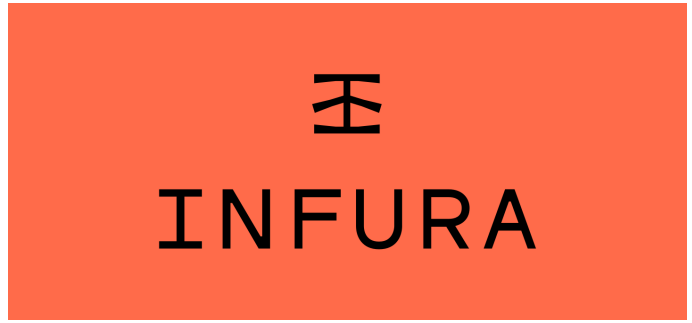
OpenZeppelin ERC 721 Token Deployment Tutorial

Made by Liam Stojanovic

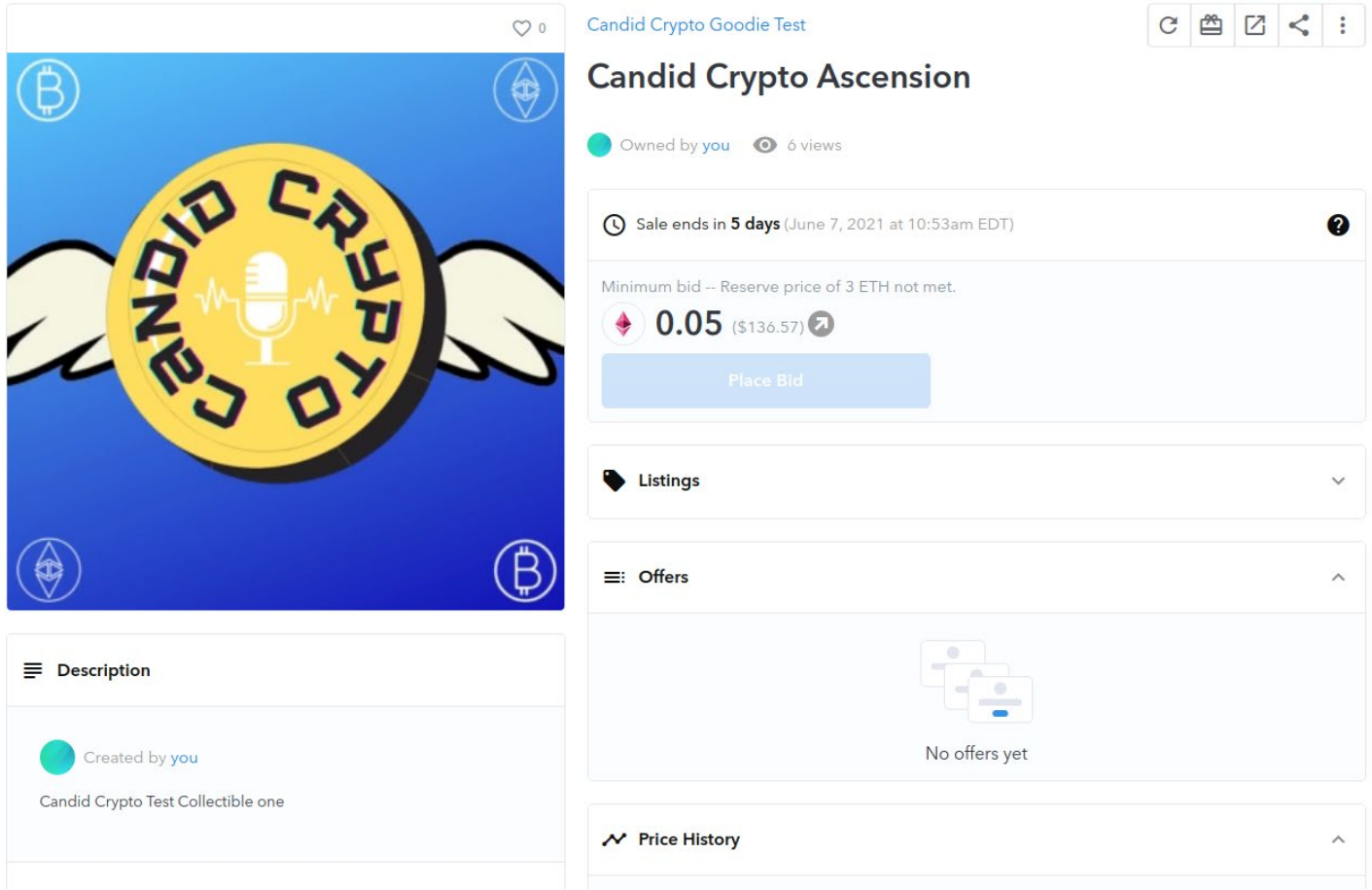
https://candidcryptopodcast.com



Summary

- This tutorial will allow you to launch an ERC721 token (a non-fungible token) on the Ethereum testnet, Rinkeby
- The NFT will be visible on Opensea.io
- In theory, this method can be ported to the Ethereum main network

End result


0
Candid Crypto Goodie Test
Candid Crypto Ascension
Owned by you
6 views
Sale ends in 5 days (June 7, 2021 at 10:53am EDT)
Minimum bid -- Reserve price of 3 ETH not met.
0.05 ($136.57)
Place Bid
Listings
Offers
No offers yet
Price History
Description
Created by you
Candid Crypto Test Collectible one

Motivation

- Building NFTs from scratch builds character
  - You learn how these things are really built and their mechanisms
- Filtering out the noise from endless (and oftentimes outdated) tutorials
- Not having to rely on services and their markups

High-level objectives

- Create development environment
- Edit/customize token fields
  - Add metadata to NFT
- Deploy smart contract to Rinkeby testnet

Requirements

- Familiarity with Linux command line
- A metamask wallet with testnet Ether (Rinkeby)
- RESTful API with token metadata
- Infura.io account

Software

- npm
  - Node package manager
- node
  - If installing from apt, ensure that you update npm and node to latest versions.
- truffle
  - Ethereum development tool
- openzeppelin contracts
  - The contracts leveraged to deploy the ERC721 token
- Metamask
  - Generate seed phrase, receive test network ether

Step 1: Create development environment

- Ubuntu VM
  - 20.04
- npm
  - https://www.npmjs.com/package/npm
- node
  - https://github.com/nodesource/distributions/blob/master/README.md

Step 2: Initialize project directory

$ mkdir helloNft && cd helloNft

$ npm init -y

Install openzeppelin contracts

$ npm install –save-dev @openzeppelin/contracts

$ npm install truffle

Step 2: Initialize project directory, cont…

Setup solidity project

$ npx truffle init

$ mkdir –p build/contracts/

$ cp node_modules/@openzeppelin/contracts/build/contracts/* build/contracts/

Step 3: Adjust parameters for ERC721 token

- Create migrations/2_deploy.js with the following content

```
const ERC721PresetMinterPauserAutoId = artifacts.require("ERC721PresetMinterPauserAutoId");


module.exports = function(deployer) {
    deployer.deploy(ERC721PresetMinterPauserAutoId, "Candid Crypto Goodie Test", "CCT", "https://candidcryptopodcast.com/api/tokens/");
};
```

Adjust the arguments to match your NFT info

Step 4: Prep for Testnet

Install @truffle/hdwallet-provider

npm install @truffle/hdwallet-provider

Create secrets.json

touch secrets.json

nano secrets.json

Step 4: Prep for Testnet, cont…

Configure truffle-config.js for Rinkeby network

Add HDWalletProvider

```
const HDWalletProvider = require('@truffle/hdwallet-provider’);
```

Add secrets

```
const secrets = require(“./secrets”);
```

Step 4: Prep for Testnet, cont…

Configure truffle-config.js for Rinkeby network

Add Rinkeby network config

```
networks: {
      rinkeby: {
            provider:  () => new HDWalletProvider(secrets.mnemonic,
`https://rinkeby.infura.io/v3/${secrets.projectId}`),
            network_id: 4,
            gas: 8500000,
            gasPrice: 1000000000,
            confirmations: 2,
            timeoutBlocks: 200,
            skipDryRun: true
      },
```

Step 5: Deploy

$ npx truffle console --network rinkeby

truffe(rinkeby)> migrate

Step 6: Mint

truffle(rinkeby)> nft = await ERC721PresetMinterPauserAutoId.deployed()

undefined

truffle(rinkeby)> accounts

truffle(rinkeby)> await nft.mint(“addr”)

truffle(rinkeby)> nft.address

Step 7: Validate deployment

- Opensea API for checking metadata
  - https://rinkeby-api.opensea.io/asset/[your_address]/0/validate/
- Your NFT should now appear on the Opensea testnet market

Further reading

- **Opensea docs metadata standards**
  - https://docs.opensea.io/docs/metadata-standards
- **JSON placeholder**
  - https://jsonplaceholder.typicode.com/

Sources

- **OpenZeppelin Docs**
  - https://docs.openzeppelin.com/contracts/3.x/erc721#Presets
- **OpenZeppelin forum post**
  - https://forum.openzeppelin.com/t/create-an-nft-and-deploy-to-a-public-testnet-using-truffle/2961
- **OpenSea Docs**
  - https://docs.opensea.io/docs/getting-started
- **OpenSea sample repository**
  - https://github.com/ProjectOpenSea/opensea-creatures
- **Json-server**
  - https://www.npmjs.com/package/json-server