

# **SOCIETE GENERALE HACKATHON**

(15<sup>th</sup> JULY 2024)

**(PSG COLLEGE OF TECHNOLOGY - COIMBATORE)**

## **Student Details:**

**NAME** : RAMYA P  
**DEGREE** : M.E BIOMETRICS AND CYBERSECURITY  
**PHONE NO** : +91 9482096308  
**OFFICIAL MAIL** : 23pb07@psgtech.ac.in  
**PERSONAL MAIL** : ramyap0455@gmail.com

## **Problem Statement Chosen:**

### **DEVELOPING A BLOCKCHAIN-BASED EVAULT FOR LEGAL RECORDS**

**GITHUB REPOSITORY LINK:** <https://github.com/23pb07ramya/E-Vault>

Given:

#### **Objective**

- Develop a blockchain-based eVault system for legal records that can ensure security, transparency, and accessibility for all stakeholders.
- The system should be able to store, manage, and share legal records securely and efficiently, with the potential to integrate with existing legal databases.

#### **Requirements**

- The eVault system should be based on a blockchain platform such as Ethereum, Hyperledger, or Corda, should use smart contracts to manage access, permissions, and transactions.
- The system should have user-friendly interfaces for the stakeholders to interact with the eVault, with features such as uploading and retrieving documents, tracking changes, and sharing information.
- The system should ensure the privacy and confidentiality of legal records, with appropriate access controls, encryption, and authentication mechanisms.

- The system should allow for seamless integration with existing databases, to ensure interoperability and ease of use.
- The system should be scalable and adaptable to accommodate future changes and upgrades.

THE FOLLOWING DOCUMENT CONSISTS OF AN ILLUSTRATED AND DETAILED PROCEDURE OF THE WORKING OF THE DESIGNED BLOCKCHAIN MODEL

1. A functional prototype of the blockchain-based eVault system for legal records, with a user-friendly interface and features such as document upload, retrieval, and sharing.
2. A detailed design document outlining the architecture, features, and technical specifications of the eVault system.
3. A business plan outlining the potential impact, market opportunities, and revenue models for the eVault system.
4. A presentation of the prototype, design document, and business plan.

## INTRODUCTION

Complete and accurate school records are a great tool for individuals. In recent years, educational documents have been digitalized. Nonetheless, there are still two significant issues that need to be resolved. The first is to save educational records in a secure and private manner; the second is to comprehend how to share records and make sure that the sharing procedure is protected. In order to build a dependable and secure environment, we provide in this paper a plan for educational records based on blockchain storage and sharing. This plan combines blockchain, storage databases, and cryptographic techniques. Our approach makes use of Ethereum blockchain technology to guarantee data storage security and dependability, with smart contracts on the blockchain controlling the storage and sharing process.

## TECHNOLOGY STACK

### Technology Stack:

- Blockchain
- Ethereum
- Web3
- Solidity
- IPFS
- Truffle
- Ganache

### Front End (Web DApp)

- React JS
- Bootstrap

### Back End

- NodeJS

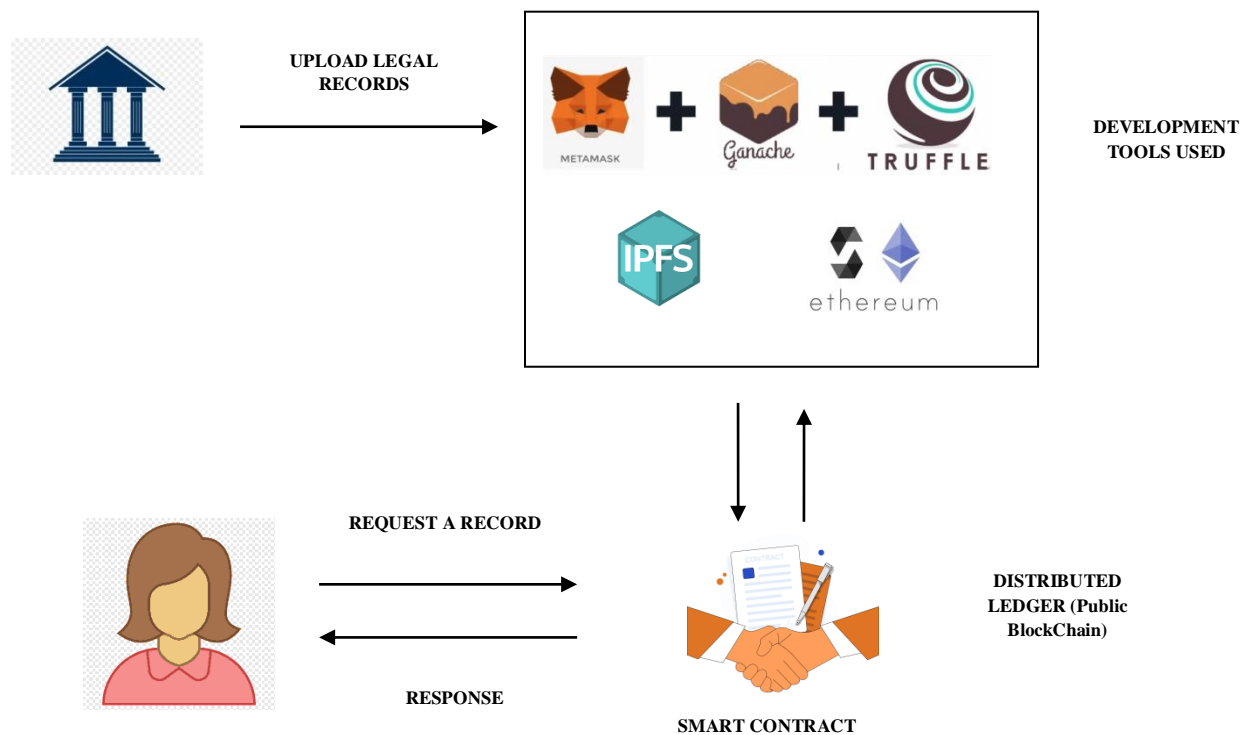
## Database

- OrbitDB

## Hosting Services

- Heroku (React DApp)
- Infura (Blockchain)
- Metamask

## PROPOSED SYSTEM DESIGN AND ARCHITECTURE



## DATA STORAGE IN PROPOSED SYSTEM

- IPFS

Blockchain platforms typically have limitations on the amount and type of data that can be stored directly on-chain due to scalability and cost concerns.

IPFS provides a decentralized and distributed storage solution where files can be stored off-chain but referenced on-chain.

## **Potential Impact**

### **1. Security and Transparency:**

The proposed system inherent security features (e.g., cryptographic hashing, immutability) can significantly enhance the security and integrity of legal records stored in the eVault. This reduces the risk of tampering or unauthorized access, which is crucial for maintaining trust in legal documentation.

### **2. Efficiency in Record Management:**

By leveraging blockchain and smart contracts, the eVault can automate and streamline processes related to record storage, access control, and sharing. This improves efficiency and reduces administrative overhead for legal professionals and organizations.

### **3. Reduced Costs:**

Moving towards digital and decentralized storage can potentially reduce costs associated with physical document storage, retrieval, and maintenance. It can also mitigate risks related to physical document loss or damage.

## **Market Opportunities**

1. The legal industry is increasingly adopting digital solutions to streamline operations and improve client service. The proposed system addresses the industry's need for secure and efficient record management systems.

2. The proposed eVault ensures data security and privacy compliance can cater to this market segment.

3. Integration with existing legal tech solutions, such as case management systems or document automation platforms, presents opportunities for partnerships and collaborative solutions that enhance overall efficiency and functionality.

## **Revenue Models**

1. Offer subscription-based access to the blockchain-based eVault, where legal firms pay a recurring fee based on usage or storage capacity.

2. Implement transaction fees for accessing or sharing legal records stored in the eVault, particularly for cross-border transactions or high-security documents.

3. Charge for premium features such as advanced encryption options, customizable access control settings, or integration with specialized legal software.

### Instructions to be followed :

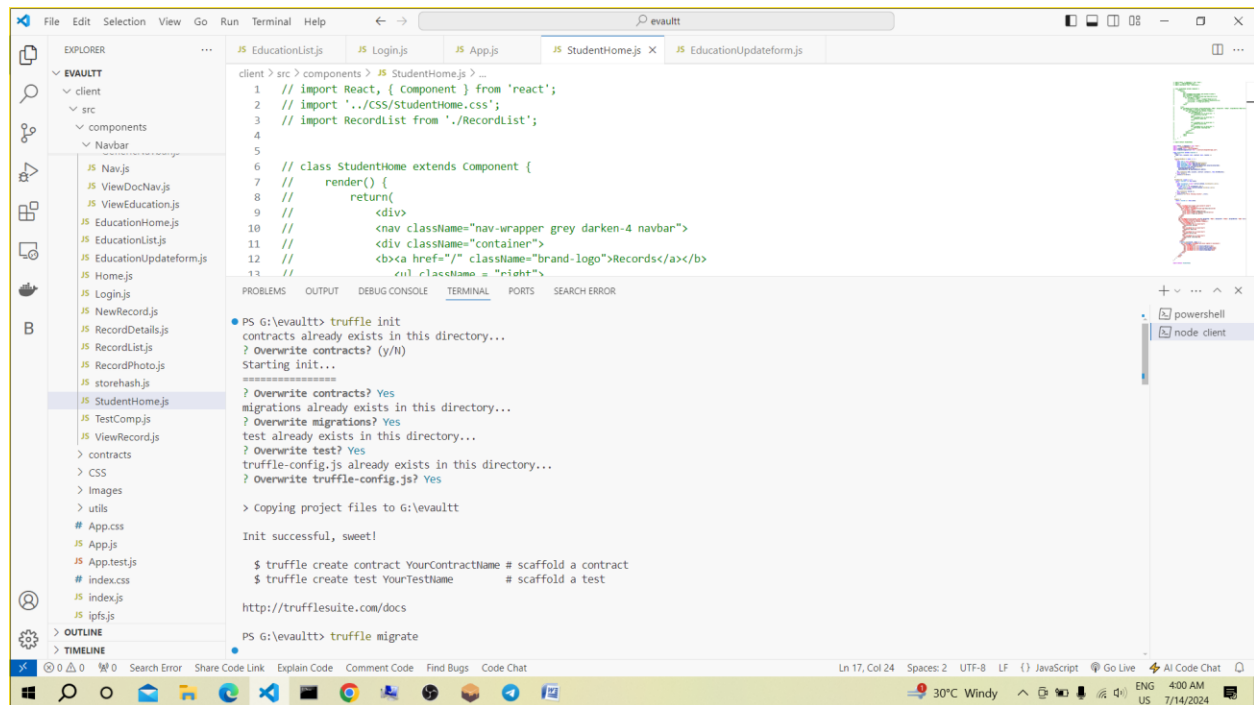
1. Open the extracted zip file in Visual studio code via open folder and access a new terminal in it.

```
client > src > components > EducationList.js > EducationList > componentDidMount
155 class EducationList extends Component {
156   state = {
161     contract: null
162   };
163
164   componentDidMount = async () => {
165     try {
166       // Get network provider and web3 instance.
167       const web3 = await getWeb3();
168
169       // Use web3 to get the user's accounts.
170       const accounts = await web3.eth.getAccounts();
171
172       // Get the contract instance.
173       const networkId = await web3.eth.net.getId();
174       const deployedNetwork = SimpleStorageContract.networks[networkId];
175       (IdployedNetwork) {
176         throw new Error(`Contract not deployed on network with ID ${networkId}`);
177       }
178       const instance = new web3.eth.Contract(
179         SimpleStorageContract.abi,
180         deployedNetwork && deployedNetwork.address,
181       );
182
183       // Set web3, accounts, and contract to the state, and then proceed with an
184       // example of interacting with the contract's methods.
185       this.setState({ web3, accounts, contract: instance }, this.getVal);
186     } catch (error) {
187
188     }
189   }
190 }
```

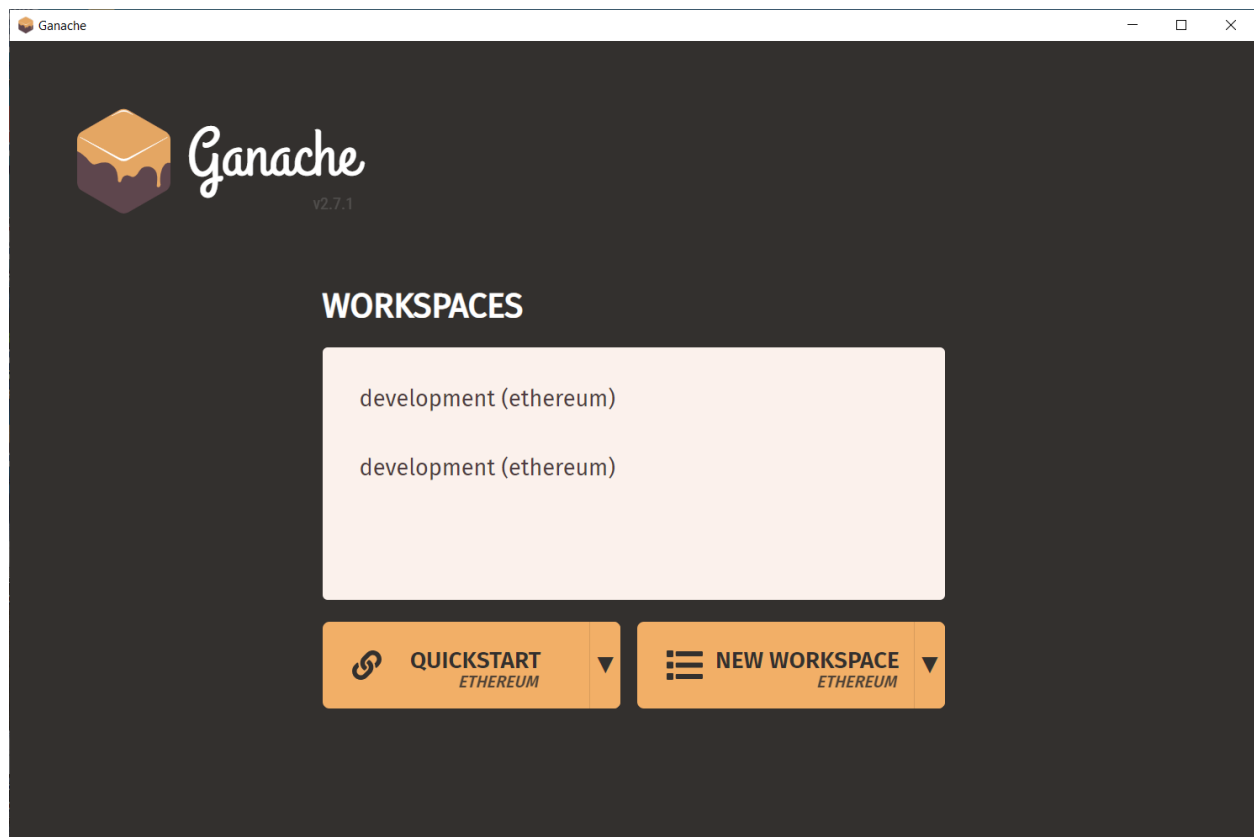
ERROR in [eslint] plugin "react" was conflicted between "package.json" > G:\evaultt\node\_modules\eslint-config-react-app\base.js and "BaseConfig" > C:\Users\RAMYA\AppData\Roaming\npm\node\_modules\react-scripts\node\_modules\eslint-config-react-app\base.js".

webpack compiled with 1 error and 7 warnings

1. Use *npm install* to install dependencies
2. Use *npm start* to start react-scripts
3. Visit *localhost:3000* to view the app running



Install *Metamask* and *Ganache* and create a free account



Ganache

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK  
43

GAS PRICE  
2000000000

GAS LIMIT  
6721975

HARDFORK  
MERGE

NETWORK ID  
5777

RPC SERVER  
HTTP://127.0.0.1:7545

MINING STATUS  
AUTOMINING

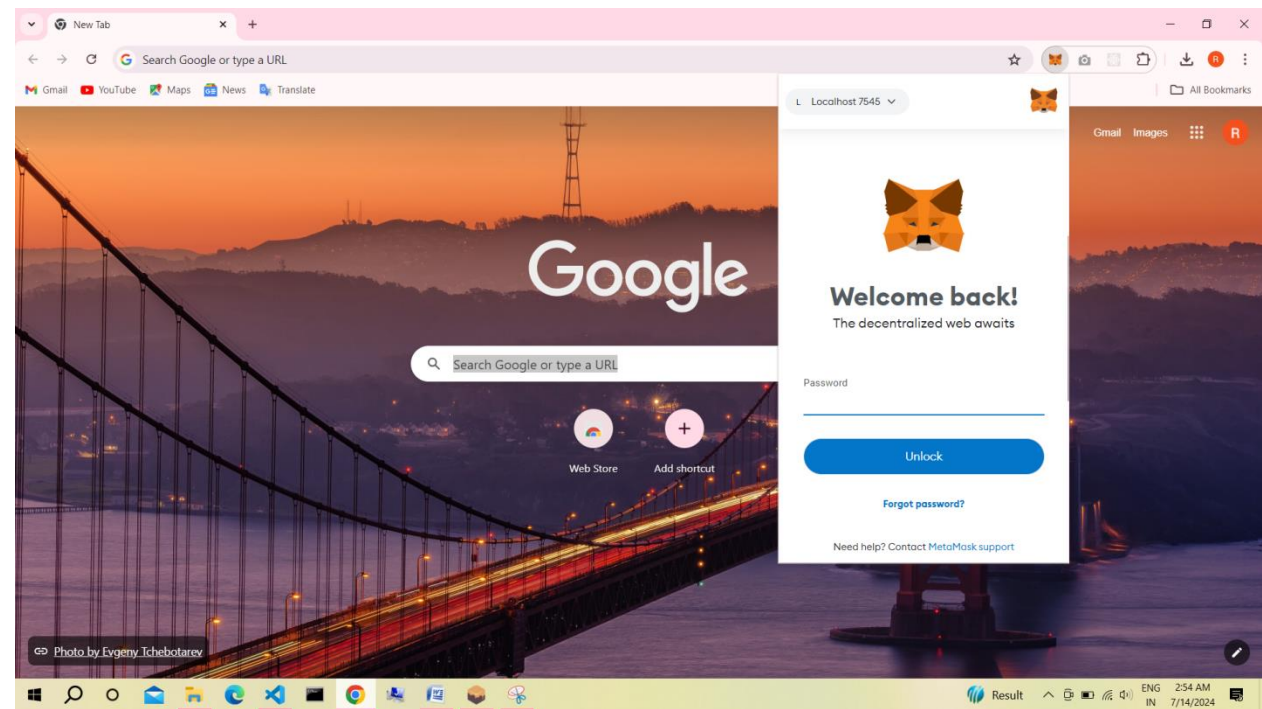
WORKSPACE  
DEVELOPMENT

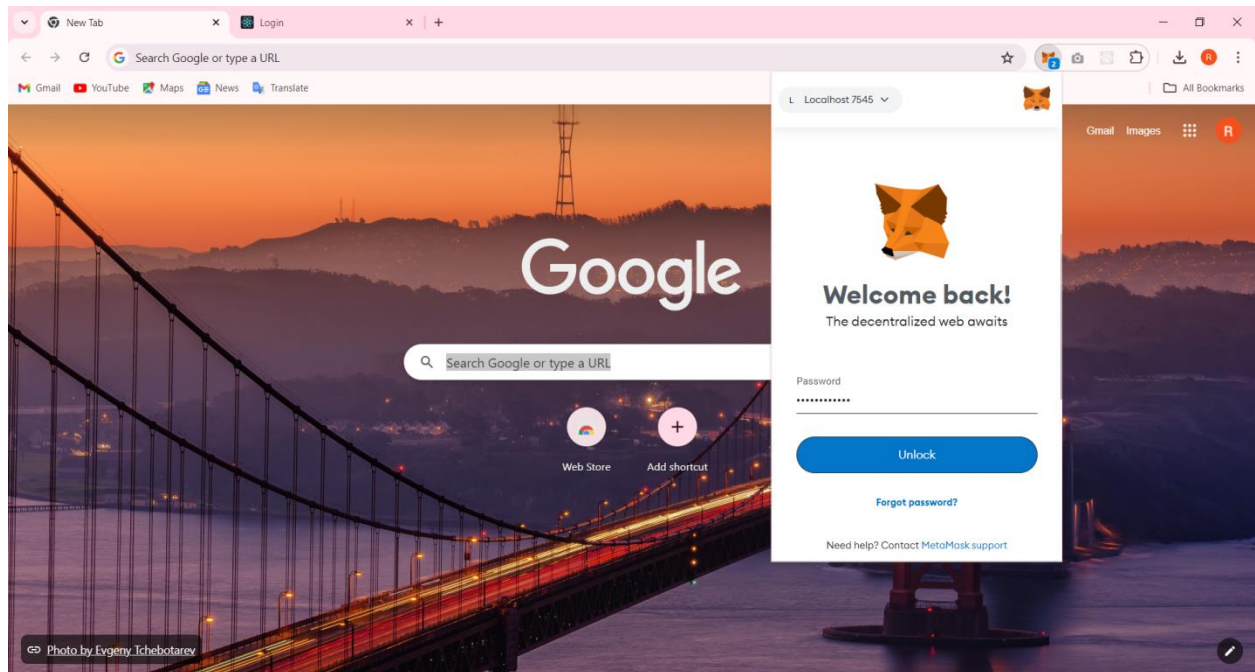
SWITCH

MNEMONIC ?  
surge recipe rapid cushion snake room excess similar place cause dream enable

HD PATH  
m44'60'0'0account\_index

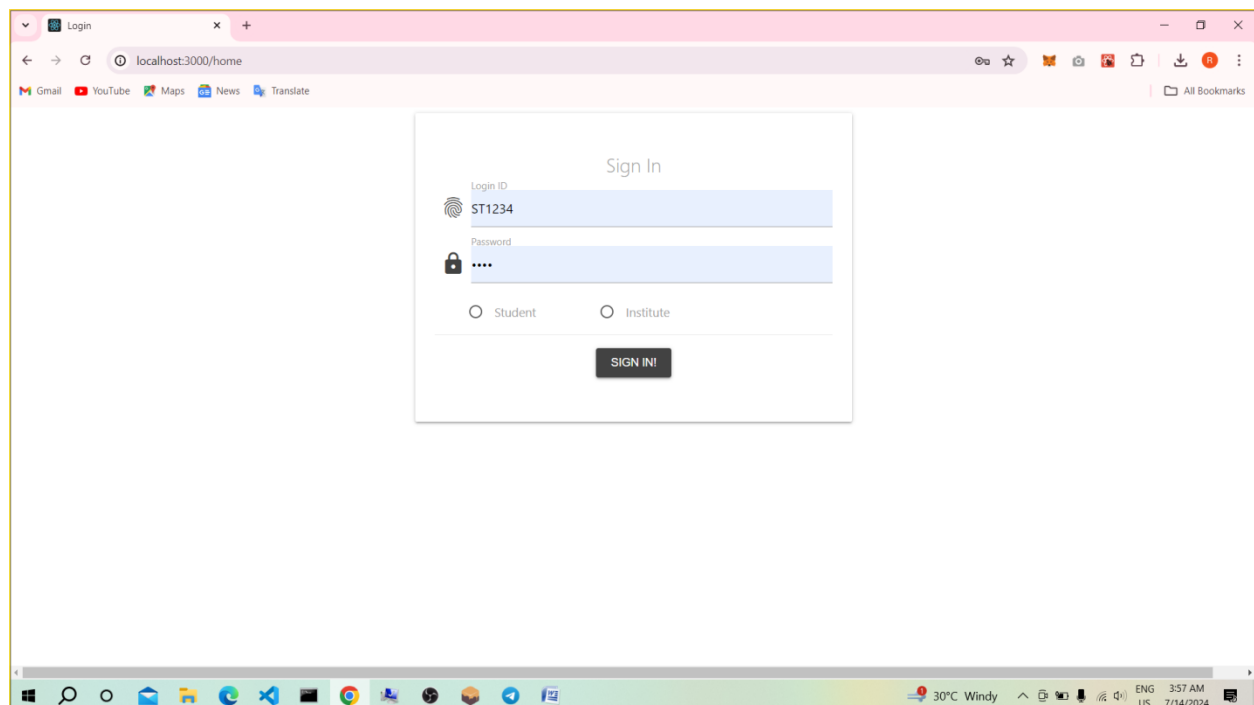
ADDRESS 0x0A0A8Bc0ECd415fF6d7A77F040EB491fDa11425A	BALANCE 99.89 ETH	TX COUNT 26	INDEX 0	
ADDRESS 0xe03976172bf0b1EfC96b099E4D6d80c09a4D344e	BALANCE 100.00 ETH	TX COUNT 2	INDEX 1	
ADDRESS 0x73D949C29D87E09eB1195641Cc35F3460D6d1bE3	BALANCE 100.00 ETH	TX COUNT 3	INDEX 2	
ADDRESS 0xbab85A4FCCD1789619160E0E654c7Fa40A16B3e3	BALANCE 100.00 ETH	TX COUNT 3	INDEX 3	
ADDRESS 0x6D48b3cEe993ADB070506033a2e0FFBc063285eC	BALANCE 100.00 ETH	TX COUNT 0	INDEX 4	
ADDRESS 0x51973adBf02486dce0c4b8578abdEb0fEf4D6A08	BALANCE 100.00 ETH	TX COUNT 1	INDEX 5	
ADDRESS 0x794e1435D55c4c6Ed1462AE36060c3e0E76F5D42	BALANCE 100.00 ETH	TX COUNT 8	INDEX 6	





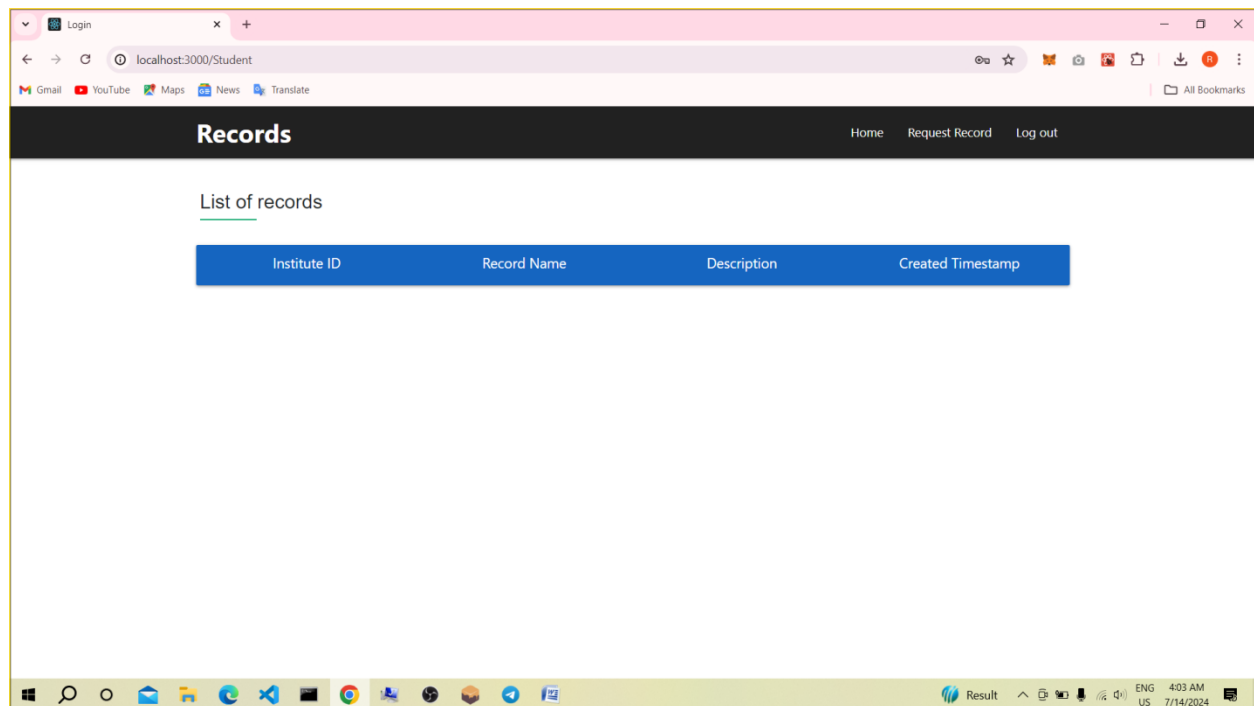
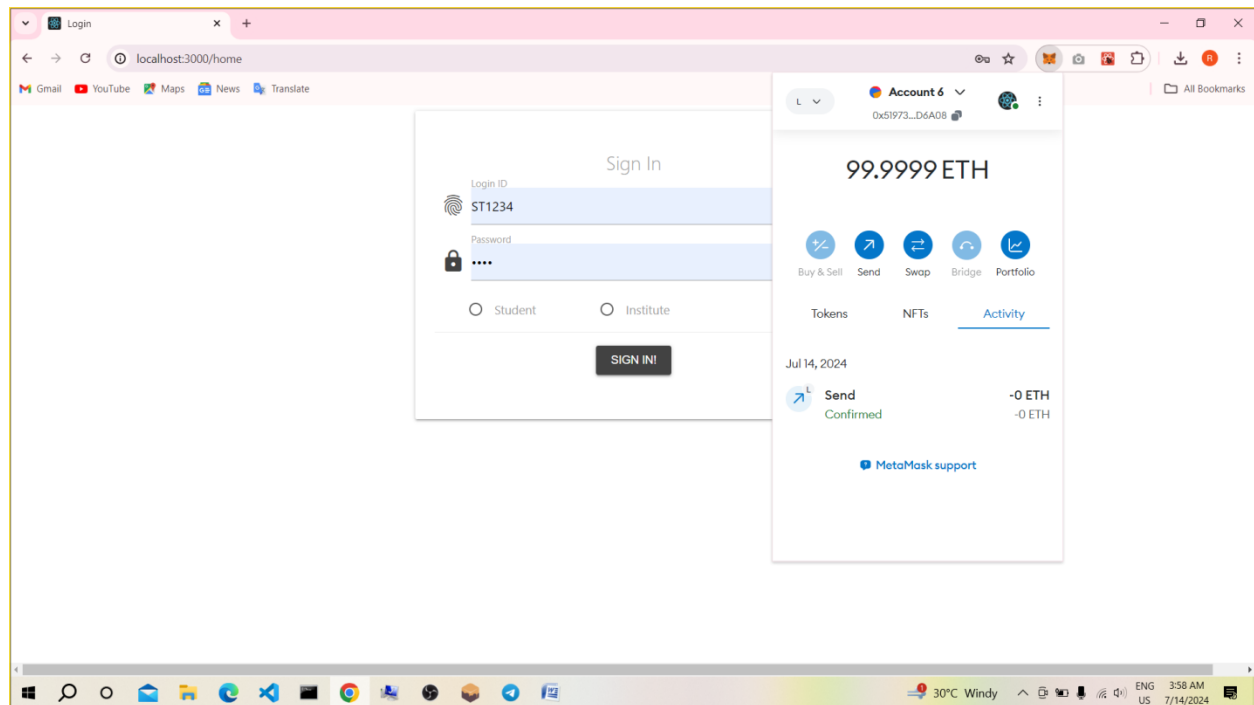
*Import Ganache into Metamask*

**First we are directed towards the sign-in page**

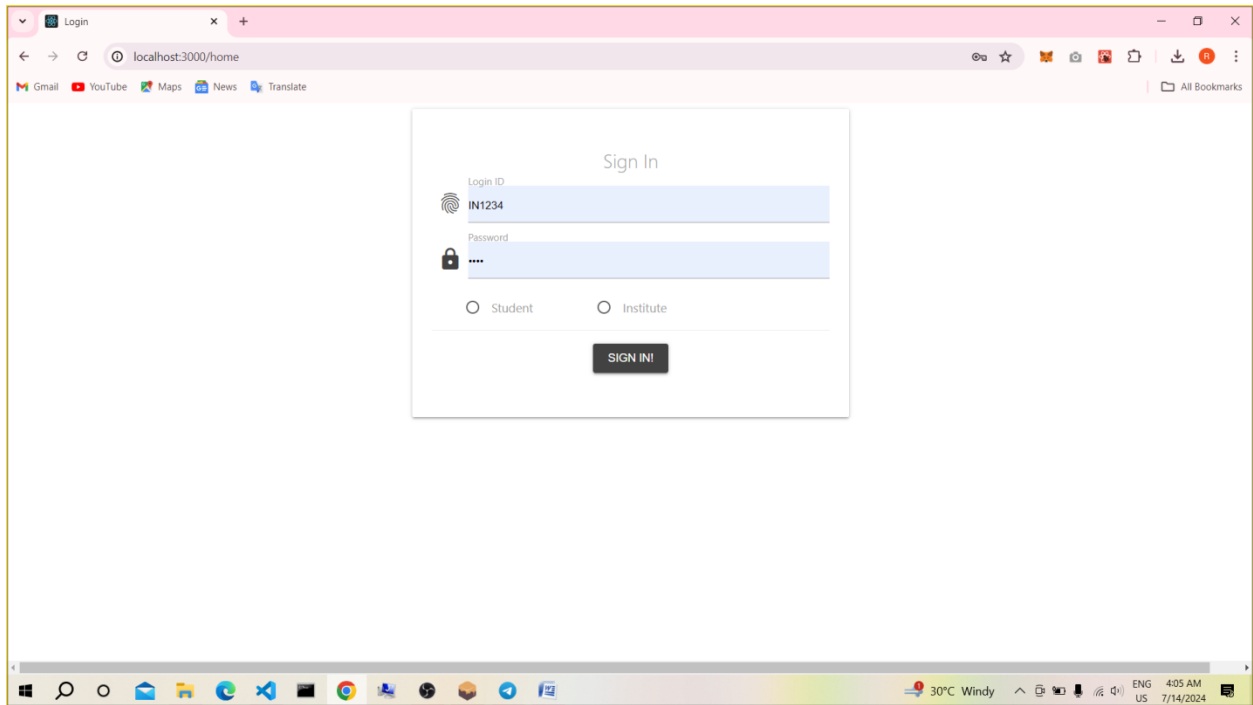


Upon signing into the account as Student we are unable to view any records as no records have been made available for visibility





Hence we login as the institution for accessing and granting permission to looking into the records



A fixed or set amount of ether(gas) is sent and received for every so called transaction

## Records Home Request Record Log out

List of records

Institute ID	Record Name	Description	Created Timestamp
--------------	-------------	-------------	-------------------

No records!

In order to add records we use the Add records section

Records Home Log out

New Record

Student ID

Institute Address

2024-07-14 03:04:49

Record Name

Upload to Blockchain

Enter Description

Records Home

New Record

Student ID

1711T270

Institute Address

0x51973adBf02486dce0c4b8578abdEb0fE4D6A08

2024-07-14 03:50:05

Record Name

Academic certificate

Upload to Blockchain

Enter Description

Required for scholarship application

MetaMask

Localhost 7545

Account 5 0x6E653a...

SENDING ETH

0

DETAILS HEX

You are sending 0 eth.

Estimated fee 0.00006053

0.00006053 ETH

Site suggested -15 sec Max fee: 0.00006073 ETH

Total 0.00006053

0.00006053 ETH

Amount + gas Max amount: 0.00006073 ETH

fee

Reject Confirm

Once the given record has been added over the blockchain it is displayed as

## Records

[Home](#) [Log out](#)

### List of Records

Student ID	Record Name	Description	Created Timestamp
171IT270	Academic Certificate	Required for scholarship application	2020-11-03 21:13:10
<a href="#">Add</a>			

We can also add records over the blockchain in order to keep them from being duplicated hence enhancing security

## Records

[Home](#) [Logout](#)

### Upload Education Reports

INSTITUTE ID\*

NITK

STUDENT ADDRESS KEY\*

0x4E35d1C1d881EF1e4c012746c50fc48E422B1CeB

RECORD NAME \*

Academic Certificate

DESCRIPTION\*

Verified the certificate of Student ID:171IT248

DOCUMENTS (UPLOAD IN .ZIP OR .RAR FORMAT)\*

[Choose File](#) certificate.png.zip

TIMESTAMP\*

2020-11-03 21:16:31

[Upload to Blockchain](#)

Using IPFS the record is stored over the database for future access as well

Records				Home	Request Record	Log out
<a href="#">List of records</a>						
Institute ID	Record Name	Description	Created Timestamp			
NITK	Academic Certificate	Verified the certificate of Student ID:171IT248	2020-11-03 21:16:31			
		IPFS	Download			

**One request all the files can be viewed on the Blockchain via Request record but a certain user is allowed to do so only if he/she is a part of the blockchain.**

## CONTRACTS USED

EducationContract.sol

Migrations.sol

SimpleStorage.sol

## Usecase satisfied:

- Integration with Existing Legal Databases
- Smart Contract-Based Access Control
- User Authentication and Encryption
- User-Friendly Interface
- Audit Trail and Reporting
- Scalable Architecture
- Interoperability with Legal Frameworks
- Disaster Recovery and Backup