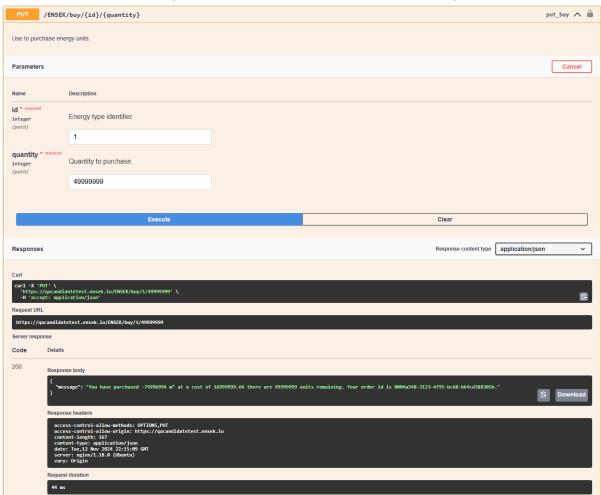
Ensek API Testing Feedback

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



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



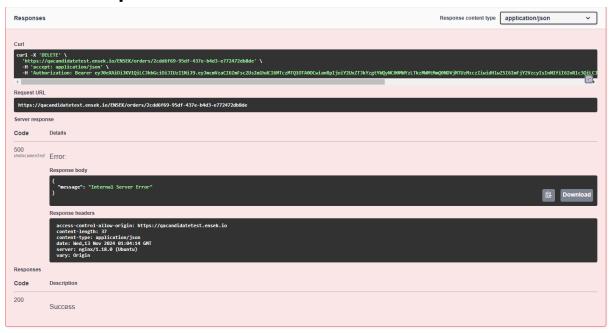
The JSON object returned by the buy endpoint should in my opinion return two fields. The message for the front end to use for the client and the ID for the front end to utilise on the rest of the site. For example, viewing orders, removing orders, and any other endpoints that might require the order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

<u>Issue 3 - Incorrect response on Buy endpoint for ordering</u> <u>Energy Type with no stock left</u>



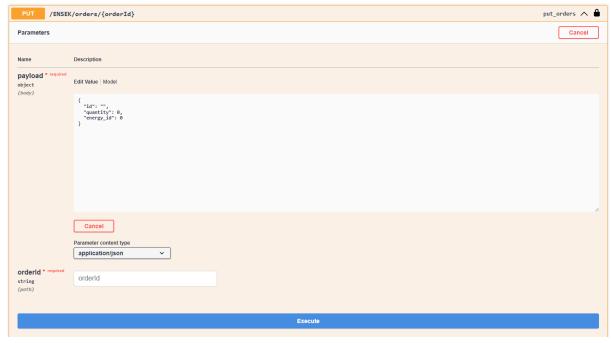
When the user tries to order an energy that has no stock left an error code of 400 should be returned as the order cannot be processed.

<u>Issue 4 - Delete endpoint returns 500 error for valid</u> <u>deletion request</u>



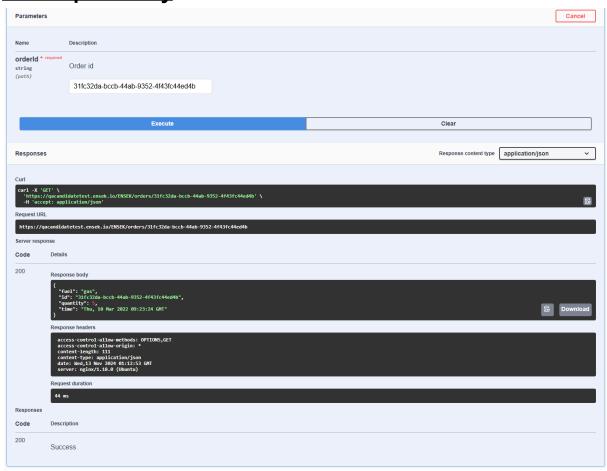
The test I have written for the delete endpoint within my API testing suite currently fails due to the endpoint not working correctly. In the screenshot above I have taken an order ID from the orders endpoint and attempted to delete it, using this endpoint results in an error 500 response. The test I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

<u>Issue 5 - Put order endpoint creates duplicate/invalid data</u>



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.

<u>Issue 6 - Test data generation method creates testing</u> <u>interdependency</u>



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.