Centurion University
Shaping Lives... Empowering Communities...

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

Algorithm:

1. Upload the image to Pinata and copy the generated IPFS link.
2. Create a metadata JSON file, include the image link inside it, then upload the file to Pinata and copy the metadata IPFS link.
3. Write and deploy an NFT smart contract on a blockchain (e.g., Ethereum Sepolia Testnet).
4. Mint the NFT by calling the `mint` function with:
5. Your wallet address
6. The metadata IPFS URI
7. Open MetaMask and check the **NFTs** section to confirm the NFT.

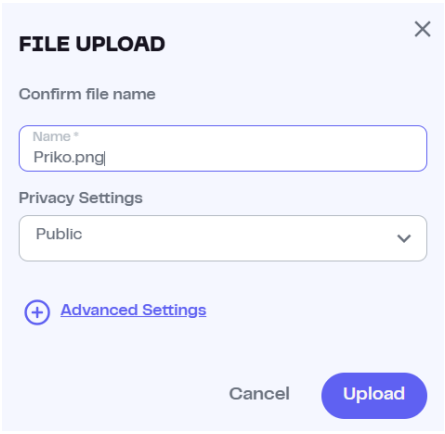
* Softwares used

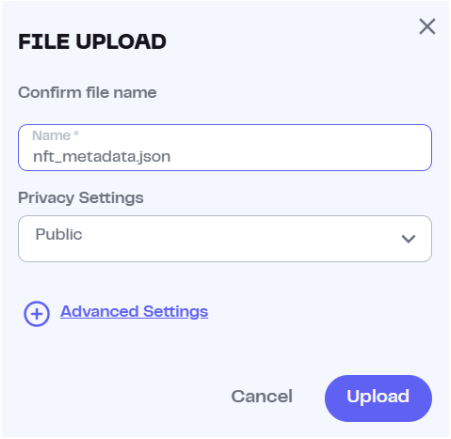
1.Pinata
2.MetaMask
3.Remix IDE
4.Ethereum Test Network (Sepolia)

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

* Implementation Phase: Final Output (no error)

First go to your pinata account and upload your image/logo of NFT and then write a metadata of your NFT in.json file and also add the .json file to pinata account






```
{
  "name": "Priko",
  "description": "My digital picture",
  "image": "https://pink-junior-landfowl-354.mypinata.cloud/ipfs/bafkreig54ixi4eonagy6s324ywp5iubuyrlqjtpgwuv37nbkpraza6b3ia",
  "attributes": [
    {
      "trait_type": "Creator",
      "value": "Priko"
    },
    {
      "trait_type": "Category",
      "value": "Personal"
    },
    {
      "trait_type": "Type",
      "value": "Digital Picture"
    }
  ]
}
```

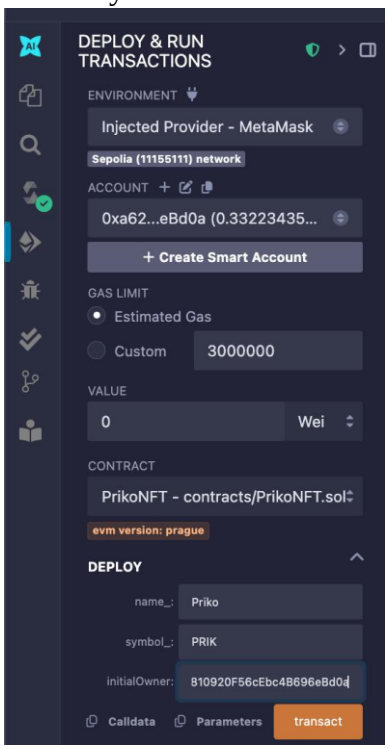
After uploading the logo and .json file in remix IDE write your Smart contract for NFT creation.

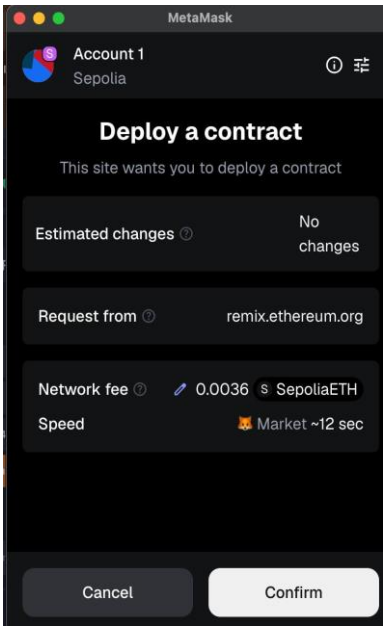
```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.24;

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 PrikoNFT is ERC721URIStorage, Ownable {
    uint256 private _nextId;

    // Pass the owner address on deploy (OpenZeppelin v5 pattern)
    constructor(string memory name_, string memory symbol_, address initialOwner)    infinite gas 1930600 gas
        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) {    infinite gas
        _nextId += 1;
        uint256 tokenId = _nextId;
        _safeMint(to, tokenId);
        _setTokenURI(tokenId, metadataURI);
        return tokenId;
    }

    function totalMinted() external view returns (uint256) {    2454 gas
        return _nextId;
    }
}
```

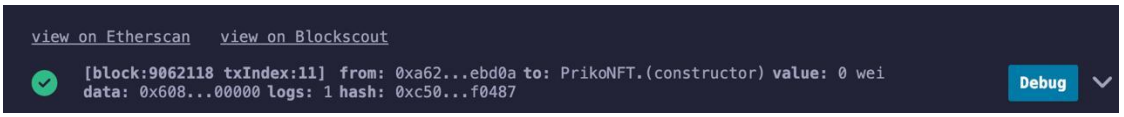

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

* Implementation Phase: Final Output (no error)

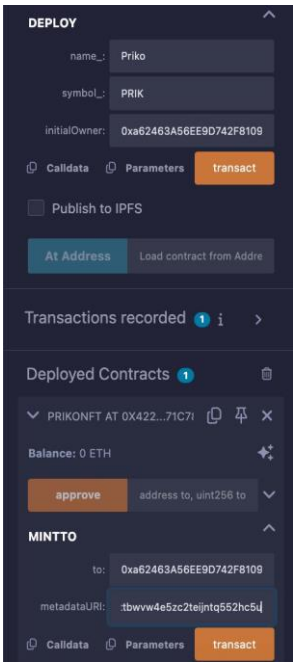
After compile the .sol file then deploy the smart contract in deploy section give name of the token and Symbl and the address of your wallet







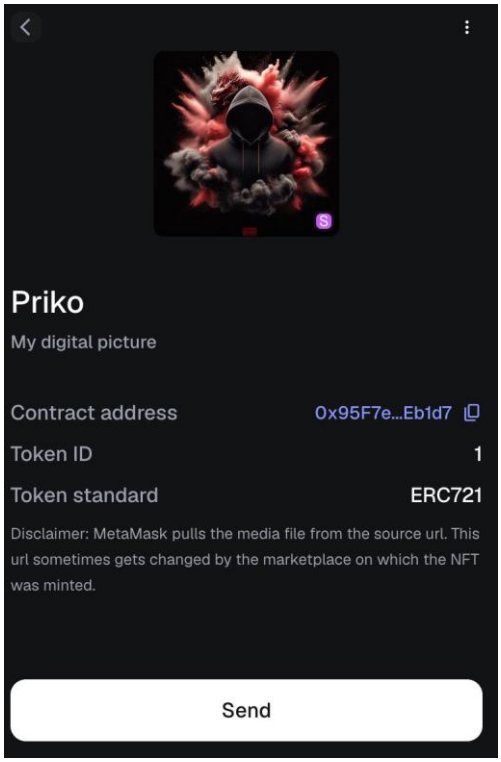
After Deployment then in MINT To section give your wallet address and metadata URI



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

* Implementation Phase: Final Output (no error)

After this you can see the NFT is successfully added to the meta maskwallet



* Observations

The NFT image and metadata were uploaded to IPFS via Pinata, ensuring decentralized storage. A smart contract was then deployed on the Sepolia Testnet using Remix and MetaMask. During the minting process, the NFT was successfully linked to the specified wallet address through the IPFS metadata URI. Finally, the NFT appeared in MetaMask, confirming its successful creation and deployment.

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.*