Centurion
University

School: .... Campus: ....

Academic Year: .... Subject Name: .... Subject Code: ....

Semester: .... Program: .... Branch: .... Specialization: ....

Date: ....

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiement : Mint it Yourself _ NFT Creation and Deployment

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

□ **Write the Smart Contract**

- Create an ERC721 contract using OpenZeppelin (ERC721URIStorage, Ownable).
- Add a mintTo(address to, string memory metadataURI) function.

□ **Compile the Contract**

- Open **Remix IDE**.
- Paste the contract code and click **Compile**.

□ **Deploy the Contract**

- Connect **MetaMask (Sepolia Testnet)**.
- Deploy with name, symbol, and your wallet address as the initial owner.
- Confirm deployment in MetaMask.

□ **Upload Metadata to IPFS (Pinata)**

- Upload NFT image (e.g., JPEG/PNG).
- Create a **metadata.json** file with:
- metadata.json to IPFS and copy its CID.

□ **Mint the NFT**

- Call mintTo your_wallet_address in Remix.
- Confirm transaction in MetaMask.

□ **Verify NFT in Wallet**

- Open MetaMask → NFTs tab.
- Import the contract address if NFT doesn’t appear automatically.

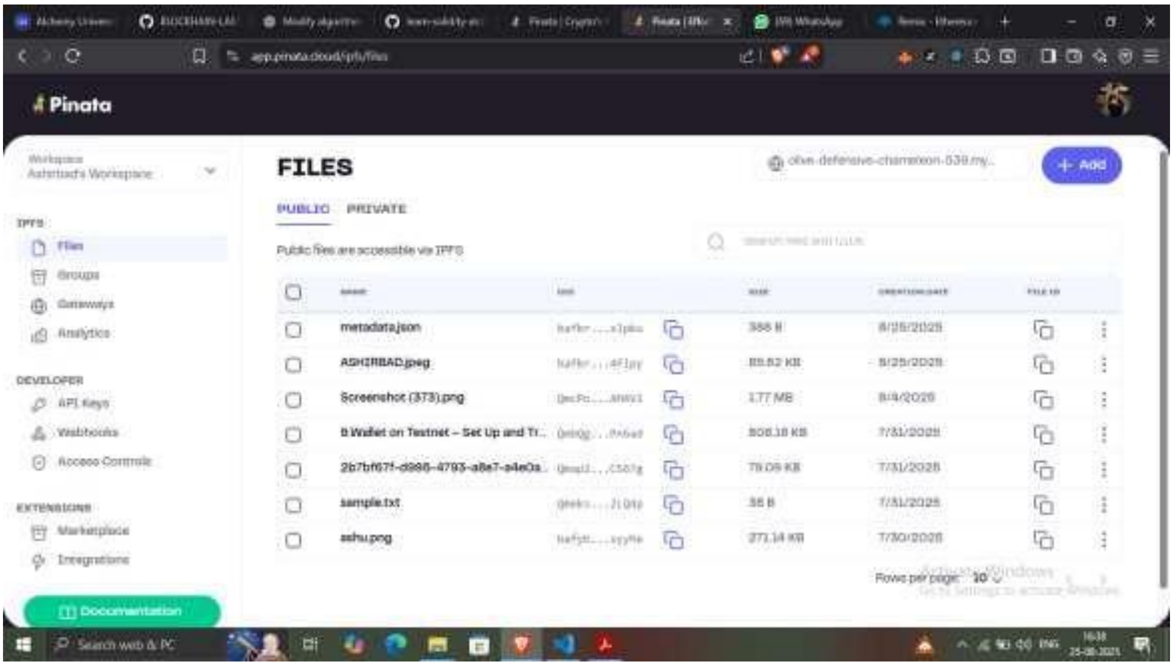
Software used

1. MetaMask Wallet
2. Remix IDE.
3. MS Word.
4. Brave for researching.
5. https://pinata.cloud/

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

* Implementation Phase: Final Output (no error)

- NFT successfully minted on **Sepolia Testnet**.
- Metadata stored on **IPFS (Pinata)**.
- NFT visible in MetaMask wallet under **NFTs section**.
- Example metadata includes **name, description, image, and attributes**.

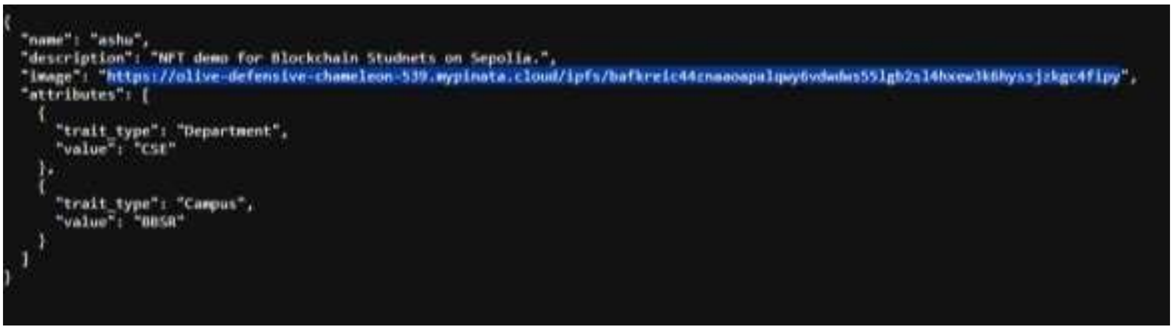


FILES

Public files are accessible via IPFS

□	NAME	CID	SIZE	CREATION DATE	FILE ID
□	metadata.json	bafkr....x1pku	388 B	8/25/2025	
□	ASHIRBAD.jpeg	bafkr....4f1py	85.52 KB	8/25/2025	
□	Screenshot (373).png	QmcPo....WRWV1	1.77 MB	8/4/2025	
□	8.Wallet on Testnet – Set Up and Tr...	QmbQg....fh6ad	806.18 KB	7/31/2025	
□	2b7bf67f-d986-4793-a8e7-a4e0a...	Qmupl....C5S7g	79.06 KB	7/31/2025	
□	sample.txt	Qmekv....Ji Q8p	38 B	7/31/2025	
□	ashu.png	bafyb....uyyhe	271.14 KB	7/30/2025	

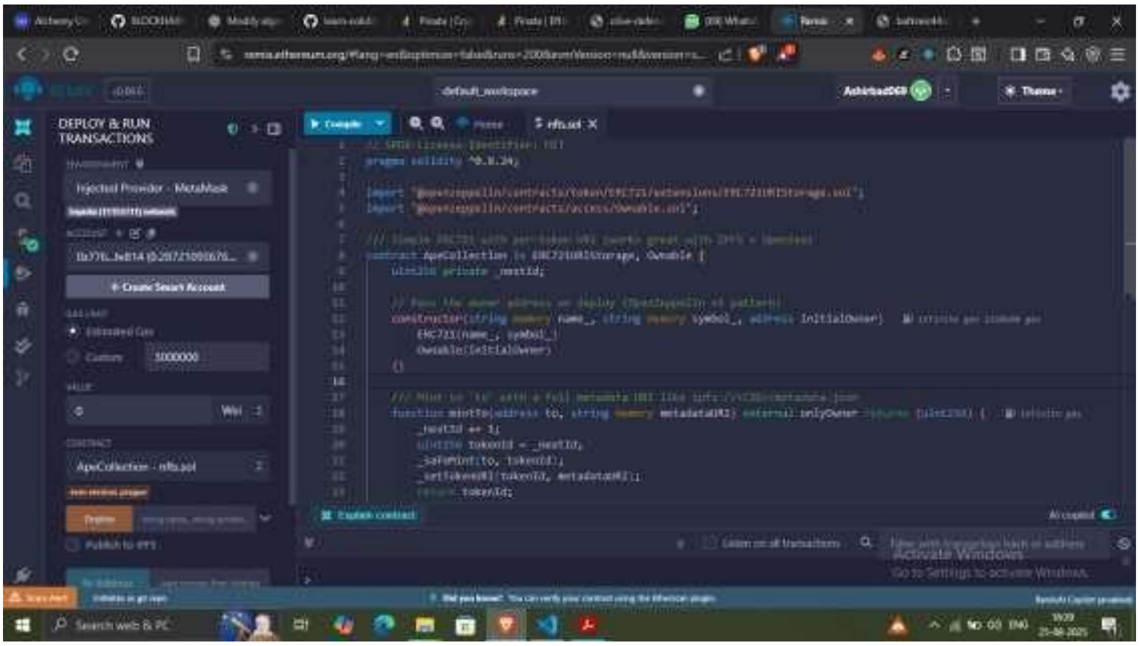
Rows per page: 10



```
{
  "name": "ashu",
  "description": "NFT demo for Blockchain Studnets on Sepolia.",
  "image": "https://olive-defensive-chameleon-539.mypinata.cloud/ipfs/bafkreic44znaaoapalqwy6vdwdws55lgb2sl4hxew3k6hyssjzkgc4fipy",
  "attributes": [
    {
      "trait_type": "Department",
      "value": "CSE"
    },
    {
      "trait_type": "Campus",
      "value": "BBSR"
    }
  ]
}
```

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

* Implementation Phase: Final Output (no error)



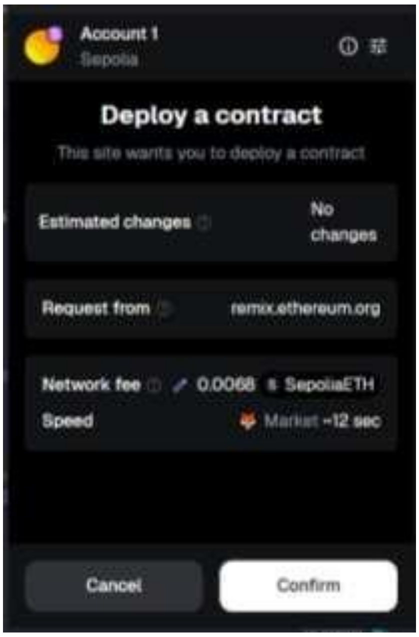
```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.24;

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

/// Simple ERC721 with per-token URI (works great with IPFS + OpenSea)
contract ApeCollection is ERC721URIStorage, Ownable {
    uint256 private _nextId;

    // Pass the owner address on deploy (OpenZeppelin v5 pattern)
    constructor(string memory name_, string memory symbol_, address initialOwner)
        ERC721(name_, symbol_)
        Ownable(initialOwner)
    {}

    /// Mint to `to` with a full metadata URI like ipfs://<CID>/metadata.json
    function mintTo(address to, string memory metadataURI) external onlyOwner returns (uint256) {
        _nextId += 1;
        uint256 tokenId = _nextId;
        _safeMint(to, tokenId);
        _setTokenURI(tokenId, metadataURI);
        return tokenId;
```



Account 1
Sepolia

Deploy a contract

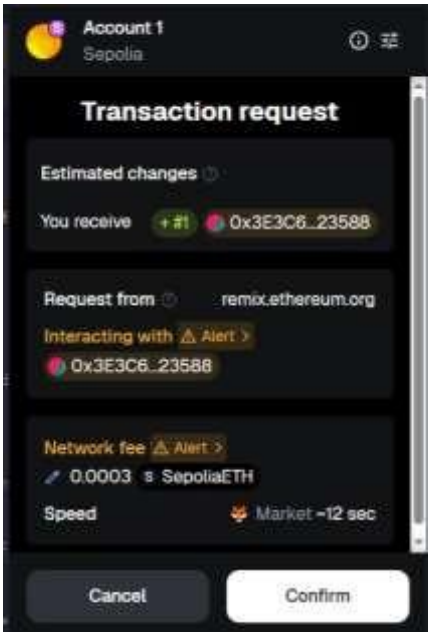
This site wants you to deploy a contract

Estimated changes: No changes

Request from: remix.ethereum.org

Network fee: 0.0068 SepoliaETH

Speed: Market ~12 sec



Account 1
Sepolia

Transaction request

Estimated changes

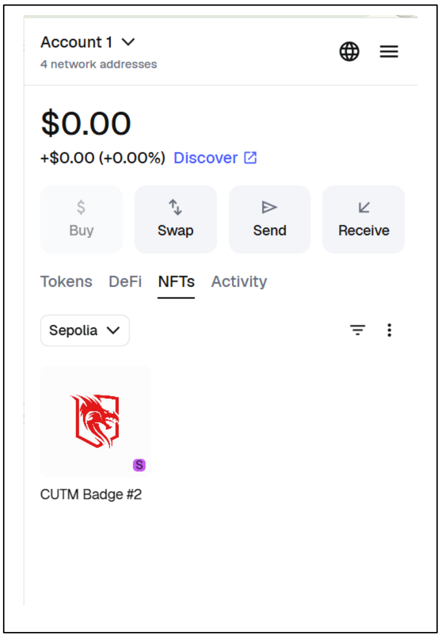
You receive: +#1 0x3E3C6...23588

Request from: remix.ethereum.org

Interacting with ⚠ Alert: 0x3E3C6...23588

Network fee ⚠ Alert: 0.0003 SepoliaETH

Speed: Market ~12 sec



Account 1
4 network addresses

$0.00
+$0.00 (+0.00%) Discover

CUTM Badge #2

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

* Observations:

□ Smart contract + IPFS are both essential for NFT creation.

□ mintTo assigns ownership and metadata to a unique token ID.

□ Metadata must follow ERC721 JSON format.

□ Sepolia ETH is required for transactions.

□ IPFS ensures decentralized, permanent storage.

□ NFTs are **non-fungible** and unique.

□ Wallets like MetaMask allow easy viewing of minted NFTs.

ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	**50**		

Signature of the Student:

Name :

Regn. No. :

Signature of the Faculty:

*
As applicable according to the experiment.

Two sheets per experiment (10-20) to be used.