

README Guide for Cryptocurrency Exchange Project

Welcome to the Cryptocurrency Exchange Project! This guide will help you set up and run the project on your local machine for development and testing purposes.

Prerequisites

Before you begin, ensure you have the following installed:

- [Node.js](#) (version 12 or later)
- [Git](#)
- [MetaMask](#) extension installed in your browser

Installation

Clone the Repository Open your terminal and run:

```
git clone https://github.com/preravitch/Cryptocurrency-Exchange.git  
cd your-project-name
```

Install Dependencies. Inside the project directory, install the necessary dependencies:

```
npm install
```

Set Up Hardhat. This project uses Hardhat for local Ethereum network simulation. To set it up:

```
npx hardhat node
```

This command will start a local Ethereum network. Deploy Smart Contracts
In a new terminal window, deploy the smart contracts to the local network (run on different terminal)

```
npx hardhat run scripts/deploy.js --network localhost
```

Note the deployed contract addresses output by this command.

Configure MetaMask

Open MetaMask and connect to the `Localhost 8545` network.
Import one of the accounts provided by Hardhat (displayed in the terminal where you ran `npx hardhat node`) using its private key.

Update the frontend configuration with the deployed contract addresses. Modify the relevant files in the `src` directory with the addresses you noted earlier.

```
13   const platformAddress = "0x9fE46736679d2D9a65F0992F2272dE9f3c7fa6e0";
14   const contractAddress = "0x9fE46736679d2D9a65F0992F2272dE9f3c7fa6e0";

102  const tokenAddresses = {
103    BTC: "0x5FbDB2315678afecb367f032d93F642f64180aa3",
104    USDT: "0xe7f1725E7734CE288F8367e1Bb143E90bb3F0512",
105  };
106  You, 22 hours ago via PR #1 • update UI
```

Start the Frontend Application

Run the frontend application:

```
npm start
```

This will launch the app in your browser.

Usage

- Interacting with the Exchange
 - Use the interface to perform token exchanges, add or remove liquidity, and other functionalities.
 - Transactions will prompt MetaMask for confirmation.
- Testing
 - Run tests to ensure smart contracts are functioning as expected:
 - `npx hardhat test`

Troubleshooting

- If you encounter any issues with transactions, ensure MetaMask is connected to the `localhost 8545` network and you have sufficient ETH in your Hardhat account.
- For smart contract changes, re-deploy the contracts and update the frontend configuration accordingly.