DeFi Options Trading

Team S-K

XIAO, Li	LAKHANI, Harsh Sunil	DONG, Wenyu
20925438	20910249	20930158
lxiaoae@connect.ust.hk	hslakhani@connect.ust.hk	wdongai@connect.ust.hk

1 Problem Statement

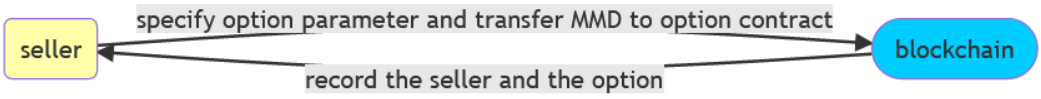
DeFi options are a low-barrier borderless instrument that can be easily traded among peers on the platform without involving brokers, brokerage or any third-party. In the recent years, DeFi option trading has gained the attention of traders due to the nature of option contracts being a tool for risk aversion and being highly liquid, especially among hedgers and speculators. In general, there are two types of option contracts: call and put. A call option allows the user to buy a specified amount of the underlying asset at a fixed price within a specific period of time, which is also the main theoretical basis of this paper. Conversely, a put option gives the holder the right to sell the underlying asset at a specified price on or before the expiration date. In this project, we establish an on-chain peer to peer option trading protocol built on MMD and cMMD(mock).

2 Design Process

2.1 Backend

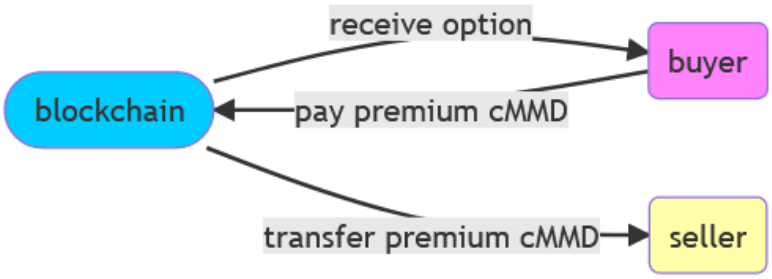
The stablecoin, unstablecoin and option contract are stored in the backend and serve as the three main contracts for the protocol. The former two contracts extend **ERC20**, while the latter performs the primary role of implementing transactions.

2.1.1 Write Option



In the write option function, the seller specifies the option parameters that will be stored on the blockchain, including name of the coin, strike price, premium, expiry date and token amount. After which, the blockchain correspondingly records and stores the option parameter information.

2.1.2 Buy Option



After the seller successfully uploads the option contract on the blockchain, the buyer is able to browse a list of available options and they are also able to buy the option contract, which requires them to pay the premium in cMMD. If the buyer successfully purchases the option contract, the data on the blockchain will be updated and the premium will be transferred to the seller.

2.1.3 Exercise Option

Within the expiry date, the buyer is able to exercise the option, i.e., pays seller the strike price with cMMD using the blockchain as a hub. After which, the buyer will receive the allocated MMD from the blockchain.

2.1.4 Cancel Option

The functions of cancellation and retrieval are the process of active or passive cancellation of option transactions by the seller, wherein the seller will get the allocated MMD tokens back and the status of option will be changed to cancelled and will not be available to buy. Although this function has been implemented in the backend, the function isn't implemented in the frontend.

2.2 Frontend

It is important to establish a connection between the portal and the MetaMask wallet, from where the stablecoin and unstablecoin are acquired. From the perspective of the user, there is a landing page describing the portal along with two buttons to redirect them to the

appropriate page depending on whether the user is a buyer or seller. On the page for the seller, the user is able to type in the parameter information for the option contract option they desire to sell. In contrast, on the BEC page, the user is able to view the options available to be bought or exercised.

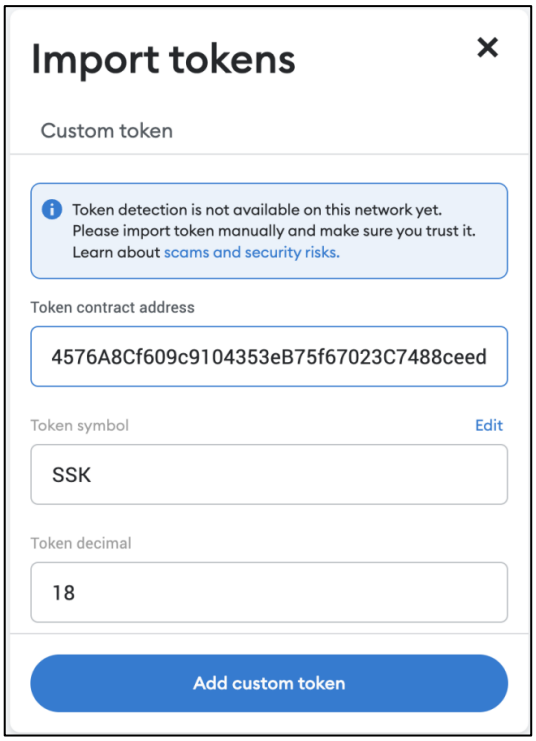
3 Final Result

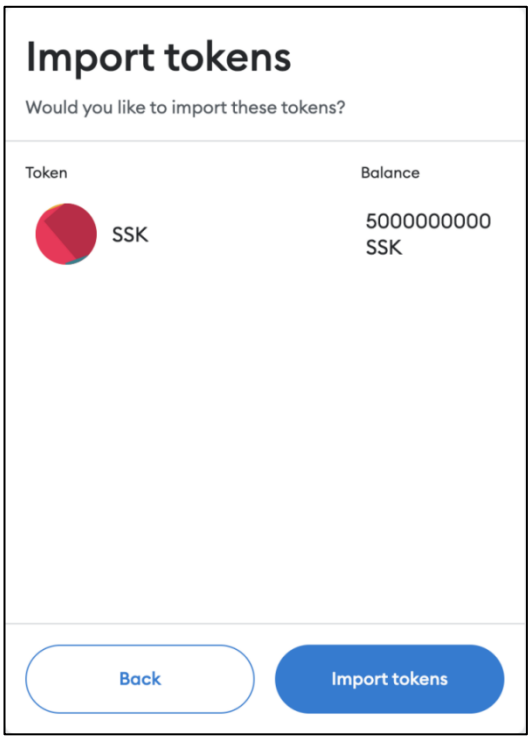
We implemented the DeFi options trading protocol based on ERC20 and it can be used for actual transactions. Categorized by back-end contract functions, this section will start with the sequence of sellers selling options, buyers buying the options, and finally buyers exercising their options.

3.1 Deploy Smart Contract

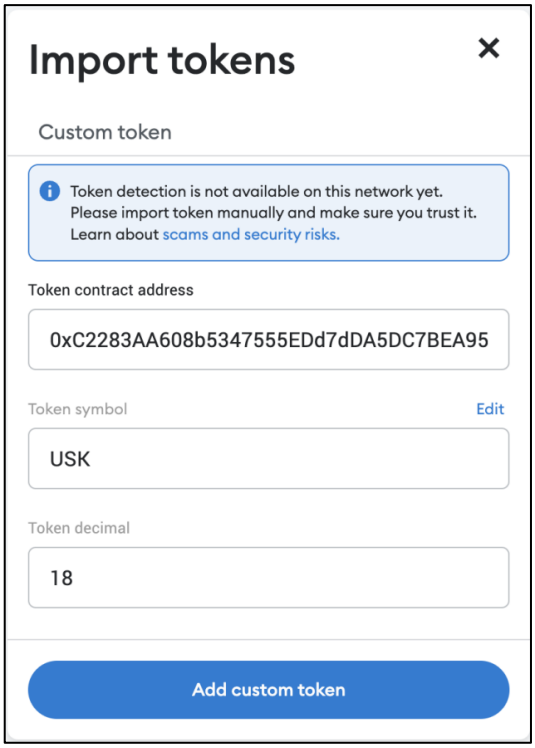
3.2 Stablecoin and unstablecoin contract

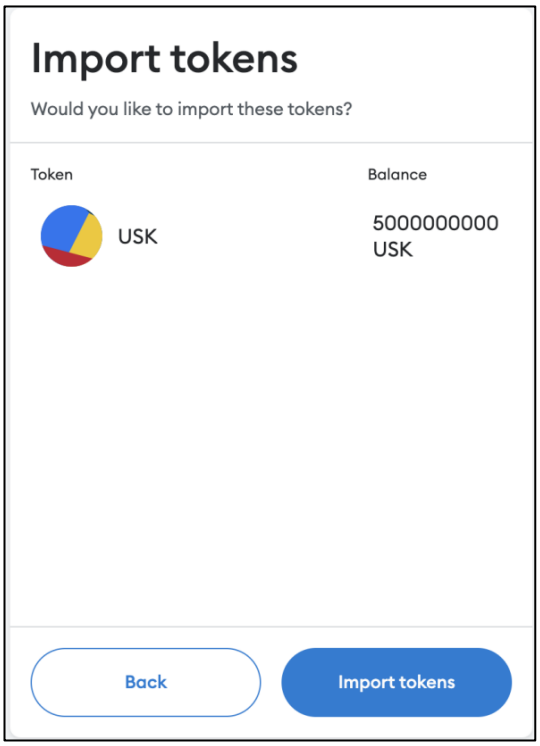
Import defined stablecoin(SSK).



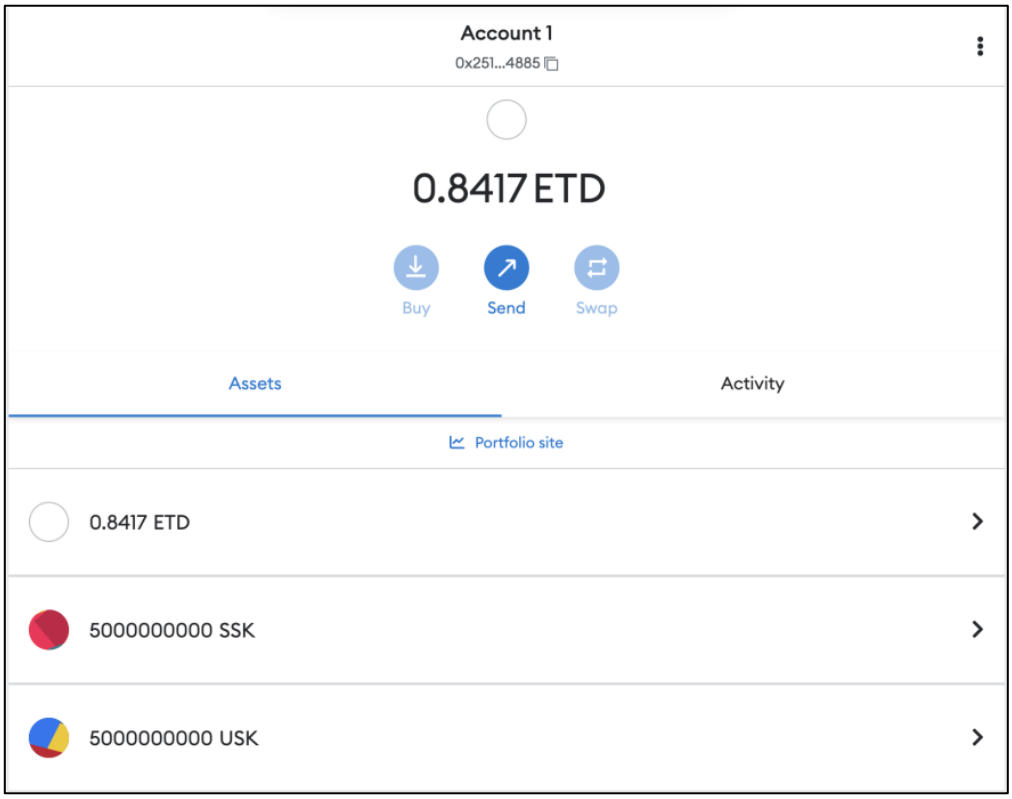


Import defined unstablecoin(USK).



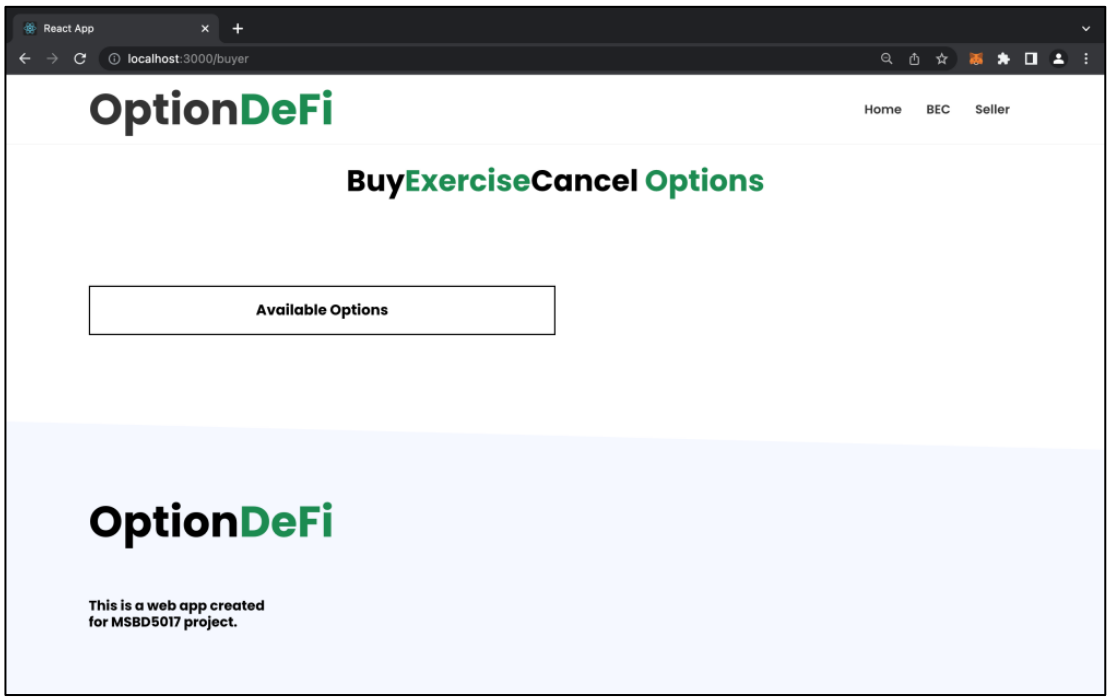


View the coins present in the wallet.

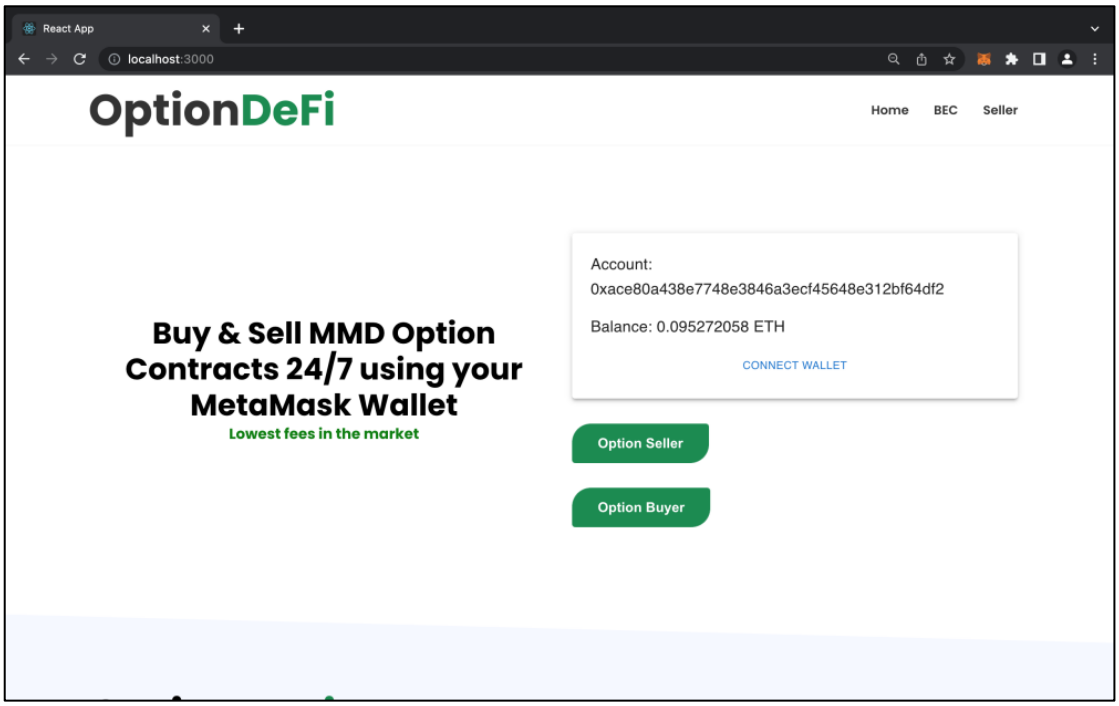

Account 1
0x251...4885
0.8417 ETD
Buy
Send
Swap
Assets
Activity
Portfolio site
0.8417 ETD
5000000000 SSK
5000000000 USK

3.3 Option Contract

When we click on the BEC page, there is currently no option contract on the blockchain.

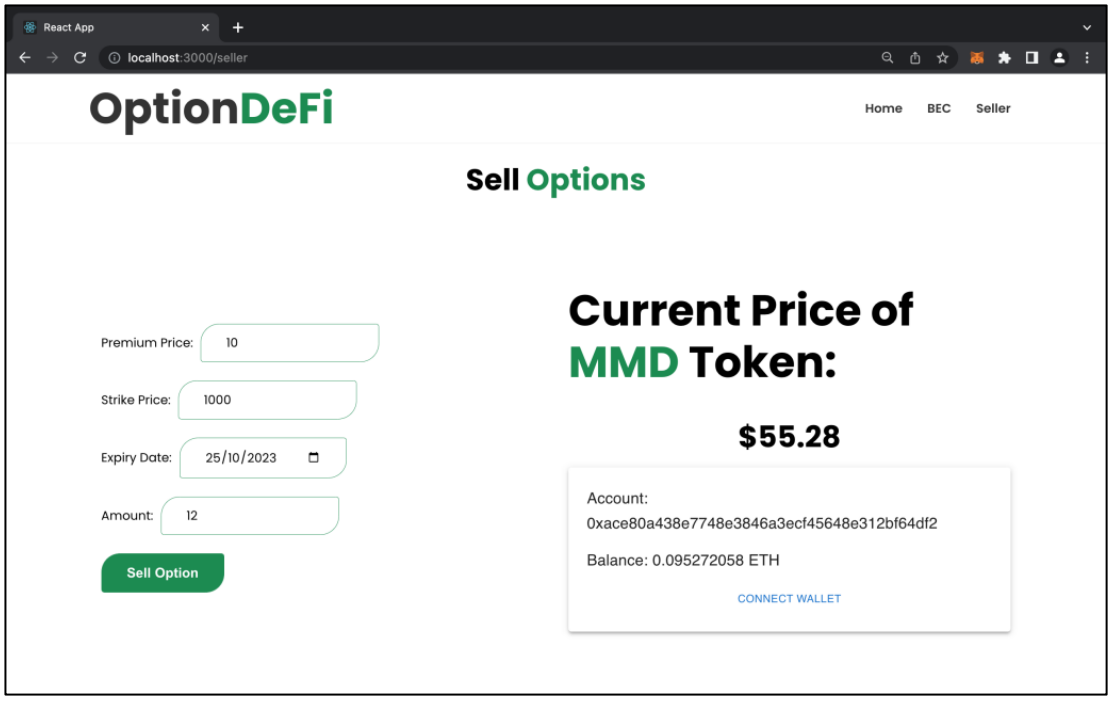


Click the “connect wallet” button to establish a connection between the portal and the MetaMask wallet extension and view balance, which calls the “*connect wallet”* function of front-end.

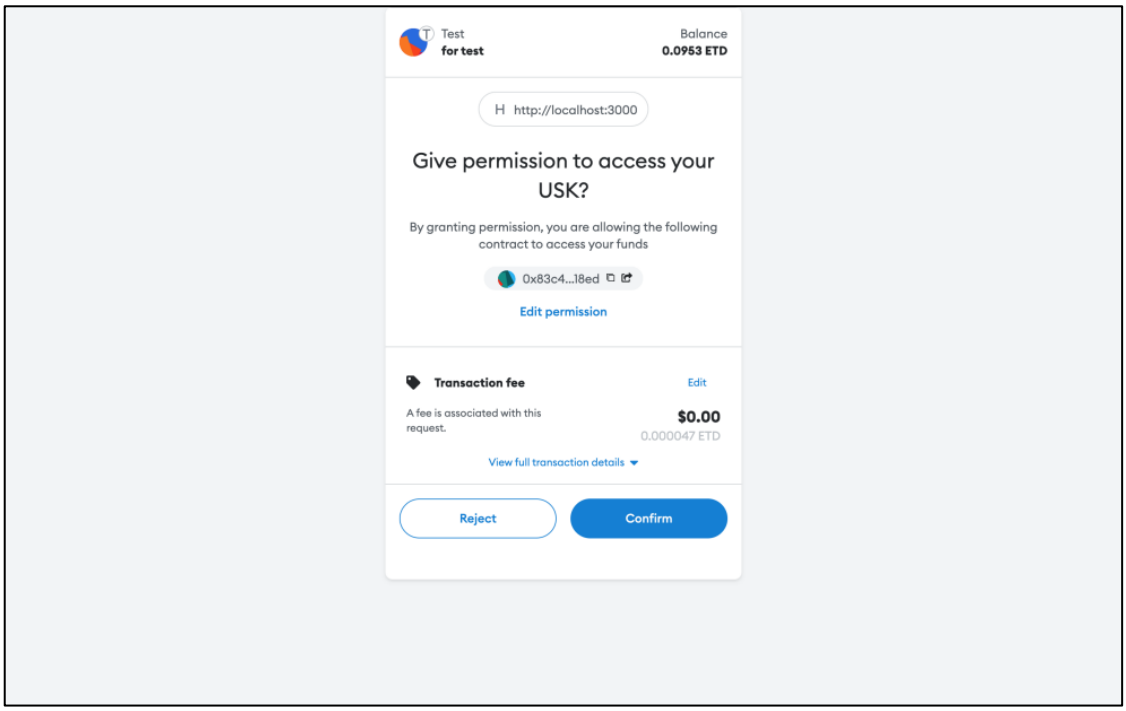


3.3.1 Sell Option

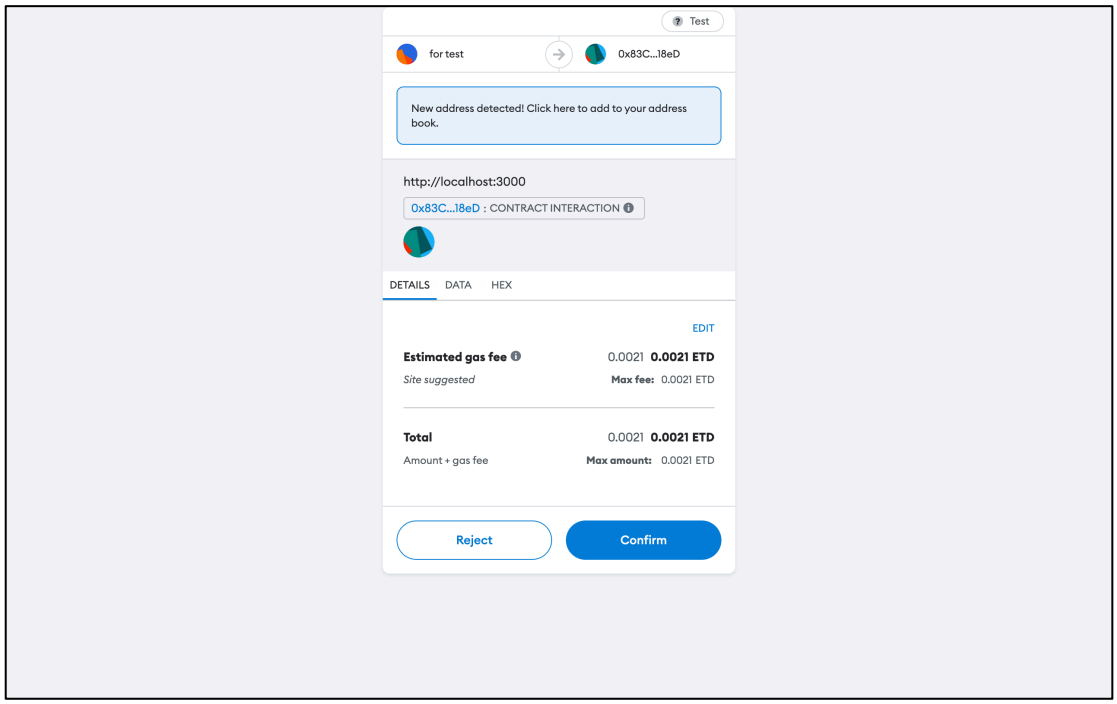
Writer specifies parameters of option which includes premium, strike price, expiry date and amount.



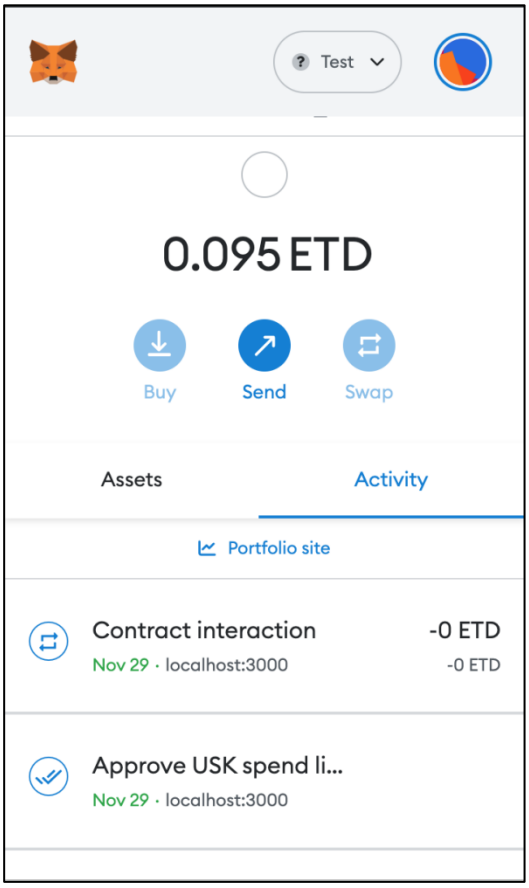
The user is prompted to approve unstablecoin from the MetaMask wallet to contract, using "*approve*" function from smart contract, which extends **ERC20.**



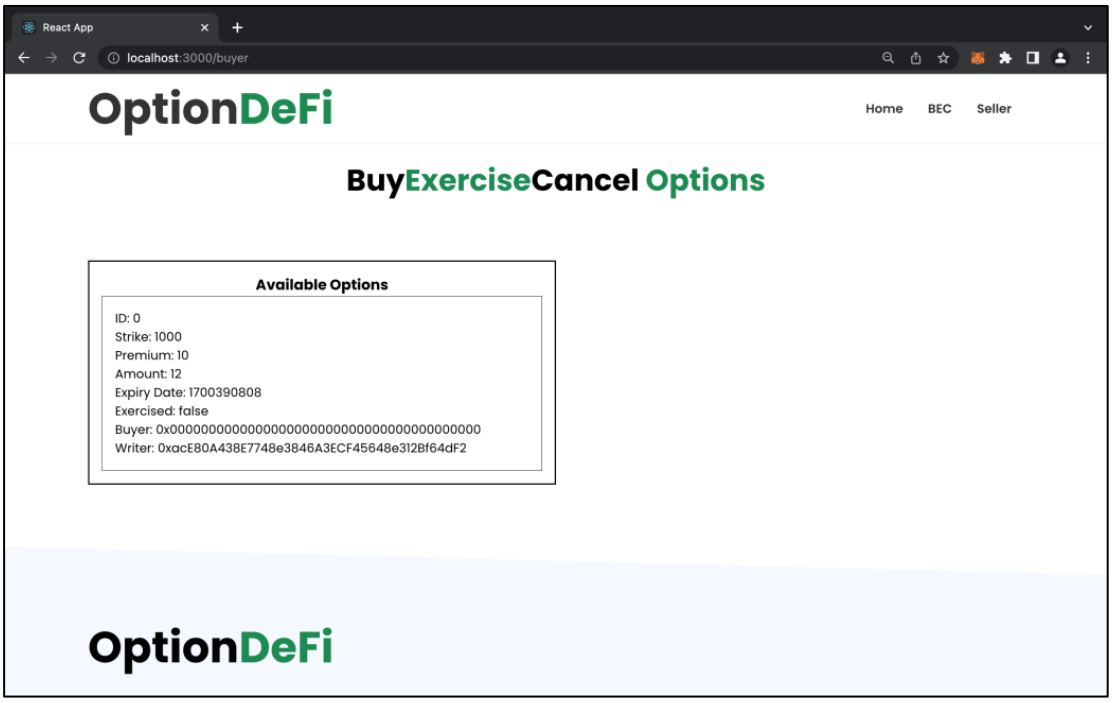
The user is then prompted to confirm the transaction to transfer the unstablecoin from writer to option contract.



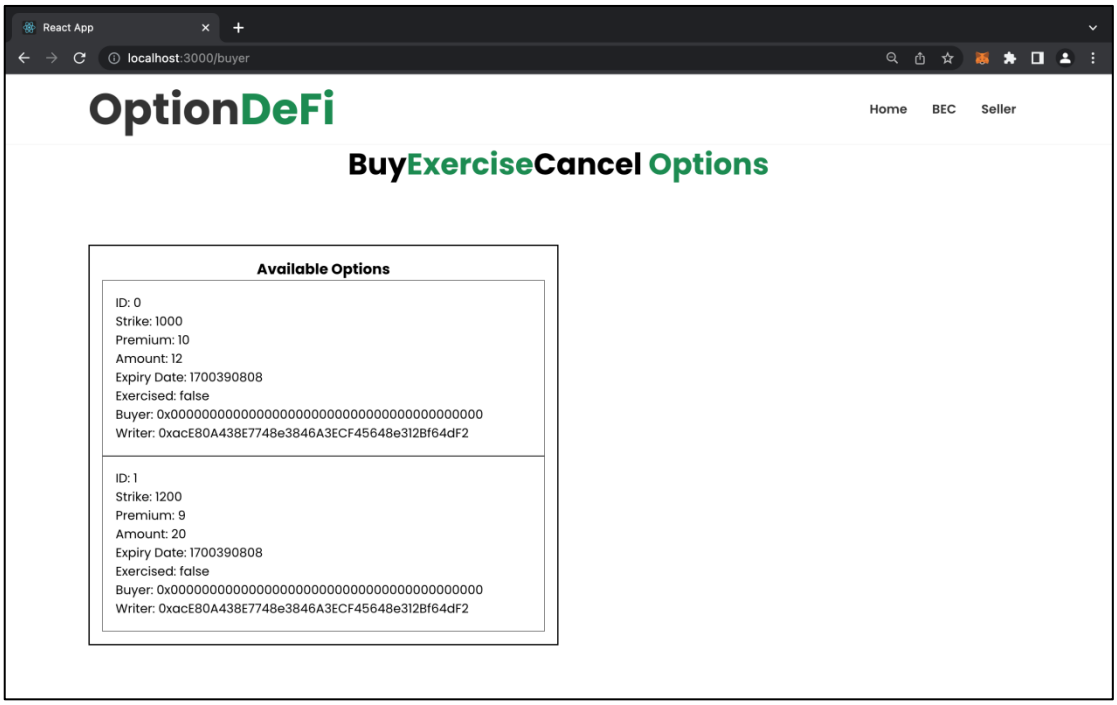
Verify whether the functions of "*approve*" and "*transfer*" work, by viewing the activity in the MetaMask wallet.



The BEC page now displays the details of the written option contract available on the blockchain to be bought or exercised.

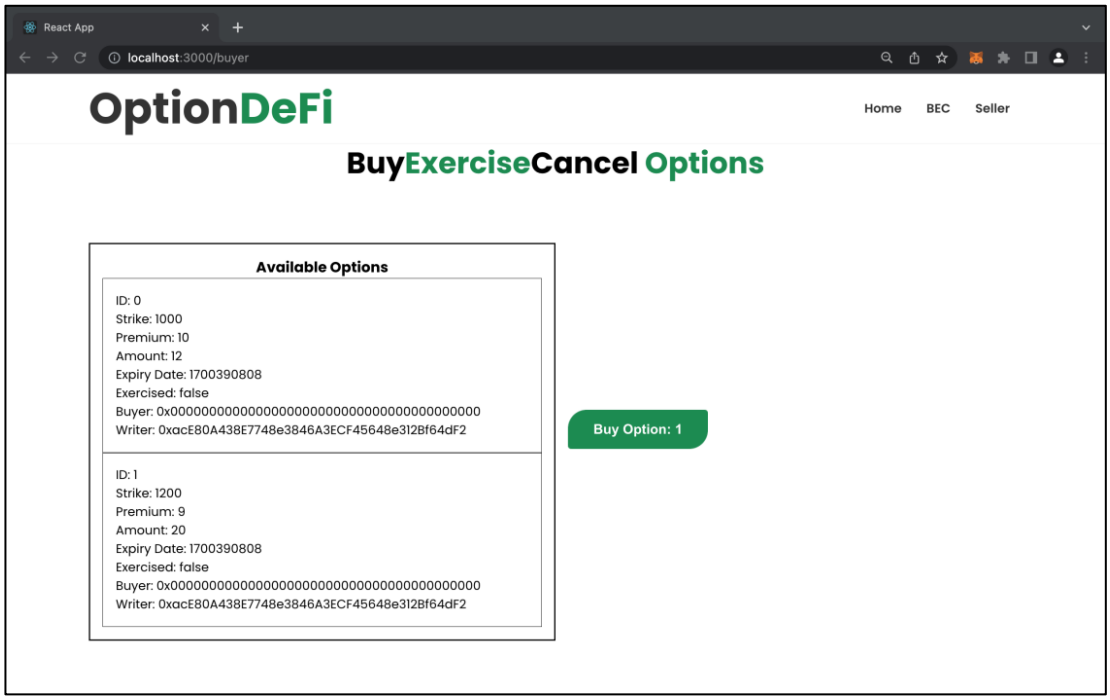


Example of writing another option and displaying it on the BEC page.

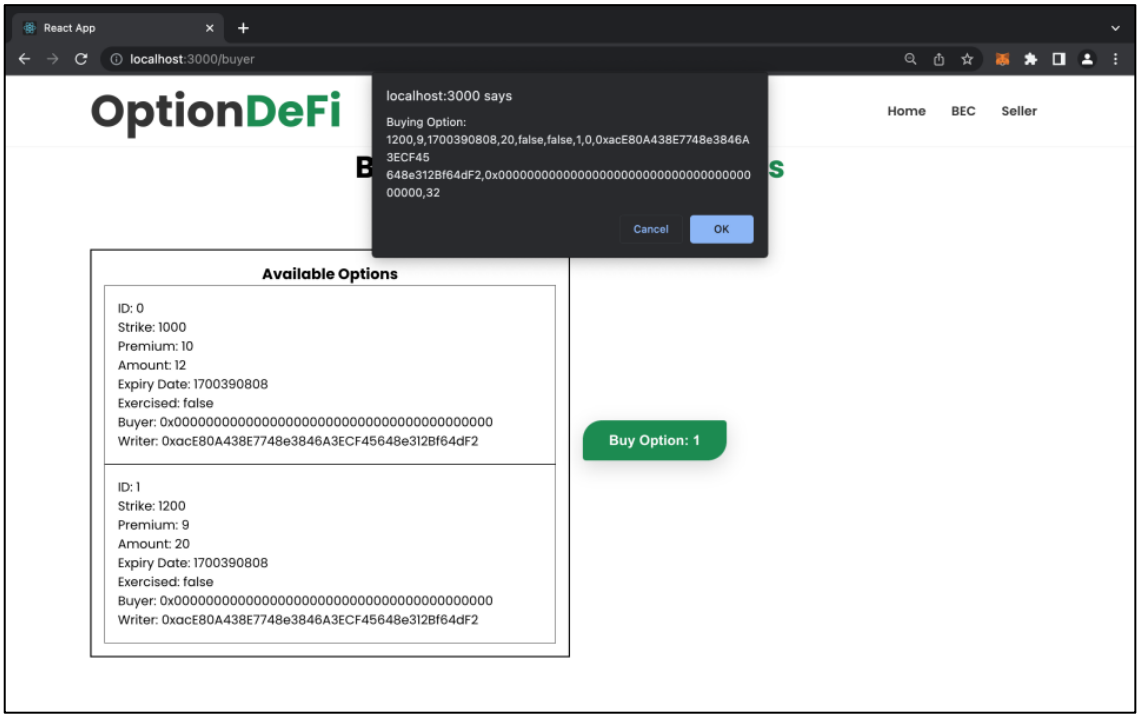


3.3.2 Buy Option

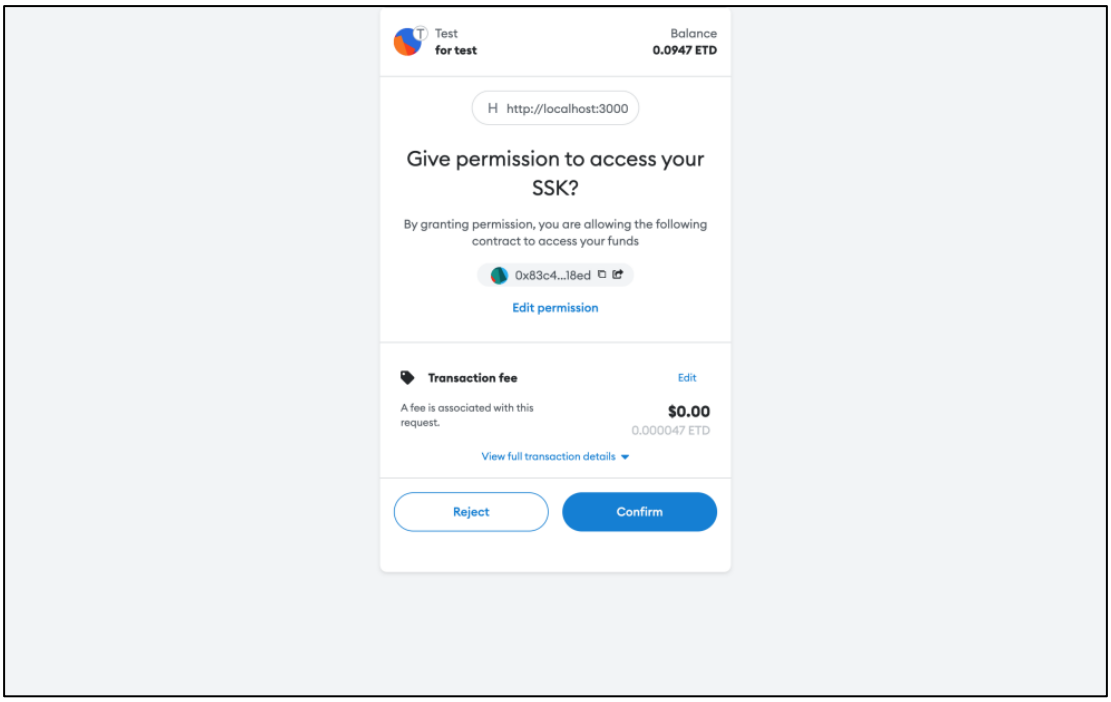
From this list, the user is able to buy the option desired. Below is an example of buying the second option (ID: 1).



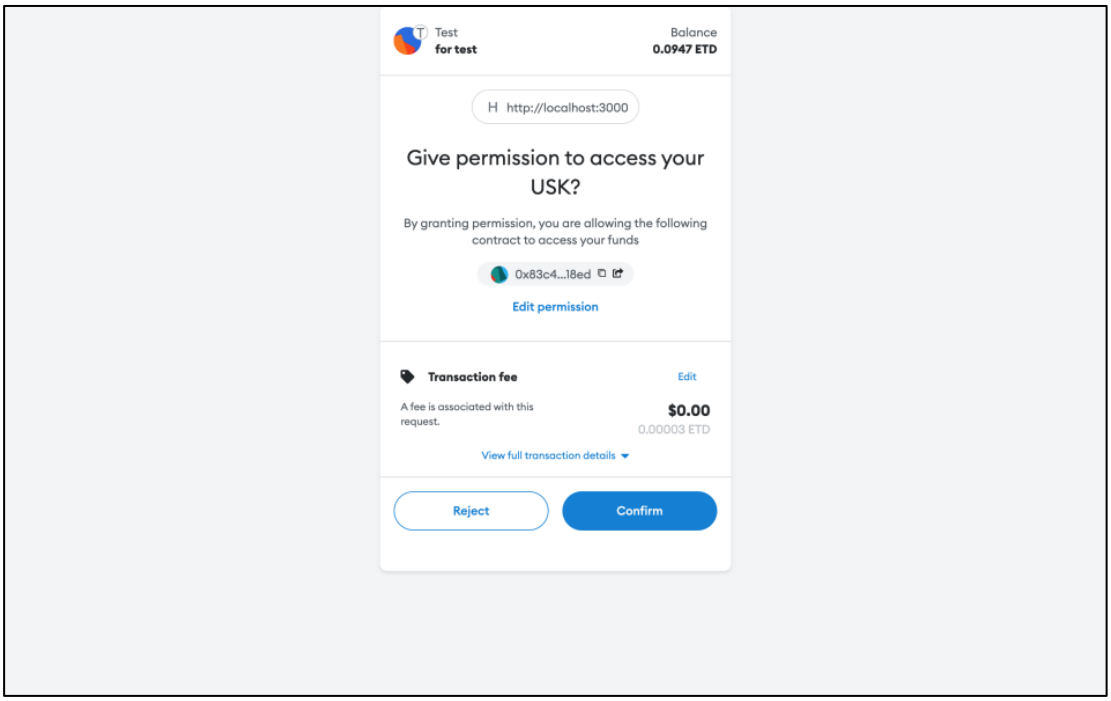
After clicking "*Buy Option*", the system asks user to confirm.

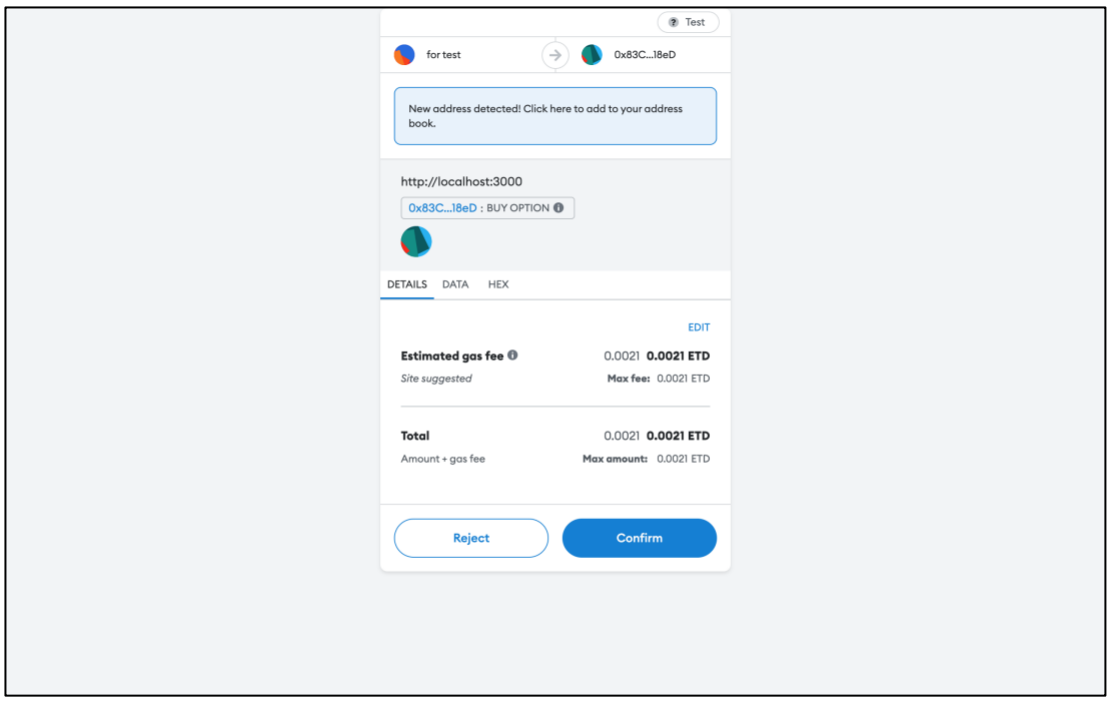


After the user confirms, the user is prompted to approve the transfer of the stablecoin from the wallet to option contract.

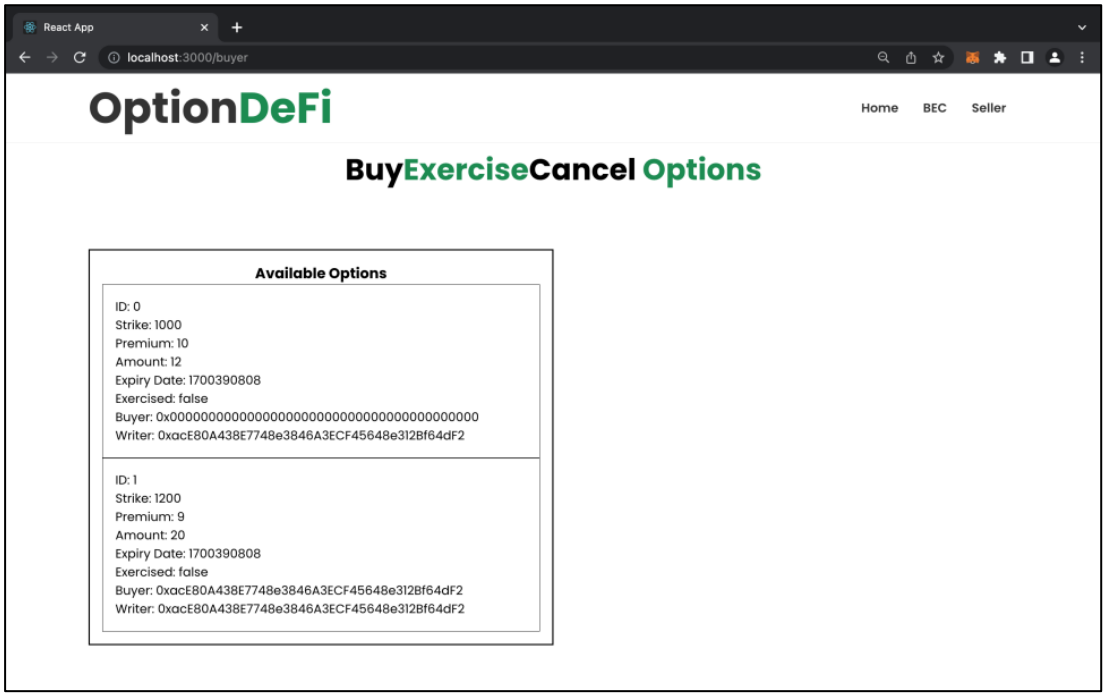


The user is then prompted to confirm the transfer of the stablecoin from buyer to the option contract.



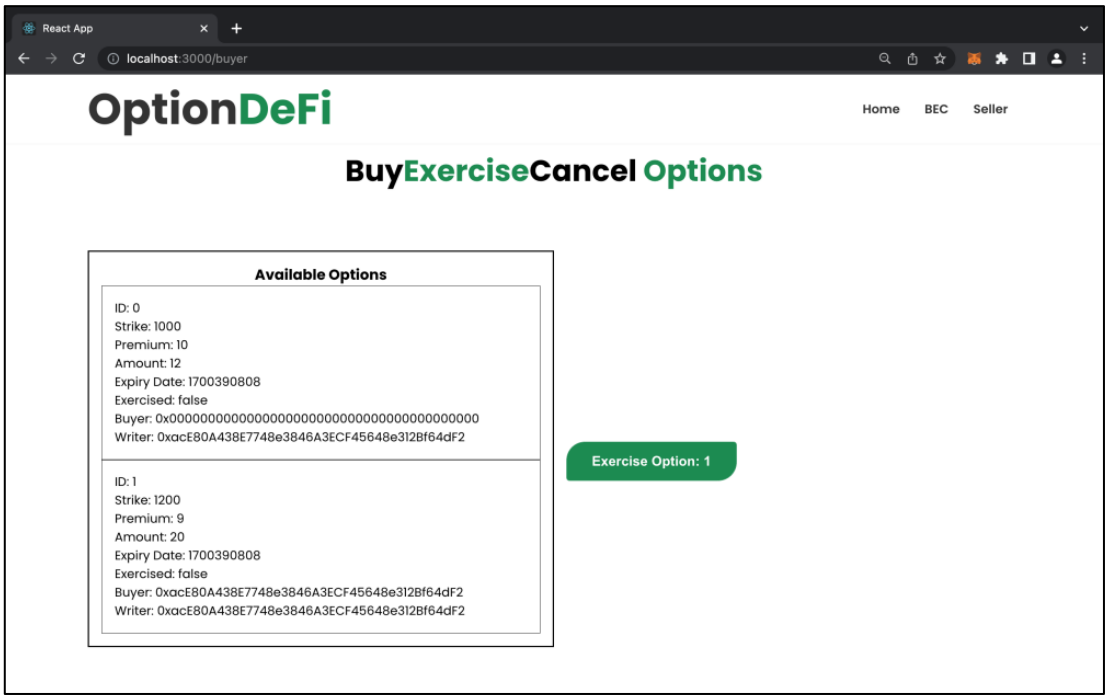


Verifying whether the *“Buy Option”* function works as intended. As expected, the “Buyer “ address field of the contract bought (ID: 1) has now been updated.

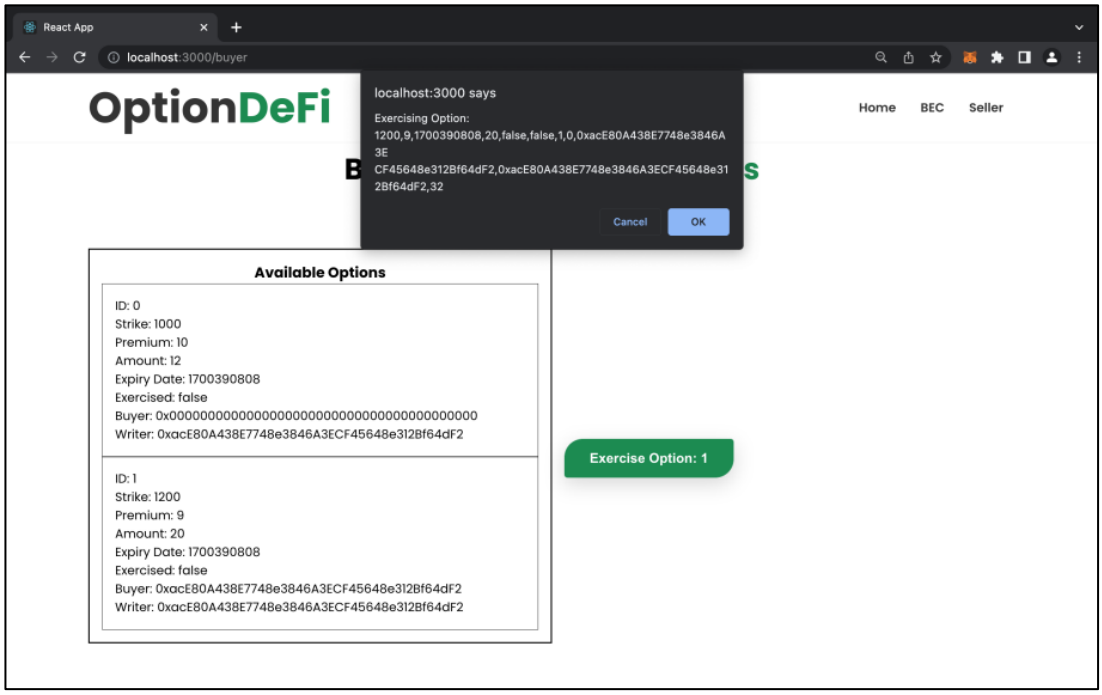


3.3.3 Exercise option

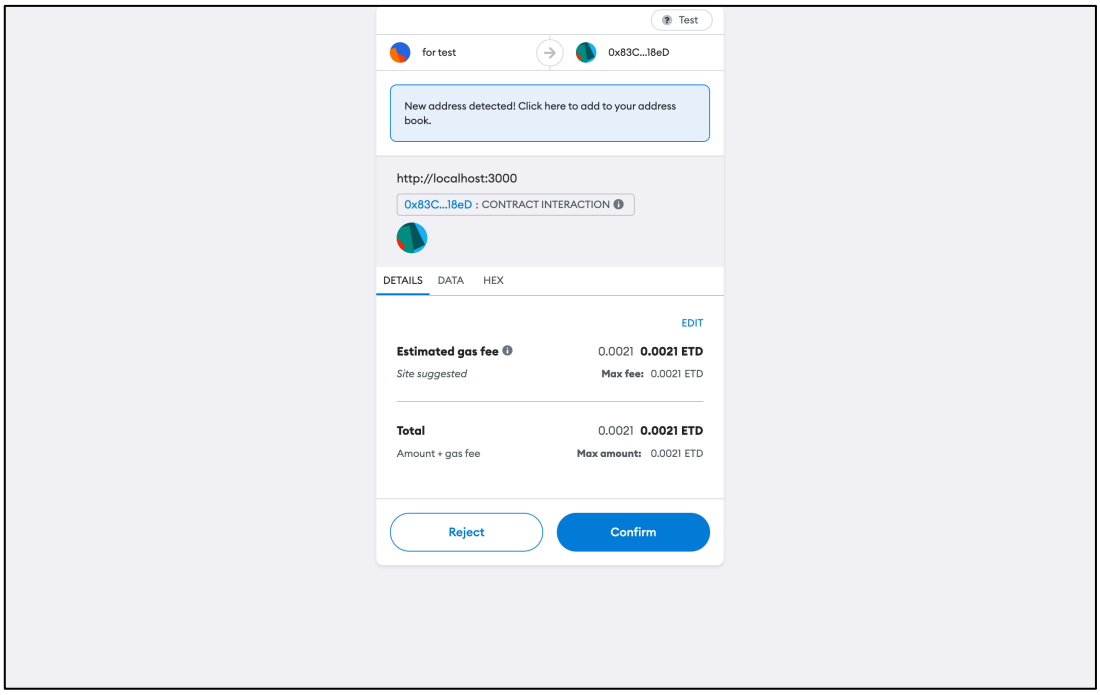
Example of exercising an option that has already been bought (ID: 1).



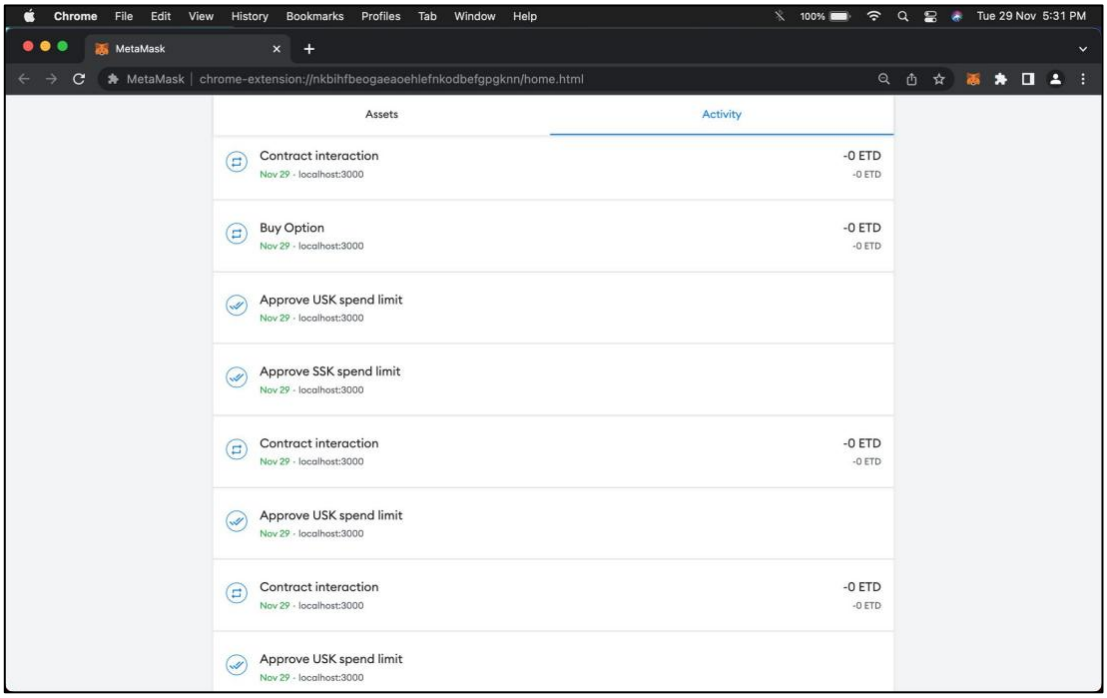
After clicking the *"Exercise Option"* button, the system prompts the user to confirm.



The user is then prompted to confirm the transaction where the stablecoin is transferred from the buyer's wallet to the contract.



View the list of all the transaction activity on the MetaMask wallet.



In conclusion, our project aims to give consumers access to a low-risk, open-source, decentralized, secure platform with the main functions of selling, buying, and exercising options.

4 Experience & Harvest

With the assistance of the teacher's assistant, we gradually grasped the solutions to the problems we met in practice, and were able to develop a deeper understanding of the use of DeFi on blockchain. The difficulties and obstacles we faced during the development of the front-end and the back-end are listed below.

4.1 Backend

1. Calling the ERC20 interface in smart contract for transferring funds.
2. Designing option contracts as an array with different attributes in smart contract.
3. Writing the test case to test whether the functions in the smart contract work as intended.
4. Deploying a smart contract to blockchain.

4.2 Frontend

1. Establishing a connection between the frontend and the MetaMask wallet.
2. Connecting the smart contract (abi) to the frontend.
3. Calling the backend functions of option contracts.
4. Displaying the list of options written on the blockchain.

5 Future Plan

Due to technical and token limitations, this project can be improved in several aspects. As future improvement, we aim to achieve the following:

1. cMMD should be pegged to the USD.
2. Exchange rate between cMMD and MMD should be floating.
3. Implement a liquidity pool to allow buyers to decide the parameters of option they want to buy.
4. Implement a pricing function of option under 3.
5. Enable the use of the 'Cancel option' function on the frontend.
6. Logic for frontend interactions must be optimized, and quick feedback on errors must be provided.
7. Support different MMDs with Uniswap.

6 Acknowledgements

Building an options trading platform on the blockchain took approximately two months, during which we gradually came to understand the opportunities and challenges of web3. It was a very enriching experience that helped us develop many skills and deepend our knowledge of the web3 space. From selecting the direction of the project to drafting the report, the attentive guidance and assistance from Professor Lei were essential. Moreover, whenever we were in need of guidance whether it was regarding the ideas of the project or code bugs, the teacher's assistant, Qi Wei was always available to provide us his time

and patience to help us solve these problems. We sincerely appreciate the help from Professor Zhibin Lei and his assistant Qi Wei.