

# Ensek API Testing Feedback

Below I have collated my thoughts on the API endpoints provided as well as any feedback I might have to the architect/developer to improve the overall useability and quality of the API.

## Issue 1 - Ordering more than available on Buy endpoint

PUT

/ENSEK/buy/{id}/{quantity}

put\_buy

Use to purchase energy units.

Parameters

Cancel

Name	Description
<b>id</b> * required integer (path)	Energy type identifier.
<input type="text" value="1"/>	
<b>quantity</b> * required integer (path)	Quantity to purchase.
<input type="text" value="49999999"/>	

Execute

Clear

Responses

Response content type application/json

Curl

curl -X 'PUT' \nhttps://qa candidatetest.ensek.io/ENSEK/buy/1/49999999' \n-H 'accept: application/json'

Request URL

https://qa candidatetest.ensek.io/ENSEK/buy/1/49999999

Server response

Code

Details

200

Response body

{\n "message": "You have purchased -7996994 m³ at a cost of 16999999.66 there are 49999999 units remaining. Your order id is 0004a548-3123-4f95-bc68-b64cd388305b."\n}

Download

Response headers

access-control-allow-methods: OPTIONS,PUT\naccess-control-allow-origin: https://qa candidatetest.ensek.io\ncontent-length: 167\ncontent-type: application/json\ndate: Tue, 12 Nov 2024 22:15:09 GMT\nserver: nginx/1.18.0 (Ubuntu\nvary: Origin

Request duration

44 ms

Ordering a quantity over the current amount results in broken data. This should be caught in validation for this endpoint only allowing a user to purchase no more than the remaining quantity available.

## Issue 2 - Incorrect data structure returned by Buy endpoint

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

- Responses:** A dropdown menu set to `application/json`.
- Curl:**

```
curl -X 'PUT' \
  'https://qa candidatetest.ensek.io/ENSEK/buy/1/1' \
  -H 'accept: application/json'
```
- Request URL:** `https://qa candidatetest.ensek.io/ENSEK/buy/1/1`
- Server response:**
  - Code:** 200
  - Response body:**

```
{
  "message": "You have purchased 2999 m³ at a cost of 0.34 there are 1 units remaining. Your order id is b7980dab-7853-467c-8f4b-7a0ee2a561f5."
}
```
  - Response headers:**

```
access-control-allow-methods: OPTIONS,PUT
access-control-allow-origin: https://qa candidatetest.ensek.io
content-length: 148
content-type: application/json
date: Wed, 13 Nov 2024 01:18:09 GMT
server: nginx/1.18.0 (Ubuntu)
vary: Origin
```
  - Request duration:** 188 ms

The JSON object returned by the buy endpoint should in my opinion return two fields. The message for the front end to use for the client and the ID for the front end to utilise on the rest of the site. For example, viewing orders, removing orders, and any other endpoints that might require the order ID.

## Issue 3 - Incorrect response on Buy endpoint for ordering Energy Type with no stock left

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

- Code:** 200
- Response body:**

```
{
  "message": "There is no nuclear fuel to purchase!"
}
```
- Response headers:**

```
access-control-allow-methods: OPTIONS,PUT
access-control-allow-origin: https://qa candidatetest.ensek.io
content-length: 52
content-type: application/json
date: Tue, 12 Nov 2024 23:48:14 GMT
server: nginx/1.18.0 (Ubuntu)
vary: Origin
```
- Request duration:** 45 ms

When the user tries to order an energy that has no stock left an error code of 400 should be returned as the order cannot be processed.

### Issue 4 - Delete endpoint returns 500 error for valid deletion request

Responses

Response content typeapplication/json

Curl

```
curl -X 'DELETE' \
'https://qa candidatetest.ensek.io/ENSEK/orders/2cdd6f69-95df-437e-b4d3-e772472db8de' \
-H 'accept: application/json' \
-H 'Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiIsInR5cCI6ImlhdCI6MTczMTQ1OTAwCiwianRpIjo1Y2UxZTJkYzgtYmQyMC00MwZlTkzjMmMtMmQNDVJhTUozRzcziIiwidHlwZSI6ImFjcyIiwiaWF0IjoiNjE1IiwiaW1rIjoiQ01CLCJ9'

```

Request URL

https://qa candidatetest.ensek.io/ENSEK/orders/2cdd6f69-95df-437e-b4d3-e772472db8de

Server response

Code

Details

500

Undocumented Error:

Response body

```
{
  "message": "Internal Server Error"
}
```

Download

Response headers

```
access-control-allow-origin: https://qa candidatetest.ensek.io
content-length: 37
content-type: application/json
date: Wed, 12 Nov 2024 01:04:14 GMT
server: nginx/1.18.0 (Ubuntu)
vary: Origin

```

Responses

Code

Description

200

Success

The test I have written for the delete endpoint within my API testing suite currently fails due to the endpoint not working correctly. In the screenshot above I have taken an order ID from the orders endpoint and attempted to delete it, using this endpoint results in an error 500 response. The test I have written will pass once this functionality is introduced.

### **Issue 5 - Put order endpoint creates duplicate/invalid data**

PUT

/ENSEK/orders/{orderId}

put\_orders

Parameters

Name

Description

payload

required

object

(body)

Edit Value

Model

```
{
  "id": "",
  "quantity": 0,
  "energy_id": 0
}
```

Cancel

Parameter content type

application/json

orderid

required

string

(path)

orderid

Execute

This PUT endpoint currently does not function correctly, when a request is sent it takes two IDs when it should only take one as either a parameter or within the JSON. When the endpoint is run it duplicates the order you are trying to modify, creating one with a field that should not exist.

## Issue 6 - Test data generation method creates testing interdependency

Parameters

Cancel

Name	Description
<b>orderid</b> <small>* required</small>	Order id
string (path)	
	31fc32da-bccb-44ab-9352-4f43fc44ed4b

Execute

Clear

Responses

Response content type application/json

Curl

```
curl -X 'GET' \
  'https://qacandidatetest.ensek.io/ENSEK/orders/31fc32da-bccb-44ab-9352-4f43fc44ed4b' \
  -H 'accept: application/json'
```

Request URL

https://qacandidatetest.ensek.io/ENSEK/orders/31fc32da-bccb-44ab-9352-4f43fc44ed4b

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{   "fuel": "gas",   "id": "31fc32da-bccb-44ab-9352-4f43fc44ed4b",   "quantity": 1,   "time": "Thu, 10 Mar 2022 09:23:24 GMT" }</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>access-control-allow-methods: OPTIONS,GET access-control-allow-origin: * content-length: 111 content-type: application/json date: Wed, 13 Nov 2024 01:12:53 GMT server: nginx/1.18.0 (Ubuntu)</pre></div><div><div>Request duration</div><div>44 ms</div></div></div>

Responses

Code	Description
200	Success

The get order endpoint is not testable unless the IDs created in the test data are consistent. Because the IDs are generated randomly on every test data reset you have to call the get all orders endpoint to get the currently inserted GUIDs, this created an interdependence on endpoints and therefore is an invalid test.

## **Other General Suggestions and Observations**

- All endpoints that involve editing and creating orders should require an Authentication token.
- A JSON array for energy data type endpoint should be used to simplify deserialization.
- The PUT Order endpoint should not take an ID as a parameter (or within the JSON) the ID should be created in the controller for the endpoint to simplify the request.
- Errors are not caught on the PUT Orders endpoint when an invalid request is sent (500 is returned).
- In the task I was told to confirm how many orders were created before the current date, I don't understand this question as the number of orders made before the current date will always be the same as returned by the orders endpoint.