project Development Phase

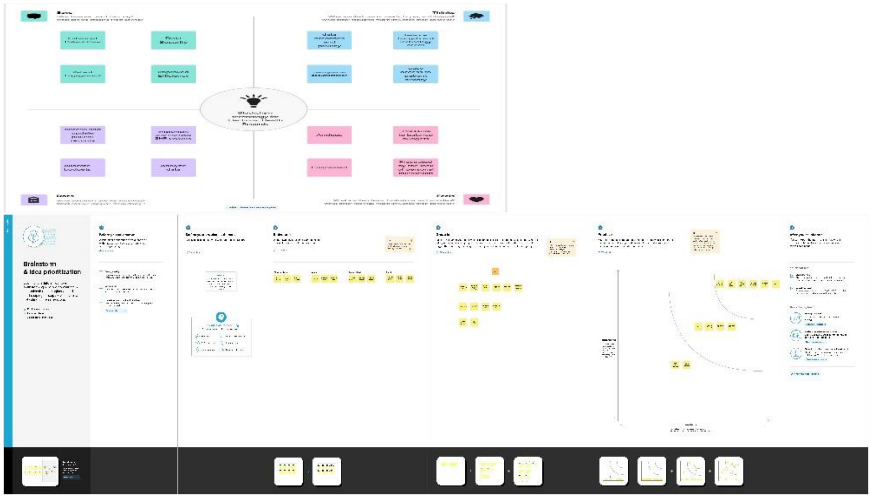
Model Performance Test

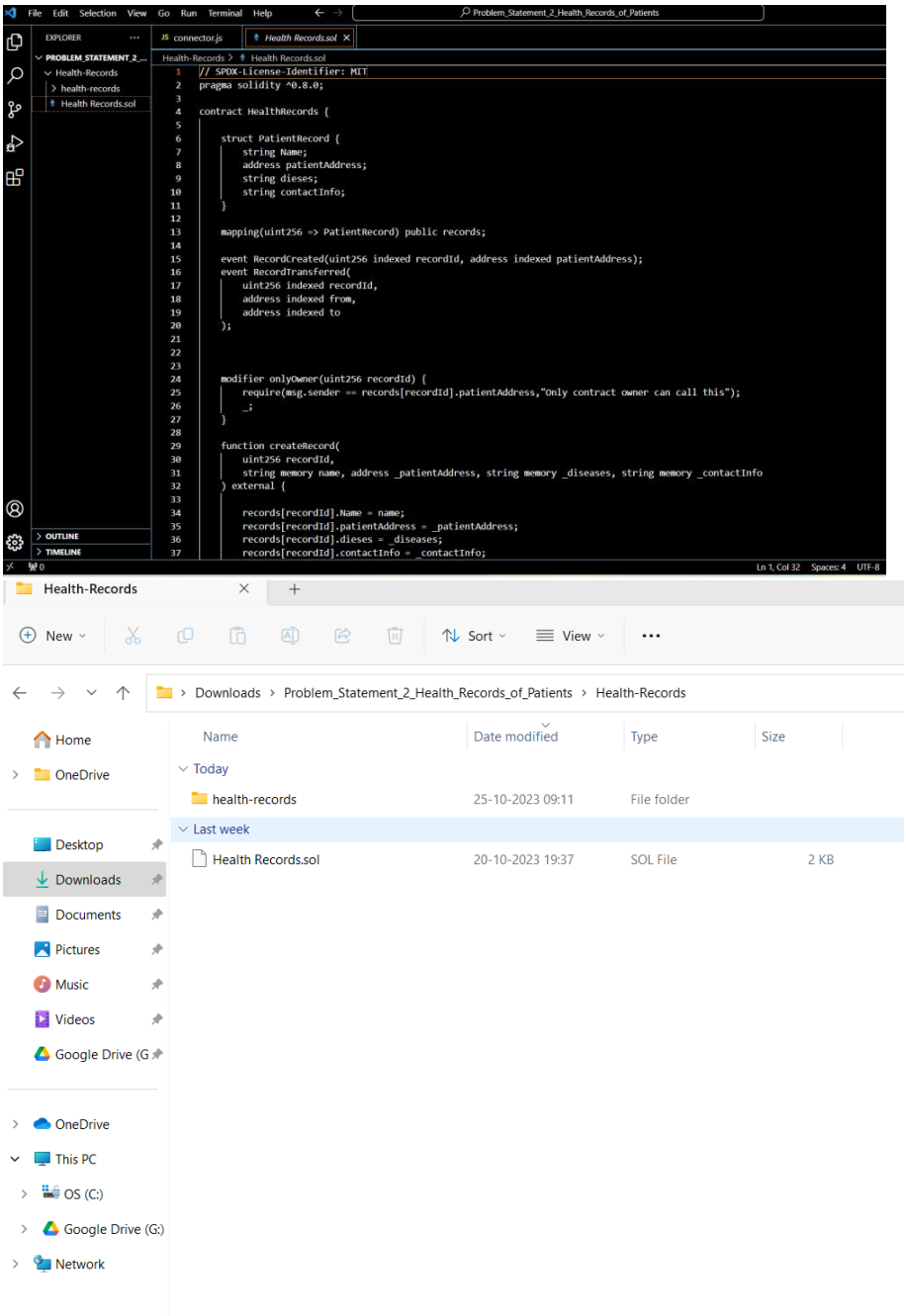
Date	25-10-2023
Team ID	NM2023TMID00147
Project Name	Blockchain Technology for Electronic Health Records
Maximum Marks	10 Marks

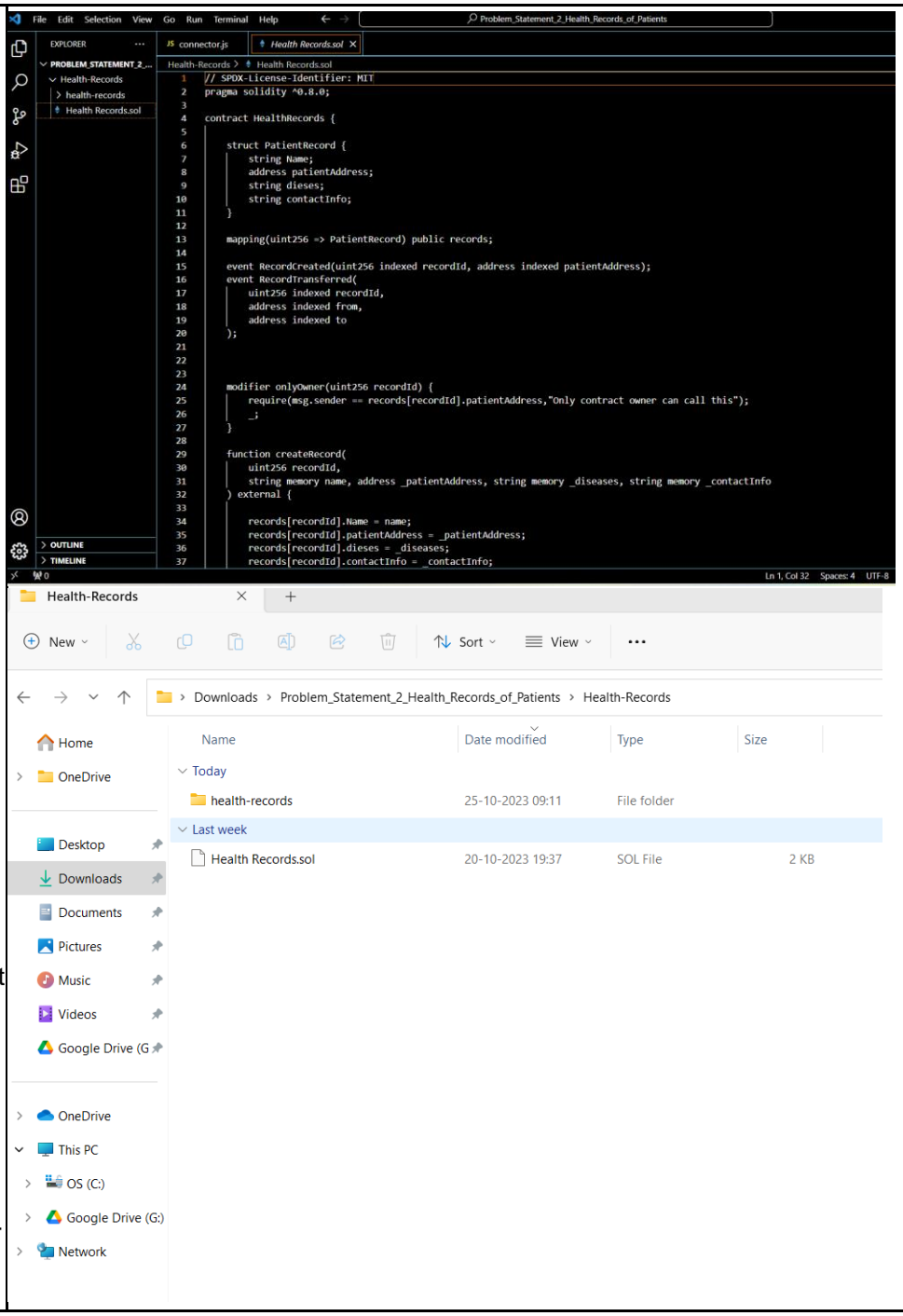
Model Performance Testing:

Project team shall fill the following information when working for blockchain.

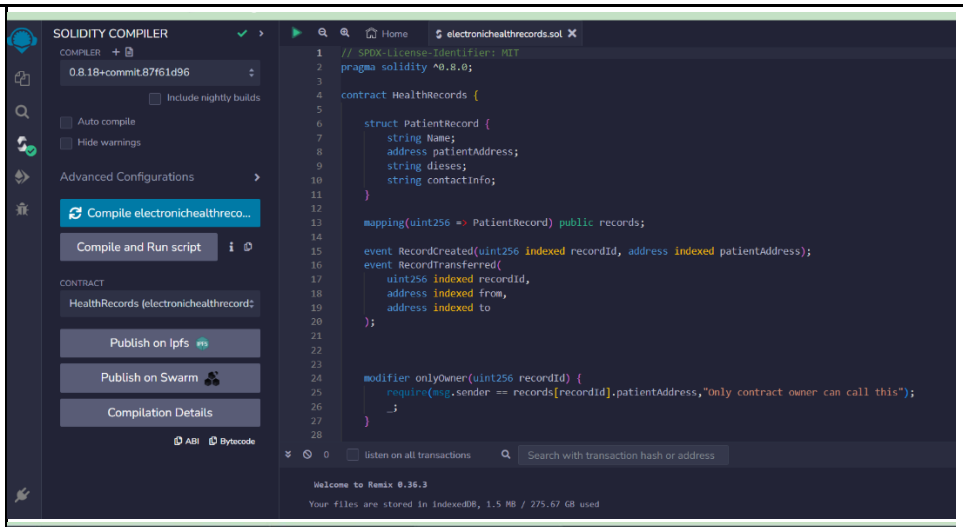
S.No.	Parameter	Values	Screenshot

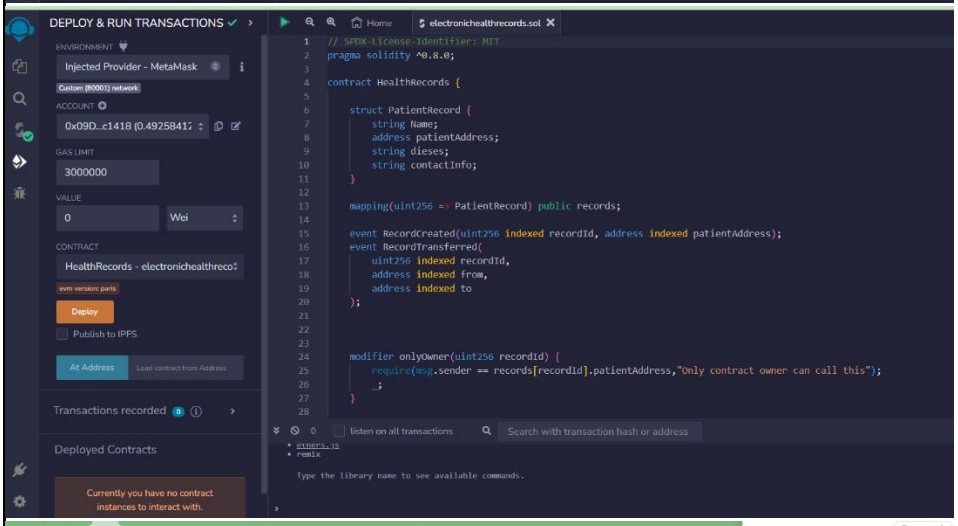
1.	Information gathering	Setup all the Prerequisite:	

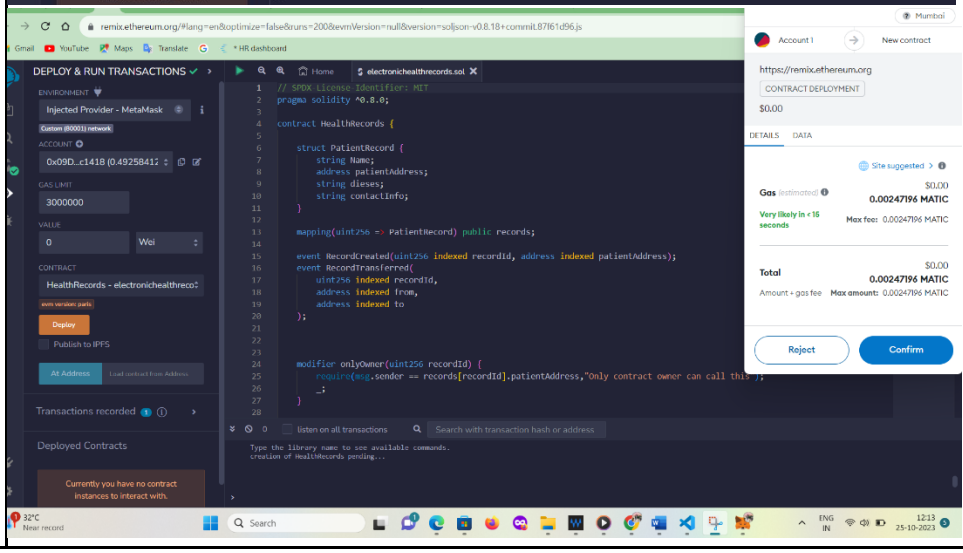
2.	Extract the zip files	Open to vs code	

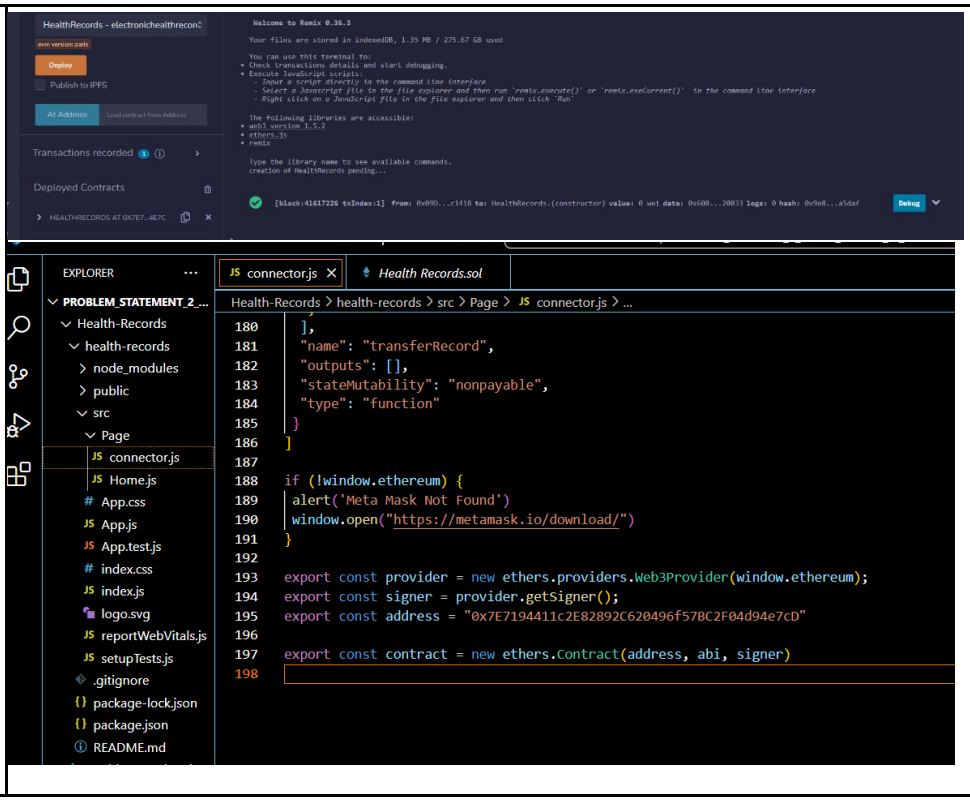
3.	Remix Ide platform explorting	Deploy the smart contract code Deploy and run the transaction. By selecting the environment - inject the MetaMask.	

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract HealthRecords {

    struct PatientRecord {
        string Name;
        address patientAddress;
        string dieses;
        string contactInfo;
    }

    mapping(uint256 => PatientRecord) public records;

    event RecordCreated(uint256 indexed recordId, address indexed patientAddress);
    event RecordTransferred(
        uint256 indexed recordId,
        address indexed from,
        address indexed to
    );



    modifier onlyOwner(uint256 recordId) {
        require(msg.sender == records[recordId].patientAddress,"Only contract owner can call this");
        _;
    }

    function createRecord(
        uint256 recordId,
        string memory name, address _patientAddress, string memory _diseases, string memory _contactInfo
    ) external {

        records[recordId].Name = name;
        records[recordId].patientAddress = _patientAddress;
        records[recordId].dieses = _diseases;
        records[recordId].contactInfo = _contactInfo;
```

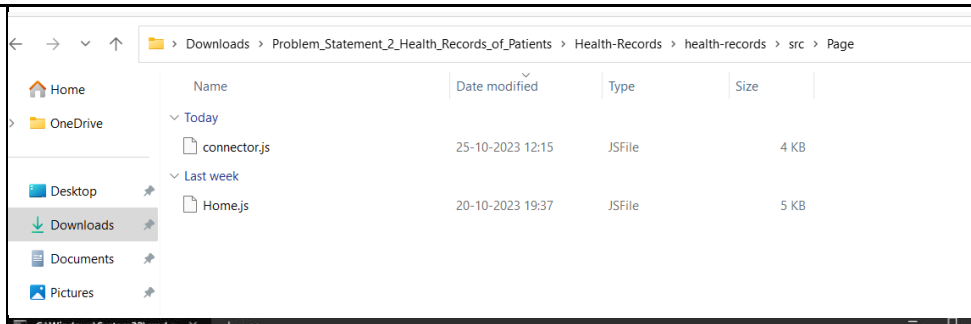
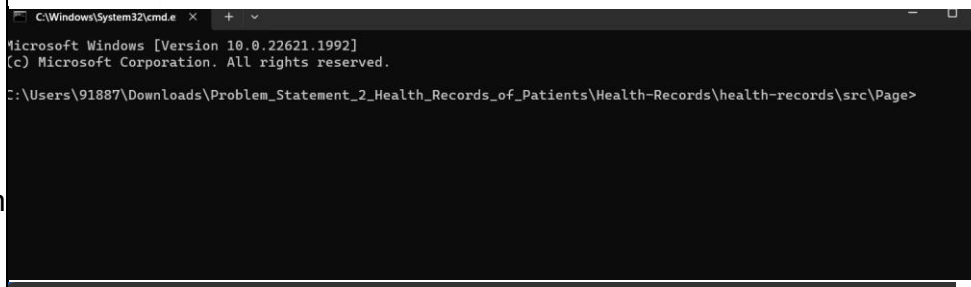
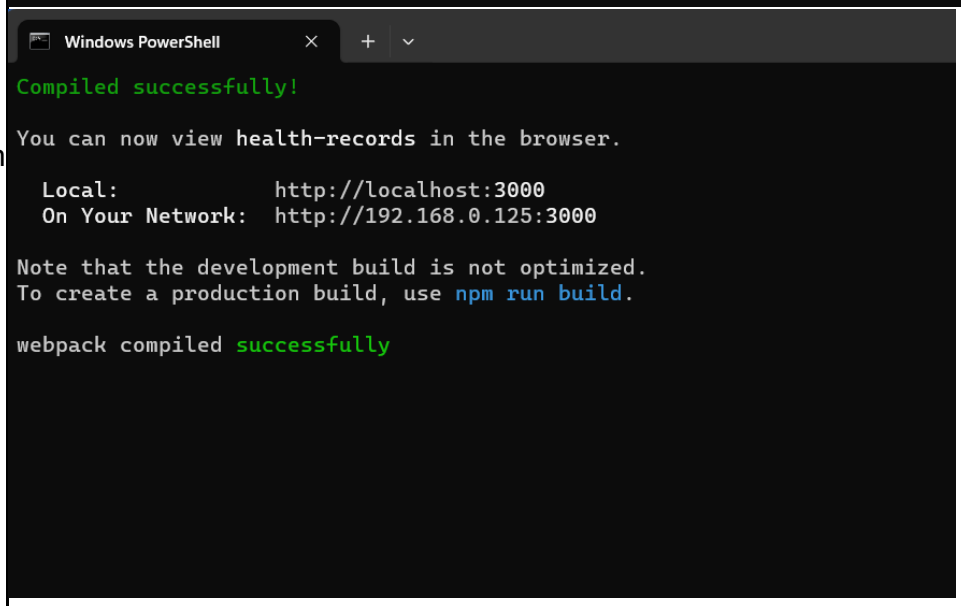

SOLIDITY COMPILER
COMPILER
0.8.18+commit.87f61d96
Include nightly builds
Auto compile
Hide warnings
Advanced Configurations
Compile electronichealthreco...
Compile and Run script
CONTRACT
HealthRecords (electronichealthrecord
Publish on Ipfs
Publish on Swarm
Compilation Details
ABI
Bytecode
Home
electronichealthrecords.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract HealthRecords {
struct PatientRecord {
string Name;
address patientAddress;
string dieses;
string contactInfo;
}
mapping(uint256 => PatientRecord) public records;
event RecordCreated(uint256 indexed recordId, address indexed patientAddress);
event RecordTransferred(
uint256 indexed recordId,
address indexed from,
address indexed to
);
modifier onlyOwner(uint256 recordId) {
require(msg.sender == records[recordId].patientAddress,"Only contract owner can call this");
_;
}
listen on all transactions
Search with transaction hash or address
Welcome to Remix 0.36.3
Your files are stored in IndexedDB, 1.5 MB / 275.67 GB used


DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
Injected Provider - MetaMask
Custom (80001) network
ACCOUNT
0x09D...c1418 (0.4925841
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT
HealthRecords - electronichealthreco
evm version: paris
Deploy
Publish to IPFS
At Address
Load contract from Address
Transactions recorded 0
Deployed Contracts
Currently you have no contract instances to interact with.
Home
electronichealthrecords.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract HealthRecords {
struct PatientRecord {
string Name;
address patientAddress;
string dieses;
string contactInfo;
}
mapping(uint256 => PatientRecord) public records;
event RecordCreated(uint256 indexed recordId, address indexed patientAddress);
event RecordTransferred(
uint256 indexed recordId,
address indexed from,
address indexed to
);
modifier onlyOwner(uint256 recordId) {
require(msg.sender == records[recordId].patientAddress,"Only contract owner can call this");
_;
}
listen on all transactions
Search with transaction hash or address
ethers.js
remix
Type the library name to see available commands.


remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS
Injected Provider - MetaMask
Deploy
Transactions recorded
Deployed Contracts
Currently you have no contract instances to interact with.
Mumbai
Account 1
New contract
https://remix.ethereum.org
CONTRACT DEPLOYMENT
$0.00
DETAILS
DATA
Site suggested
Gas
$0.00
0.00247196 MATIC
Very likely in < 15 seconds
Max fee: 0.00247196 MATIC
Total
$0.00
0.00247196 MATIC
Amount + gas fee
Max amount: 0.00247196 MATIC
Reject
Confirm
Type the library name to see available commands.
creation of HealthRecords pending...
32°C
Search
ENG
IN
12:13
25-10-2023


HealthRecords - electronichealthrecon
Deploy
Publish to IPFS
At Address
Transactions recorded
Deployed Contracts
Welcome to Remix 0.36.3
Debug

EXPLORER
connector.js
Health Records.sol
Health-Records > health-records > src > Page > connector.js > ...
PROBLEM_STATEMENT_2_...
Health-Records
health-records
node_modules
public
src
Page
connector.js
Home.js
App.css
App.js
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md
180],
181 "name": "transferRecord",
182 "outputs": [],
183 "stateMutability": "nonpayable",
184 "type": "function"
185 }
186]
187
188 if (!window.ethereum) {
189 alert('Meta Mask Not Found')
190 window.open("https://metamask.io/download/")
191 }
192
193 export const provider = new ethers.providers.Web3Provider(window.ethereum);
194 export const signer = provider.getSigner();
195 export const address = "0x7E7194411c2E82892C620496f57BC2F04d94e7cD"
196
197 export const contract = new ethers.Contract(address, abi, signer)
198

4	Open file explorer	Open the extracted file and click on the folder. Open src, and search for utiles. Open cmd enter commands 1.npm install 2.npm bootstrap 3. npm start	  

5	{LOCALHOSTIP ADDRESS	copy the address and open it to chrome so you can see the front end of your project.