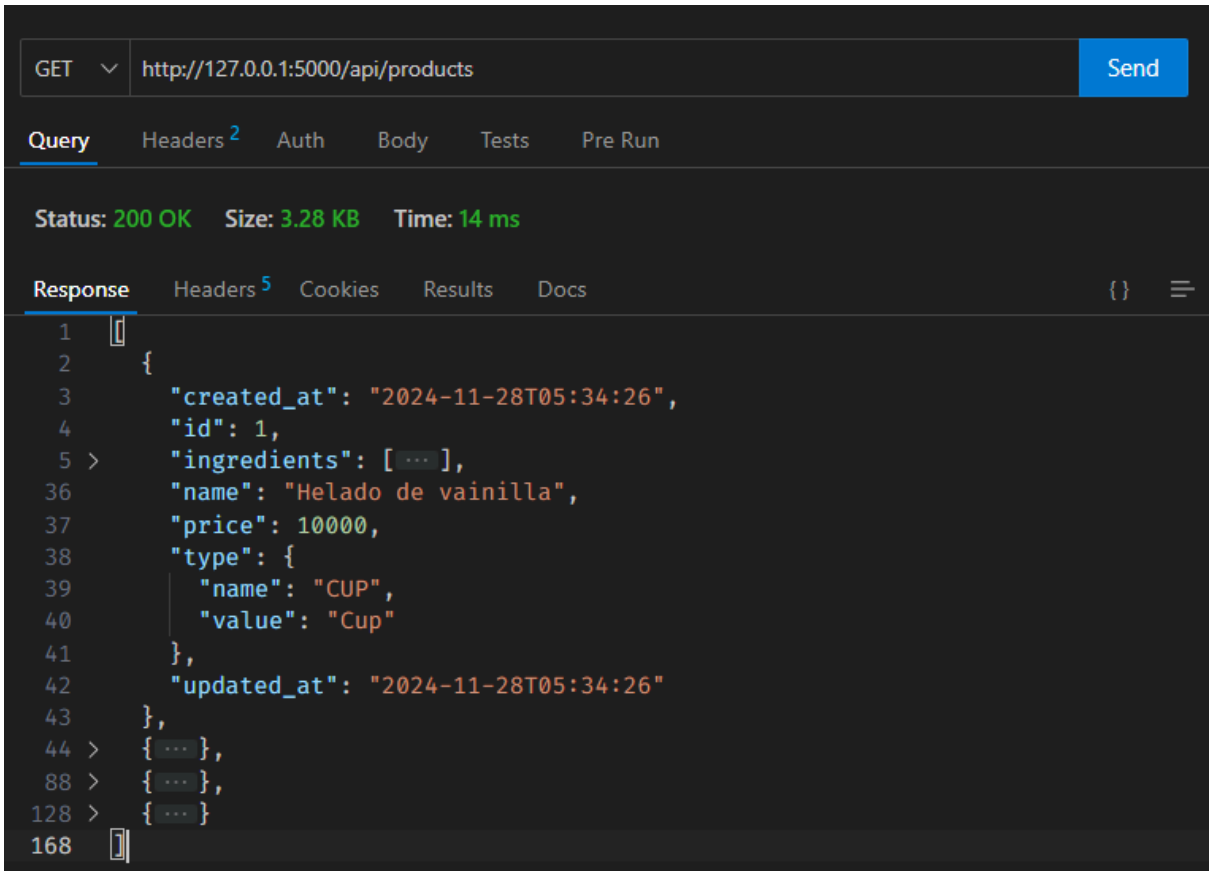


Módulo 3 Taller 2

Consultar todos los productos

url **/api/products**
method GET
params Ninguno
returns List[Product]



Consultar un producto según su ID

url /api/products/**get_by_id**
method GET
params id:int
returns Product (200) | ValueError (404) | Error (Code)

GET

http://127.0.0.1:5000/api/products/get_by_id/1

Send

Query

Headers 2

Auth

Body

Tests

Pre Run

Status: 200 OK

Size: 779 Bytes

Time: 30 ms

Response

Headers 5

Cookies

Results

Docs

{}

≡

1

{

2

"created_at": "2024-11-28T05:48:20",

3

"id": 1,

4

> "ingredients": [...],

35

"name": "Helado de vainilla",

36

"price": 10000,

37

"type": {

38

"name": "CUP",

39

"value": "Cup"

40

},

41

"updated_at": "2024-11-28T05:48:20"

42

}

Copy

GET

http://127.0.0.1:5000/api/products/get_by_id/4

Send

Query

Headers 2

Auth

Body

Tests

Pre Run

Status: 200 OK

Size: 718 Bytes

Time: 9 ms

Response

Headers 5

Cookies

Results

Docs

{}

≡

1

{

2

"id": 4,

3

> "ingredients": [...],

34

"name": "Malteada de café",

35

"price": 9500,

36

"type": {

37

"name": "MILK_SHAKE",

38

"value": "Milk_Shake"

39

}

40

}

Copy

Consultar un producto según su nombre

url /api/products/**get_by_name**
method GET
params name:string
returns Product (200) | ValueError (404) | Error (Code)

GET

http://127.0.0.1:5000/api/products/get_by_name/Helado%20de%20vainilla

Send

Query

Headers2

Auth

Body

Tests

Pre Run

Status: 200 OK

Size: 779 Bytes

Time: 8 ms

Response

Headers5

Cookies

Results

Docs

{}

≡

1

{

2

"created_at": "2024-11-28T06:05:24",

3

"id": 1,

4 >

"ingredients": [...],

35

"name": "Helado de vainilla",

36

"price": 10000,

37

"type": {

38

"name": "CUP",

39

"value": "Cup"

40

},

41

"updated_at": "2024-11-28T06:05:24"

42

}

GET

http://127.0.0.1:5000/api/products/get_by_name/Malteada%20de%20Chocolate

Send

Query

Headers2

Auth

Body

Tests

Pre Run

Status: 200 OK

Size: 719 Bytes

Time: 10 ms

Response

Headers5

Cookies

Results

Docs

{}

≡

1

{

2

"id": 3,

3 >

"ingredients": [...],

34

"name": "Malteada de chocolate",

35

"price": 8500,

36

"type": {

37

"name": "MILK_SHAKE",

38

"value": "Milk_Shake"

39

}

40

}

Consultar las calorías de un producto según su ID

url /api/products/**calories_by_id**
method GET
params id:int
returns Product (200) | ValueError (404) | Error (Code)

GET ⌵ http://127.0.0.1:5000/api/products/calories_by_id/1 Send

Query Headers ² Auth Body Tests Pre Run

Status: 200 OK Size: 35 Bytes Time: 7 ms

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "calories": 241.3,  
3   "id": 1  
4 }
```

Copy

GET ⌵ http://127.0.0.1:5000/api/products/calories_by_id/4 Send

Query Headers ² Auth Body Tests Pre Run

Status: 200 OK Size: 36 Bytes Time: 7 ms

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "calories": 379.55,  
3   "id": 4  
4 }
```

Consultar la rentabilidad de un producto según su ID

url /api/products/**calculate_profit_by_id**
method GET
params id:int
returns Product (200) | ValueError (404) | Error (Code)

GET ⌵ http://127.0.0.1:5000/api/products/calculate_profit_by_id/2 Send

Query Headers ² Auth Body Tests Pre Run

Status: 200 OK Size: 33 Bytes Time: 7 ms

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "id": 2,  
3   "profit": 14000  
4 }
```

GET ⌵ http://127.0.0.1:5000/api/products/calculate_profit_by_id/4 Send

Query Headers ² Auth Body Tests Pre Run

Status: 200 OK Size: 32 Bytes Time: 9 ms

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "id": 4,  
3   "profit": 8210  
4 }
```

Consultar el costo de producción de un producto según su ID

url /api/products/**calculate_cost_by_id**
method GET
params id:int
returns Product (200) | ValueError (404) | Error (Code)

GET ⌵ http://127.0.0.1:5000/api/products/calculate_cost_by_id/1 Send

Query Headers ² Auth Body Tests Pre Run

Status: **201 CREATED** Size: **29 Bytes** Time: **7 ms**

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "cost": 840,  
3   "id": 1  
4 }
```

GET ⌵ http://127.0.0.1:5000/api/products/calculate_cost_by_id/3 Send

Query Headers ² Auth Body Tests Pre Run

Status: **201 CREATED** Size: **30 Bytes** Time: **7 ms**

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "cost": 1310,  
3   "id": 3  
4 }
```

Copy

Vender un producto según su ID

url /api/sales/**sale_product_by_id**
method GET
params id:int
returns Product (200) | ValueError (404) | ProductNotAvailableError (409 |
 Error (Code)

GET http://127.0.0.1:5000/api/sales/sale_product_by_id/1 Send

Query Headers² Auth Body Tests Pre Run

Status: 200 OK Size: 65 Bytes Time: 245 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {
2   "id": 1,
3   "message": "Producto vendido con éxito !!"
4 }
```

Cuando el producto se agotó, se emite un error personalizado 409

GET http://127.0.0.1:5000/api/sales/sale_product_by_id/1 Send

Query Headers² Auth Body Tests Pre Run

Status: 409 CONFLICT Size: 65 Bytes Time: 8 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {
2   "error": "¡No hay producto disponible para la venta!"
3 }
```

Consultar todos los Ingredientes

url **/api/ingredients/**
method GET
params Ninguno
returns List[Ingredient] (200) | ValueError (404) | Error (Code)

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** http://127.0.0.1:5000/api/ingredients/
- Status:** 200 OK
- Size:** 1.83 KB
- Time:** 8 ms
- Response:** A JSON array of ingredients. The first ingredient is "Base Café" and the second is "Chips de Chocolate".

```
1 {  
2   {  
3     "calories": 30,  
4     "extra_data": {  
5       "flavor": "Café"  
6     },  
7     "id": 1,  
8     "is_vegetarian": true,  
9     "name": "Base Café",  
10    "price": 300,  
11    "units": 0.2  
12  },  
13  { ... },  
24  { ... },  
35  { ... },  
46  { ... },  
57  {  
58    "calories": 99,  
59    "extra_data": {},  
60    "id": 6,  
61    "is_vegetarian": true,  
62    "name": "Chips de Chocolate",  
63    "price": 200,  
64    "units": 1.0  
65  },  
66  { ... },  
75  { ... },  
84  { ... },  
93  { ... },  
102 { ... }  
111 }
```


Consultar un ingrediente según su ID

url /api/ingredients/**get_by_id**
method GET
params id:int
returns Ingredient (200) | ValueError (404) | Error (Code)

GET ⌵ http://127.0.0.1:5000/api/ingredients/get_by_id/1 Send

Query Headers ² Auth Body Tests Pre Run

Status: 200 OK Size: 166 Bytes Time: 29 ms

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "calories": 30,  
3   "extra_data": {  
4     "flavor": "Café"  
5   },  
6   "id": 1,  
7   "is_vegetarian": true,  
8   "name": "Base Café",  
9   "price": 300,  
10  "units": 0.2  
11 }
```

GET ⌵ http://127.0.0.1:5000/api/ingredients/get_by_id/3 Send

Query Headers ² Auth Body Tests Pre Run

Status: 200 OK Size: 168 Bytes Time: 5 ms

Response Headers ⁵ Cookies Results Docs {} ≡

```
1 {  
2   "calories": 55,  
3   "extra_data": {  
4     "flavor": "Limón"  
5   },  
6   "id": 3,  
7   "is_vegetarian": true,  
8   "name": "Base Limón",  
9   "price": 350,  
10  "units": 0.6  
11 }
```

GET

⌵

http://127.0.0.1:5000/api/ingredients/get_by_id/1500000

Send

Query

Headers 2

Auth

Body

Tests

Pre Run

Status: 404 NOT FOUND

Size: 43 Bytes

Time: 5 ms

Response

Headers 5

Cookies

Results

Docs

{}

≡

1

{

2

"error": "Ingrediente no encontrado"

3

}

Consultar un ingrediente según su nombre

url /api/ingredients/**get_by_name**
method GET
params name:string
returns Ingredient (200) | ValueError (404) | Error (Code)

The screenshot shows a REST client interface with a dark theme. At the top, the method is set to 'GET' and the URL is 'http://127.0.0.1:5000/api/ingredients/get_by_name/Base%20Chocolate'. A blue 'Send' button is on the right. Below the URL bar, there are tabs for 'Query', 'Headers', 'Auth', 'Body', 'Tests', and 'Pre Run'. The 'Query' tab is selected. The response status is '200 OK', size is '166 Bytes', and time is '5 ms'. Below this, there are tabs for 'Response', 'Headers', 'Cookies', 'Results', and 'Docs'. The 'Response' tab is selected, showing a JSON object with the following structure:

```
1  {  
2    "calories": 30,  
3    "extra_data": {  
4      "flavor": "Café"  
5    },  
6    "id": 1,  
7    "is_vegetarian": true,  
8    "name": "Base Café",  
9    "price": 300,  
10   "units": 0.2  
11 }
```

Consultar si un ingrediente es sano según su ID

url /api/ingredients/**is_healthy**
method GET
params name:string
returns Ingredient (200) | ValueError (404) | Error (Code)

GET ⌵ http://127.0.0.1:5000/api/ingredients/is_healthy/1 Send

Query Headers² Auth Body Tests Pre Run

Status: 200 OK Size: 37 Bytes Time: 8 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "_is_healthy": true,  
3   "id": 1  
4 }
```

GET ⌵ http://127.0.0.1:5000/api/ingredients/is_healthy/10 Send

Query Headers² Auth Body Tests Pre Run

Status: 200 OK Size: 38 Bytes Time: 6 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "_is_healthy": true,  
3   "id": 10  
4 }
```

Reabastecer un ingrediente según su ID

url /api/ingredients/**supply**
method GET
params name:string
returns Ingredient (200) | ValueError (404) | Error (Code)

GET ⌵ `http://127.0.0.1:5000/api/ingredients/supply/5` Send

Query Headers² Auth Body Tests Pre Run

Status: 200 OK Size: 54 Bytes Time: 67 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "id": 5,  
3   "new_units": 6.2,  
4   "old_units": 1.2  
5 }
```

GET ⌵ `http://127.0.0.1:5000/api/ingredients/supply/10` Send

Query Headers² Auth Body Tests Pre Run

Status: 200 OK Size: 56 Bytes Time: 85 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "id": 10,  
3   "new_units": 15.0,  
4   "old_units": 5.0  
5 }
```

Renovar el inventario de un ingrediente según su ID

url /api/ingredients/**reset_units**
method GET
params name:string
returns Ingredient (200) | ValueError (404) | Error (Code) |
 MethodDoesNotApplyError (409)

GET Send

Query Headers² Auth Body Tests Pre Run

Status: 201 CREATED Size: 54 Bytes Time: 5 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "id": 8,  
3   "new_units": 0.0,  
4   "old_units": 3.0  
5 }
```

Copy

GET Send

Query Headers² Auth Body Tests Pre Run

Status: 201 CREATED Size: 55 Bytes Time: 8 ms

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "id": 10,  
3   "new_units": 0.0,  
4   "old_units": 5.0  
5 }
```

Copy

GET Send

Query Headers² Auth Body Tests Pre Run

Status: 200 OK Size: 89 Bytes Time: 2.28 s

Response Headers⁵ Cookies Results Docs {} ≡

```
1 {  
2   "error": "¡Sólo se puede abastecer un ingrediente del tipo complemento!"  
3 }
```