

# Defi Options Trading

Team S-K

XIAO, Li    LAKHANI, Harsh Sunil    DONG, Wenyu

01 Dec 2022

## 1 Problem statement

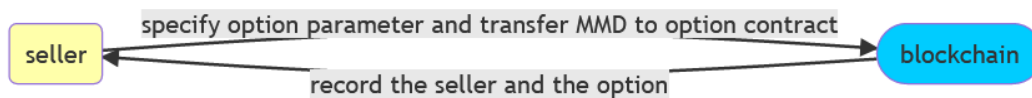
DeFi options are low-barrier borderless instruments that can be easily traded among peers on the platform without involving brokers or paying brokerage. In recent years, defi options have gained the attention of traders due to their risk aversion and high liquidity, especially among hedgers and speculators. In general, there are two types of option trading: call and put. Call options allows the owner to buy a specified amount of 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 gives the holder the right to sell the underlying asset at a specified price on or before expiration. In our project, we will establish an on-chain peer to peer option trading protocol built on MMD and cMMD(mock).

## 2 Design process

### 2.1 Back end

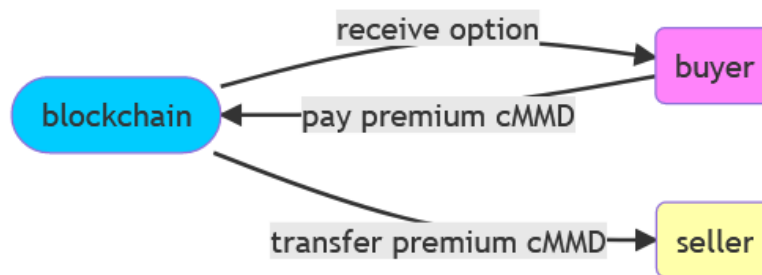
Stablecoin, Unstablecoin and Option contract stored on the back-end are three main contracts for this project. The former two contracts extend **ERC20**, while the latter performs the primary role of implementing transactions.

#### 2.1.1 Seller write option



In the function of writing option, seller specifies option parameters stored on blockchain, including coin, strike price, premium, expiry and token amount. Blockchain correspondingly record option information.

### 2.1.2 Buy option



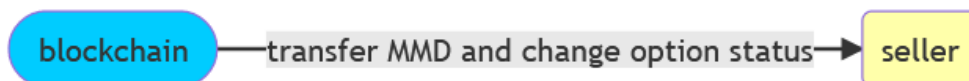
After seller uploads the option, buyers can browse the list of available options and pay premium cMMD if they decide to buy one. Meanwhile, the updated data and premium will be transferred to the seller.

### 2.1.3 Exercise option



Within the expiry date, buyer pays seller the strike price with cMMD using blockchain as a hub. Then buyer will receive MMD from the blockchain.

### 2.1.4 Seller cancel/retrieve option(to do)



The functions of cancellation and retrieval are the process of active or passive cancellation of option transactions by the seller, which the seller will get MMD back and the status of option will be changed. Although it has been implemented at the back end, the front end does not display this part.

## 2.2 Front end

It is important to establish a connection to the MetaMask wallet, where stablecoin and unstablecoin come from. From the perspective of users, what comes to mind is a homepage with buyer and seller triggers. When clicking the button of option seller, users can see parameters of the option information, current price of MMD and account. In contrast, users will view the written options in the buyer page. Then, the exercise page contains not only purchased options but also further confirmation. In the cancel or retrieve page, the information and status of written option is necessary, which will be done later.

### 3 Final result

We implemented defi option 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

```
~/Dow/zzz_important/S_K/Back-end main !1 1m 27s base 19:53:24
npx hardhat run scripts/deploy.ts --network etherdata
Options contract deployed to: 0xc62Ad04Dc061B2F1D69f8fFC147C741478031660
SKCoin_stable_coin contract deployed to: 0xAfE899f14c38A327E5D29f3E9a9B423938F8824C
SKCoin Unstable Asset contract deployed to: 0x7d59ab5546B4d301AeBD1EE151325E3746F6A3De
```

#### 3.2 Stablecoin and unstablecoin contract

Import defined stablecoin(SSK)

Import tokens

Custom token

Token detection is not available on this network yet.  
Please import token manually and make sure you trust it.  
[Learn about scams and security risks.](#)

Token contract address

4576A8Cf609c9104353eB75f67023C7488ceed

Token symbol

SSK

Edit

Token decimal

18

Add custom token

Import tokens

Would you like to import these tokens?

Token	Balance
<div><div></div>SSK</div>	5000000000 SSK

Back

Import tokens

Import defined unstablecoin(USK)

Import tokens

Custom token

Token detection is not available on this network yet. Please import token manually and make sure you trust it. Learn about [scams and security risks](#).

Token contract address

0xC2283AA608b5347555EDd7dDA5DC7BEA95

Token symbol

USK

Edit

Token decimal

18

Add custom token

Import tokens

Would you like to import these tokens?

Token

Balance

USK

5000000000  
USK

Back

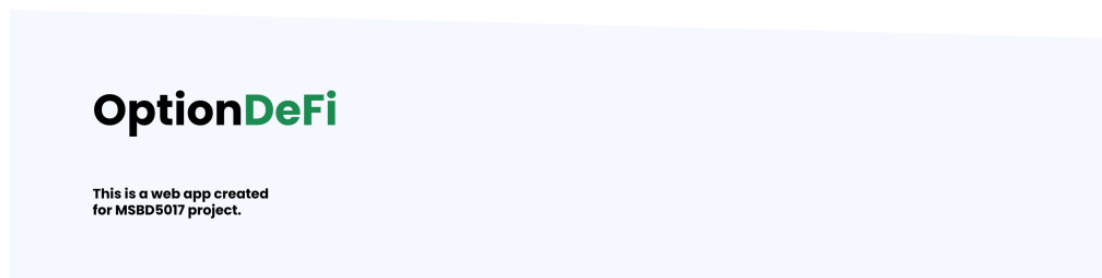
Import tokens

All coins in the wallet

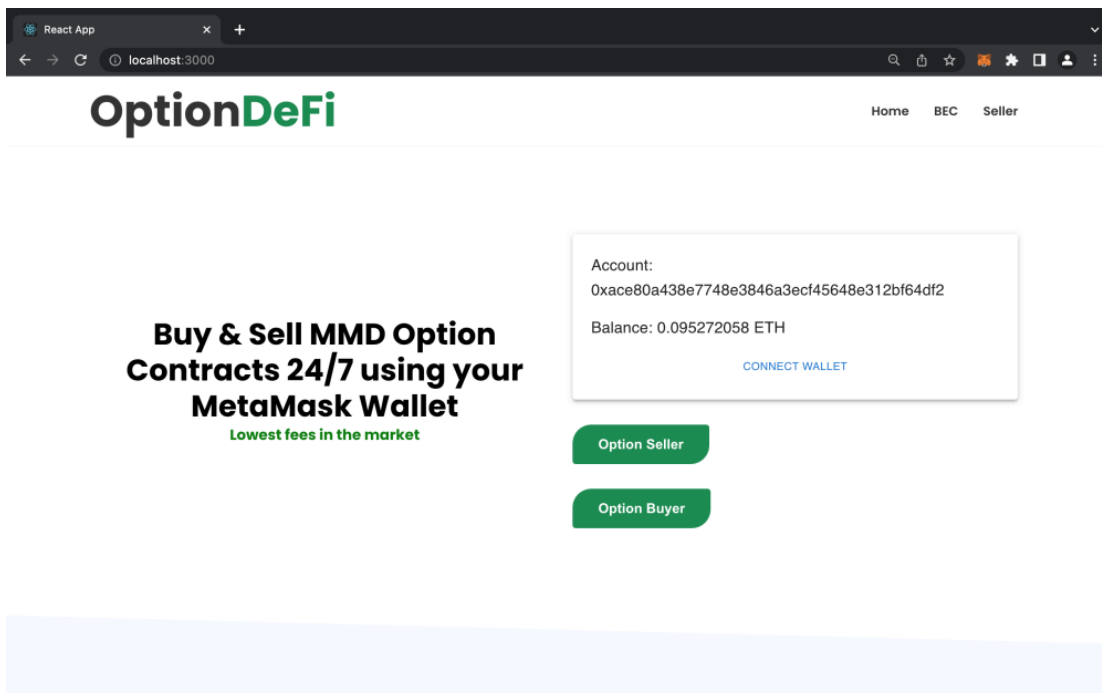
Account 1 0x251...4885	
0.8417 ETD	
<div>BuySendSwap</div>	
<div>AssetsActivity</div>	
<div>Portfolio site</div>	
<div>0.8417 ETD</div>	>
<div>5000000000 SSK</div>	>
<div>5000000000 USK</div>	>

### 3.3 Option contract

When clicking on the homepage, there is currently no option contract on the blockchain.



Connect to 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, premium, strike, expiry and amount.

React App x +

localhost:3000/seller

# OptionDeFi

Home BEC Seller

## Sell Options

Premium Price: 10

Strike Price: 1000

Expiry Date: 25/10/2023

Amount: 12

**Sell Option**

### Current Price of MMD Token:

**\$55.28**

Account:  
0xace80a438e7748e3846a3ecf45648e312b64df2

Balance: 0.095272058 ETH

[CONNECT WALLET](#)

Approve unstablecoin from wallet to contract, using “*approve*” function from smart contract, which extends **ERC20**.

Test for test Balance 0.0953 ETD

H http://localhost:3000

### Give permission to access your USK?

By granting permission, you are allowing the following contract to access your funds

0x83c4...j8ed

[Edit permission](#)

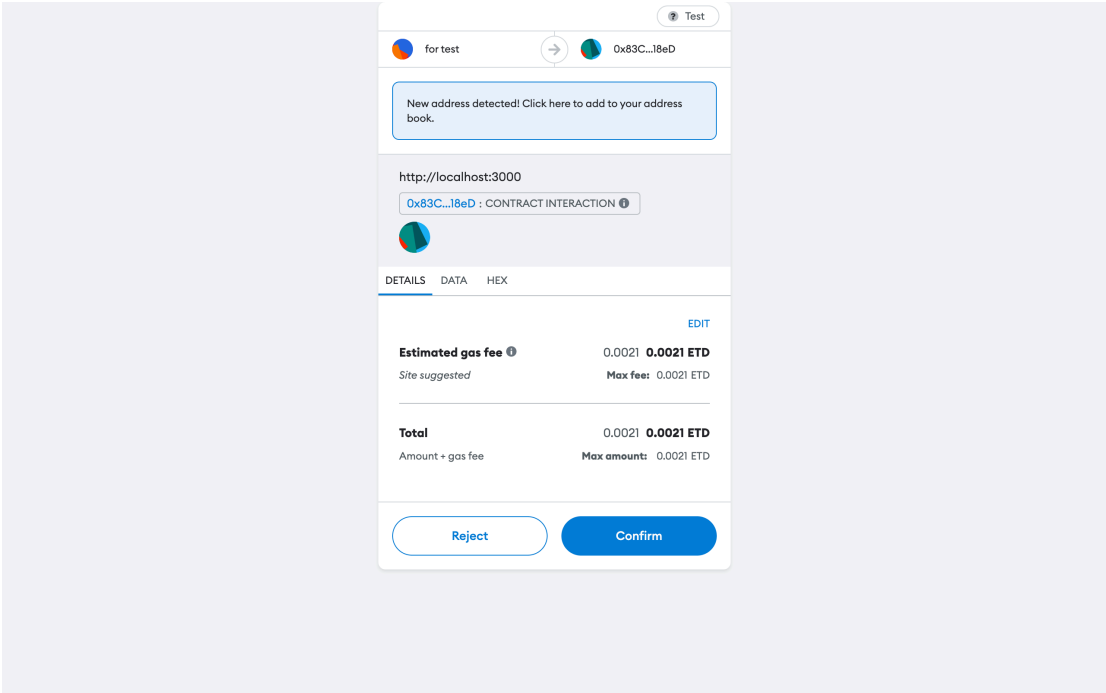
**Transaction fee** Edit

A fee is associated with this request. **\$0.00** 0.000047 ETD

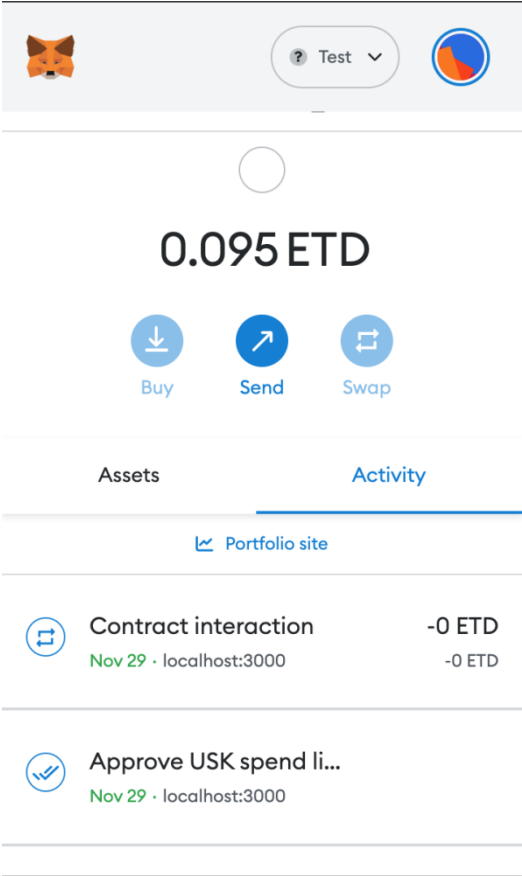
[View full transaction details](#)

[Reject](#) [Confirm](#)

Transfer unstablecoin from writer to option contract.



Check whether the functions of “*approve*” and “*transfer*” work, from the unstablecoin contract which extends *ERC20*.



Details of the written option contract available on the blockchain

React App

localhost:3000/buyer

OptionDeFi

HomeBECSeller

BuyExerciseCancelOptions

Available Options

ID: 0  
Strike: 1000  
Premium: 10  
Amount: 12  
Expiry Date: 1700390808  
Exercised: false  
Buyer: 0x00  
Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2

OptionDeFi

Write another option

React App

localhost:3000/buyer

OptionDeFi

HomeBECSeller

BuyExerciseCancelOptions

Available Options

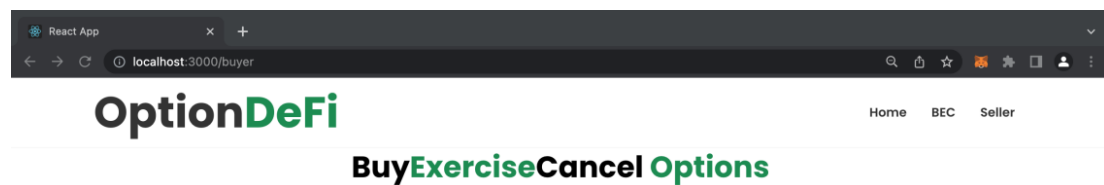
ID: 0  
Strike: 1000  
Premium: 10  
Amount: 12  
Expiry Date: 1700390808  
Exercised: false  
Buyer: 0x00  
Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2

ID: 1  
Strike: 1200  
Premium: 9  
Amount: 20  
Expiry Date: 1700390808  
Exercised: false  
Buyer: 0x00  
Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2

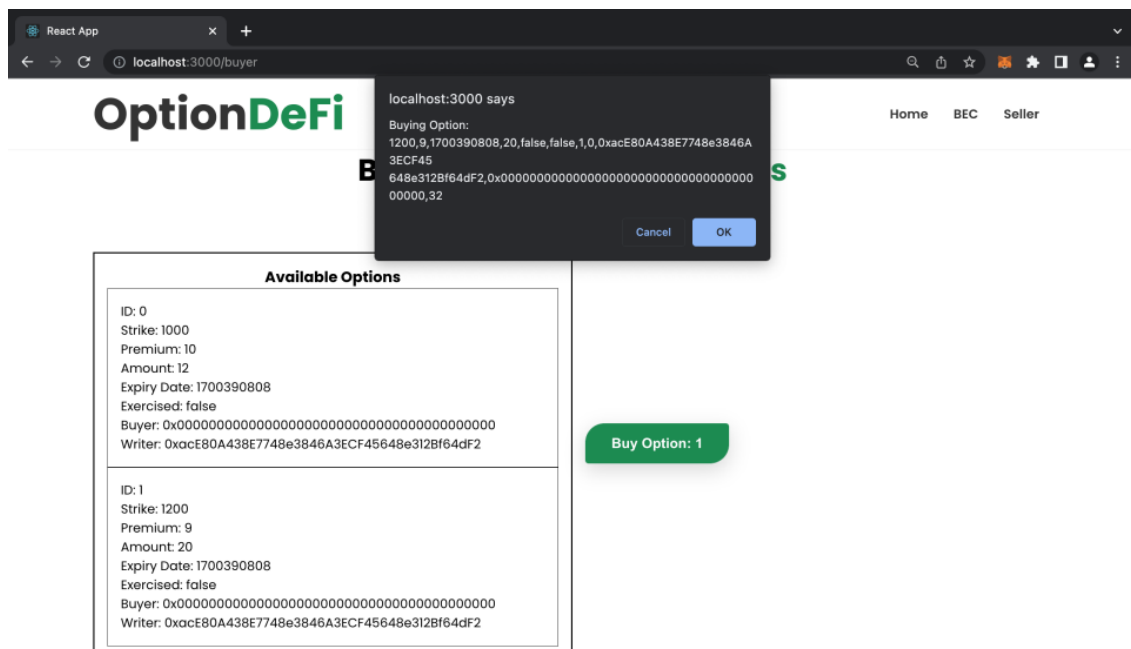


### 3.3.2 Buy option

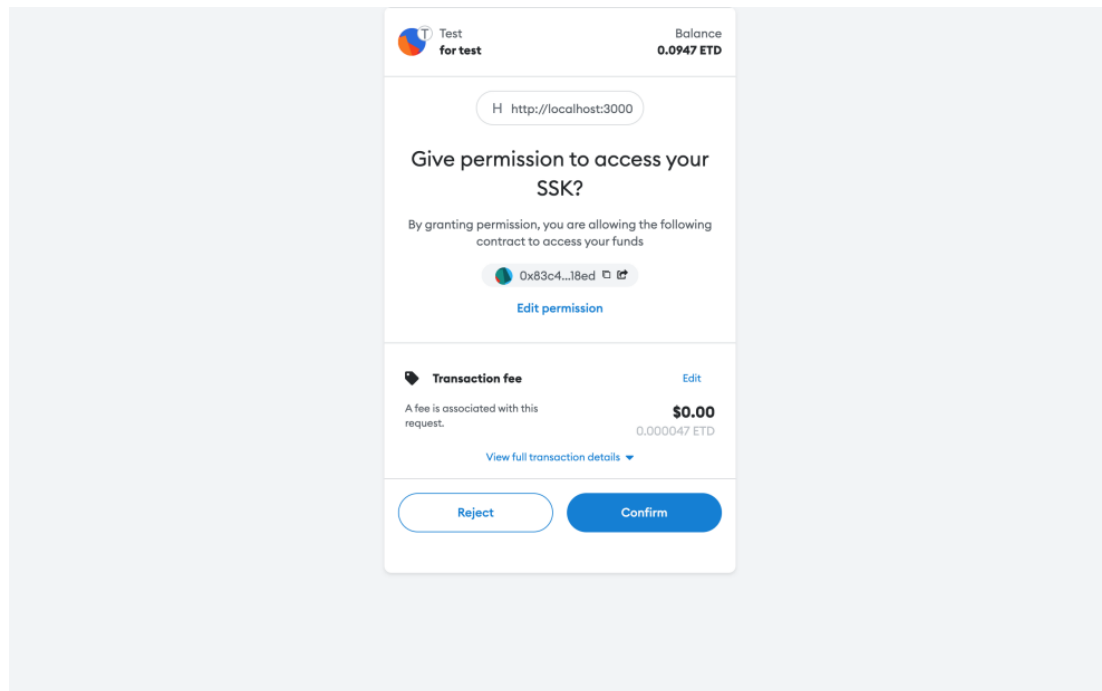
Buy the second option(ID: 1). In this screen, you can buy the option you want from the list.



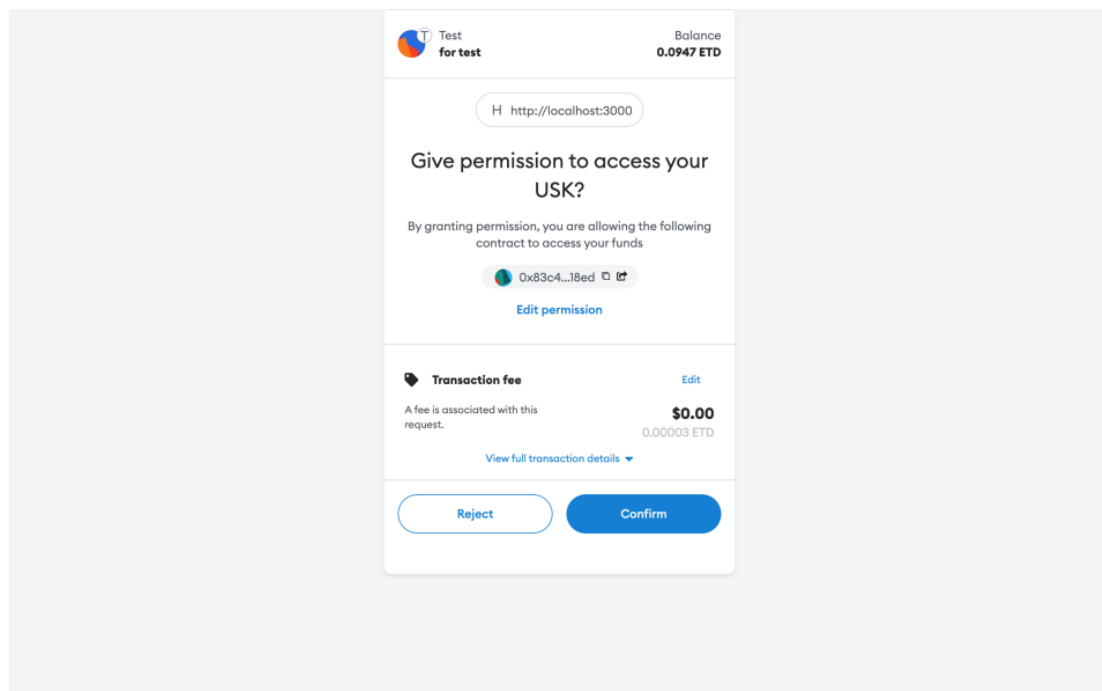
After clicking “Buy Option”, the system asks user to confirm.



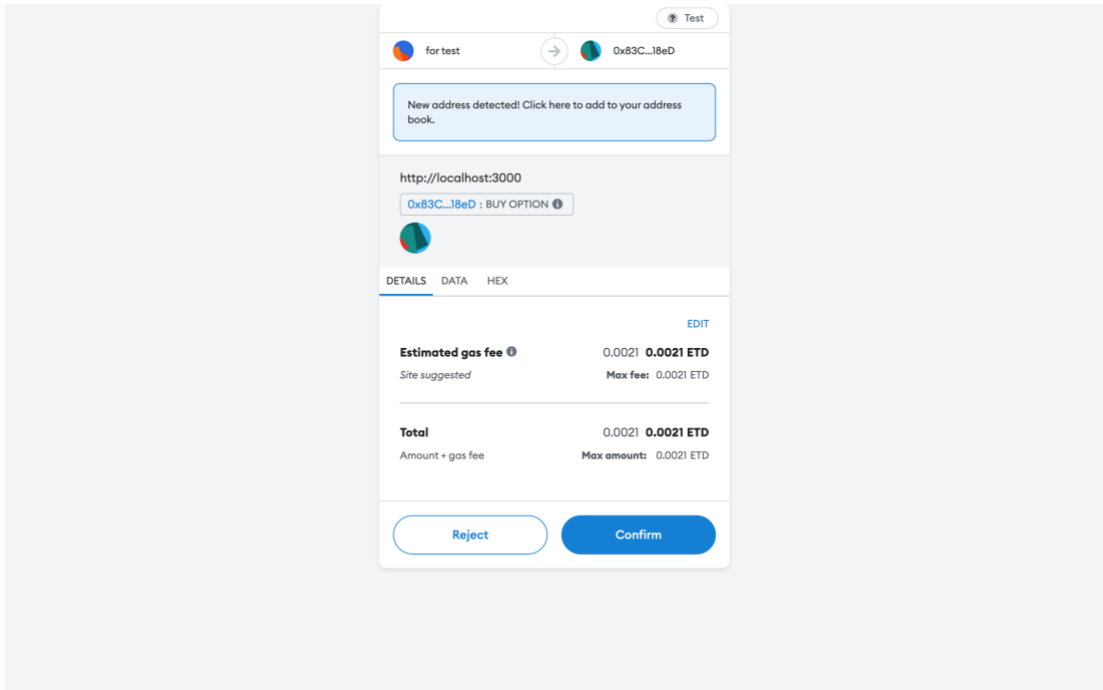
Approve stablecoin from the wallet to option contract.



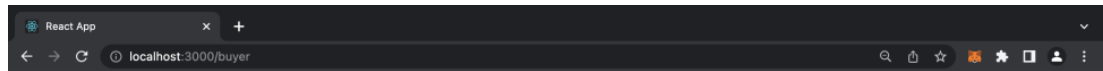
Transfer stablecoin from buyer to contract



Transfer stablecoin from contract to writer



Verify the correctness of “buy option”. As expected, the buyer address of the contract bought has now been updated



**OptionDeFi** Home BEC Seller

**BuyExerciseCancel Options**

Available Options	
ID: 0	
Strike: 1000	
Premium: 10	
Amount: 12	
Expiry Date: 1700390808	
Exercised: false	
Buyer: 0x00	
Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2	
ID: 1	
Strike: 1200	
Premium: 9	
Amount: 20	
Expiry Date: 1700390808	
Exercised: false	
Buyer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2	
Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2	

### 3.3.3 Exercise option

Exercise the second option (ID: 1)

The screenshot shows the OptionDeFi web application interface. At the top, there's a navigation bar with the logo 'OptionDeFi' and links for 'Home', 'BEC', and 'Seller'. Below the navigation bar, there's a header with the text 'BuyExerciseCancel Options'. The main content area features a panel titled 'Available Options' which lists two options:

- Option 0:**
  - ID: 0
  - Strike: 1000
  - Premium: 10
  - Amount: 12
  - Expiry Date: 1700390808
  - Exercised: false
  - Buyer: 0x00
  - Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2
- Option 1:**
  - ID: 1
  - Strike: 1200
  - Premium: 9
  - Amount: 20
  - Expiry Date: 1700390808
  - Exercised: false
  - Buyer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2
  - Writer: 0xacE80A438E7748e3846A3ECF45648e312Bf64dF2

To the right of the 'Available Options' panel, there is a green button labeled 'Exercise Option: 1'.

After clicking “Exercise Option”, the system asks user to confirm

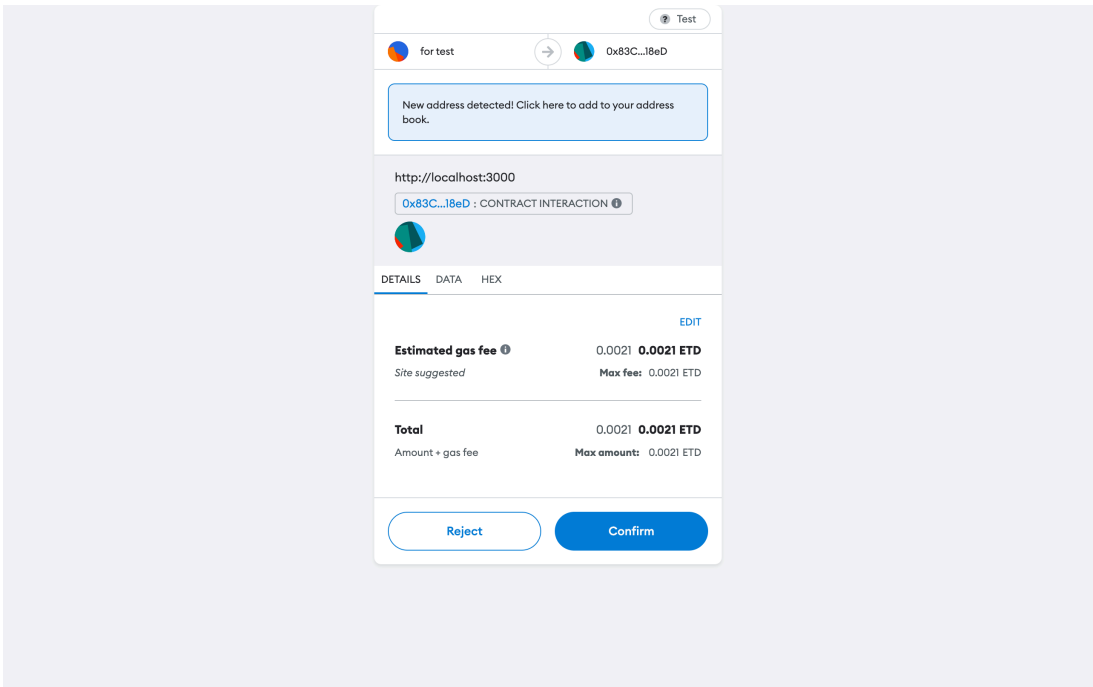
The screenshot shows the OptionDeFi web application interface with a confirmation dialog box overlaid. The dialog box contains the following text:

localhost:3000 says  
Exercising Option:  
1200,9,1700390808,20,false,false,1,0,0xacE80A438E7748e3846A3E  
CF45648e312Bf64dF2,0xacE80A438E7748e3846A3ECF45648e31  
2Bf64dF2,32

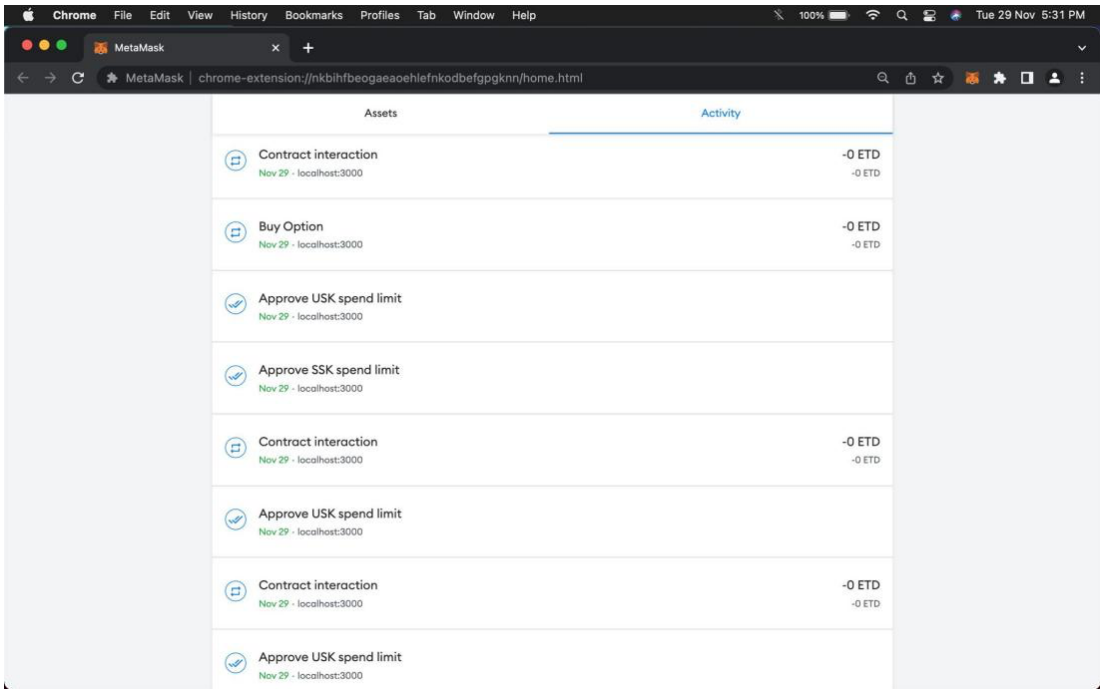
At the bottom of the dialog box, there are two buttons: 'Cancel' and 'OK'.

Below the dialog box, the 'Available Options' panel is visible, showing the same two options as in the previous screenshot. To the right of the panel, there is a green button labeled 'Exercise Option: 1'.

Do transaction, which transfers stablecoin from the buyer's wallet to the contract



The whole transaction activities



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

## **4 Experience and harvest**

With the assistance of our assistant professors, we gradually grasped the solutions to the problems we met in the practice, and we developed a deeper understanding of the use of defi on blockchain. The difficulties and obstacles we faced during front-end and back-end training are listed below.

### **4.1 Back end**

1. Call the ERC20 interface in smart contract for transferring funds
2. Design option as an array with different attributes in smart contract
3. Write test case to test whether those functions in smart contract work
4. Deploy smart contract to blockchain

### **4.2 Front end**

1. Connect MetaMask wallet in the front-end page
2. Connect smart contract (abi, contract address) in the front-end page
3. Transfer transaction with defined token in the front-end page
4. Call the function of option contract in the front-end page
5. Show option list in front-end page

## **5 Future Plan**

Due to technical and token limitations, our research has several flaws. our goal in the future is to accomplish 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 pricing function of option under 3
5. Logic for front-end interactions must be optimized, and quick feedback on errors must be provided.
6. 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. From choosing the topic's direction through drafting the report, the attentive guidance and assistance from Professor Lei were essential in this endeavor. Moreover, whenever we encounter code bugs or discuss our group's ideas , Qi wei always gives us timely and patient help. Therefore, we sincerely appreciate the help from Professor Zhibin Lei and TA Qi wei.

## 7 Github Repository

[https://github.com/Joe-Bradley/S\\_K](https://github.com/Joe-Bradley/S_K)