.1. Service Attributes

Service Property	Description
Service Name	GetInventoryProductCode
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

15.2. Interface description

This API can be used to Get the Inventory Product Code in Price Monitoring based on the base parameters defined below.

15.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}\InventoryProduct\GetInventoryProductCode

15.4. Request Body

Property	Type	Required	Description
----------	------	----------	-------------

15.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

15.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
ProductId	int	Unique Reference ID
ProductNameEN	string	English Name For Product
ProductNameAR	string	Arabic Name For Product
<i>ProductCodes</i>		
productCode	String	Product Code
productCodeTypeId	Int	Product Code Type Reference Id
supportUnits	List	List of units that can be used with this product.



15.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

15.8. Request and Response

15.8.1 Request Sample

```
{
  "name": "GetInventoryProductCode",
  "request": {
    "method": "GET",
    "header": [],
    "url": {
      "raw": "{{BaseUrl}}InventoryProduct/GetInventoryProductCode",
      "host": [
        "{{BaseUrl}} InventoryProduct "
      ],
      "path": [
        "GetInventoryProductCode"
      ],
      "response": []
    }
  }
}
```

15.8.2 Success Response Sample

```
{
  "confirm": true,
  "message": "string",
  "modelStateErrors": [ {
    "name": "string",
    "value": "string" }],
  "success": true,
  "value": {
    {
      "productId": "862e25be-1e2c-4585-9280-0059194076b2",
      "productNameAr": "كيلو دجاج محلي طازج فقيه",
      "productNameEn": "",
      "productCodes": [
        {
          "productCode": "ITFC03",
          "productCodeTypeId": 2
        }],
      "supportUnits": [
        {
          "unitId": 80,
          "nameAr": "لتر",
          "nameEn": "Litre"
        },
        {
          "unitId": 85,
          "nameAr": "حبة",
          "nameEn": "Piece"
        }
      ]
    }
  }
}
```



```
        },
    ],
},
"correlationId": "string" }
```

15.8.3 Error Response Sample

```
{"confirm": false,
"message": "string",
"ModelStateErrors": [],
"success": false,
"value": {
},
"correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

16. Post Add Product Inventory Stage

16.1. Service Attributes

Service Property	Description
Service Name	Post Add Product Inventory Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

16.2. Interface description

This API can be used to Post the Product Inventory price in Price Monitoring based on the base parameters defined below.

16.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}InventoryProduct/AddProductInventoryStage

16.4. Request Body

Property	Type	Required	Description
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
availableQuantity	Number	Yes	Available quantity is for vendor's internal use only and not meant for direct sale to end customers.
totalQuantity	Number	Yes	Total Quantity
itemWeight	Number	Yes	Item weight

expectedQuantity	Number	Yes	Expected quantity
expectedDate	Datetime	Yes	Expected date of delivery
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

16.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

16.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	

16.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

13.6. Request and Response

16.8.1 Request Sample

```
{
  "productId": "ab775585-3b61-4880-984d-04c340550401",
  "regionId": 6,
  "cityId": 28,
  "unitId": 83,
  "productCodeTypId": 2,
  "productCode": "ITFC03",
  "availableQuantity": 5000,
  "totalQuantity": 15000,
  "itemWeight": 1,
```



```
"expectedQuantity": 8000,  
"expectedDate": "2023-05-24",  
"addDate": "2023-01-24",  
"updateDate": null  
}
```

16.8.2 Success Response Sample

```
{  
    "confirm": false,  
    "message": null,  
    "modelStateErrors": [],  
    "success": true,  
    "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",  
    "correlationId": null  
}
```

16.8.3 Error Response Sample

```
{  
    "confirm": false,  
    "message": "city is incorrect",  
    "modelStateErrors": [],  
    "success": false,  
    "value": "",  
    "correlationId": "2abb80c9-800d-408e-906f-b8fe69e2a57e" }
```

17. Post Update Product Inventory Stage

17.1. Service Attributes

Service Property	Description
Service Name	Post update Product Inventory Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

17.2. Interface description

This API can be used to Post the Inventory product in Price Monitoring based on the base parameters defined below.

17.3. Service Endpoint (Base URL)

Environment	URL
Staging	 {{BaseUrl}}/InventoryProduct/UpdateProductInventoryStage

17.4. Request Body

Property	Type	Required	Description
id	String	Yes	The reference number from the add step
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3



productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
availableQuantity	Number	Yes	Available quantity is for vendor's internal use only and not meant for direct sale to end customers.
totalQuantity	Number	Yes	Total Quantity
itemWeight	Number	Yes	Item weight
expectedQuantity	Number	Yes	Expected quantity
expectedDate	Datetime	Yes	Expected date of delivery
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

17.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

17.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	Id

17.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

17.8. Request and Response

17.8.1 Request Sample

```
{
  "id": "f3029623-d1fc-43fb-bcf3-af2e0101622e",
  "productId": "862e25be-1e2c-4585-9280-0059194076b2",
  "regionId": 6,
  "cityId": 28,
```



```
{"unitId": 83,  
"productCodeTypeId": 2,  
"productCode": "ITFC03",  
"availableQuantity": 5000,  
"totalQuantity": 15000,  
"itemWeight": 1,  
"expectedQuantity": 8000,  
"expectedDate": "2023-05-24",  
"addDate": "2023-01-24",  
"updateDate": null  
}
```

17.8.2 Success Response Sample

```
{  
    "confirm": false,  
    "message": null,  
    "modelStateErrors": [],  
    "success": true,  
    "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",  
    "correlationId": null  
}
```

17.8.3 Error Response Sample

```
{ "confirm": false,  
  "message": "string",  
  "modelStateErrors": [],  
  "success": false,  
  "value": {  
  },  
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"  
}
```

18. Get import Product Code

18.1. Service Attributes

Service Property	Description
Service Name	GetimportProductCode
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Get
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

18.2. Interface description

This API can be used to Get the import Product Code in Price Monitoring based on the base parameters defined below.

18.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}importProduct/GetimportProductCode

18.4. Request Body

Property	Type	Required	Description

18.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

18.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	

Description	Success	
Media Type	text/json	
<i>Response Object</i>		
ProductId	int	Unique Reference ID
ProductNameEN	string	English Name For Product
ProductNameAR	string	Arabic Name For Product
ProductCodes		
productCode	String	Product Code
productCodeTypeId	Int	Product Code Type Reference Id
supportUnits	List	List of units that can be used with this product.

18.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

18.8. Request and Response

18.8.1 Request Sample

```
{
  "name": "GetimportProductCode",
  "request": {
    "method": "GET",
    "header": [],
    "url": {
      "raw": "{{BaseUrl}}importProduct/GetimportProductCode",
      "host": [
        "{{BaseUrl}} importProduct "
      ],
      "path": [
        "GetimportProductCode "
      ]
    },
    "response": []
  }
}
```

18.8.2 Success Response Sample

```
{
```



```

"confirm": true,
"message": "string",
"modelStateErrors": [ {
    "name": "string",
    "value": "string" }],
"success": true,
"value": {
    {
        "productId": "862e25be-1e2c-4585-9280-0059194076b2",
        "productNameAr": "كيلو ١ دجاج محلي طازج فقيه",
        "productNameEn": "",
        "productCodes": [
            {
                "productCode": "ITFC03",
                "productCodeTypeId": 2
            }],
        "supportUnits": [
            {
                "unitId": 80,
                "nameAr": "لتر",
                "nameEn": "Litre"
            },
            {
                "unitId": 85,
                "nameAr": "حبة",
                "nameEn": "Piece"
            }
        ]
    }],
    "correlationId": "string" }
}

```

18.8.3 Error Response Sample

```
{
    "confirm": false,
    "message": "string",
    "modelStateErrors": [],
    "success": false,
    "value": {
    },
    "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"
}
```

19. Post Add Product Import Stage

19.1. Service Attributes

Service Property	Description
Service Name	Post Add Product import Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

19.2. Interface description

This API can be used to Post the Product import price in Price Monitoring based on the base parameters defined below.

19.3. Service Endpoint (Base URL)

Environment	URL
Staging	{{BaseUrl}}importProduct/AddProductImportStage

19.4. Request Body

Property	Type	Required	Description
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
exitPort	String	Yes	The exit port
importedQuantity	Number	Yes	imported Quantity



importedPrice	Number	Yes	imported price
importedUnit	String	Yes	imported unit
importedCountry	String	Yes	imported country
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

19.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

19.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	

19.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

19.8. Request and Response

19.8.1 Request Sample

```
{
  "productId": "abc22647-6185-4104-b846-0118b2964eb2",
  "regionId": 6,
  "cityId": 28,
  "unitId": 83,
  "productCodeTypeId": 2,
  "productCode": "ITFC03",
  "exitPort": "Riyadh Exit",
  "importedQuantity": 2500,
  "importedPrice": 16.5,
  "importedUnit": "Kilo",
```



```
{
  "importedCountry": "USA",
  "addDate": "2023-01-23",
  "updateDate": "2023-01-23"
}
```

19.8.2 Success Response Sample

```
{
  "confirm": false,
  "message": null,
  "modelStateErrors": [],
  "success": true,
  "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",
  "correlationId": null
}
```

19.8.3 Error Response Sample

```
{
  "confirm": false,
  "message": "city is incorrect",
  "modelStateErrors": [],
  "success": false,
  "value": "",
  "correlationId": "2abb80c9-800d-408e-906f-b8fe69e2a57e" }
```

20. Post Update Product Import Stage

20.1. Service Attributes

Service Property	Description
Service Name	Post update Product import Stage
Version	1.0
Service Owner	PriceMonitoring
Architectural Style	Restful
Supported Restful Verbs	Post
Support Message Exchange Patterns	Request-Response (Synchronous)
Supported Transport Protocol	HTTPS
Request Authentication	Bearer Token

20.2. Interface description

This API can be used to Post the import product in Price Monitoring based on the base parameters defined below.

20.3. Service Endpoint (Base URL)

Environment	URL
Staging	 {{BaseUrl}}importProduct/UpdateProductimportStage

20.4. Request Body

Property	Type	Required	Description
id	String	Yes	The reference number from the add step
productid	GUID	Yes	Product id
productCodeTypeId	int	Yes	Product CodeType Id that was received from Step 4
regionId	int	Yes	Region id received from Step 2
cityId	int	Yes	City Id received from Step 5
unitId	int	Yes	Unit id received from Step 3
productCode	string	Yes	The international barcode from the company system and if not found add the HS Code or Facility Code
exitPort	String	Yes	The exit port

importedQuantity	Number	Yes	imported Quantity
importedPrice	Number	Yes	imported price
importedUnit	String	Yes	imported unit
importedCountry	String	Yes	imported country
addDate	datetime	Yes	Add date {yyyy-MM-dd HH:mm}
updateDate	datetime	Yes	Update date {yyyy-MM-dd HH:mm}

20.5. HTTP Headers

Property Name	Required?	Allowed Values	Description
Authorization	Yes	string	

20.6. Response Body

Property	Type	Description
<i>Response Status</i>		
Code	200	
Description	Success	
Media Type	text/json	
<i>Response Object</i>		
Id	GUID	Id

20.7. Response Codes

Category	HTTP Code	Error Code	Error Description
Success	200	0	Success
Client Error	400	1	Client Error. Please verify the request and try again with appropriate request parameters.
Server Error	500	2	Internal server error or technical error.

20.8. Request and Response

20.8.1 Request Sample

```
{
  "id": "71db0977-6c85-4bef-8b90-fd456454c194",
  "productId": "862e25be-1e2c-4585-9280-0059194076b2",
  "regionId": 6,
  "cityId": 28,
```



```
"unitId": 83,  
"productCodeTypeId": 2,  
"productCode": "ITFC03",  
"enterPort": "Riyadh Exit",  
"importedQuantity": 2500,  
"importedPrice": 16.5,  
"importedUnit": "Kilo",  
"importedCountry": "USA",  
"addDate": "2023-01-23",  
"updateDate": "2023-01-23"  
}
```

20.8.2 Success Response Sample

```
{  
    "confirm": false,  
    "message": null,  
    "modelStateErrors": [],  
    "success": true,  
    "value": "2abb80c9-800d-408e-906f-b8fe69e2a57e",  
    "correlationId": null  
}
```

20.8.3 Error Response Sample

```
{ "confirm": false,  
  "message": "string",  
  "modelStateErrors": [],  
  "success": false,  
  "value": {  
  },  
  "correlationId": "2fabbf64-5717-4562-b3fc-2c963f66afcc"  
}
```

21. Supply-Chain Stages Definition

1. Importation

- This stage involves the receiving of raw materials, components, or finished goods from international suppliers into the country and then to the local market.
NOTE: Frequency of sharing the data: On-Action

2. Production

- This stage involves transforming inputs and raw materials into finished goods through manufacturing process or assembly.
NOTE: Frequency of sharing the data: (Daily - On-Action)
-

3. Inventory

- This stage involves the management of raw materials, work-in-progress, and storage of finished goods.
- Available quantity at this stage is for vendor's **internal use only and not meant for direct sales to end customers.**

NOTE: Frequency of sharing the data: Daily

4. Retail

- This stage involves making products available to end customers through various sales channels like physical stores or e-commerce. This may include business models like B2B, B2C, and D2C.
- Available quantity at this stage is for vendor's **external use only and is meant for direct sale to end customers.**

NOTE: Frequency of sharing the data: Daily

5. Exportation

- This stage involves the process of shipping finished goods from the home country to international customers, distributors, or warehouses.

NOTE: Frequency of sharing the data: On-Action



22. FAQ

- **How do I add new products to MC Service?** Currently, products are added manually. Please send an Excel file of your products within our tracking category.
- **Do we need to integrate with all supply chains (import, inventory, production, retail, export)?** You only need to integrate with the supply chains you actively work with.
- **What is the purpose of "ProductCode" and how does it relate to "ProductCodeTypeId"?** "ProductCode" is a string value, and "ProductCodeTypeId" specifies the type of code it represents. We recommend using the international barcode.
- **When should we sync data with MC?** The sync frequency depends on the supply chain. For retail and inventory, we recommend a daily sync at a convenient time. For other types, data should be synced when an action occurs, such as when you import something.
- **What should we do if there is a problem with the integration?** Our system provides clear responses for both successful and failed requests. If you receive a failed message or a bad request, you should try sending the request again until you receive a successful response. If the integration is down for an extended period, we recommend adding the requests to a queue and retrying them later in ascending order by date.
- **Do I need to integrate all of my products?** Yes. If you have a product that is not currently in the MC list, please let us know so we can add it.
- **What does "ItemWeight" indicate?** "ItemWeight" should be indicated in kilograms (kg) for products with weight. For other products, it can be used for liters.
- **What is the difference between "AddDate" and "UpdateDate"?** "AddDate" refers to the initial date a product was sent, while "UpdateDate" refers to the date of the most recent update.
- **Does "ItemPrice" include VAT?** Yes, "ItemPrice" includes VAT.

API & Environments

- **When should I use the "Add...x" and "Update...x" API endpoints?** The "Add" service should be used only once to add a new item. After a successful call, you will receive a new ID that should be used for all subsequent updates via the "Update" service.

- **How many environments does MC support, and what are their differences?** MC supports two environments:
 - **Stage:** This is a testing environment where you are free to experiment with different values.
 - **Prod:** This is the live environment for real-world data. You should only use this after you have thoroughly tested your integration and are ready to go live.
- **Is there a way to retrieve uploaded data?** Yes. An individual who has been assigned the "inventory delegates service" from the Ministry of Commerce at <https://business.sa> can log in to <https://pqpo.mc.gov.sa> to retrieve the uploaded data.
- **Can I still manually upload stock data if needed?**

Yes, a manual approach is available, and vendors can submit stock levels manually. However, API integration is the **preferred and primary method** to ensure real-time accuracy.
- **Is my data secure?**

Yes. All data transfers through the API are encrypted, and only authorized vendors with valid API credentials can send inventory updates.
- **What happens if my submission is rejected?**

Your data will not be processed. You'll receive information about why it was rejected, and you may reapply by correcting and resubmitting your details.