



# **Case - Sistema de Gestão de Portfólio de Investimentos**

## **Documentação de utilização**

Desenvolvido e documentado por:

Vitor Hugo Camargo de Freitas

## Products

1- GET: Retorna os produtos registrados no arquivo .XML.

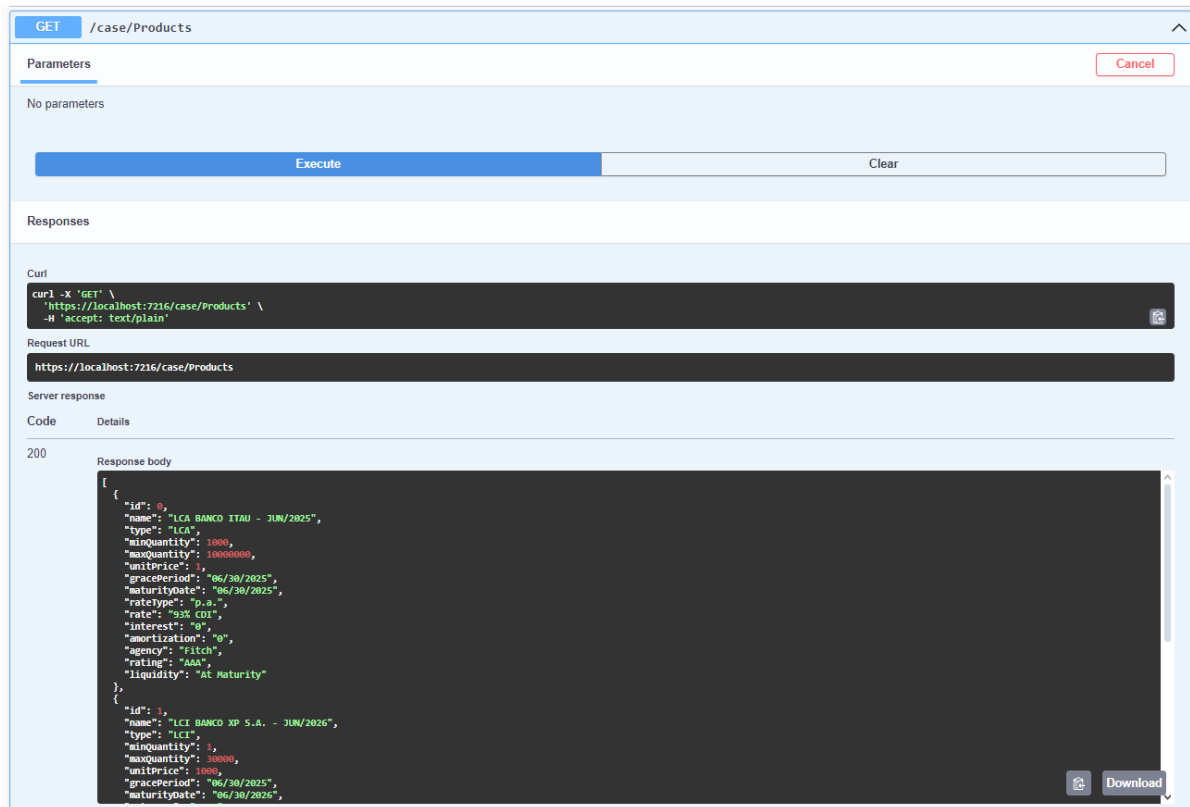
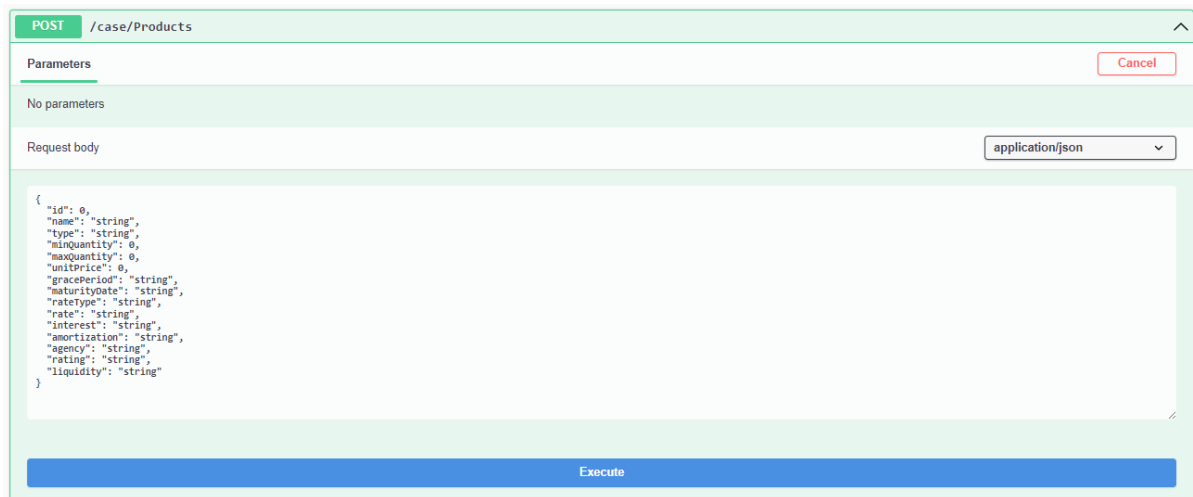


Figura 1: Output da função GET, listando os produtos documentados em productlist.xml

2- POST: Adiciona um produto ao banco de dados.



POST /case/Products

Parameters

No parameters

Request body

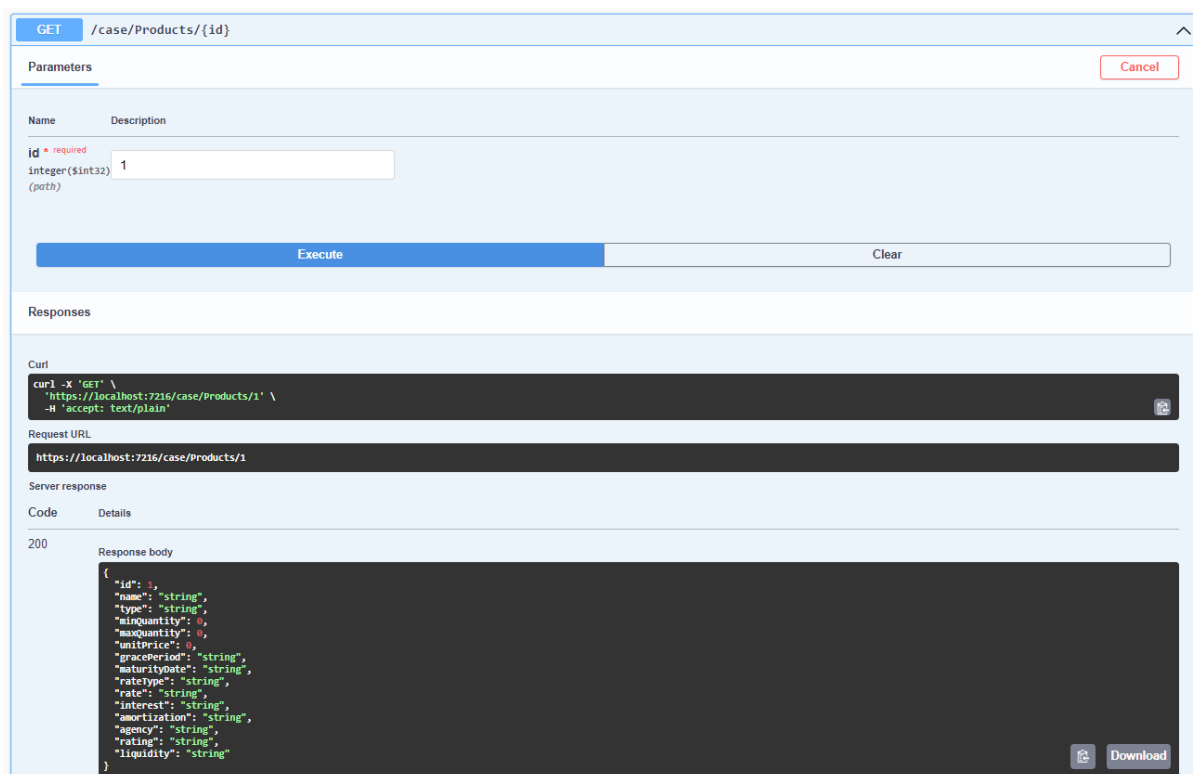
application/json

```
{
  "id": 0,
  "name": "string",
  "type": "string",
  "minQuantity": 0,
  "maxQuantity": 0,
  "unitPrice": 0,
  "gracePeriod": "string",
  "maturityDate": "string",
  "rateType": "string",
  "rate": "string",
  "interest": "string",
  "amortization": "string",
  "agency": "string",
  "rating": "string",
  "liquidity": "string"
}
```

Execute

Figura 2: Caixa de input de dados da função POST.

3- GET [id]: Seleciona um produto específico com base no id.



GET /case/Products/{id}

Parameters

Name Description

id \* required  
integer(\$int32)  
(path)

1

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  https://localhost:7216/case/Products/1 \
  -H 'accept: text/plain'
```

Request URL

```
https://localhost:7216/case/Products/1
```

Server response

Code Details

200

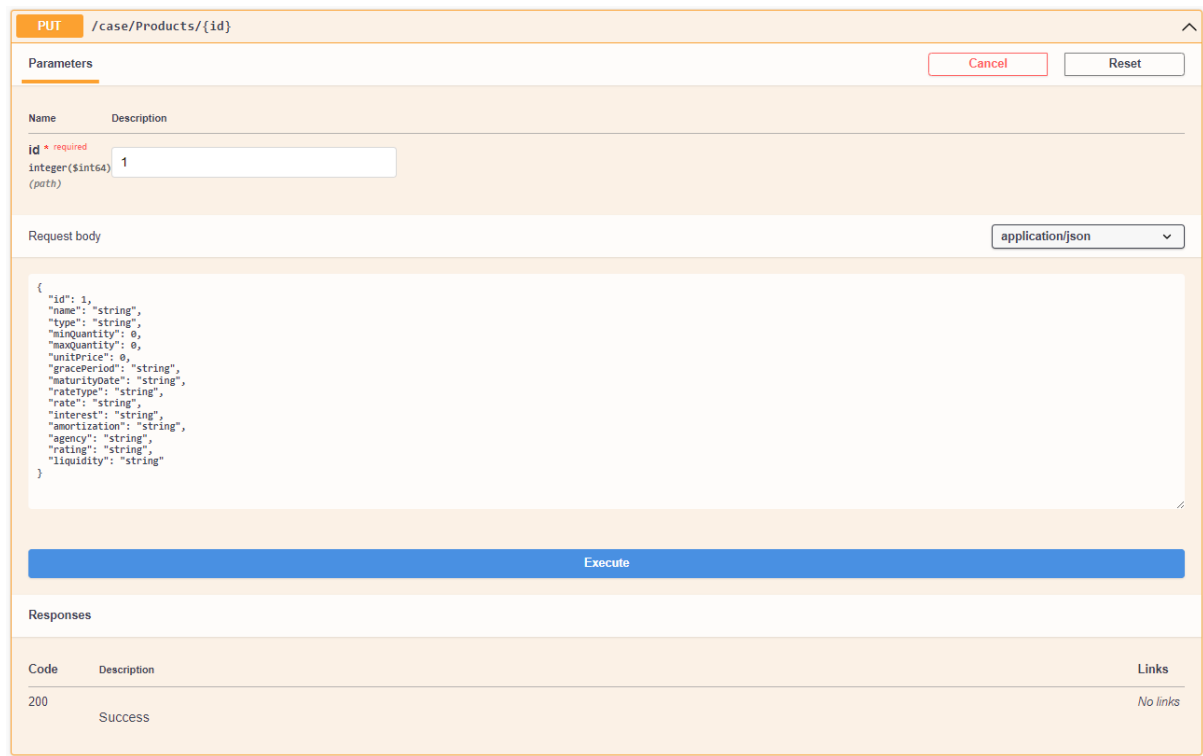
Response body

```
{
  "id": 1,
  "name": "string",
  "type": "string",
  "minQuantity": 0,
  "maxQuantity": 0,
  "unitPrice": 0,
  "gracePeriod": "string",
  "maturityDate": "string",
  "rateType": "string",
  "rate": "string",
  "interest": "string",
  "amortization": "string",
  "agency": "string",
  "rating": "string",
  "liquidity": "string"
}
```

Download

Figura 3: Output da busca pelo id 1.

4- PUT [id]: Altera os dados de um produto já existente.



PUT /case/Products/{id}

Parameters

Cancel Reset

Name	Description
id * required	integer(int64) (path)

Request body

application/json

```
{
  "id": 1,
  "name": "string",
  "type": "string",
  "minquantity": 0,
  "maxquantity": 0,
  "unitPrice": 0,
  "graceperiod": "string",
  "maturityDate": "string",
  "rateType": "string",
  "rate": "string",
  "interest": "string",
  "amortization": "string",
  "agency": "string",
  "rating": "string",
  "liquidity": "string"
}
```

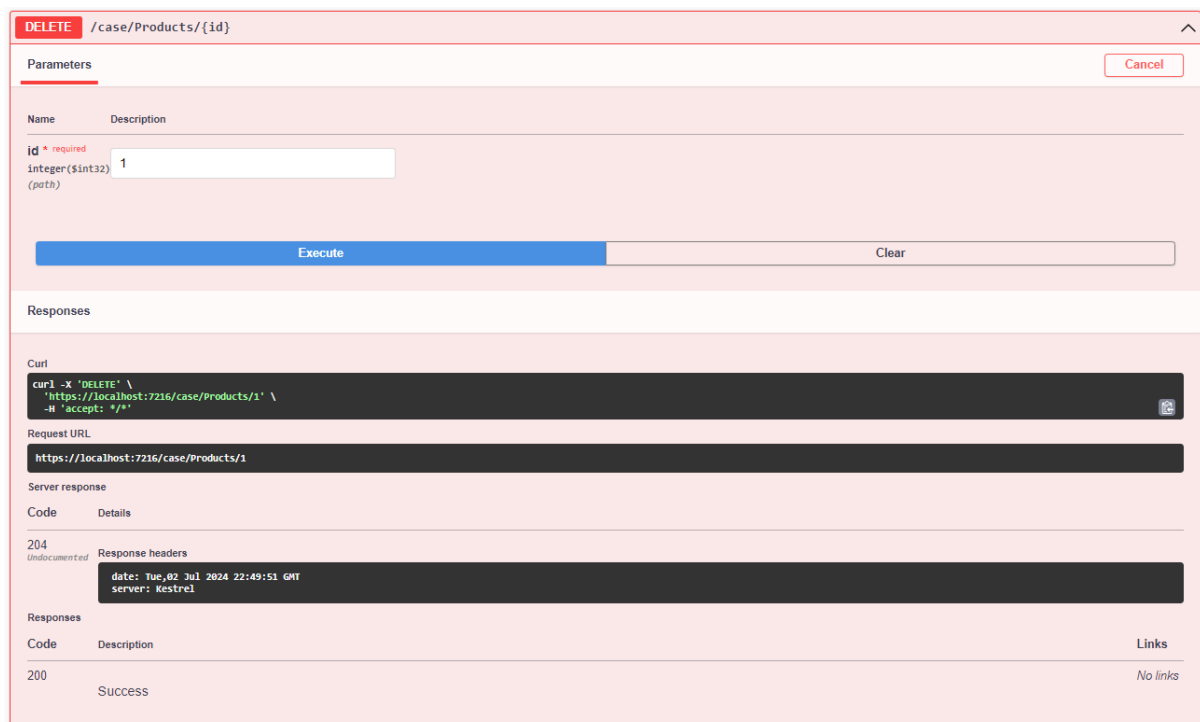
Execute

Responses

Code	Description	Links
200	Success	No links

Figura 4: Caixa de input de alteração do id 1.

5- DELETE [id]: Remove um produto específico com base no id.



DELETE /case/Products/{id}

Cancel

Name	Description
id * required	integer(int32) (path)

Execute Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7216/case/Products/1' \
-H 'accept: */*'
```

Request URL

https://localhost:7216/case/Products/1

Server response

Code	Details
204	Response headers date: Tue, 02 Jul 2024 22:49:51 GMT server: Kestrel

Responses

Code	Description	Links
200	Success	No links

Figura 5: Remoção do produto de id 1.

## Users

1- As funções da aba “Users” seguem a mesma lógica exibidas anteriormente.

Users		^
GET	/case/Users	▼
POST	/case/Users	▼
GET	/case/Users/{id}	▼
PUT	/case/Users/{id}	▼
DELETE	/case/Users/{id}	▼

Figura 6: Menu de funções da aba Users.