

[/v3/api-docs](#)[Explore](#)

productoservice



1.0 OAS 3.0

[/v3/api-docs](#)

Microservicio de Producto

[Génesis Montero - Website](#)[Send email to Génesis Montero](#)

Servers

[http://localhost:8082 - Generated server url](#)

Productos

Operaciones CRUD para el microservicio de productos

**GET****/api/productos** Obtener todos los productos de la base de datos

Devuelve todos los productos almacenados en la base de datos

Parameters

[Cancel](#)

No parameters

[Execute](#)[Clear](#)

Responses

Curl

```
curl -X 'GET' \  
  'http://localhost:8082/api/productos' \  
  -H 'accept: */*'
```

Request URL

http://localhost:8082/api/productos

Server response

Code	Details
200	<div><div>Response body</div><pre>[{ "productoId": 1, "nombre": "Armando Armani", "precio": 1000, "stock": 5 }, { "productoId": 2, "nombre": "Cloud 5", "precio": 2000, "stock": 2 }, { "productoId": 3, "nombre": "Perfume buenisimo", "precio": 3000, "stock": 4 }]</pre><div>Download</div></div> <div><div>Response headers</div><pre>connection: keep-alive content-type: application/json date: Tue,08 Jul 2025 05:18:59 GMT keep-alive: timeout=60 transfer-encoding: chunked</pre></div>

Responses

Code	Description	Links
200	<div>consulta exitosa</div> <div><div>Media type</div><div><div>*/*</div></div><div>Controls Accept header.</div></div> <div><div>Example Value</div><div>Schema</div><pre>[{ "productoId": 1, "nombre": "Cloud de Ariana Grande", "precio": 30000, "stock": 4 }]</pre></div>	No links

POST /api/productos Crear un nuevo producto

Agrega un nuevo producto a la base de datos

Parameters**Cancel****Reset**

No parameters

Request body required

application/json

```
{
  "nombre": "Super perfume",
  "precio": 90000,
  "stock": 7
}
```

Execute**Clear****Responses****Curl**

```
curl -X 'POST' \
  'http://localhost:8082/api/productos' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "nombre": "Super perfume",
    "precio": 90000,
    "stock": 7
  }'
```

Request URL

http://localhost:8082/api/productos

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "productoId": 4, "nombre": "Super perfume", "precio": 90000, "stock": 7}</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>connection: keep-alive content-type: application/json date: Tue,08 Jul 2025 05:15:35 GMT keep-alive: timeout=60 transfer-encoding: chunked</pre></div></div>

Responses

Code	Description	Links
200	<div>Creacion exitosa</div> <div><div>Media type</div><div><div>*/*</div></div><div>Controls Accept header.</div></div> <div><div>Example Value</div><div>Schema</div><div><pre>{ "productoId": 1, "nombre": "Cloud de Ariana Grande", "precio": 30000, "stock": 4}</pre></div></div>	No links

Code	Description	Links
400	Error validacion	No links
<div>Media type</div> <div><input type="text" value="*/"/></div> <div>Example Value Schema</div> <div><pre>{ "productoId": 1, "nombre": "Cloud de Ariana Grande", "precio": 30000, "stock": 4 }</pre></div>		

DELETE /api/productos/{id} Eliminar producto

Elimina un producto de la base de datos

Parameters

Cancel

Name	Description
<div><div>id * required</div><div>integer(\$int64)</div><div>(path)</div></div>	<div><input type="text" value="4"/></div>

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
  'http://localhost:8082/api/productos/4' \
  -H 'accept: */*'

```

Request URL

```
http://localhost:8082/api/productos/4

```

Server response

Code	Details	
200	<div>Response headers</div> <div><pre>connection: keep-alive content-length: 0 date: Tue,08 Jul 2025 05:18:42 GMT keep-alive: timeout=60</pre></div>	
Responses		
Code	Description	Links
200	Eliminación exitosa	No links
404	Producto a eliminar no existe	No links

PATCH

/api/productos/{id} Actualizar producto

^

Actualiza los datos de un producto en la base de datos

Parameters

Cancel

Reset

Name	Description
id * required	
integer(\$int64)	4
(path)	

Request body required

application/json

```
{
  "nombre": "perfume FOME"
}
```

//

Execute

Clear

Responses

Curl

```
curl -X 'PATCH' \
  'http://localhost:8082/api/productos/4' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "nombre": "perfume FOME"
  }'
```

Request URL

http://localhost:8082/api/productos/4

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "productoId": 4, "nombre": "perfume FOME", "precio": 90000, "stock": 7 }</pre></div><div><div>Response headers</div><div>connection: keep-alive content-type: application/json date: Tue,08 Jul 2025 05:18:30 GMT</div></div></div>

Download

Code

Details

keep-alive: timeout=60
transfer-encoding: chunked

Responses

Code

Description

Links

200

Actualización exitosa

No links

Media type

/

Controls Accept header.

Example Value Schema

```
{
  "productoId": 1,
  "nombre": "Cloud de Ariana Grande",
  "precio": 30000,
  "stock": 4
}
```

400

Datos proporcionados son inválidos

No links

Media type

/

Example Value Schema

```
{
  "productoId": 1,
  "nombre": "Cloud de Ariana Grande",
  "precio": 30000,
  "stock": 4
}
```

GET

/api/productos/usuario/{id} Buscar usuario por ID



Busca al usuario por id en la base de datos

Parameters

Cancel

id * required

1

Clear

Curl

Request URL

Server response

200

```
connection: keep-alive
content-length: 0
date: Tue,08 Jul 2025 05:16:18 GMT
keep-alive: timeout=60
```

Code

Links

200

No links

Example Value	Schema
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32
33	33
34	34
35	35
36	36
37	37
38	38
39	39
40	40
41	41
42	42
43	43
44	44
45	45
46	46
47	47
48	48
49	49
50	50
51	51
52	52
53	53
54	54
55	55
56	56
57	57
58	58
59	59
60	60
61	61
62	62
63	63
64	64
65	65
66	66
67	67
68	68
69	69
70	70
71	71
72	72
73	73
74	74
75	75
76	76
77	77
78	78
79	79
80	80
81	81
82	82
83	83
84	84
85	85
86	86
87	87
88	88
89	89
90	90
91	91
92	92
93	93
94	94
95	95
96	96
97	97
98	98
99	99
100	100

```
{
  "id": 1,
  "nombre": "Juan Perez, juanchito@gmail.com, Administrador",
  "correo": "Juan Perez, juanchito@gmail.com, Administrador"
```

Code	Description	Links
	<pre>"rol": "Juan Perez, juanchito@gmail.com, Administrador" }</pre>	
402	No se encontró al usuario	No links
	Media type <div><input type="text" value="*/"/></div>	
	Example Value Schema	
	<pre>{ "id": 1, "nombre": "Juan Perez, juanchito@gmail.com, Administrador", "correo": "Juan Perez, juanchito@gmail.com, Administrador", "rol": "Juan Perez, juanchito@gmail.com, Administrador" }</pre>	

GET

/api/productos/producto/{id} Busca a un producto

^

Busca un producto en la base de datos y si existe lo convierte en un DTO para integrarlo con el carrito cuando este lo llame

Parameters

Cancel

Name	Description
<div><div>id * required</div><div>integer(\$int64)</div><div>(path)</div></div>	<div><input type="text" value="4"/></div>

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8082/api/productos/producto/4' \
  -H 'accept: */*'

```

Request URL

http://localhost:8082/api/productos/producto/4

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "productoId": 4, "nombre": "Super perfume", "cantidad": 0, "precio": 90000}</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>connection: keep-alive content-type: application/json date: Tue,08 Jul 2025 05:16:43 GMT keep-alive: timeout=60 transfer-encoding: chunked</pre></div></div>

Responses

Code	Description	Links
200	<div>Producto encontrado</div> <div><div>Media type</div><div><div>*/*</div></div><div>Controls Accept header.</div></div> <div><div>Example Value</div><div>Schema</div><div><pre>{ "productoId": 1, "nombre": "Cloud de Ariana Grande", "cantidad": 9, "precio": 30000}</pre></div></div>	No links

Code	Description	Links
404	<p>Producto no existe</p> <p>Media type</p> <div><input type="text" value="*/"/></div> <p>Example Value Schema</p> <pre>{ "productoId": 1, "nombre": "Cloud de Ariana Grande", "cantidad": 9, "precio": 30000}</pre>	No links

GET

/api/productos/buscar/{id}

Buscar producto por ID

^

Busca al producto por id en la base de datos

Parameters

Cancel

Name	Description
<div><div>id * required</div><div>integer(\$int64)</div><div>(path)</div></div>	<div><input type="text" value="4"/></div>

Execute

Clear

Responses

Curl

curl -X 'GET' \ 'http://localhost:8082/api/productos/buscar/4' \ -H 'accept: */*'

Request URL

http://localhost:8082/api/productos/buscar/4

Server response

Code

Details

200

Response body

```
{
  "productoId": 4,
  "nombre": "Super perfume",
  "precio": 90000,
  "stock": 7
}
```

Download

Response headers

```
connection: keep-alive
content-type: application/json
date: Tue,08 Jul 2025 05:16:53 GMT
keep-alive: timeout=60
transfer-encoding: chunked
```

Responses

Code

Description

Links

200

Se encontró al producto

No links

Media type

/

Controls Accept header.

Example Value Schema

```
{
  "productoId": 1,
  "nombre": "Cloud de Ariana Grande",
  "precio": 30000,
  "stock": 4
}
```

402

No se encontró al producto

No links

Media type

/

Example Value Schema

```
{
  "productoId": 1,
  "nombre": "Cloud de Ariana Grande",
  "precio": 30000,
  "stock": 4
}
```

Schemas

**Producto** {

description: Entidad que representa un Producto

productoId **integer(\$int64)**
example: 1
ID autogenerado con IDENTITY

nombre **string**
example: Cloud de Ariana Grande
Nombre del producto

precio **number(\$double)**
example: 30000
Precio del producto

stock **integer(\$int32)**
example: 4
Stock disponible del producto

}

Usuario {

description: Objeto de transferencia de datos para entidad usuario

id **integer(\$int64)**
example: 1
ID del usuario correspondiente

nombre **string**
example: Juan Perez, juanchito@gmail.com, Administrador
Datos del usuario

correo **string**
example: Juan Perez, juanchito@gmail.com, Administrador
Datos del usuario

rol **string**
example: Juan Perez, juanchito@gmail.com, Administrador
Datos del usuario

}

```
ProductoDTO {  
  description: Objeto de transferencia de datos para entidad producto  
  
  productId integer($int64)  
              example: 1  
              ID del producto correspondiente  
  
  nombre string  
          example: Cloud de Ariana Grande  
          Nombre del producto  
  
  cantidad integer($int32)  
            example: 9  
            Cantidad el producto  
  
  precio number($double)  
          example: 30000  
          Precio del producto  
  
}
```