Web3 & Blockchain Applications for Real World Problems ---- ChainTalk

SCSE23 – 0200

Presented by:
Poon Yan Xin Melise (U2022504B)

13 May 2024

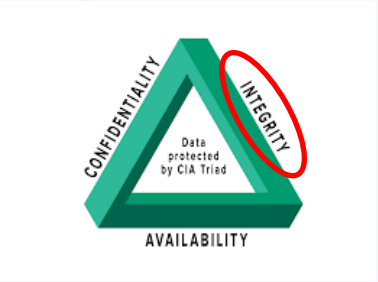
Table of contents

Source: Flaticon

Blockchain for Messaging

1) Decentralization → Availability
    - Rising concerns of data ownership
    - Ensures that no single party can unilaterally sell or exploit users' personal data without their consent
    - Centralized server is also susceptible to server downtime → loss of availability.

Blockchain for Messaging



2) Immutability & Transparency → Integrity
    - Ensure permanence of digital communications
    - e.g. in formal discussions, neither party can make changes to agreement without other party's knowledge.

Blockchain for Messaging

3) Secure → Confidentiality

- Centralized servers are susceptible to data breaches and cyber-attacks → loss of confidentiality
- Cryptographic techniques ensures only authorized parties can access data → enhanced security

Motivation

- Messaging application is the single most used social media application
- With more users → larger attack surface
- Potential for usage of blockchain to build a secure, private and user-centric messaging platform
- Aims to redefine the standards of secure communication in the digital era

SoulBound Tokens (SBT)

- SBT: publicly verifiable and non-transferable tokens that represents commitments, credentials and affiliations.
- Web3 lacks primitives to represent social identity → relies on centralized Web2 structures
- Soul that stores SBTs that represent their credentials, employment history and many more → provide more credibility to a person's existence in Web3

ERC – 735 standard

- Standard on Ethereum Blockchain for issuing and maintaining identities through smart contracts
- Functions: adding, removing and holding of claims

Claims

- Topics: attestation of skills, qualifications, user personal data etc.
- Attested from third - parties or self - attested
- Address KYC (Know your Customer) issues in blockchain

Existing Applications

Aspect	Typical Application (Whatsapp)	Blockchain based (ChainTalk)
Data Storage	Centralised storage on Whatsapp servers	Decentralised storage across Blockchain & IPFS

Introducing ChainTalk

- Blockchain-integrated messaging platform with that enables users to engage in instant messaging
    - Prioritises Security, Privacy, Durability in modern communication
- Key Features
    - Create Account
    - Add Friends
    - Exchange real - time messages

Introducing ChainTalk

- No intermediaries involved
- Leverages Blockchain for on-chain storage and IPFS for off-chain storage (layered architecture)
  - Only message content-identifiers (CIDs) is stored on-chain.
- Usage of SoulBound Tokens (SBTs) and ERC-735 standard
  - Creates digital identity for users
  - Enhances user autonomy in identity and minimises risk of security information disclosure

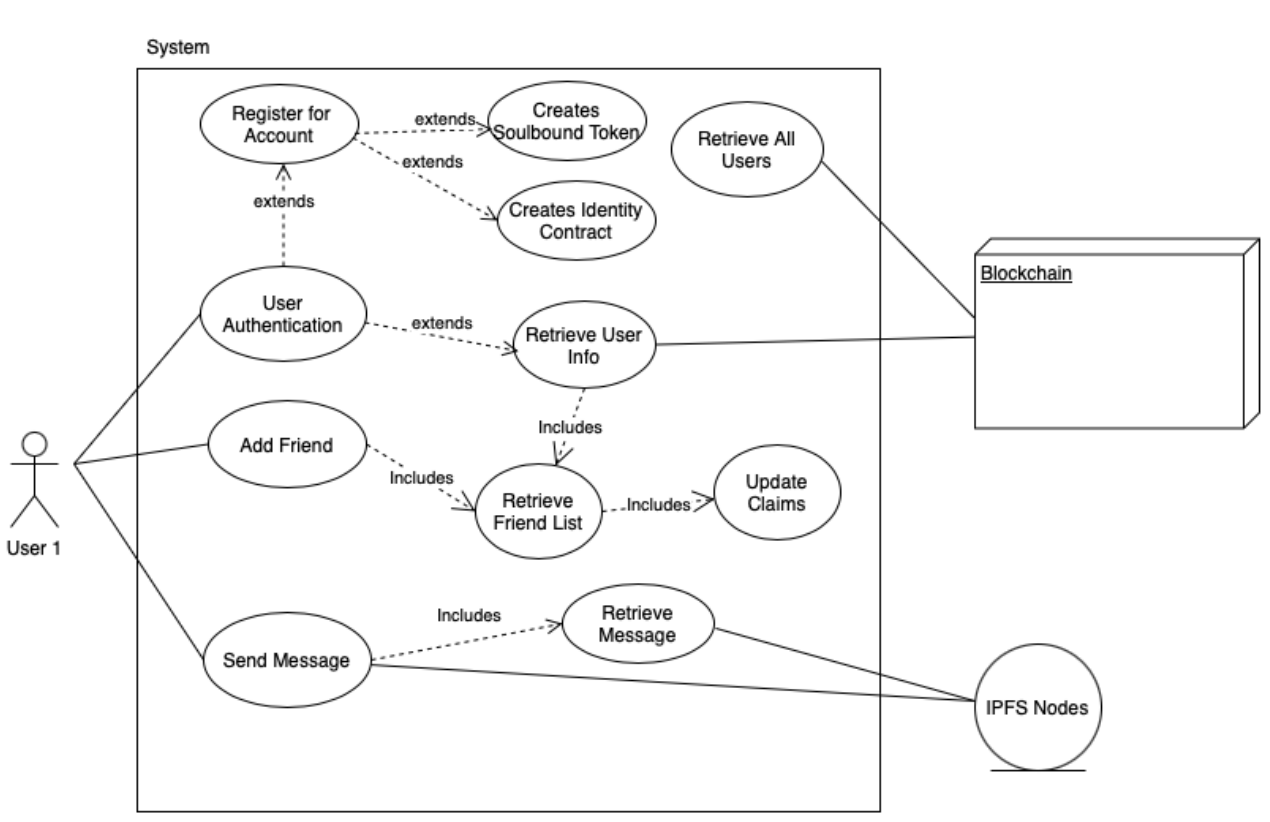
Target Users

- Business Professionals: Sensitive communications
- Individuals: heightened data privacy concerns
- Security conscious users

Functional Requirements

Function	Explanation
User Registration	Creates a new user based on valid Ethereum address, username, email and password provided.

Use Case Diagram



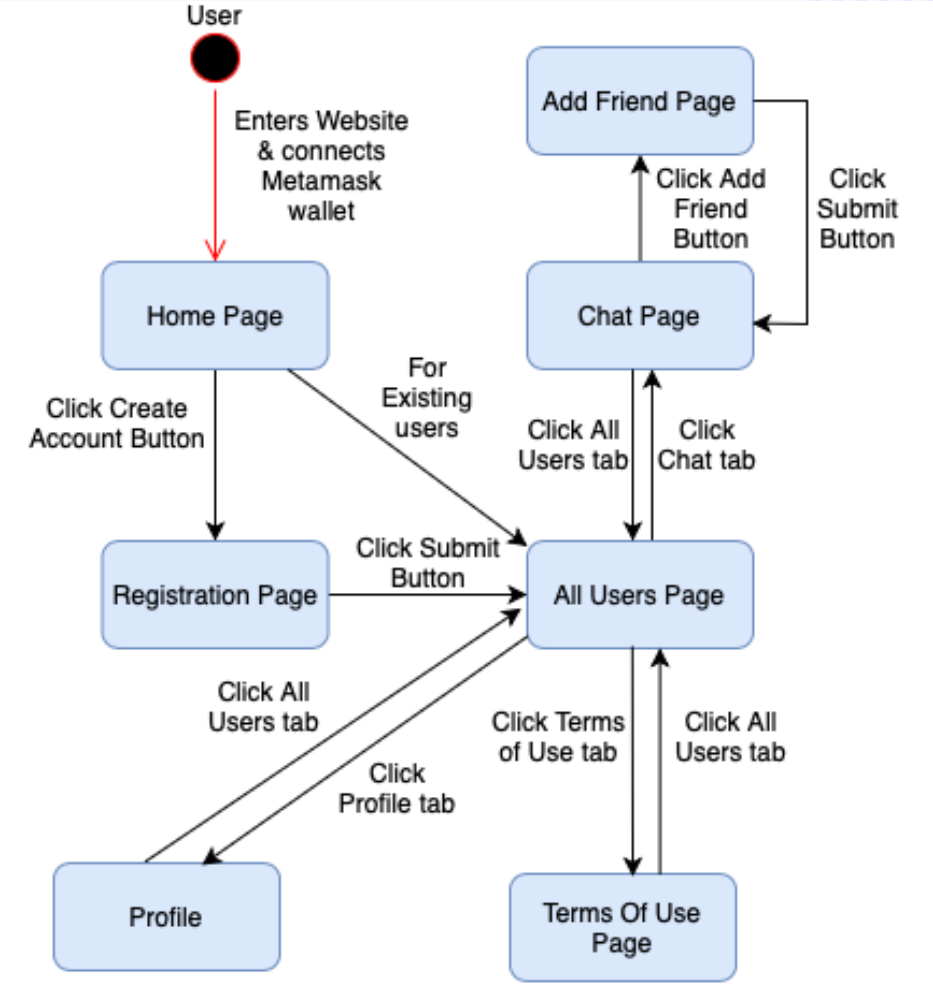
11 Use Cases

1 Actor

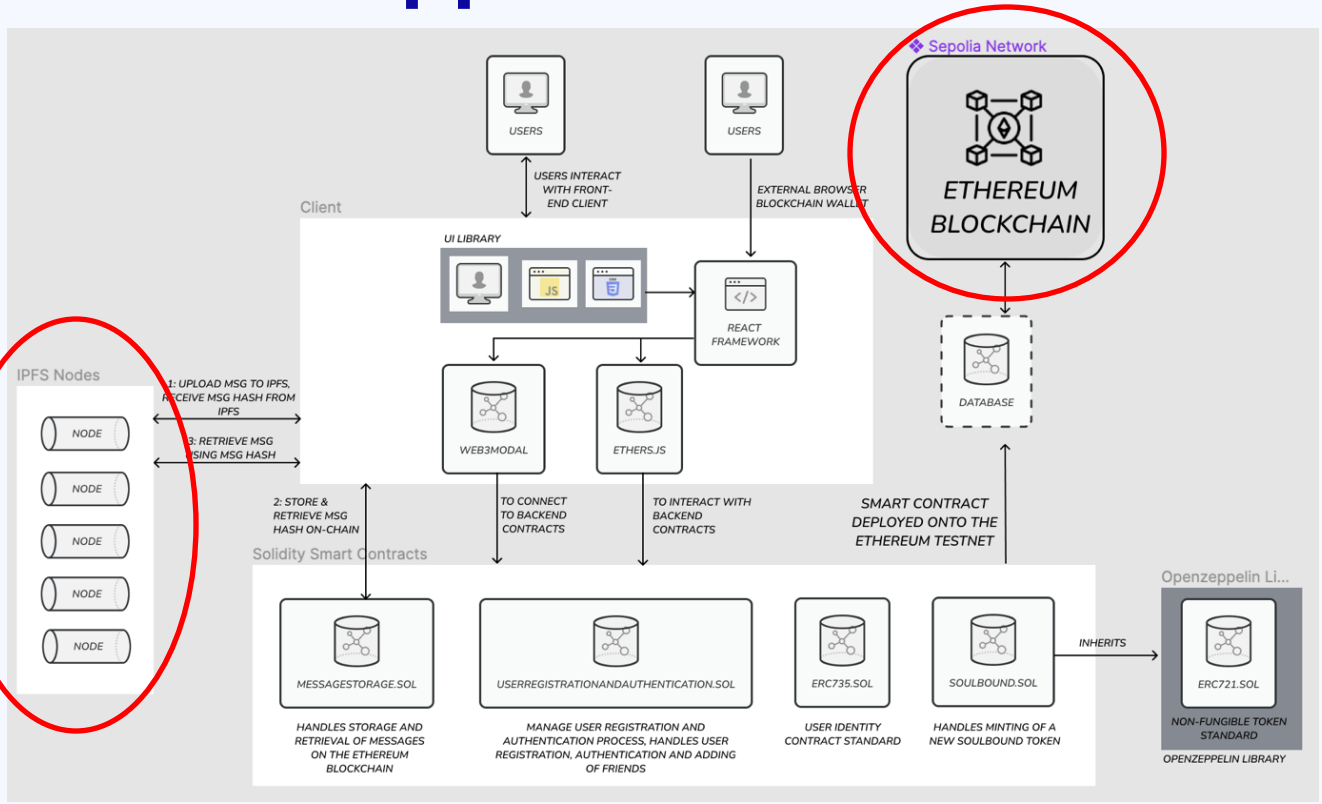
Blockchain

IPFS Nodes

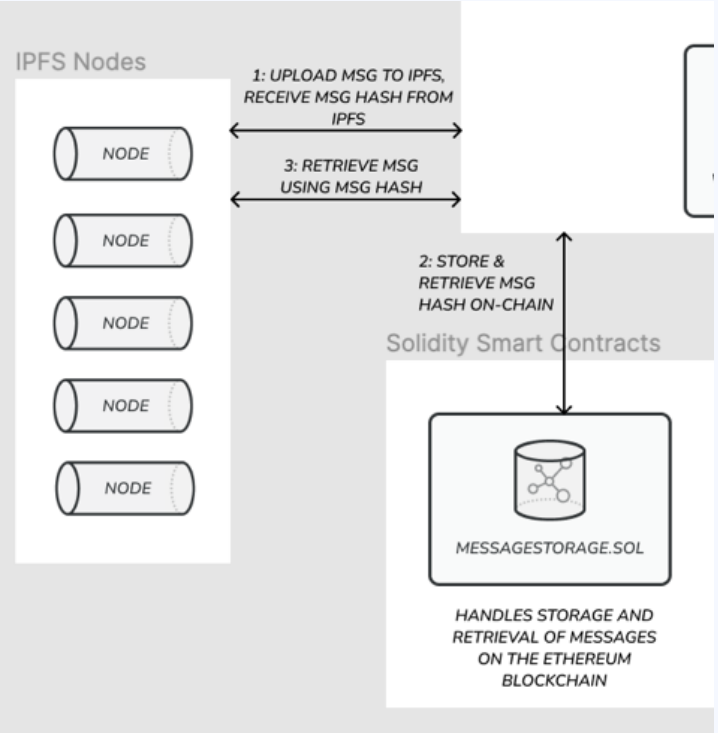
Dialog Map


User
Enters Website & connects Metamask wallet
Add Friend Page
Click Add Friend Button
Click Submit Button
Home Page
Chat Page
Click Create Account Button
For Existing users
Click All Users tab
Click Chat tab
Registration Page
Click Submit Button
All Users Page
Click All Users tab
Click Terms of Use tab
Click All Users tab
Click Profile tab
Profile
Terms Of Use Page

Overall Application Architecture

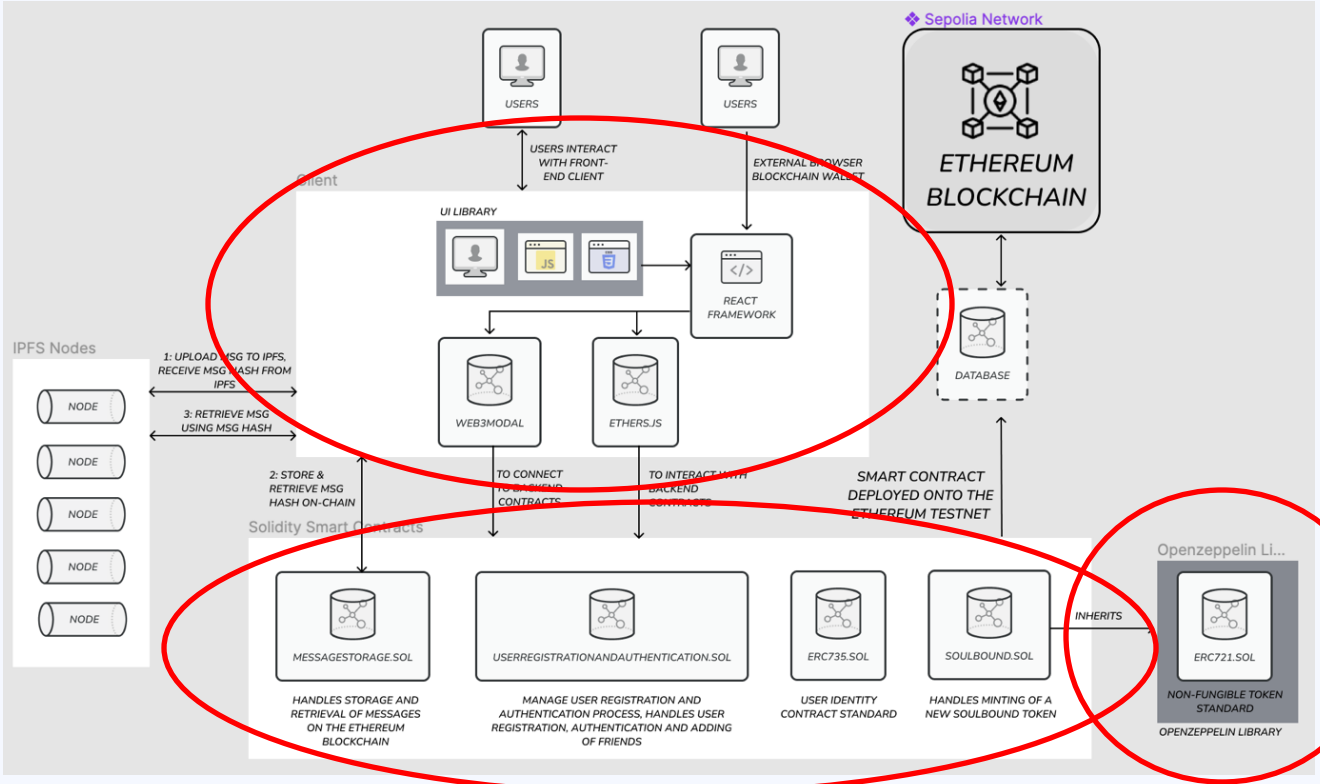


Overall Application Architecture

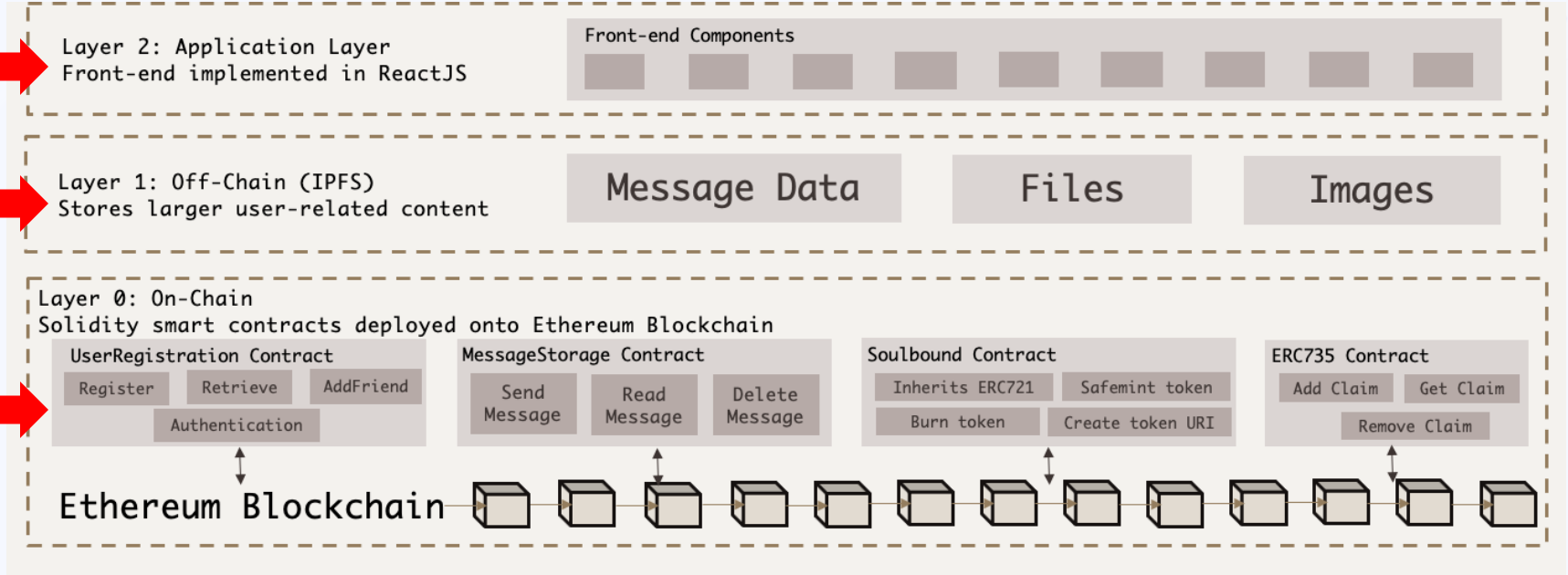


- Used for off-chain storage of user data
- Only IPFS Content Identifiers (CIDs) will be stored on chain
- Reduce transactional costs

Overall Application Architecture



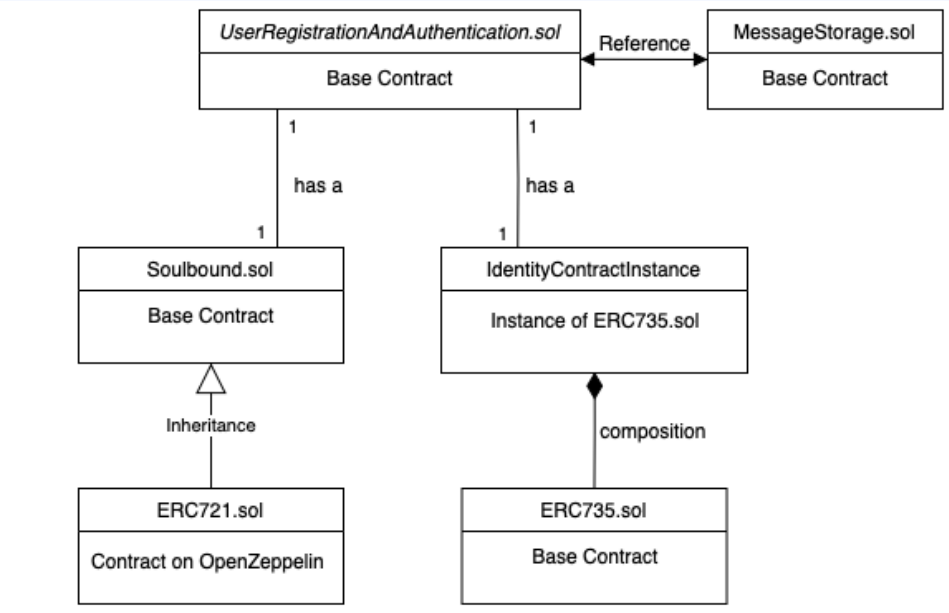
Layered Architecture



Benefits of Layered Architecture

Aspect	Explanation
Abstraction & Modularity	Easier development and maintenance of each layer independently.

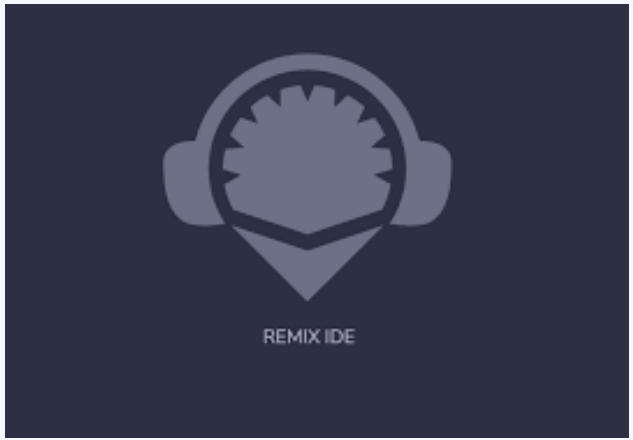
Smart Contract Architecture



4 contracts

- Each User registered from has a Soulbound token (SBT)
- *soulbound.sol* inherits functions from *ERC721.sol.*
- Each User registered has an instance of Identity Contract.
- Identity contract instance has a composition relationship with *ERC735.sol*

Technology Stack



SOLIDITY

IPFS

Hardhat

React JS

ethers.js

Visual Studio Code

Front - End Development

Metamask connection

```
export const connectWallet = async () => {
    try {
        if (!window.ethereum) return console.log("Install MetaMask")

        const accounts = await window.ethereum.request({
            method: "eth_requestAccounts",
        })
        const firstAccount = accounts[0]
        return firstAccount
    } catch (error) {
        console.log(error)
    }
}
```

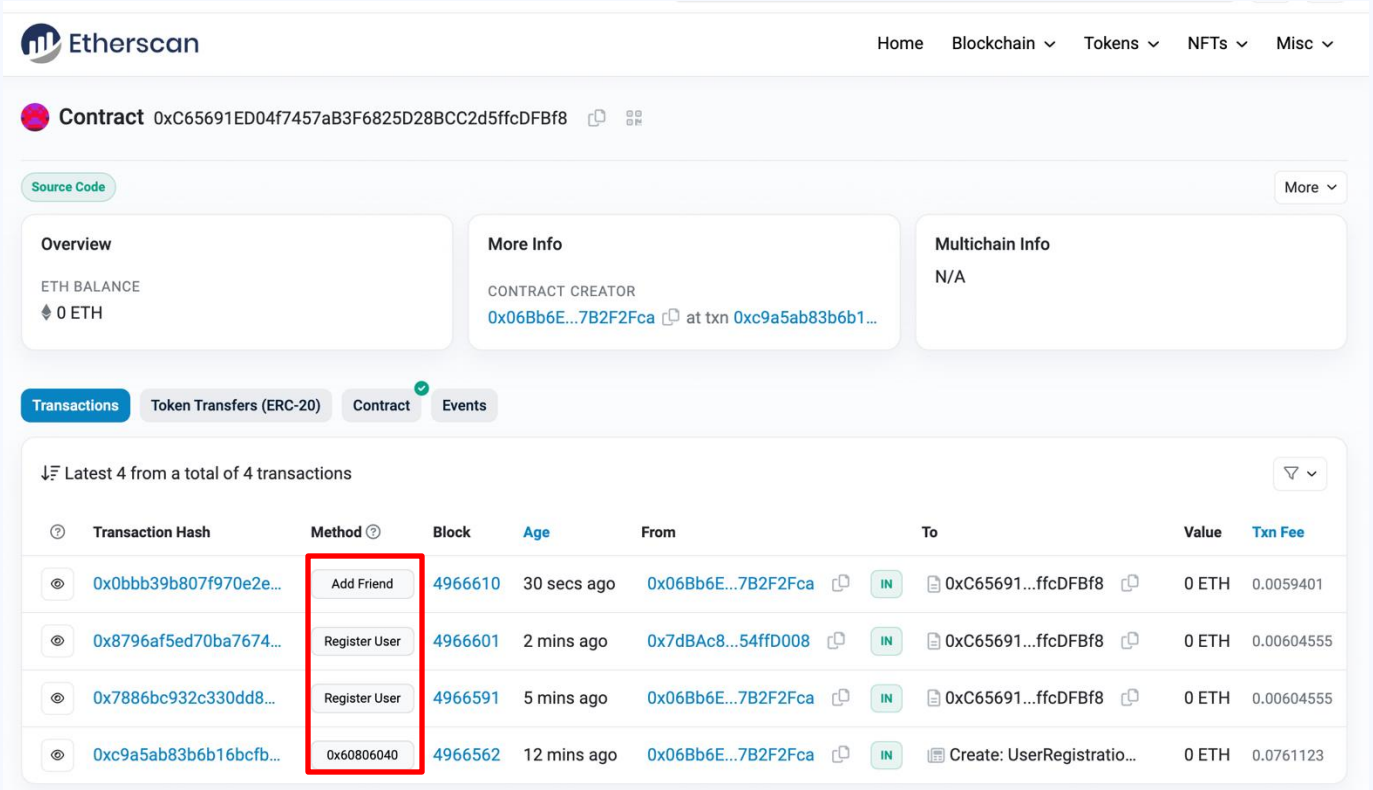
Front - End Development

Fetching Contracts

```
export const connectingWithContract1 = async () => {
    try {
        const web3modal = new Web3Modal()
        const connection = await web3modal.connect()
        const provider = new ethers.providers.Web3Provider(connection)
        const signer = await provider.getSigner()
        const contract1 = fetchContract(signer, userRegAddressLH, userRegABI)

        return contract1
    } catch (error) {
        console.log(error)
        throw error
    }
}
```

Deployment of Smart Contracts

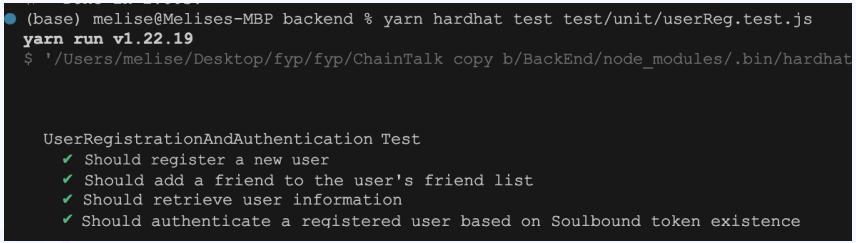


Demo

Black Box Testing (Using Scripts)

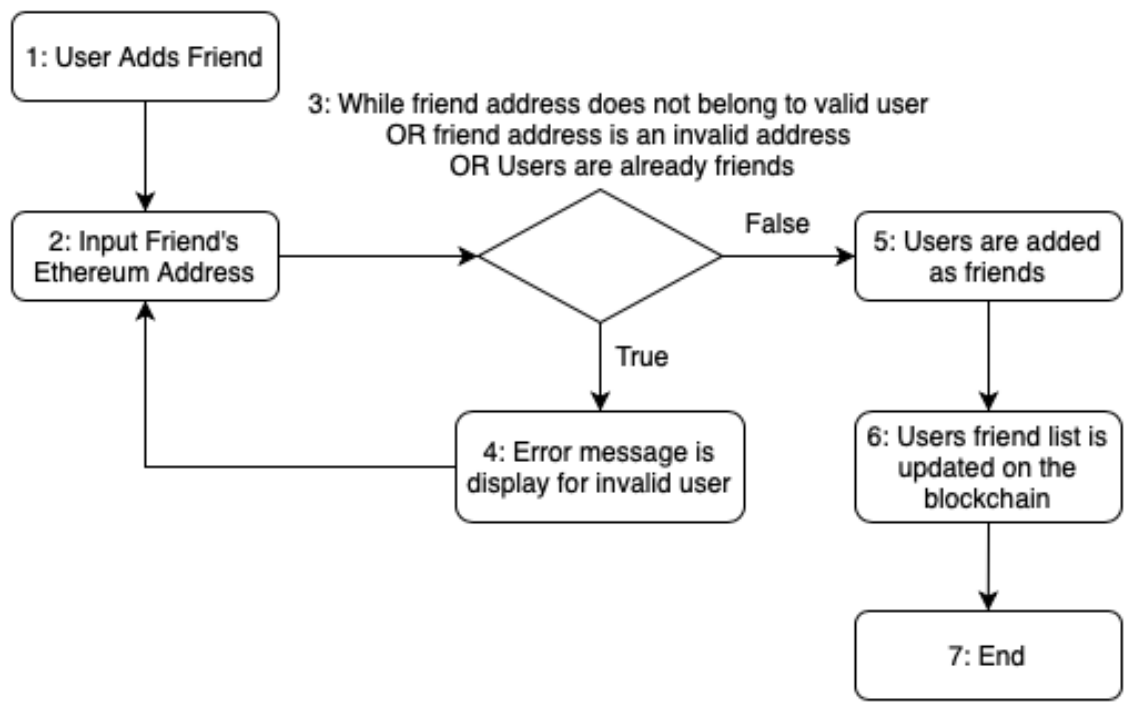
Testing User Contract

Test	Input	Function tested
Register a new user	Name: *user1* Email: *user1@gmail.com* Password: *"0x3a2f...21b8"*	registerUser
Add a friend	Friend's address	addFriend, getMyFriendList
Retrieve user info	-	getMyUserInfo
Authenticate user base on SBT	-	authenticate, deployIdentityContract

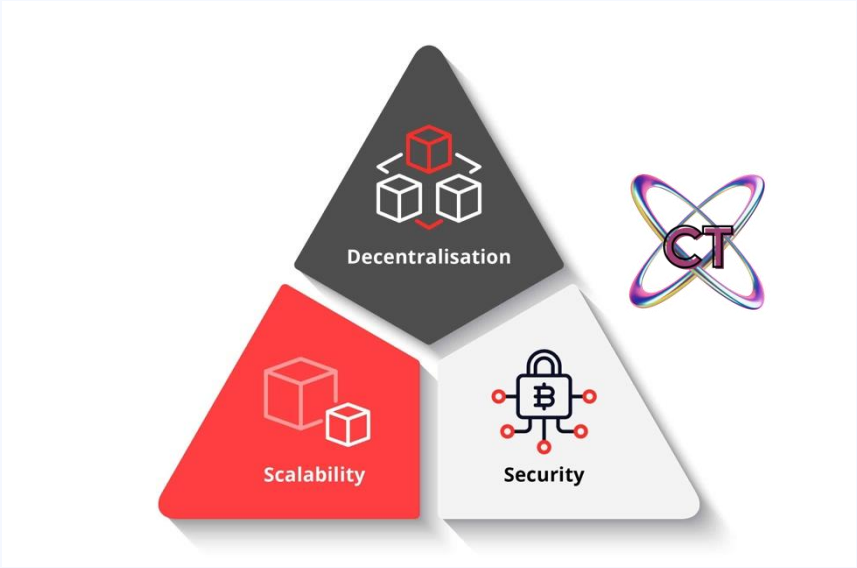


White Box Testing

Testing the addFriend function



The Blockchain Trilemma



- Decentralisation, Security and Scalability
- Cannot achieve all 3

Source: CoinMarketCap

Future Work

Pros of IPFS

- Facilitates seamless, secure and low latency data distribution

Limitations of IPFS

- Database-as-a-service model
- No economic incentives put in place → impractical for long term usage
- Inherently free, nodes not volunteering to store large amount of data for free for a long time
- Results in limited storage capacity and duration.

Future Work

Solution

- Layer 2 technologies
- Improve efficiency and reduce transaction cost for sending a message
- E.g. State channels, Side-chains, Optimistic Rollups, ZK-Rollups

Source: PolygonTechnology

Conclusion

- Explored the landscape of decentralised messaging applications by leveraging blockchain technology.
- Integration of IPFS → pivotal aspect
- Digital Identity using SoulBound Tokens and ERC-735
- Aims to offer a secure, efficient and scalable messaging solution in a decentralised environment.

Thank You


BLOCKCHAIN
TECHNOLOGY
COULD CHANGE
OUR WORLD
MORE THAN PEOPLE IMAGINE
– Jack Ma
Co-Founder of Alibaba
CoinsCapture
CoinsCapture

References

[1] “Blockchain Trilemma”, Coinmarketcap Website. Available on: https://coinmarketcap.com/academy/glossary/blockchain-trilemma#
[2] Image retrieved from Polygon Technology Website. Available on: https://polygon.technology/blog/final-approach-last-testnet-for-an-upgraded-polygon-zkevm
[3] Google Images
[4] Slidesgo