

Ordering API Documentation

Requests & Responses

- Version: 1.0 –

- Date: 11/07/2023 –

- Prepared by : Jean Hervé Ingabire -

1. Category

A- Add Category

Endpoint: localhost:8081/api/v1/categories

Method : POST

Successful Case - (Http Status = 201):

The screenshot displays a REST client interface with a tab for the endpoint `localhost:8081/api/v1/categories`. The request method is set to **POST**. The request body is a JSON object:

```
1 {
2   "name": "Cat5",
3   "description": "This is a Cat5"
4 }
```

The response status is **201 Created**, with a time of 622 ms and a size of 264 B. The response body is shown in the **Body** tab, formatted as JSON:

```
1 {
2   "id": 5,
3   "name": "Cat5",
4   "description": "This is a Cat5",
5   "createdAt": "2023-07-11T13:47:40.6106846"
6 }
```

Failed Case - (Http Status = 409):

The screenshot displays a REST client interface with a tab for `localhost:8081/api/v1/categories`. The request method is `POST`. The request body is a JSON object:

```
1 {
2   "name": "Cat5",
3   "description": "This is a Cat5"
4 }
```

The response status is `409 Conflict`. The response body is a JSON object:

```
1 {
2   "timestamp": "2023-07-11T11:54:07.232+00:00",
3   "status": 409,
4   "error": "Conflict",
5   "message": "Category:[Cat5] is already exist",
6   "path": "/api/v1/categories"
7 }
```

B- Read Categories

Endpoint: localhost:8081/api/v1/categories

Method : GET

Successful Case - (Http Status = 200):

The screenshot displays a REST client interface with a tab for the endpoint `localhost:8081/api/v1/categories`. The method is set to `GET`. The `Params` tab is active, showing a table for query parameters:

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Below the table, the `Body` tab is selected, showing the response in `JSON` format:

```
1 {
2   {
3     "id": 1,
4     "name": "Cat1",
5     "description": "This is a Cat1",
6     "createdAt": "2023-07-10T17:22:44"
7   },
8   {
9     "id": 2,
10    "name": "Cat2",
11    "description": "This is a Cat2",
12    "createdAt": "2023-07-10T17:22:55"
13  },
14 }
```

The status bar at the bottom indicates `Status: 200 OK`, `Time: 637 ms`, and `Size: 605 B`. A `Save Response` button is also visible.

C- Read Categories By ID

Endpoint: localhost:8081/api/v1/categories/1

Method : GET

Successful Case - (Http Status = 200):

The screenshot displays a REST client interface with the following components:

- Toolbar:** Includes navigation arrows, a list of request methods (POST, GET, PUT, PATCH), a plus sign, and an environment dropdown set to "No Environment".
- Request Bar:** Shows the URL "localhost:8081/api/v1/categories/1" and a "Send" button.
- Request Tabs:** "Params", "Authorization", "Headers (6)", "Body", "Pre-request Script", "Tests", "Settings", and "Cookies".
- Query Params Table:**

KEY	VALUE	DESCRIPTION
Key	Value	Description
- Response Tabs:** "Body", "Cookies", "Headers (5)", and "Test Results".
- Response Status:** "Status: 200 OK", "Time: 269 ms", "Size: 251 B", and a "Save Response" button.
- Response Body:** A JSON object displayed in "Pretty" format:

```
1 {
2   "id": 1,
3   "name": "Cat1",
4   "description": "This is a Cat1",
5   "createdAt": "2023-07-10T17:22:44"
6 }
```

D- Read Categories By Name and Description

Endpoint: localhost:8081/api/v1/categories/search?name=Cat1&description=cat

Method : GET

Successful Case - (Http Status = 200):

The screenshot displays a REST client interface with a request configured for the endpoint `localhost:8081/api/v1/categories/search?name=Cat1&description=cat` using the GET method. The 'Query Params' section contains two parameters: 'name' with value 'Cat1' and 'description' with value 'cat'. The response status is 200 OK, with a time of 61 ms and a size of 253 B. The response body is shown in JSON format, containing an array with one object representing a category.

localhost:8081/api/v1/categories/search?name=Cat1&description=cat

GET localhost:8081/api/v1/categories/search?name=Cat1&description=cat

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	name	Cat1			
<input checked="" type="checkbox"/>	description	cat			
	Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 200 OK Time: 61 ms Size: 253 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 1,
4     "name": "Cat1",
5     "description": "This is a Cat1",
6     "createdAt": "2023-07-10T17:22:44"
7   }
8 ]
```

Failed Case - (Http Status = 404):

The screenshot shows a REST client interface with a list of requests at the top. The selected request is a GET request to `localhost:8081/api/v1/categories/search?name=Cat 2&description=cat2`. The interface includes tabs for Params, Authorization, Headers (6), Body, Pre-request Script, Tests, and Settings. The Params tab is active, showing query parameters for 'name' (Cat 2) and 'description' (cat2). Below the params tab, there is a table for query parameters:

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	name	Cat 2			
<input checked="" type="checkbox"/>	description	cat2			
	Key	Value	Description		

Below the Params tab, there are tabs for Body, Cookies, Headers (5), and Test Results. The Body tab is active, showing the response status and body. The status is `404 Not Found` with a time of `73 ms` and size of `337 B`. The response body is a JSON object:

```
1 {
2   "timestamp": "2023-07-11T12:08:36.073+00:00",
3   "status": 404,
4   "error": "Not Found",
5   "message": "Category with name[Cat 2] does not exist",
6   "path": "/api/v1/categories/search"
7 }
```

E- Update Category

Endpoint: localhost:8081/api/v1/categories/1

Method : PUT

Successful Case - (Http Status = 200):

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

- URL:** localhost:8081/api/v1/categories/1
- Method:** PUT
- Body (raw):**

```
1 {  
2   "name": "Cat 44",  
3   "description": "This is Cat 1 modified"  
4 }
```
- Status:** 200 OK
- Time:** 1647 ms
- Size:** 261 B
- Response Body (Pretty):**

```
1 {  
2   "id": 1,  
3   "name": "Cat 44",  
4   "description": "This is Cat 1 modified",  
5   "createdAt": "2023-07-10T17:22:44"  
6 }
```


F- Update Category

Endpoint: localhost:8081/api/v1/categories/1

Method : PATCH

Successful Case - (Http Status = 200):

The screenshot displays a REST client interface with a sequence of requests in the top bar: POST loc, POST loc, GET loca, GET loca, GET loca, PUT loca, and PATCH lo. The active request is a PATCH to localhost:8081/api/v1/categories/1. The request body is a JSON object: {"description": "This is Cat 1 modified using Patch"}. The response status is 200 OK, with a time of 293 ms and a size of 273 B. The response body is a JSON object: {"id": 1, "name": "Cat 44", "description": "This is Cat 1 modified using Patch", "createdAt": "2023-07-10T17:22:44"}. The interface includes tabs for Params, Authorization, Headers (8), Body, Pre-request Script, Tests, and Settings. The Body tab is selected, showing the request and response in JSON format. The response is also available in Pretty, Raw, Preview, and Visualize formats.

localhost:8081/api/v1/categories/1

PATCH localhost:8081/api/v1/categories/1

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 {"description": "This is Cat 1 modified using Patch"

2 }

3 }

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 293 ms Size: 273 B Save Response

Pretty Raw Preview Visualize JSON

1 {"id": 1,

2 "name": "Cat 44",

3 "description": "This is Cat 1 modified using Patch",

4 "createdAt": "2023-07-10T17:22:44"

5 }

6 }

G- Update Category

Endpoint: localhost:8081/api/v1/categories/4

Method : DELETE

Successful Case - (Http Status = 200):

The screenshot displays a REST client interface with a list of requests at the top. The selected request is a DELETE method to the endpoint `localhost:8081/api/v1/categories/4`. The interface shows the request details, including the method, endpoint, and a 'Send' button. Below this, the 'Params' tab is active, showing a table for query parameters. The table has columns for KEY, VALUE, and DESCRIPTION. A single parameter is listed: 'Key' with a value of 'Value' and a description of 'Description'. At the bottom, the 'Body' tab is active, showing the response status as '200 OK', a time of '5.38 s', and a size of '123 B'. The response is displayed in 'Pretty' JSON format.

localhost:8081/api/v1/categories/4

DELETE localhost:8081/api/v1/categories/4

Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (4) Test Results

Status: 200 OK Time: 5.38 s Size: 123 B Save Response

Pretty Raw Preview Visualize JSON

1

2. Product

A- Add Product

Endpoint: localhost:8081/api/v1/products

Method : POST

Successful Case - (Http Status = 201):

The screenshot displays a REST client interface with a POST request configured for the endpoint `localhost:8081/api/v1/products`. The request body is a JSON object:

```
1 {
2   "name": "Product5",
3   "description": "This is a product5",
4   "category": {
5     "id": 1
6   }
7 }
```

The response is shown in the bottom panel, indicating a status of 201 (Created) with a time of 1099 ms and a size of 287 B. The response body is a JSON object:

```
1 {
2   "id": 7,
3   "name": "Product5",
4   "description": "This is a product5",
5   "createdAt": "2023-07-11T14:18:20.8746476",
6   "categoryId": 1
7 }
```

B- Add Product

Endpoint: localhost:8081/api/v1/products

Method : POST

Failed Case - (Http Status = 404):

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

- Endpoint:** localhost:8081/api/v1/products
- Method:** POST
- Body (JSON):**

```
1 {
2   "name": "Product5",
3   "description": "This is a product5",
4   "category": "1",
5   "id": "12"
6 }
```
- Status:** 404 Not Found
- Time:** 62 ms
- Size:** 323 B
- Response Body (JSON):**

```
1 {
2   "timestamp": "2023-07-11T12:20:51.589+00:00",
3   "status": 404,
4   "error": "Not Found",
5   "message": "Category with id[12] does not exist",
6   "path": "/api/v1/products"
7 }
```

C- Read Products

Endpoint: localhost:8081/api/v1/products

Method : GET

Failed Case - (Http Status = 200):

The screenshot shows a REST client interface with a GET request to `localhost:8081/api/v1/products`. The response is a JSON array of two product objects, with a status of 200 OK.

Request:

- Method: GET
- URL: localhost:8081/api/v1/products
- Body: none

Response:

```
1 {
2   {
3     "id": 1,
4     "name": "Product1",
5     "description": "This is a product1",
6     "createdAt": "2023-07-10T17:25:00",
7     "categoryId": 1
8   },
9   {
10    "id": 2,
11    "name": "Product2",
12    "description": "This is a product2",
13    "createdAt": "2023-07-10T17:25:50",
14    "categoryId": 2
15  },
16 }
```

Status: 200 OK, Time: 106 ms, Size: 609 B

D- Read Product By Id

Endpoint: localhost:8081/api/v1/products/1

Method : GET

Failed Case - (Http Status = 200):

The screenshot displays a REST client interface with a tab for the endpoint `localhost:8081/api/v1/products/1`. The request method is set to `GET`. The response status is `200 OK`, with a response time of `101 ms` and a size of `274 B`. The response body is shown in JSON format, containing the following data:

```
1 {
2   "id": 1,
3   "name": "Product1",
4   "description": "This is a product1",
5   "createdAt": "2023-07-10T17:25:00",
6   "categoryId": 1
7 }
```

E- Read Products By Id

Endpoint: localhost:8081/api/v1/products/search?description=Product2

Method : GET

Failed Case - (Http Status = 200):

The screenshot shows a REST client interface with a GET request to `localhost:8081/api/v1/products/search?description=Product2`. The request is configured with the following parameters:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> description	Product2	
Key	Value	Description

The response is a JSON object with status 200 OK, Time: 24 ms, and Size: 276 B. The response body is displayed in the 'Body' tab, showing the following JSON structure:

```
1 {
2   "id": 2,
3   "name": "Product2",
4   "description": "This is a product2",
5   "createdAt": "2023-07-10T17:25:50",
6   "categoryId": 2
7 }
8
9
```

F- Update Product

Endpoint: localhost:8081/api/v1/products/2

Method : PUT

Failed Case - (Http Status = 200):

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

- URL:** localhost:8081/api/v1/products/1
- Method:** PUT
- Body:**

```
1 {
2   "name": "Product1",
3   "description": "This is a product1 updated",
4   "category": {
5     "id": 2
6   }
7 }
8
```
- Response:** Status: 200 OK, Time: 163 ms, Size: 282 B. The response body is:

```
1 {
2   "id": 1,
3   "name": "Product1",
4   "description": "This is a product1 updated",
5   "createdAt": "2023-07-10T17:25:00",
6   "categoryId": 2
7 }
```


G- Update Product - PATCH

Endpoint: localhost:8081/api/v1/products/2

Method : PATCH

Failed Case - (Http Status = 200):

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

- URL:** localhost:8081/api/v1/products/1
- Method:** PATCH
- Body:**

```
{
  "name": "Product11"
}
```
- Response:** Status: 200 OK, Time: 186 ms, Size: 283 B. The response body is empty.

H- Delete Product

Endpoint: localhost:8081/api/v1/products/2

Method : DELETE

Failed Case - (Http Status = 200):

The screenshot shows a REST client interface with a sequence of requests in the top bar: DEL local, POST loc, GET loca, GET loca, PUT loca, PATCH lo, and DEL local. The selected request is a DELETE method to the endpoint localhost:8081/api/v1/products/2. The interface includes tabs for Params, Authorization, Headers (6), Body, Pre-request Script, Tests, and Settings. The Params tab is active, showing a table with columns KEY, VALUE, and DESCRIPTION. Below the table, there is a section for Query Params. The response section at the bottom shows a status of 200 OK, with time 517 ms and size 123 B. The response body is displayed in a 'Pretty' JSON format, showing a single element '1'.

localhost:8081/api/v1/products/2

DELETE localhost:8081/api/v1/products/2

Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (4) Test Results

Status: 200 OK Time: 517 ms Size: 123 B Save Response

Pretty Raw Preview Visualize JSON

```
1
```

3. Customer

A- Add Customer

Endpoint: localhost:8081/api/v1/customers

Method : POST

Successful Case - (Http Status = 201):

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

- URL:** localhost:8081/api/v1/customers
- Method:** POST
- Body (Request):**

```
1 {
2   "name": "Jean",
3   "email": "jean@gmail.com",
4   "phone": "+48963852700",
5   "address": "Stoklosy 5",
6   "age": 26,
7   "gender": "MALE"
8 }
```
- Response (Status: 201 Created):**

```
1 {
2   "id": 4,
3   "name": "Jean",
4   "email": "jean@gmail.com",
5   "phone": "+48963852700",
6   "address": "Stoklosy 5",
7   "age": 26,
8   "gender": "MALE",
9   "createdAt": "2023-07-11T14:34:16.9706901"
10 }
```

B- Add Customer

Endpoint: localhost:8081/api/v1/customers

Method : POST

Failed Case(Email already taken) - (Http Status = 400):

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

- URL:** localhost:8081/api/v1/customers
- Method:** POST
- Body (JSON):**

```
1 {
2   "name": "Jean",
3   "email": "jean@gmail.com",
4   "phone": "+48963852700",
5   "address": "Stoklosy 5",
6   "age": 26,
7   "gender": "MALE"
8 }
```
- Response (JSON):**

```
1 {
2   "timestamp": "2023-07-11T12:35:36.795+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Email [jean@gmail.com] is already taken",
6   "path": "/api/v1/customers"
7 }
```
- Status:** 400 Bad Request
- Time:** 51 ms
- Size:** 303 B

C- Read Customers

Endpoint: localhost:8081/api/v1/customers

Method : GET

Successful Case- (Http Status = 200):

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

- URL:** localhost:8081/api/v1/customers
- Method:** GET
- Status:** 200 OK
- Time:** 81 ms
- Size:** 637 B

The response body is displayed in JSON format:

```
1 {
2   {
3     "id": 1,
4     "name": "Jasmina",
5     "email": "jasmina@gmail.com",
6     "phone": "+48963852749",
7     "address": "Jana Paul 2",
8     "age": 45,
9     "gender": "FEMALE",
10    "createdAt": "2023-07-11T00:52:19"
11  },
12  {
13    "id": 3,
14    "name": "Harvey",
15    "email": "harvey@gmail.com",
16    "phone": "+48963852746",
17    "address": "Stoklosy 5",
18    "age": 26,
```

D- Read Customers

Endpoint: localhost:8081/api/v1/customers/search?email=jasmina@gmail.com

Method : GET

Successful Case- (Http Status = 200):

The screenshot displays a REST client interface with the following components:

- URL Bar:** localhost:8081/api/v1/customers/search?email=jasmina@gmail.com
- Method:** GET
- Headers:** 6 hidden
- Body:** Pretty, Raw, Preview, Visualize (JSON selected)
- Status:** 200 OK, Time: 156 ms, Size: 327 B
- Response Body (JSON):**

```
{  "id": 1,  "name": "Jasmina",  "email": "jasmina@gmail.com",  "phone": "+48963852749",  "address": "Jana Paul 2",  "age": 45,  "gender": "FEMALE",  "createdAt": "2023-07-11T00:52:19"}
```

E- Update Customer

Endpoint: localhost:8081/api/v1/customers/1

Method : PUT

Successful Case- (Http Status = 200):

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

- URL:** localhost:8081/api/v1/customers/1
- Method:** PUT
- Request Body (JSON):**

```
1 {
2   "name": "Alan",
3   "email": "alan3@gmail.com",
4   "phone": "+48963852708",
5   "address": "Zamiany 19",
6   "age": 41,
7   "gender": "MALE"
8 }
```
- Response Status:** 200 OK, Time: 174 ms, Size: 317 B
- Response Body (JSON):**

```
1 {
2   "id": 1,
3   "name": "Alan",
4   "email": "alan3@gmail.com",
5   "phone": "+48963852708",
6   "address": "Zamiany 19",
7   "age": 41,
8   "gender": "MALE",
9   "createdAt": "2023-07-11T00:52:19"
10 }
```

The interface also shows tabs for Params, Authorization, Headers (8), Body, Pre-request Script, Tests, and Settings. The response is displayed in a Pretty view format.

F- Update Customer - PATCH

Endpoint: localhost:8081/api/v1/customers/1

Method : PATCH

Successful Case- (Http Status = 200):

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

- URL:** localhost:8081/api/v1/customers/1?email=john123@hotmail.com
- Method:** PATCH
- Body:** (Empty)
- Response:** Status: 200 OK, Time: 291 ms, Size: 321 B
- Response Body (JSON):**

```
1 {
2   "id": 1,
3   "name": "Alan",
4   "email": "john123@hotmail.com",
5   "phone": "+48963852708",
6   "address": "Zamiany 19",
7   "age": 41,
8   "gender": "MALE",
9   "createdAt": "2023-07-11T00:52:19"
10 }
```


G- Delete Customer

Endpoint: localhost:8081/api/v1/customers/1

Method : DELETE

Successful Case- (Http Status = 200):

The screenshot displays a REST client interface with the following components:

- Top Bar:** Contains a sequence of method buttons (PATCH, DEL, POST, GET, PUT, PATCH, DEL) with status indicators, a plus icon, and the text "No Environment".
- URL Bar:** Shows the endpoint "localhost:8081/api/v1/customers/1" with a "Save" button and a comment icon.
- Method and Path:** A dropdown menu is set to "DELETE" with the path "localhost:8081/api/v1/customers/1". A blue "Send" button is to the right.
- Request Tab:** The "Params" tab is active, showing a table for "Query Params".
- Query Params Table:**

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		
- Response Tab:** The "Body" tab is active, showing the response status "Status: 200 OK", "Time: 221 ms", and "Size: 123 B". A "Save Response" button is present.
- Response Format:** The "Pretty" tab is selected, showing the response in a human-readable format.