

Chapman University Themed NFT Project

By: Luc Rieffel, Michelle Zhang, and Emily Nguyen



Project Overview

- A decentralized platform to buy Chapman University themed NFT's
- The NFT project would be controlled by students



Contract Details

- Deployed image to IPFS(InterPlanetary file system)
- Makes image accessible on Web3

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

import "openzeppelin-contracts/contracts/access/Ownable.sol";
import "openzeppelin-contracts/contracts/token/ERC20/IERC20.sol";
import "openzeppelin-contracts/contracts/token/ERC1155/ERC1155.sol";

contract PantherNFT is ERC1155, Ownable {
    uint256 public constant TOKEN_ID = 0;
    string public constant TOKEN_URI = "https://ipfs.io/ipfs/QmcNd58gbiV1pf6ChaPCDfRAZfGCuphRACpVDFtPiDtuFW?filename=pete_nft_image.png";

    constructor() ERC1155(TOKEN_URI) {}

    function mint(address account, uint256 amount) public onlyOwner {
        _mint(account, TOKEN_ID, amount, "");
    }

    function burn(address account, uint256 amount) public {
        require(account == msg.sender, "You can only burn your own tokens");
        _burn(account, TOKEN_ID, amount);
    }
}
```





Results...

- Our contract is not yet deployed
- Insufficient funds in all of our metamask balances
- We compiled and ran tests successfully but need to fix the funding error

```
at /Users/lucriffeel/Desktop/298/REAL/introduction-to-smart-contracts/node_modules/@ethersproject/providers/src.ts/json-rpc-provider.ts:265:24
at processTicksAndRejections (node:internal/process/task_queues:96:5) {
  reason: 'insufficient funds for intrinsic transaction cost',
  code: 'INSUFFICIENT_FUNDS',
  error: ProviderError: insufficient funds for gas * price + value
    at HttpProvider.request (/Users/lucriffeel/Desktop/298/REAL/introduction-to-smart-contracts/node_modules/hardhat/src/internal/core/providers/ht
tp.ts:88:21)
    at processTicksAndRejections (node:internal/process/task_queues:96:5)
    at async EthersProviderWrapper.send (/Users/lucriffeel/Desktop/298/REAL/introduction-to-smart-contracts/node_modules/@nomiclabs/hardhat-ethers/
src/internal/ethers-provider-wrapper.ts:13:20),
  method: 'sendTransaction',
  transaction: undefined
}
```

Findings

- Deploying a smart contract takes a significant amount of domain knowledge
- Environment setup is key
- Need sufficient funds in order to deploy contract





Real World Implications

- Security & transparency
- Showcase and sell artwork to a global audience
- Reduced costs & fees



Final Deployment

Contract

0xbb17416aE3c2FdAf829234Ef5466EEDf9bADC396

Source Code

More

Overview

ETH BALANCE
0 ETH

More Info

CONTRACT CREATOR
0xb2d3c4...a6EF410C at txn 0x69004d18aa2c74b4...

Multi Chain

MULTICHAIN ADDRESSES
N/A

Transactions

Token Transfers (ERC-20)

Contract

Events

Code

Read Contract

Write Contract

Search Source Code

Contract Source Code Verified (Exact Match)

Contract Name:

PantherNFT

Optimization Enabled:

No with 200 runs

Compiler Version

v0.8.10+commit.fc410830

Other Settings:

default evmVersion

Introduction to Smart Contracts

Goerli Testnet Only

Connected Wallet Address:
0xb2d3c40AD0484eED029FdF5ECf61EE97a6EF410C

Disconnect

0.0001

Send

In ETH

Transaction Hash: 0xd3bd28e956c8356c6813b63bfb01eff844f81530236ad4563f70c842d503a2cd