Truffle Box For ERC4907¶

Requirements¶

The Truffle Box For ERC4907 has the following requirements:

- Node.js
- 14.x or later
- NPM
- version 5.2 or later
- Truffle
- Ganache
- Windows, Linux or MacOS

Helpful, but optional: - AnInfura account and Project ID - AMetaMask account

Setup¶

Installation¶

1. First ensure you are in a new and empty directory.
2. truffle
3. unbox
4. emojidao/ERC4907Box
5. If you want to create a directory, you can also run:truffle
6. unbox
7. emojidao/ERC4907Box
8. <Directory
9. Name>
10. Running theunbox
11. command should install the required dependencies.
12. Now, run the development console. This will spin up and allow you to interact withganache
13. , a local test chain onlocalhost:9545
14. .
15. truffle
16. develop
17. Compile and migrate the smart contracts. Runningmigrate
18. will do both. Note inside the development console we don't have to preface commands withtruffle
19. .
20. migrate
21. In theclient
22. directory, we run the React app. Smart contract changes must be manually recompiled and migrated.
23. // in another terminal (i.e. not in the truffle develop prompt)
24. cd
25. client
26. npm
27. install
28. npm
29. run
30. start
31. After migrating your contracts, head to theclient
32. directory and runnpm run start
33. to view the application in yourhttp://localhost:3000/
34. .
35. To mint your first NFT, you'll need to connect your wallet to a network. To do so, make sure your development console is still running. Then, connect to that network using your wallet. Afterwards, you'll need to use a funded account. You can import an account to your MetaMask wallet by using a private key outputted by the development console.
36. If you minted your NFT onlocalhost:9545
37. on the developer console, you should be able to see the mint on your MetaMask wallet. If you deploy toGoerli
38. , you will be able to see the NFT on their test-net site here

39. .
40. To build the application for production, use the build script. A production build will be in theclient/build
41. folder.
42. // ensure you are inside the client directory when running this
43. npm
44. run
45. build

Deployment¶

To deploy your contracts to a public network (such as a testnet or mainnet) there are two approaches. The first uses Truffle Dashboard which provides "an easy way to use your existing MetaMask wallet for your deployments". The second, requires copying your private key or mnemonic into your project so the deployment transactions can be signed prior to submission to the network.

Using Truffle Dashboard (recommended)¶

Truffle Dashboard ships with Truffle and can be started withtruffle dashboard . This in turn loads the dashboard at http://localhost:24012 and beyond that you'll just need to run your migration (truffle migrate --network dashboard). A more detailed guide to using Truffle Dashboard is availablehere .

Using the env File¶

You will need at least one mnemonic to use with the network. The.dotenv npm package has been installed for you, and you will need to create a.env file for storing your mnemonic and any other needed private information.

The.env file is ignored by git in this project, to help protect your private data. In general, it is good security practice to avoid committing information about your private keys to github. Thetruffle-config.js file expects aMNEMONIC value to exist in.env for running commands on each of these networks, as well as a defaultMNEMONIC for the Arbitrum network we will run locally.

If you are unfamiliar with using.env for managing your mnemonics and other keys, the basic steps for doing so are below:

1) Usetouch .env in the command line to create a.env file at the root of your project. 2) Open the.env file in your preferred IDE 3) Add the following, filling in your own Infura project key and mnemonics:

MNEMONIC="" INFURA_KEY="" RINKEBY_MNEMONIC="" MAINNET_MNEMONIC="" 4) As you develop your project, you can put any other sensitive information in this file. You can access it from other files withrequire('dotenv').config() and refer to the variable you need withprocess.env['] .

Support¶

Support for this box is available via the Truffle community available here .