NON-FUNGIBLE TOKEN (NFT) DEPLOYMENT IN ETHEREUM NETWORK

Abstract

The Non-Fungible Token (NFT) industry has exploded in recent years both in the industry and also in academic research. The notion of Non-Fungible Token (NFT) is based on an Ethereum token standard that aims to distinguish each token with identifiable indications. As their unique identifiers, these tokens can be coupled with virtual/digital attributes. All marked properties can be freely sold via NFTs, with customized values based on their ages, rarity, liquidity, and so on. It has significantly boosted the growth of the decentralized application (DApp) sector. The thousand fold return on its expanding market attracts a lot of interest all across the world. However, the NFT ecosystem is still in its early stages, and NFT technologies are still in their infancy. As a result, we are working on NFT deployment as well as learning about security problems and NFT-related topics. We have to also remember that newer applications of NFTs in the Metaverse is also making NFTs state of the art.

Keywords: Non-Fungible Token, Etherium, Security, Metaverse

Authors

Shayan Nandy
Computer Science and Engineering,
Institute of Engineering and Management,
Kolkata, West Bengal, India.
shayannandy99@gmail.com

Rajdeep Chakraborty
Computer Science and Engineering,
Institute of Engineering and Management,
Kolkata, West Bengal, India.
iamrajdeep2019@gmail.com

Indraneel Mukhopadhyay
Amity Institute of Information and Technology
Amity University
Kolkata, West Bengal, India.
imukhopadhyay@gmail.com

I. INTRODUCTION

NFTs are non-interchangeable units of data recorded on block chain that are immune to tampering, destruction, or duplication from a technical standpoint. NFTs can be validated using block chain technology, giving them extrinsic value. Non-Fungible Token (NFT) is a distinct type of digital asset on the Block chain. Unlike other types of crypto currency, these digital assets are not interchangeable. It enables ownership verification. Each NFT is a one-of-a-kind asset that cannot be duplicated or reproduced. NFT types include picture, file, cartoon, virtual real estate, pet, video, and trading card. A one-of-a-kind commercial platform for artists and producers to promote their products or collections, allowing newbies to get involved in the digital business industry.

In response, a revolution is allowing artists to create and profit from their work while also offering collectors with total transparency about the provenance and origin of their purchases. NFT assets include digital art, collectibles, creative extensions of music, a fusion of all three, and wholly original and unheard-of compositions. Creators continue to push the boundaries of innovation by changing NFTs in novel and imaginative ways. NFTs are significant because they offer up a whole new realm of digital ownership and commerce. Users can own digital goods that are truly unique and have actual worth with NFTs. This includes anything from digital paintings and treasures to virtual land and in-game stuff.

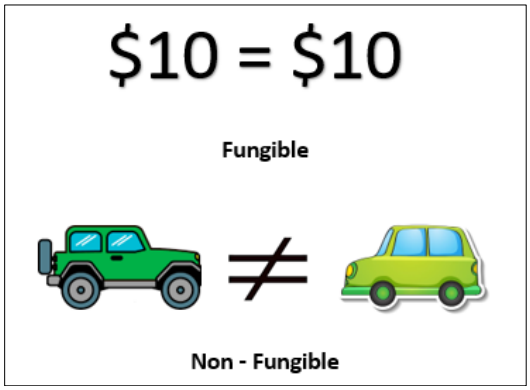

Figure 1: Fungible vs Non-Fungible Token

Before we do a deep dive into NFT lets clear what is the NFT Standard is all about. ERC-721 is a token standard for non-fungible tokens (NFTs), which are a sort of cryptographic token that, by definition, are not interchangeable. Because of its unique requirements, one token cannot be exchanged for another. As a result, NFTs are used to represent digital collectibles, game objects, digital art, event tickets, domain names, and physical asset ownership records. Figure 1 shows the difference between fungible vs non-fungible token.

Due to its non-fungible nature, this particular entity possesses the potential to evolve into a viable solution for safeguarding intellectual property. This is primarily attributed to its ability to facilitate seamless trading of complete transaction histories, its robust liquidity, and its seamless compatibility with other systems. Non-fungible tokens (NFTs) are comprised of codes that possess inherent value for buyers due to their relative scarcity as digital assets.

II. HOW TO LAUNCH AN NFT?

Creating an NFT on the Ethereum blockchain typically involves the following steps:

1. **Understanding NFTs and Ethereum:** Before creating an NFT, it's essential to grasp the concept of NFTs and how they function on the Ethereum blockchain. NFTs are unique digital assets that represent ownership of a specific item, such as digital art, music, videos, virtual real estate, collectibles, etc. Ethereum, being a smart contract platform, provides the infrastructure for creating and managing NFTs through its ERC-721 or ERC-1155 standards.

2. **Smart Contract Development:** To create an NFT, you'll need to develop a smart contract that complies with either the ERC-721 or ERC-1155 standard. ERC-721 is the older standard for unique tokens, while ERC-1155 is a newer standard that allows for both unique and fungible tokens to be managed under a single contract. You can use Solidity, Ethereum's most popular smart contract programming language, to create the contract.

3. **Setting Up Development Environment: Install** the necessary development tools and frameworks to work with Ethereum smart contracts. Common tools include Truffle, Ganache (for local development blockchain), and a code editor like Visual Studio Code.

4. **Code the NFT Smart Contract:** Write the Solidity code for the smart contract that adheres to the ERC-721 or ERC-1155 standard. The contract should define the properties of the NFT, including metadata (name, description, image, etc.), ownership, and transfer functions.

5. **Compile and Deploy:** Use Truffle or a similar tool to compile the smart contract code. After compiling, you'll deploy the contract to the Ethereum network. During deployment, you'll incur gas fees for contract execution, so make sure you have enough funds in your Ethereum wallet to cover these costs.

6. **Minting NFTs:** Minting refers to the process of creating new NFT tokens and assigning them unique properties. This process can be done by the contract owner or by others if the contract allows it. Minting typically involves calling a function within the smart contract that creates new NFTs and assigns them to specific addresses.

7. **Interacting with the NFT**: After minting, the NFTs are live on the Ethereum blockchain. Users can now buy, sell, or trade these tokens through various NFT marketplaces or platforms.

It's important to note that creating an NFT involves working with real cryptocurrencies like Ethereum, and any transactions on the blockchain will incur gas fees. Additionally, security and proper testing are crucial when dealing with smart contracts to prevent potential vulnerabilities.

III. IMPLEMENTED METHODOLOGY

The alchemy website was used to deploy the NFTs on the Ethereum Blockchain. The NFT API was used to do this. For making purchases, users use the use of the Metamask bitcoin wallet. Szabo's original concept of "smart contracts," designed to speed up, verify, or carry out digital agreements, is also a crucial component of implementing NFTs. Smart contracts present a unifying technique to construct apps across a wide range of industries and allow untrusted parties and decentralized players to conduct fair transactions without the need for a trusted third party.

Most NFT methods rely on Blockchain-based smart contract platforms to guarantee the correct execution of time-sensitive operations. Cryptocurrencies rely heavily on the concepts of Blockchain addresses and transactions. Like a bank account is used to spend money at a financial institution, a Blockchain address is a unique identifier for a user to transfer and receive the assets. The combination of a public key and a private key produces a string of alphanumeric characters with a predetermined length. NFTs can only be sent to new addresses when their owner verifies possession of the private key and signs the transaction. This common practice is reflected in the ERC-777 standard as a transaction made from a cryptocurrency wallet to a smart contract.. These smart contracts and setting up of address are done using scripting languages. We are using hardhat extension of JavaScript for setting up the contract. There are different token standards in NFTs namely ERC-20, ERC-721, and ERC-1155. We are using ERC-721 for our contract. ERC-721 deploys the group of non-fungible tokens in a single contract with the same configurations.

1. Alchemy is a Blockchain developer platform and application programming interface (API) that facilitates communication with the Ethereum Blockchain without the need for us to maintain our own nodes.

2. **Make an App (And an API Key):** From your Alchemy Dashboard, go to the "Create App" page by hovering over "Apps" and clicking "Create App." Make an App Give your app a name and a short description, and then select "Ethereum" for the Chain and "Goerli" for the Network.

3. **Sign Up for an Ethereum Account**: In order to make and receive purchases, you will need to sign up for an Ethereum account. Metamask, a software wallet for managing your Ethereum address, is what you need for this.

4. Start the project fresh by running npm init. We use hardhat on the inside of our project. This will produce a hardhat.config.js file for us, in which we can specify the project's configuration. Two new directories will be made to better organize our work in progress. Our code for the NFT smart contract will be stored here. There will be scripts for deploying and interacting with our smart contract stored in this directory.

5. **Draught the Agreement:** Sol controls who can access our smart contract, ensuring that only the smart contract's owner can mint NFTs. Our bespoke NFT smart contract follows our import declarations. Open Zeppelin contracts were passed down to us, and they contain most of the necessary methods for generating NFTs, such as ownerOf, which

returns the account that generated the NFT, and transferFrom, which allows the NFT's ownership to be transferred to a new account.

6. **Integrate Metamask:** To begin, open your project folder and install the dotenv package. Then, place your private Metamask key and the URL to the HTTP Alchemy API in an.env file that you create in the project's root directory. Hardhat's Plugin system makes it a breeze to add new tools and features, such as ether.js. Then, you must refresh Hardhat.config.js. Finally, compile the source code.

7. **Deploy:** Create a script that deploys the contract by writing To do this, open a new scripts/ folder, name the file deploy.js, and paste the code inside. When you initiate a deployment by calling deploy() on a ContractFactory, you will receive a Promise that resolves to a Contract. Each of our smart contract features corresponds to a method on this object. To get the contract deploy address, go back to the project's starting directory and run the code. Choose an address to create a contact with in Metamask, and the NFT smart contract will be activated. We will be making use of Alchemy Web3, a library that improves upon Web3 by adding features like automated retries and full WebSocket support. Make a mint-nft.js file and link it up with web3.

8. **Seize the Contract Application Programming Interface:** The interface for interacting with our smart contract. The MyNFT.json file contains an ABI that was generated automatically by Hardhat.

9. To make sure our NFT is genuinely decentralized, we will first set up the metadata for it using IPFS. To do this, we will utilize Pinata, a user-friendly API and toolkit for IPFS. You can use an image you upload to Pinata as the NFT's image asset. Information about the files you upload will be displayed in a table on the "Files" page.

10. **Make an Instance of Your Contract :** We need to make an instance of the contract in our code so that we can interact with it. The contract's address can be found in the deployment logs or on Etherscan by searching for the contract's genesis address. Make sure your.env file includes your public key.

11. **Make a Deal and Put Your Name on It:** To begin setting up the transaction details, read the PRIVATE_KEY and PUBLIC_KEY from the.env file. After drafting our transaction, we'll need to put our private key to work by signing it before sending it out.

12. Make a call to MintNFT and have the Pinata hash code and metadata.json sent through it. You may now install the NFT by running nodescript. The status of your transaction can be viewed in the Alchemy mempool.

    It’s now deployed AND minted with a NFT on the Ethereum blockchain.

IV. FUTURE SCOPE

Non-Fungible Token (NFT) is an emerging technology prevailing in the blockchain market. Introduction of Metaverse and the booming of NFTs in online games is a good opportunity for creators to create unique NFTs and make earning by deploying them. We

plan on learning blockchain, cybersecurity based on NFTs and also deploy as many NFTs as we can. We also plan on learning the challenges be it security or anything else that can hinder this NFT market.

REFERENCES

[1] G. Eason, B. Noble, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (references) J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
[2] I.S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.
[3] “NFTs: Applications and Challenges” by Wajiha Rehman, Hijab e Zainab, Narmeen Bawany. (2021).
[4] “Challenges of Implementing an NFT Marketplace” by Yash Mhatre, Devansh Dixit, Ritesh Salunkhe, Dr. Sanjay Sharma. (2022)
[5] “Research Study on Non-Fungible Token (NFT)” by Devanand DV.
[6] “Blockchain Technology and Non-Fungible Tokens (NFTs) in the Music Industry” by Christopher B. Schardt and Victoria S. Summers (2021)
[7] “Blockchain-Based Non-Fungible Tokens: Recent Developments and Future Challenges” by Vipul Goyal, Puneet Gupta, and Aayush Jain (2020)
[8] “NFTs: A Blockchain Solution for Digital Rights Management” by David Cohen-Tanugi and Céline Moille (2019)
[9] “NFTs - Applications and Challenges" by Wajiha Rehman, Hijab Zainab, Narmeen Zakaria Bawany Center for Computing Research, Department of Computer Science and Software Engineering and Technology, Jinnah University for Women 2021.
[10] "Tokens, Types, and Standards - Identification and Utilization in Ethereum,” by M. di Angelo and G. Salzer, 2020 IEEE International Conference on Decentralized Applications and Infrastructures, 2020.
[11] “Blockchain Technology and Implementation: A Systematic Literature Review," by H. R. Andrian, N. B. Kurniawan and Suhardi, 2018 International Conference on Information Technology Systems and Innovation (ICITSI), 2018.

