

**Name of the Experiement : Connect the Dots – Ethers.js and MetaMask UI**

**Coding Phase : Pseudo Code/Flow Chart/Algorithm**

1. Start React project using npx create-react-app.
2. Install ether.js library.
3. Create .env file with:
4. In app.js:
   * Import Web3 and connect to MetaMask.
   * Load contract using ABI & address from .env.
   * Fetch storedData using contract.methods.get().call().
   * Send transaction using contract.methods.set(value).send().

5. Test the frontend by setting and getting values.

**Apparatus/Software Used**:

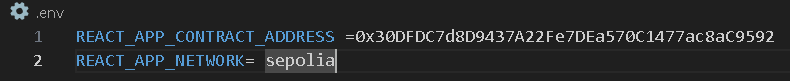
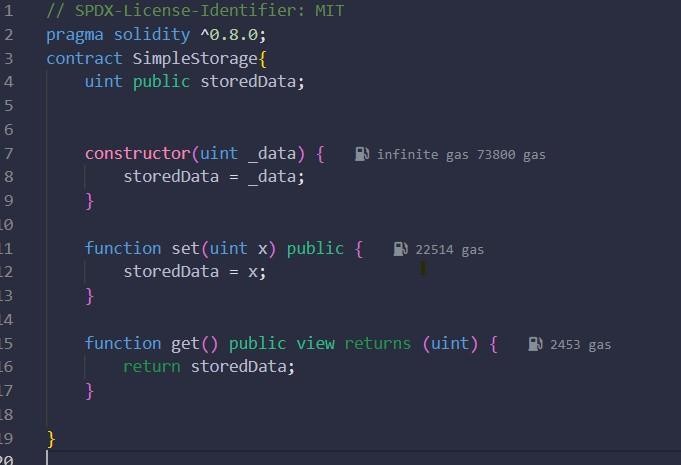
* Node.js & npm
* React.js
* MetaMask
* **Network**: Sepolia Testnet

**Testing Phase:**

* Deployed SimpleStorage contract to Sepolia using Remix.
* Noted the contract address & ABI.
* Created .env file to store sensitive data.
* Connected frontend to MetaMask.
* Verified:
* Reading stored value works.
* Writing new value updates blockchain data.

**Implementation Phase: Final Output (no error)**

Step 1: Create a smart contract in remix IDE.

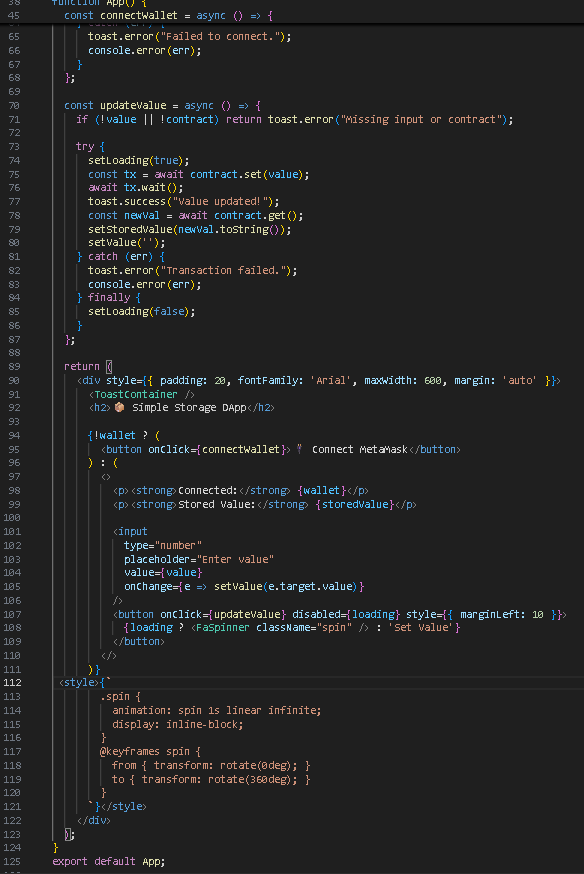
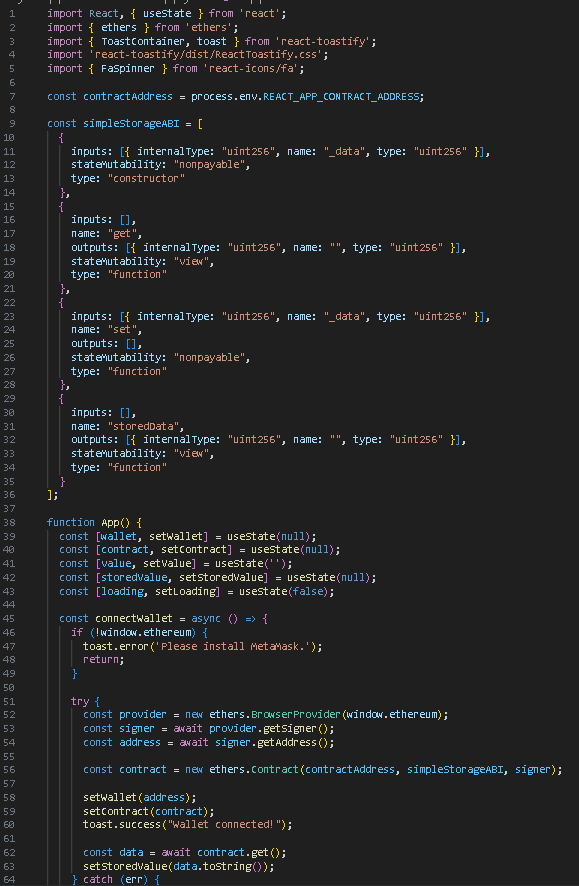


Step 2: Create a React app in VS Code.

* Open VS Code.
* Open a terminal inside of VS Code.
* Run this code (npx create-react-app simple-storage).
* Then run cd simple-storage Step 3: Create a .env File.
* Write The deployed contract address from Remix or blockchain explorer.

Step 4: Connect in src/App.js

* Replace App.js with something like:



**Step 5: Run the App**

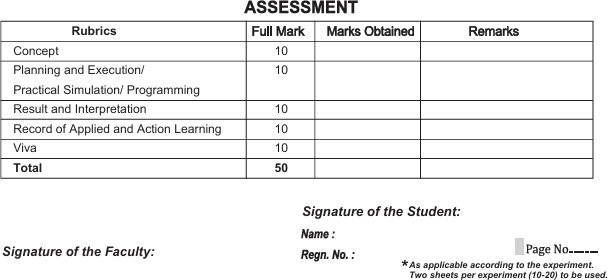
* + In terminal: npm start

**Step 6:** After run this open React app at [**http://localhost:3000**](http://localhost:3000/)

* + Then connect the meta mask.
  + Then Enter some value and set value .

**Observations :**

* Ether.js successfully connected frontend to blockchain.
* MetaMask allowed account access and transaction confirmation.
* Updating values from frontend reflected immediately on blockchain

****