**Crypto-commerce-manual**

1. Prerequisites: node.js to be installed, React to be installed, metamask (interface to Ethereum and serves as a wallet) to be installed, hardhat to be installed. (Create a directory for the project. At the root, npm install, npm hardhat install, npm react install. Metamask to be downloaded and set up.)

Add your privekey and apikey in .env file.

Add your pinata credentials in SIgnUp.js and Mint.js

There are 3 smart contracts

1. Defi.sol : Its for buying tokens and staking tokens
2. Tweets.sol : its for SignUp ,Login and functions related to tweets
3. NFTandDefiMerged:Nft fucntions and merging functions.

1. Select **Sepolia** as testnet on metamask (Sepolia is one of the default testnets available on metamask, no need to add)

2. Get some test eth from [**https://sepoliafaucet.com/**](https://www.linkedin.com/safety/go?url=https%3A%2F%2Fsepoliafaucet.com%2F&trk=flagship-messaging-web&messageThreadUrn=urn%3Ali%3AmessagingThread%3A2-MmI4MjFmYTctYzBkNy00NDlmLWEwNjAtNTU3MWE5MGJkZmVhXzAxMA%3D%3D&lipi=urn%3Ali%3Apage%3Ad_flagship3_messaging_conversation_detail%3B2VW5kti5RHGwf5XwRkNd%2Fw%3D%3D)

3. Set up a **hardhat** account:

- Go to Alchemy <https://dashboard.alchemy.com/>.

- Create app by click on CREATE APP and select  
                CHAIN : Ethereum  
                NETWORK : Sepolia  
- Then open app and click on VIEW KEY ,copy the API\_KEY.

- Replace the private key at "accounts" on hardhat.config.js file with your private key with a prefix of "0x". Then, enter URL:

<https://eth-sepolia.g.alchemy.com/v2/RBC4SxiuQkH-hB2Xqc8qODJxOOM_Hnf5>

4. Set up a **pinata** account:

- Head over to   <https://app.pinata.cloud/pinmanager>  
- Create account and log in to it.  
- Click on API Keys and select New Key  
- Give all permissions, select Max uses as 1000 and enter name of key then click on     Create Key  
- Then it will pop up, copy the API Key and API Secret .  
- Now open Mint.js file in src/Components.  
- Replace “api\_key” and “Your API Secret Key” with your copied keys.

5.**Deployment**

Deployment of **Defi.sol**  
1.place the code in deploy.js

const hre = require("hardhat");

async function main() {

    const NFTMarketPlaceAurora = awaith hre.ethers.getContractFactory('Defi');

    const contract = await NFTMarketPlaceAurora.deploy();

     await contract.deployed();

     console.log("Address of Contract : ",contract.address);

}

main().catch((error) => {

    console.error(error);

    process.exitCode = 1;

  });

2.At root, run the *npx hardhat run --network sepolia scripts/deploy.js*

*3.*Copy the address.

Deployment of **Tweets.sol**

1.Now replace code in scripts/deploy.js with  the following code

const hre = require("hardhat");

async function main() {

    const NFTMarketPlaceAurora = await hre.ethers.getContractFactory('Tweets');

    const contract = await NFTMarketPlaceAurora.deploy();

     await contract.deployed();

     console.log("Address of Contract : ",contract.address);

}

main().catch((error) => {

    console.error(error);

    process.exitCode = 1;

  });

2. At root, run the *npx hardhat run --network sepolia scripts/deploy.js*

*3.*Copy the address.  
Deployment of **NFTandDefiMerged.sol**1.Now replace code in scripts/deploy.js with  the following code

const hre = require("hardhat");

async function main() {

    const NFTMarketPlaceAurora = await hre.ethers.getContractFactory('NFTandDefiMerged');

    const contract = await NFTMarketPlaceAurora.deploy("COPIED\_ADDRESS FROM DEFI","COPIED ADDRESS FROM TWEETS");

     await contract.deployed();

     console.log("Address of Contract : ",contract.address);

}

main().catch((error) => {

    console.error(error);

    process.exitCode = 1;

  });

(MAKE SURE PAST THE TWO COPIED ADDRESS IN ABOVE CONTRACT IN ORDER)  
2. At root, run the *npx hardhat run --network sepolia scripts/deploy.js*

*3.*Copy the address.

5.Now paste the copied address in **App.js**  
6. Then, remove current **artifacts** directory under root. Then, move the artifacts directory to under src/.

\*Note that you will have to do this every time you deploy a new contract.

14. “**npm start**” on a web browser (localhost for testing purpose) where your metamask is connected to.

Functions to test:

Uploading NFT:

1. There is a UPLOAD NFT button, click on it.
2. Then you will redirected to the Upload NFT page. Enter appropriate names and values into the entries underneath the public key of the account in the wallet it is connected to.
3. Then, select a NFT to upload, and then click "List" to be followed by a popup of upload success message.
4. Confirm the trasaction of nft listing on the Metamask after a brief while.
5. You will be redirected to the "Live NFT" (where all the nft's listed for sale in the marketplace) page after the transaction is confirmed.

Buying NFT :

1. There is BUY button followed by the NFT .
2. After click on BUY, it will get added to the CART page. (You can select multiple NFTs)
3. After selecting the NFTs, go to the CART page .There you get the selected NFTs and price details .
4. If you click the PAY, then metamask will asks to confirm the transaction.
5. After successful transaction, the NFT will be added to MY NFTs page, where you can check your owned NFTs.

Resale NFT :

1. If you want to resale (to list it back for sale in the marketplace possibly at a different price with premium) your owned NFT, there is an option underneath each NFT in the MY NFTs page.
2. After click on Resale, it asks to enter the price of the NFT.
3. Then click on OK and confirm the transaction.
4. It will get added to the LIVE NFTs page after transaction is successful.

Staking tokens :

1. Buy tokens.
2. Stake tokens.
3. Unstake tokens.

A black screen with white text

Description automatically generated

A screenshot of a computer

Description automatically generated