**VIVEKANAND EDUCATION SOCIETY’S INSTITUTE OF TECHNOLOGY**

**Department of Computer Engineering**

****

Course Report on

**Case Study: NFT GenMint**

Under the subject: Blockchain Honors Degree

Year: B.E. Semester : VII

**Submitted by**

Piyush Tilokani - D20A/63

Shashwat Tripathi - D20A/64

Under the guidance of

**Subject Teacher**

Mrs. Lifna C S

(2024-2025)

**Introduction**

The project focuses on integrating blockchain technology into the digital art space by creating an NFT minting platform that generates unique images through a stable diffusion model based on user prompts. Utilizing Ethereum's ERC721 standard via a smart contract, the platform allows users to mint non-fungible tokens (NFTs) that signify ownership and authenticity of these artworks. The smart contract ensures secure transactions, establishes a minting cost, and facilitates the storage of metadata linked to each NFT. By merging generative art with blockchain, this initiative aims to empower artists, enhance digital ownership, and explore new opportunities for creativity and commerce within the digital ecosystem.

**Project Overview**

**● Project Name: NFT GenMint**

**● Technology Stack:**

* Blockchain: Ethereum
* Smart Contract Framework: OpenZeppelin (for secure and efficient contract development)
* Languages: Solidity (for smart contract development), JavaScript (for frontend and interaction with the blockchain)
* Image Generation: Stable Diffusion model for creating images based on user-defined prompts
* Frontend Framework: React (or similar) for building user-interface

**Smart Contract Design**

**Smart Contract Name:** NFT

**Functionality:** The NFT smart contract enables users to mint non-fungible tokens (NFTs) that represent the digital artworks generated by the stable diffusion model. Key features include:

* **Minting Function:** Users can mint an NFT by providing a unique token URI that points to the generated artwork, while also paying a defined cost.
* **Token Tracking:** The contract maintains a counter for tracking the total number of minted NFTs.
* **Ownership and Transferability:** The smart contract ensures that ownership of each NFT is recorded on the blockchain, allowing for secure transfers and verifiable ownership.
* **Withdraw Function:** The contract owner can withdraw accumulated funds from minting, ensuring financial control over the project.

**CODE:**

**NFT.sol**

*// SPDX-License-Identifier: UNLICENSED*

*pragma solidity ^0.8.0;*

*import "@openzeppelin/contracts/utils/Counters.sol";*

*import "@openzeppelin/contracts/token/ERC721/ERC721.sol";*

*import "@openzeppelin/contracts/token/ERC721/extensions/ERC721URIStorage.sol";*

*contract NFT is ERC721URIStorage {*

*using Counters for Counters.Counter;*

*Counters.Counter private \_tokenIds;*

*address public owner;*

*uint256 public cost;*

*constructor(*

*string memory \_name,*

*string memory \_symbol,*

*uint256 \_cost*

*) ERC721(\_name, \_symbol) {*

*owner = msg.sender;*

*cost = \_cost;*

*}*

*function mint(string memory tokenURI) public payable {*

*require(msg.value >= cost);*

*\_tokenIds.increment();*

*uint256 newItemId = \_tokenIds.current();*

*\_mint(msg.sender, newItemId);*

*\_setTokenURI(newItemId, tokenURI);*

*}*

*function totalSupply() public view returns (uint256) {*

*return \_tokenIds.current();*

*}*

*function withdraw() public {*

*require(msg.sender == owner);*

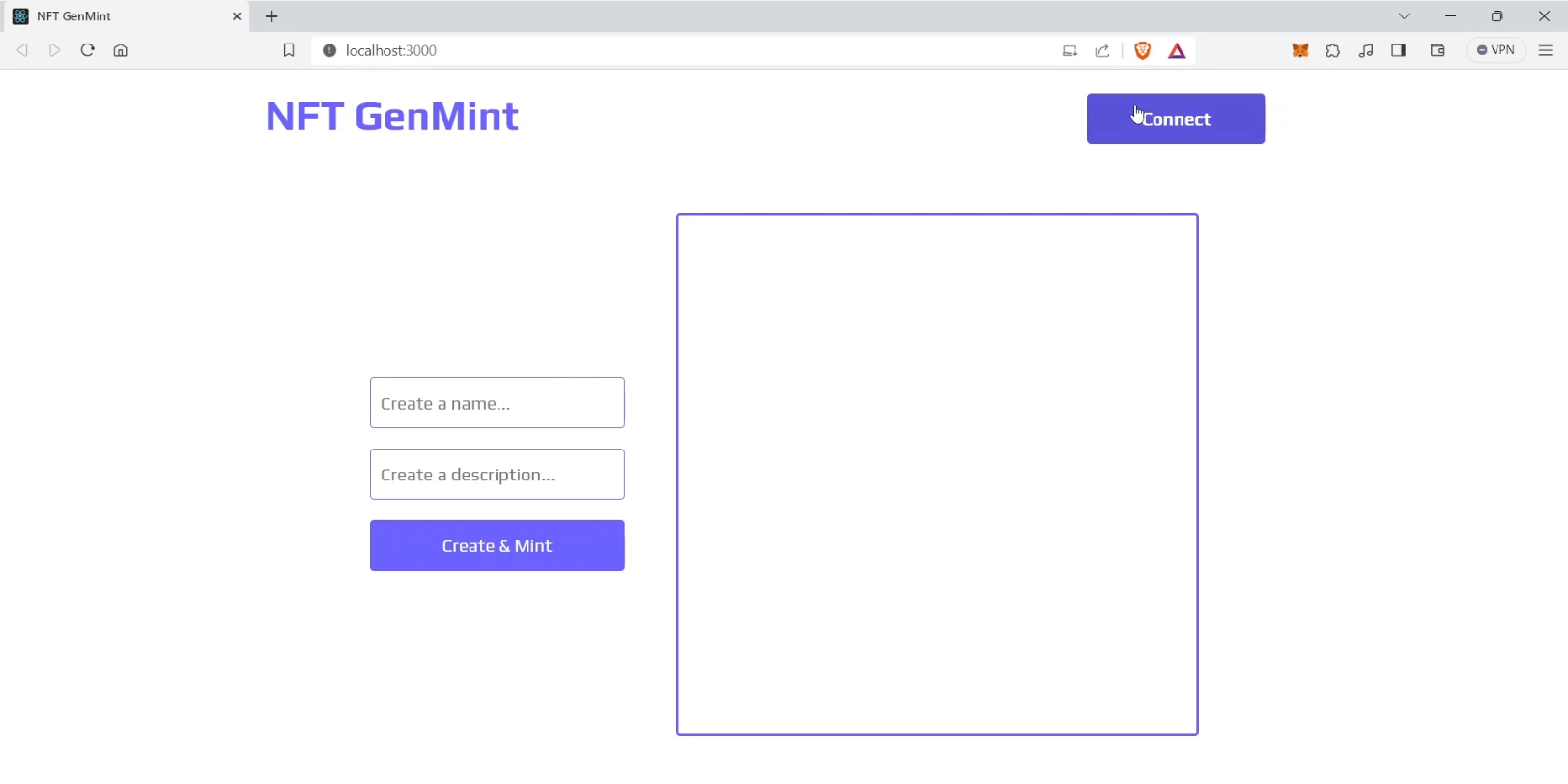
*(bool success, ) = owner.call{value: address(this).balance}("");*

*require(success);*

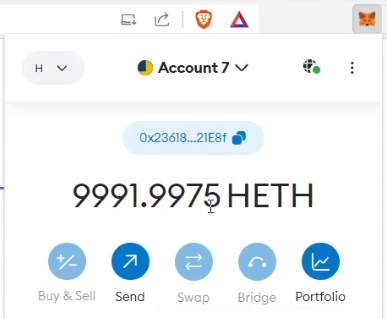
*}*

*}*

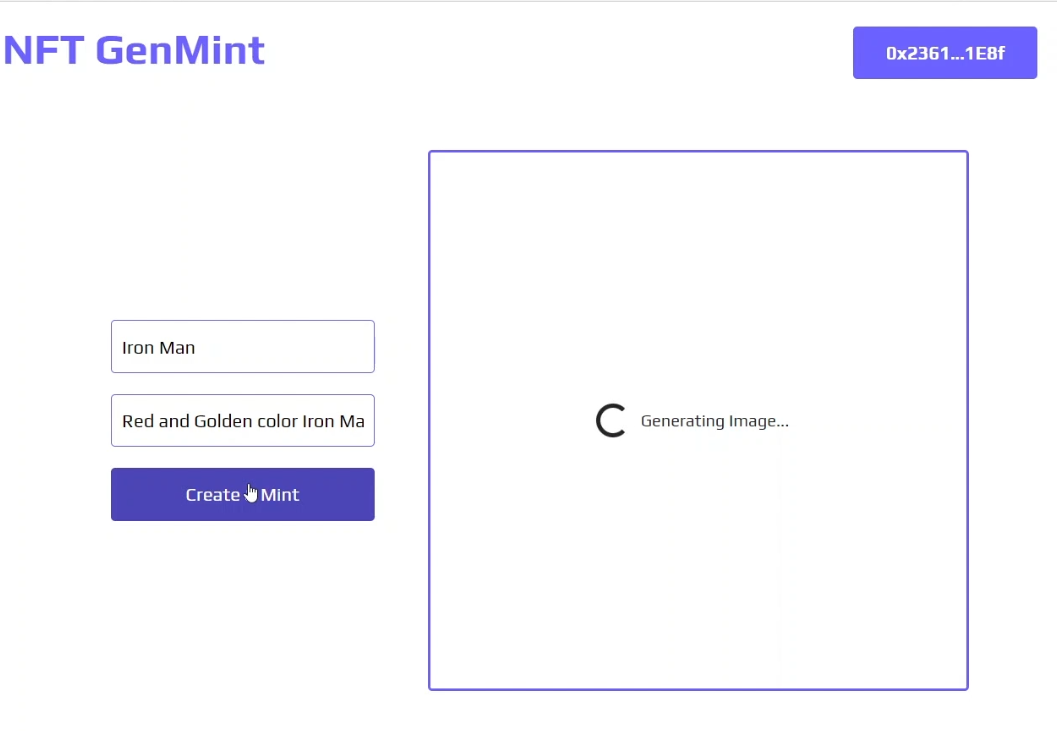
**OUTPUT:**



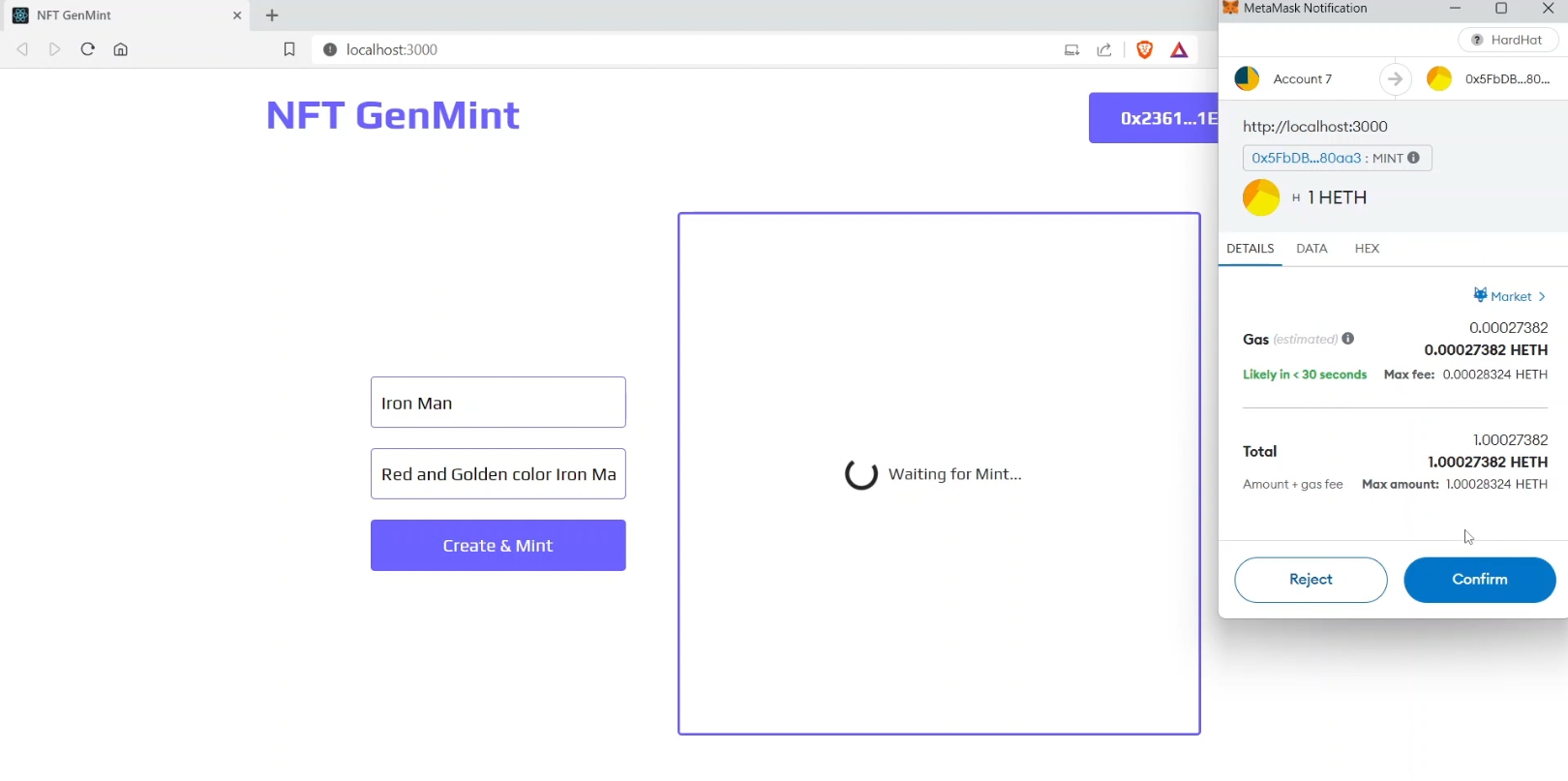
NFT generation page



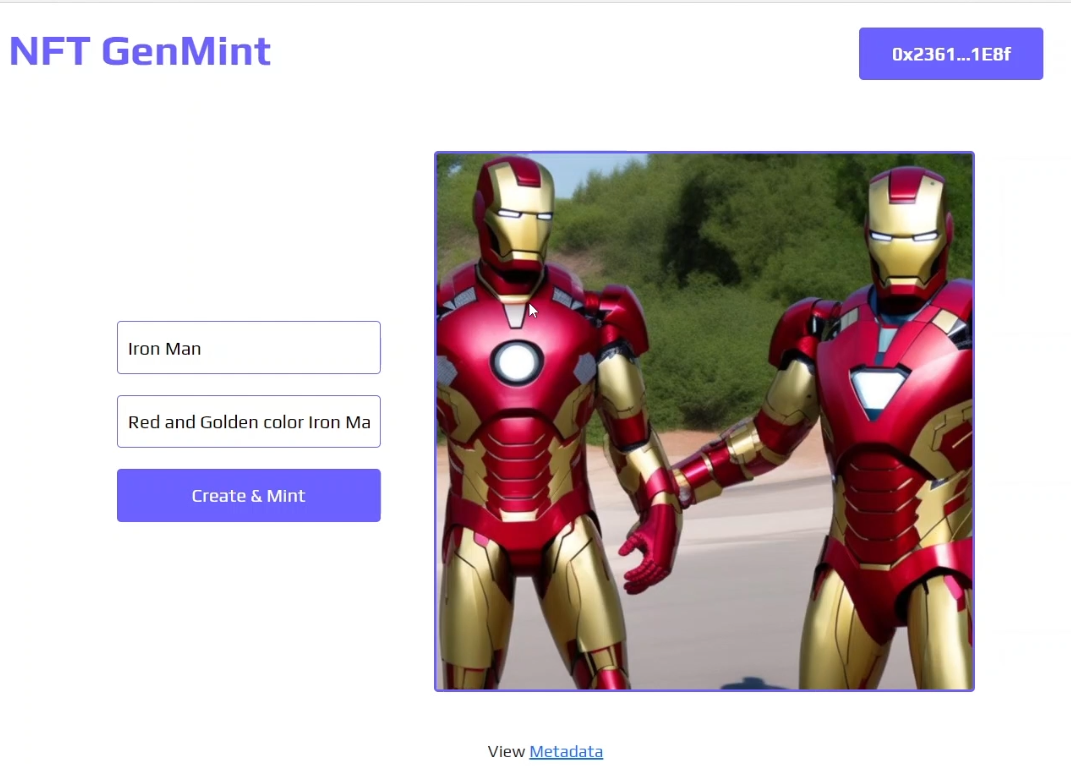
Metamask account connected



Giving prompt to generate NFT



Minting the NFT



NFT Generated and visible in Account connected

**Conclusion**

This project successfully merges blockchain technology with generative art through an NFT minting platform using a stable diffusion model. The NFT smart contract on Ethereum allows users to securely mint and trade unique digital artworks, ensuring authenticity and transparency. By empowering artists to monetize their creations, this initiative highlights the potential of blockchain in the digital art landscape.

**References**

●**Ethereum White Paper:** Vitalik Buterin. "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform."

**OpenZeppelin Documentation:** OpenZeppelin. "Secure Smart Contract Development."

**ERC721 Standard:** "ERC721 Non-Fungible Token Standard." Ethereum Improvement Proposals.

**Stable Diffusion:** Stability AI. "Stable Diffusion: A Deep Learning Text-to-Image Model."