

Endpoint Document

Note:

- `[foo]` : value corresponding to “foo” key in the request
- `:bar` : value of parameter named “bar”

Common Error Responses

1. Wrong data type

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Example of content:

```
{
  "errors": {
    "username": "Not a valid string."
  }
}
```

2. Wrong content-type

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

Content:

```
{  
  "msg": "Request is not JSON object."  
}
```

3. Wrong JSON format

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

Content:

```
{  
  "msg": "Failed to convert into JSON object."  
}
```

4. Missing required fields

- HTTP code: **400 Bad Request**
- Headers:

--	--

Key	Value
content-type	application/json

- Example of content:

```
{  
  "errors": {  
    "username": ["Missing data for required field."]  
  }  
}
```

5. Unauthorized

- HTTP code: **401 Unauthorized**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Missing Authorization Header"  
}
```

6. Wrong authorization header format

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Bad Authorization header. Expected value 'Bearer <JWT>'"  
}
```

1. POST `/login`

Submit user credentials to achieve JWT token

Request

- No parameter
- Headers:

Key	Value
content-type	application/json

- Example of content:

```
{  
  "username" : "nam_nguyen",  
  "password" : "n4mnguy3ntran"
```

```
}
```

Successful response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Example of content:

```
{
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJleHAiOjE1NTczOTcyMTQsImIhdCI6MTU1NzM5NjkxNCwibmJmIjoxNTU3Mzk20TE0LCJpZGVudGl0eSI6MX0.Xt49Z7ZqSkCWBD2eEXlUbNJRDeZoCKEFoIGuKJqU5xk"
}
```

Error responses

1. Wrong credentials

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
  "errors": {
    "username": "Invalid username."
  }
}
or
{
  "errors": {
    "password": "Invalid password."
  }
}
```

2. Wrong data type

3. Wrong content-type

4. Wrong JSON format

5. Missing required fields

2. POST /register

Create a new user with the provided information

Request

- No parameter
- Headers:

Key	Value
content-type	application/json

- Example of content:

```
{  
  "username" : "nam_nguyen",  
  "password" : "n4mnguy3ntran"  
}
```

Successful Response

- HTTP code: **201 Created**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Register successfully."  
}
```

Error Responses

1. Request content does not meet the data rules

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
  "errors": {
    "username": ["Username must contain 6 to 30 characters."]
  }
}
```

or

```
{
  "errors": {
    "username": ["Username must contain only lowercase letters, numbers."]
  }
}
```

- 2. Wrong data type**
- 3. Wrong content-type**
- 4. Wrong JSON format**

5. Missing required fields

3. GET

/categories/:category_id/items

Achieve all the items in a category with given `category_id`

Request

- Parameters:

Name	Type	Description	Example
<code>category_id</code>	Integer	Identifier of the category in which client wants to get all the items	4

- Headers:

Key	Value
authorization	Bearer <code>jwt_token</code>

- No content

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
-----	-------

content-type	application/json
--------------	------------------

- Example of response content:

```
[
  {
    "id": 3,
    "title": "Adidas Kampung",
    "description": "Adidas Kampung is a generic name for cheap black rubber shoes that are usually made in Malaysia.",
    "category_id": 1,
    "user_id": 2
  },
  {
    "id": 4,
    "title": "Ballet shoe",
    "description": "A ballet shoe, or ballet slipper, is a lightweight shoe designed specifically for ballet dancing.",
    "category_id": 1,
    "user_id": 2
  },
]
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Not Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
  "msg": "Can not find any category with id = :category_id"
}
```

2. Unauthorized

3. Wrong authorization header format

4. GET

/categories/:category_id/items/:item_id

Get item with id = **:item_id** in the category having id = **:category_id**

Request

- Parameters:

Name	Type	Description	Example
------	------	-------------	---------

<code>category_id</code>	Integer	Identifier of the category which item belongs to	4
<code>item_id</code>	Integer	Identifier of the item	3

- Headers:

Key	Value
authorization	Bearer <code>jwt_token</code>

- No content

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Example of response content:

{
"category_id": 4,
"name": "Ballet shoe",
"description": "A ballet shoe, or ballet slipper, is a lightweight shoe designed specifically for ballet dancing."
"category_id": "3",

```
"user_id": 4
```

```
}
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Nout Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find the category with id = :category_  
id"  
}
```

2. The item with given id does not exist in the category

- HTTP code: **404 Nout Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find the item with id = :item_id in the category"  
}
```

3. Unauthorized

4. Wrong authorization header format

5. POST

`/categories/:category_id/items`

Create an item in a category, the id of which is equal to `category_id`

Request

- Parameters:

Name	Type	Description	Example
<code>category_id</code>	Integer	Identifier of the category which item belongs to	4

- Headers:

Key	Value
authorization	Bearer <code>jwt_token</code>
content-type	application/json

- Example of content:

```
{  
  "title": "Ballet shoe",  
  "description": "A ballet shoe, or ballet slipper, is a  
lightweight shoe designed specifically for ballet dancing  
."  
}
```

Successful Response

- HTTP code: **201 Created**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "The item was created"  
}
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Not Found**

- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find any category with id = :category_  
id."  
}
```

2. Request content does not meet the data rules

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "errors": {  
    "title": ["Title must contain 6 to 30 characters."  
]  
  }  
}
```


or

```
{
  "errors": {
    "title": ["Title must contain only lowercase letters, numbers."]
  }
}
```

or

```
{
  "errors": {
    "title": ["The item title already exists."]
  }
}
```

or

```
{
  "errors": {
    "description": ["Description must not be empty."]
  }
}
```

3. Wrong data type

- 4. Wrong content-type
- 5. Wrong JSON format
- 6. Missing required fields
- 7. Unauthorized
- 8. Wrong authorization header format

6. PUT

/categories/:category_id/items/:item_id

Update item with id = **:item_id** in the category having id = **:category_id**

Request

- Parameters:

Name	Type	Description	Example
category_id	Integer	Identifier of the category which item belongs to	4
item_id	Integer	Identifier of the item	3

- Headers:

Key	Value
authorization	Bearer jwt_token
content-type	application/json

- Example of content:

```
{  
  "title": "Ballet shoe",  
  "description": "A ballet shoe, or ballet slipper, is a  
lightweight shoe designed specifically for ballet dancing  
."  
}
```

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "The item was updated"  
}
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Not Found**

- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find any category with id = :category_  
id."  
}
```

2. The item with given id does not exist in the category

- HTTP code: **404 Not Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find the item with id = :item_id in th  
e category"  
}
```

3. Request content does not meet the data rules

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
  "errors": {
    "title": ["Title must contain 6 to 30 characters."]
  }
}
```

or

```
{
  "errors": {
    "title": ["Title must contain only lowercase letters, numbers."]
  }
}
```

or

```
{
  "errors": {
    "title": ["The item title already exists."]
  }
}
```

or

```
{
  "errors": {
    "description": ["Description must not be empty."]
  }
}
```

4. Wrong data type

5. Wrong content-type

6. Wrong JSON format

7. Missing required fields

8. Unauthorized

9. Wrong authorization header format

7. DELETE

`/categories/:category_id/items/:item_id`

Delete item with id = `:item_id` in the category having id =

`:category_id`

- Parameters:

Name	Type	Description	Example
<code>category_id</code>	Integer	Identifier of the category which item belongs to	4
<code>item_id</code>	Integer	Identifier of the item	3

- Headers:

Key	Value
authorization	Bearer <code>jwt_token</code>

- No content

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "The item was deleted"  
}
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Nout Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find any category with id = :category_  
id."  
}
```

2. The item with given id does not exist in the category

- HTTP code: **404 Nout Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
```



```
"msg": "Can not find the item with id = :item_id in the category"
}
```

3. Unauthorized

4. Wrong authorization header format

8. GET `/categories`

Achieve all the categories

Request

- No parameter
- Headers:

Key	Value
authorization	Bearer <code>jwt_token</code>

- No content

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Example of response content:

```
[
  {
    "id": 5,
    "name": "Sport"
  },
  {
    "id": 6,
    "name": "Style"
  },
]
```

Error Responses

1. Unauthorized

2. Wrong authorization header format

9. GET `/categories/:category_id`

Get category with id = `:category_id`

Request

- Parameters:

Name	Type	Description	Example
<code>category_id</code>	Integer	Identifier of the category	5

- Headers:

Key	Value
authorization	Bearer <code>jwt_token</code>

- No content

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Example of response content:

```
{  
  "id": 5,  
  "name": "Sport"  
}
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Not Found**
- Headers:

--	--

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find the category with id = :category_  
id"  
}
```

2. Unauthorized

3. Wrong authorization header format

10. POST **/categories**

Create a new category

Request

- No parameter
- Headers:

Key	Value
authorization	Bearer jwt_token
content-type	application/json

- Example of content:

```
{  
  "name": "Sport"  
}
```

Successful Response

- HTTP code: **201 Created**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "The category was created"  
}
```

Error Responses

1. Request content does not meet the data rules

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
  "errors": {
    "name": ["Title must contain 6 to 30 characters."]
  }
}
```

or

```
{
  "errors": {
    "name": ["Title must contain only lowercase letters, numbers."]
  }
}
```

or

```
{
  "errors": {
    "name": ["The category name already exists."]
  }
}
```

2. Wrong data type

3. Wrong content-type

4. Wrong JSON format

5. Missing required fields

6. Unauthorized

7. Wrong authorization header format

11. PUT

/categories/:category_id

Update item with id = :category_id

Request

- Parameters:

Name	Type	Description	Example
category_id	Integer	Identifier of the category	4

- Headers:

Key	Value
authorization	Bearer jwt_token
content-type	application/json

- Example of content:

```
{  
  "name": "Sport facilities"  
}
```

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "The category was updated"  
}
```

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Not Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
```



```
"msg": "Can not find any category with id = :category_
id."
}
```

2. Request content does not meet the data rules

- HTTP code: **400 Bad Request**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{
  "errors": {
    "name": ["Title must contain 6 to 30 characters."]
  }
}
```

or

```
{
  "errors": {
    "name": ["Title must contain only lowercase letter
s, numbers."]
  }
}
```

or

```
{
  "errors": {
    "name": ["The category name already exists."]
  }
}
```

3. Wrong data type

4. Wrong content-type

5. Wrong JSON format

6. Missing required fields

7. Unauthorized

8. Wrong authorization header format

12. DELETE

/categories/:category_id

Delete item with id = **:category_id**

- Parameters:

Name	Type	Description	Example
category_id	Integer	Identifier of the category	4

- Headers:

--	--

Key	Value
authorization	Bearer <code>jwt_token</code>

- No content

Successful Response

- HTTP code: **200 OK**
- Headers:

Key	Value
content-type	application/json

- Content:

{
<code>"msg": "The category was deleted"</code>
}

Error Responses

1. The category with given id does not exist

- HTTP code: **404 Nout Found**
- Headers:

Key	Value
content-type	application/json

- Content:

```
{  
  "msg": "Can not find any category with id = :category_  
id."  
}
```

2. Unauthorized

3. Wrong authorization header format