

Technical Support Specialist Assessment

Section 1: Non technical

The first step I would take is to gather all the necessary information about the issues Codak Ventures is experiencing with their integration. This would involve working with the dev team to understand the root cause of the errors and develop a plan to address them.

Next, I would reach out to Codak Ventures and communicate with them in a professional and empathetic manner. I would apologize for the inconvenience caused by the errors and assure them that we are working to resolve the issues as quickly as possible.

I would also provide them with regular updates on the progress being made to fix the errors and give them an estimated time frame for when the issues will be resolved. This would help manage their expectations and ensure that they are kept informed throughout the process.

In terms of managing the expectations of Peter, I would provide him with regular updates on the progress being made to resolve the issues and assure him that we are doing everything possible to prevent the merchant from churning. I would also work with him to develop a plan to prevent similar issues from happening in the future and improve the onboarding process for new merchants.

Furthermore, I would work closely with the customer success and sales teams to ensure that Codak Ventures receives the necessary support and attention they need. This would involve taking proactive measures to keep the customer engaged and satisfied with our product or service.

The key to handling this situation would be my effective communication, transparency, and collaboration between all the stakeholders involved.

Section 2: Semi technical

1.

Hi Jesse,

I'm sorry to hear that you're experiencing issues with retrieving the latest transactions using the "<https://api.okra.ng/v2/transactions/getById>" endpoint. I would be happy to help you with this issue.

Please note that the "[getById](#)" endpoint is specifically used to retrieve a specific transaction by its ID, rather than a range of transactions by date. If you're looking to retrieve the latest transactions, you may want to use the "[getByDate](#)" endpoint instead, located at

"<https://api.okra.ng/v2/transactions/getByDate>", and specify a date range that covers the period you're interested in.

Additionally, can you please confirm if you are passing in the correct transaction ID when making the API request? It's possible that the transaction you are trying to retrieve is actually a week old, and that's why you're only getting transactions from a week ago.

If you're still experiencing issues after verifying the above, please let me know and we can investigate further. Thank you for reaching out to Okra support, and I look forward to helping you resolve this issue.

Best regards,
Jenomh Kurah

2.

Dear Mike,

Thank you for reaching out to us. To indicate programmatically whether the connection of the account was successful or not, you can utilize the callbacks that are available in our API. You can implement the "onSuccess" and "onError" functions to handle the different events that can happen during the widget flow.

Here is the JavaScript code snippet of how you can use the "onSuccess" function to indicate that the connection was successful:

```
import Okra from 'npm-okrajs';

const OkraHandler = Okra.buildWithOptions({
  ...,
  onSuccess: (data) => {
    console.log('You have linked an account!', data)
  }
});
```

You can also use the "onError" function below to indicate the connection was unsuccessful:

```
import Okra from 'npm-okrajs';

const OkraHandler = Okra.buildWithOptions({
  ...,
```

```
onError: (json, error) => {  
  console.log('Connection unsuccessful!', json)  
}  
});
```

You can find more information on available callbacks in the Okra Widget and their functionalities in the Okra documentation at this link: <https://docs.okra.ng/docs/widget-options>.

I hope this helps. Let me know if you have any further questions or concerns.

Best regards,
Jenomh Kurah

3.

Dear Blessing,

Thank you for your interest in Okra! I would be glad to assist you with your testing. Our STARTERS plan is ideal for startups, building and testing out new products, and it is completely free. You will receive 100 free API calls, unlimited test credentials, developer support, widget design customization, and real-time Slack notifications. This is a great way to test our product and its functionality without incurring any costs.

If you would like to learn more about the different plans we offer, including the free STARTERS plan, please visit our pricing page at <https://okra.ng/pricing>.

Please let me know if you have any further questions or if there is anything else I can assist you with.

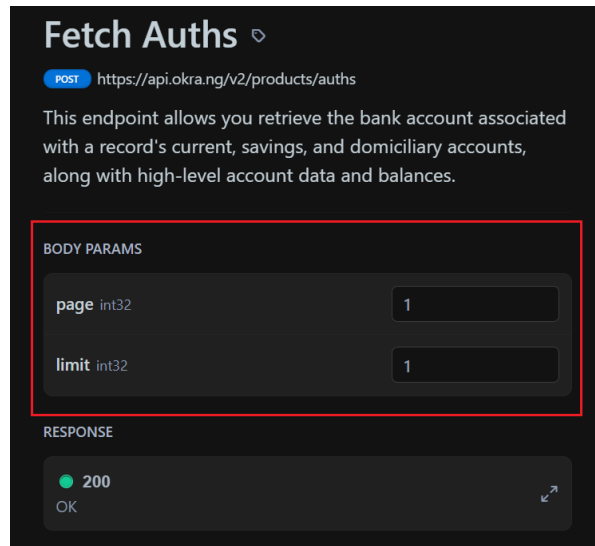
Best regards,
Jenomh Kurah

Section 3: Technical

Problem 1:

The issue is that the customer is not receiving the expected response from the *"fetchauths"* endpoint when sending a POST request with the provided payload containing only the *"page"* parameter.

To resolve this issue, the customer would have to review the documentation for the “*fetchauths*” endpoint and ensure that they are sending a valid request payload with both the “*page*” and “*limit*” parameters.

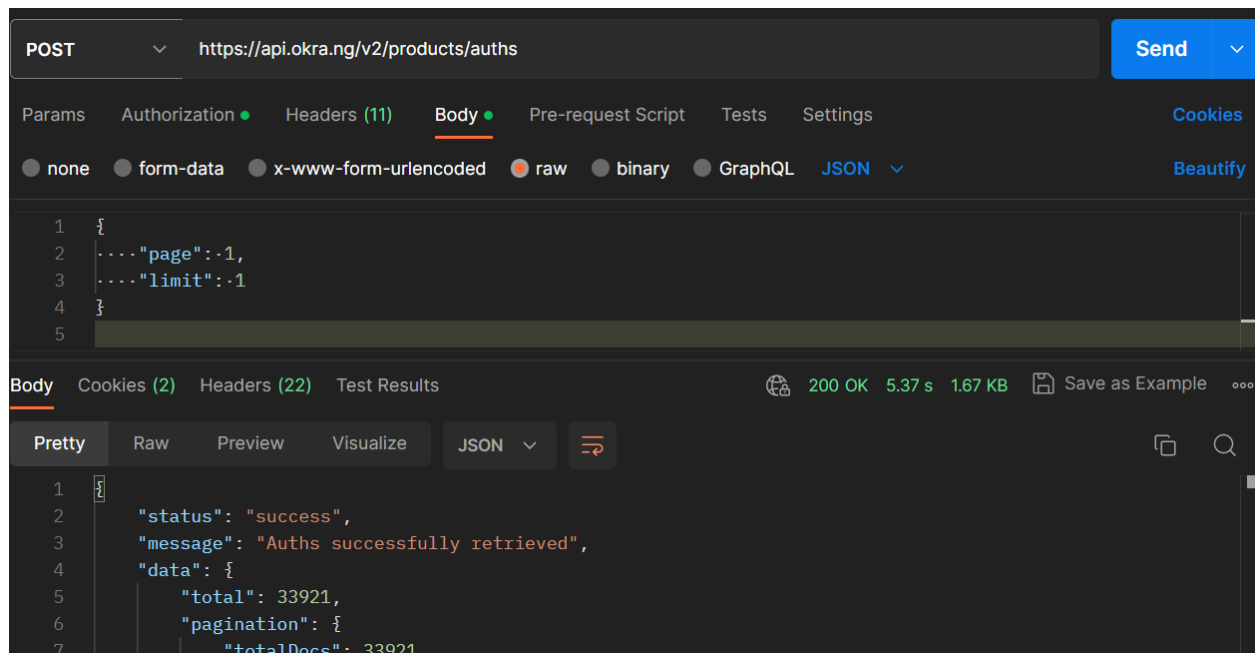


Pic 1. Shows The required parameters of the endpoint from the API Document

Therefore, the payload should be in the following format:

```
{  
  "page": 1,  
  "limit": 1  
}
```

The result will be as shown in the image below:



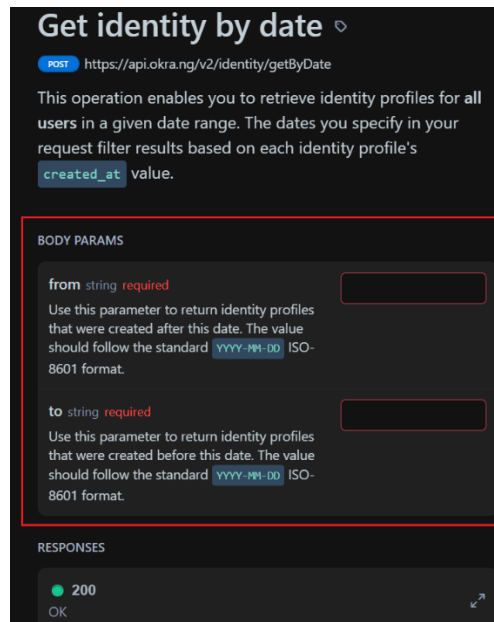
Pic 2. Status shows Success

Also, the customer should ensure that they are including the required authentication token in the header of their request to the API. Once these steps are taken, the customer should receive the expected response from the *“fetchauths”* endpoint.

Problem 2

The issue with the customer's request is that they are not providing the required parameters for the *“getByDate”* endpoint. According to the documentation, the *“getByDate”* endpoint requires the *“from”* and *“to”* parameters to be specified in the request body. However, in the customer's payload, only the *“page”* parameter is provided.

To resolve this issue, the customer should update their payload to include the *“from”* and *“to”* parameters with valid values in the YYYY-MM-DD ISO-8601 format.

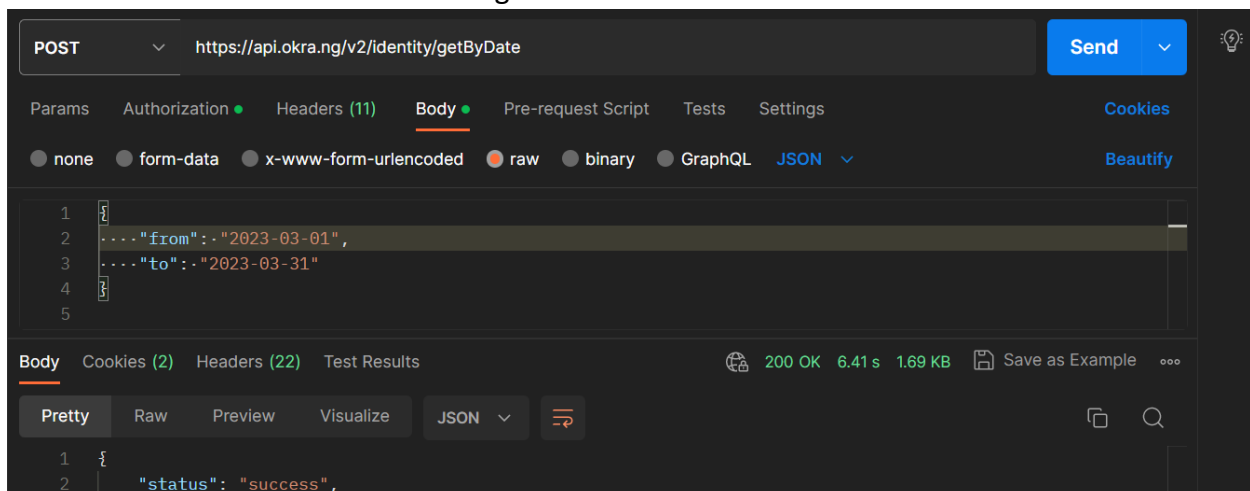


Pic 3. Shows The required parameters/format of the endpoint from the API Document

For example, the payload should be in the following format:

```
{
  "from": "2023-03-01",
  "to": "2023-03-31"
}
```

The result will be shown as in the image below:



Pic 4. Status shows success