



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 Experiment :** Read the Chain – Web3.js Basics

### Objective/Aim:

To learn how to **connect a web frontend (React)** to an **Ethereum smart contract** using **Web3.js**, and read data (like count value) from the blockchain.

### Apparatus/Software Used:

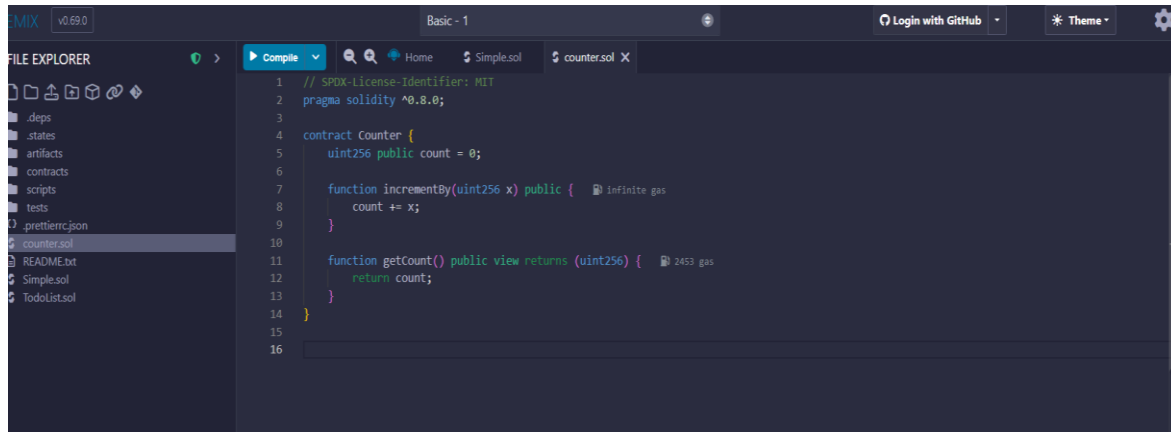
- Laptop/PC
- MetaMask
- Remix IDE
- Web3.js
- Vs code
- React

### Theory/Concept:

- 1. Blockchain:** A decentralized database where data (like variables in a smart contract) is stored across a network.
- 2. Smart Contract:** A program deployed on blockchain that holds logic and variables.
- 3. ABI (Application Binary Interface):** A JSON file that helps frontend apps understand how to talk to a contract.
- 4. Web3.js:** A JavaScript library that connects web apps to Ethereum.
- 5. MetaMask:** A browser extension wallet to sign blockchain transactions.
- 6. React:** A JavaScript library for building user interfaces.

## Procedure:

1. Go to remix Ethereum IDE and create a new file (eg. Counter.sol) after creating file write the smart Contract.

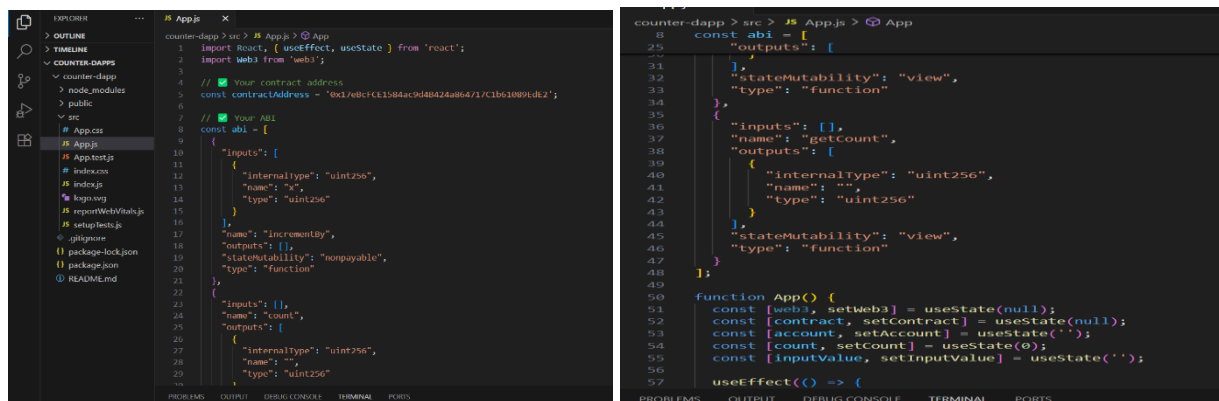


```

1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract Counter {
5     uint256 public count = 0;
6
7     function incrementBy(uint256 x) public {
8         count += x;
9     }
10
11     function getCount() public view returns (uint256) {
12         return count;
13     }
14 }
15
16

```

2. After that compile it and deploy using Injected Web3 with MetaMask.
3. After deployment Copy the deployed contract address and ABI.
4. Now go to Vs code and write frontend in App.js and past the contract address and ABI.

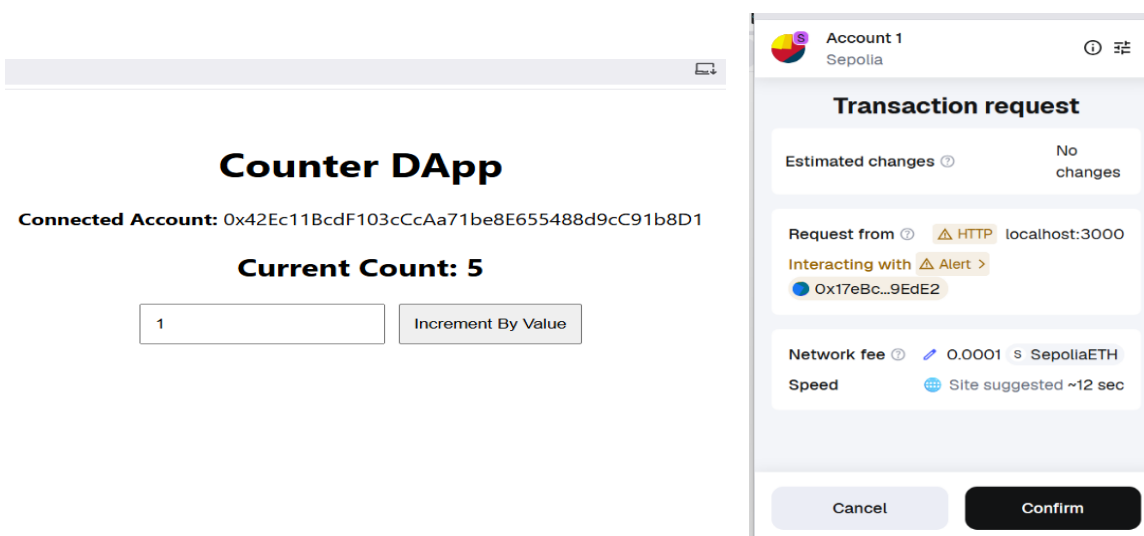


```

1 import React, { useEffect, useState } from 'react';
2 import Web3 from 'web3';
3
4 // Your contract address
5 const contractAddress = '0x17eBc...9EdE2';
6
7 // Your ABI
8 const abi = [
9     {
10         "inputs": [
11             {
12                 "internalType": "uint256",
13                 "name": "x",
14                 "type": "uint256"
15             }
16         ],
17         "name": "incrementBy",
18         "outputs": [],
19         "stateMutability": "nonpayable",
20         "type": "function"
21     },
22     {
23         "inputs": [],
24         "name": "count",
25         "outputs": [
26             {
27                 "internalType": "uint256",
28                 "name": "",
29                 "type": "uint256"
30             }
31         ],
32         "stateMutability": "view",
33         "type": "function"
34     }
35 ];
36
37 function App() {
38     const [web3, setWeb3] = useState(null);
39     const [contract, setContract] = useState(null);
40     const [account, setAccount] = useState('');
41     const [count, setCount] = useState(0);
42     const [inputValue, setInputValue] = useState('');
43
44     useEffect(() => {
45         // ...
46     });
47
48     function App() {
49         // ...
50     }
51 }
52
53

```

5. After this go to the terminal and run the code.
6. In the output first we have to connect with MetaMask then enter a number in input then Click "Increment By Value" then confirm the transaction in MetaMask and see the updated value from the blockchain.



**Counter DApp**

Connected Account: 0x42Ec11BcdF103cCcAa71be8E655488d9cC91b8D1

Current Count: 5

1  Increment By Value

**Transaction request**

Estimated changes: No changes

Request from: HTTP localhost:3000

Interacting with: Alert

0x17eBc...9EdE2

Network fee: 0.0001 SepoliaETH

Speed: Site suggested ~12 sec

Cancel Confirm

## Procedure:

### Counter DApp

Connected Account: 0x42Ec11BcdF103cCcAa71be8E655488d9cC91b8D1

Current Count: 7

## Observation:

Using Web3.js, we were able to successfully connect to the blockchain, read smart contract data, and verify that the expected output matched the deployed contract's state.

### ASSESSMENT

| Rubrics  | Full Mark | Marks Obtained | Remarks |
|--|-----------|----------------|---------|
| Concept  | 10        |                |         |
| Planning and Execution/<br>Practical Simulation/ Programming | 10        |                |         |
| Result and Interpretation                                    | 10        |                |         |
| Record of Applied and Action Learning                        | 10        |                |         |
| Viva   | 10        |                |         |
| <b>Total</b>   | <b>50</b> |                |         |

**Signature of the Student:**

Name :

Regn. No. :

**Signature of the Faculty:**

Page No.....

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