Test Plan for Testing an API for Toco Transactions

1. **Test objectives**

We have a RESTful API that allows our users to buy and sell Tocos, our currency backed by carbon mitigation assets, and make transactions with other users. The API has several endpoints including:

POST /users: Create a new user
POST /buy: Buy Tocos using fiat currency, and add them to a user's account
POST /sell: Sell Tocos and convert them back into fiat currency
POST /transactions: Make a transaction from one user's account to another
GET /transactions/{userId}: Get the transaction history for a user

The test objectives:
- to confirm that users can be created
- to confirm that users are able to buy and add tocos to the Tocos wallet
- to confirm that users are able to sell tocos
- to confirm transactions between accounts are possible
- to confirm there is correct transaction history from the user
- to confirm edge and negative cases are included and tested in the Test Plan

2. **Test scope**

API features included in the testing scope:

- Users
- Transactions
- Buy/sell functions

Environments where testing should be conducted:

- Staging
- Production

3. **Testing approach**

For testing an API revolving around transactions we will divide test cases into chapters that represent features to be tested.

For the automation framework, we are going to use a page-object approach and divide API testing files by the endpoints we are going to test, that way we have a clear structure and overview of the tests.

Test cases created should cover the basic functions and possible problems like malformed data or throttled networks.

Regarding the Bug Report, after the framework was approved by the other team members, the automation framework can be run through a pipeline so that in case there are errors we would be notified about the pipeline failing and we can run the tests ourselves to debug where the issue is.

4. Resources needed

Technical resources:
- Available API
- Environment to conduct testing on
- Test case document
- Testing framework

Staff needed:
- QA engineer to create the test cases and automation framework
- Developer/Product Owner to review the automation framework

5. Test cases document

The test case document can be found here