

Using File Connector Js

Team id	NM2023TMID04387
Project name	Project – Ethgeruem Deceentralised Smart Contract

```
const { ethers } = require("ethers");
```

```
const abi = [  
  {  
    "inputs": [],  
    "stateMutability": "nonpayable",  
    "type": "constructor"  
  },  
  {  
    "anonymous": false,  
    "inputs": [  
      {  
        "indexed": true,  
        "internalType": "address",  
        "name": "owner",  
        "type": "address"  
      },  
      {  
        "indexed": false,  
        "internalType": "string",  
        "name": "identityId",  
        "type": "string"  
      },  
      {  
        "indexed": false,  
        "internalType": "string",
```

```
    "name": "name",
    "type": "string"
  },
  {
    "indexed": false,
    "internalType": "string",
    "name": "email",
    "type": "string"
  },
  {
    "indexed": false,
    "internalType": "uint256",
    "name": "registrationTimestamp",
    "type": "uint256"
  }
],
"name": "IdentityRegistered",
"type": "event"
},
{
  "inputs": [
    {
      "internalType": "address",
      "name": "userAddress",
      "type": "address"
    }
  ],
  "name": "getIdentityDetails",
  "outputs": [
    {
      "internalType": "string",
      "name": "",
```

```
    "type": "string"
  },
  {
    "internalType": "string",
    "name": "",
    "type": "string"
  },
  {
    "internalType": "string",
    "name": "",
    "type": "string"
  },
  {
    "internalType": "string",
    "name": "",
    "type": "string"
  },
  {
    "internalType": "uint256",
    "name": "",
    "type": "uint256"
  }
],
"stateMutability": "view",
"type": "function"
},
{
  "inputs": [
    {
      "internalType": "address",
      "name": "",
      "type": "address"
    }
  ]
}
```

```
    }
  ],
  "name": "identities",
  "outputs": [
    {
      "internalType": "string",
      "name": "identityId",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "name",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "email",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "contactAddress",
      "type": "string"
    },
    {
      "internalType": "uint256",
      "name": "registrationTimestamp",
      "type": "uint256"
    }
  ],
  "stateMutability": "view",
  "type": "function"
```

```
},
{
  "inputs": [],
  "name": "owner",
  "outputs": [
    {
      "internalType": "address",
      "name": "",
      "type": "address"
    }
  ],
  "stateMutability": "view",
  "type": "function"
},
{
  "inputs": [
    {
      "internalType": "string",
      "name": "identityId",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "name",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "email",
      "type": "string"
    }
  ],
  {
```

```
    "internalType": "string",
    "name": "_address",
    "type": "string"
  }
],
"name": "registerIdentity",
"outputs": [],
"stateMutability": "nonpayable",
"type": "function"
}
]
```

```
if (!window.ethereum) {
  alert('Meta Mask Not Found')
  window.open("https://metamask.io/download/")
}
```

```
export const provider = new
ethers.providers.Web3Provider(window.ethereum);
export const signer = provider.getSigner();
export const address =
"0x418e6373e103F7bf5D73f51b6A43a0cA2EC9d9a5"
```

```
export const contract = new ethers.Contract(address, abi,
signer)
```