**(REVIEW-3)**

**DECENTRALIZED CHAT APPLICATION USING BLOCKCHAIN**

****

**REVIEW-1 REPORT SUBMITTED FOR CAPSTONE PROJECT**

**(CSE 4099)**

*by*

**V Venkata Narendra**

**18BCE0515**

**6304307077**

**venkatanarendra2018@vitstudent.ac.in**

**Sri Sai Manikanta Vishnu Shasank**

**18BCE2131**

**6303312841**

**srisai.manikanta2018@vitstudent.ac.in**

*Under the guidance of*

**Dr. Jagalingam P**

**Assistant Professor Sr. Grade 1**

**9080012798**

**jagalingam.p@vit.ac.in**

**B.Tech.**

**in**

**Computer Science and Engineering**

**School of Computer Science & Engineering**

**®**



**Abstract:**

Blockchain ensures that communication is carried out in a decentralized manner. A decentralized chat application is very much needed to ensure security, cost-effectiveness, control of data, faster processing, immutability and traceability in the today’s world. Decentralized chat application overcomes the disadvantages caused by conventional messaging applications like central node failure, data manipulation, insecure message channelling, third party intervention etc.

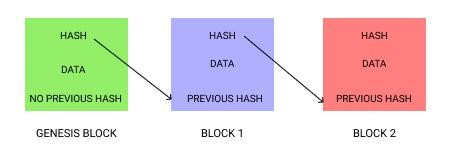
***Key words:*** Solidity, Decentralization, Immutable Ledger, Smart Contract, Digital Signature.

**Introduction:**

Blockchain based Decentralized applications (Daaps) are emerging, adapted and most hyped phenomenon in the current decade. Blockchain is the core technology behind cryptocurrency. It was first implemented by Satoshi Nakamoto in 2009.It could not draw attention in earlier stages but now Bitcoin which the most popular cryptocurrency has market worth 10 billion dollars. Blockchain now a days has emerged as revolution in the peer-to-peer communication. Due to its decentralized architecture, cost-effectiveness, immutability and trustworthy qualities blockchain has a future great potential in playing a key role in real-time communications.

The goal of this project is to implement blockchain technology in real-time chat applications. Traditionally, chat applications like Facebook, WhatsApp, Telegram use centralized servers for storing their data. In this architecture everything is central server dependant. If central server fails all the user data stored will be lost. There is also a chance of data tampering and leaking in centralized server storages. Decentralized applications (Daaps) overcome these issues by in cooperating blockchain technology which use cryptographic hash functions to build an immutable distributed ledger/database which is distributed among all the participant nodes (computers) which are core components of proof of stake in the network. This ledger runs on a P2P network of multiple nodes connected in a network. Each node in network takes part in trusted consensus algorithm using Proof-of-Work. Block contains block information in the header and functional data which chains the information on the block to its previous one. If some someone tries to change the data it in one of the blocks, they have to make changes to copy of block data in all the participant nodes. which is practically impossible hence making the blockchain network unaffected and tamper-proof. Transactions on blockchain are updated and stored across all nodes which all nodes can view at any time which makes block chain technology transparent. Blockchain in an app advances data availability maintaining security and transparency of the application.

Dapps rely on the execution of smart contracts on the top blockchain which avoids third-party intervention for transaction handling. A smart contract is computer program which includes predefined agreement between both parties (sender and receiver). It automatically executes when pre-defined conditions are met. Therefore, incorporating a real-time chat application with blockchain is must now days to ensure proof of work, security, efficiency and transparency.



**Objective:**

**T**he main objective of this project is to gain better knowledge about block chain and to implement it in the peer-to-peer communication. These days the popularity and the usage of block chain has been increasing so rapidly, so making our own project by learning about block chain is helpful.

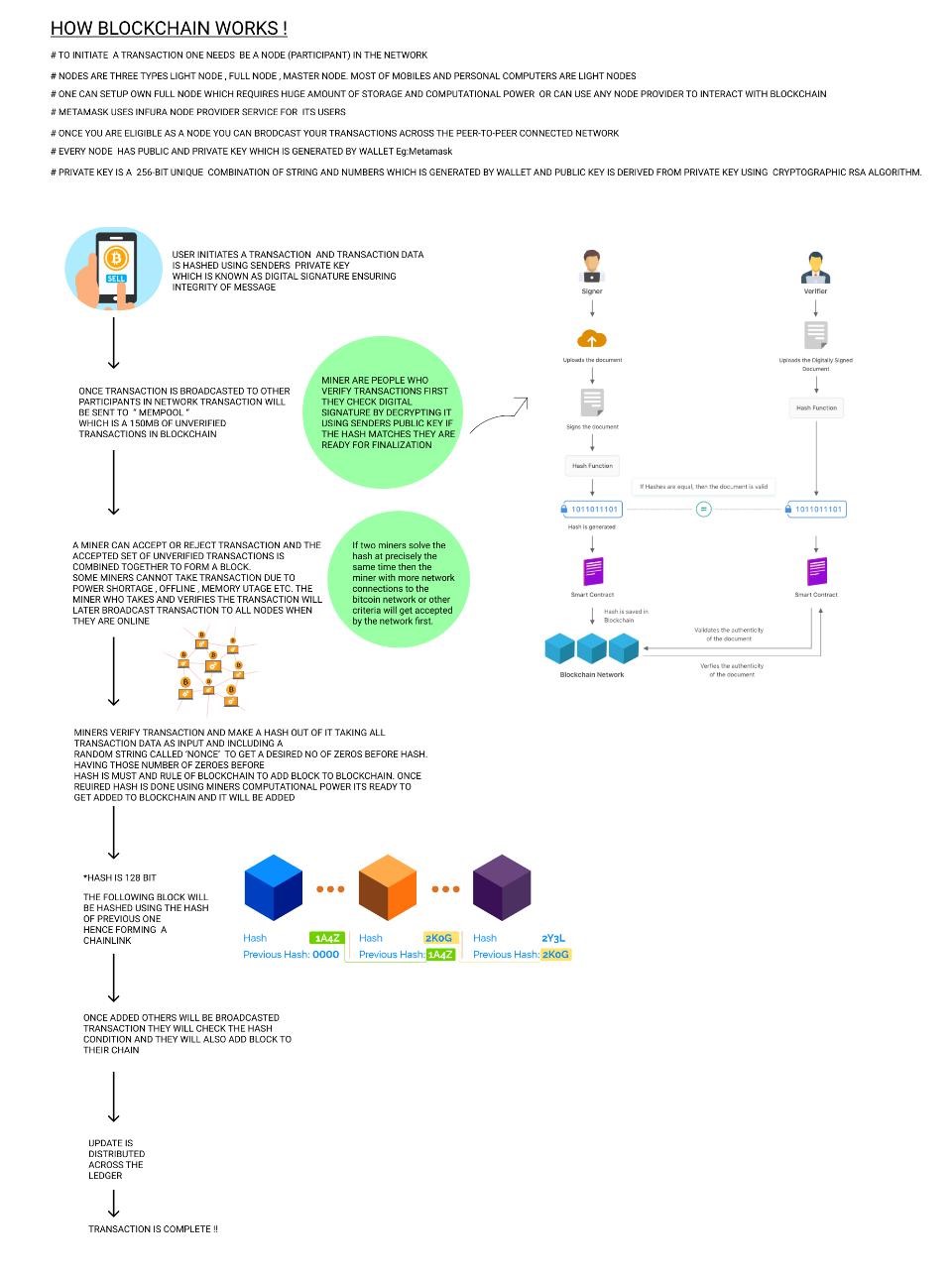
The objectives we are going to provide to the users using our project are: -

* Personal Room chat with peers
* Creating Rooms
* Transaction based chat
* Payment making instantly
* Unrevealing the person details while chatting
* Easy to use interface
* Less overhead
* Privacy preserving and fast working with low overhead

**Problem Statement:**

In recent time security has been a major concern as the intruders are increasing day by day. Most often people like to message each other rather than calling each other this increased the demand for chatting applications in the world. So, the social media platforms like Facebook, what’s app, Instagram and Snap-chat have gained a huge popularity in the recent years. But security, privacy have been greater issues. These chatting app have changed their policies in the recent years which made even data or transactions made on these apps can be modified by intruders. Blockchain which is an immutable ledger-based technology helps to get rid of these modifications. Considering its great potentiality, it provides a great security, transparency, Integrity and resilience to its users than existing conventional centralized server-based technologies.

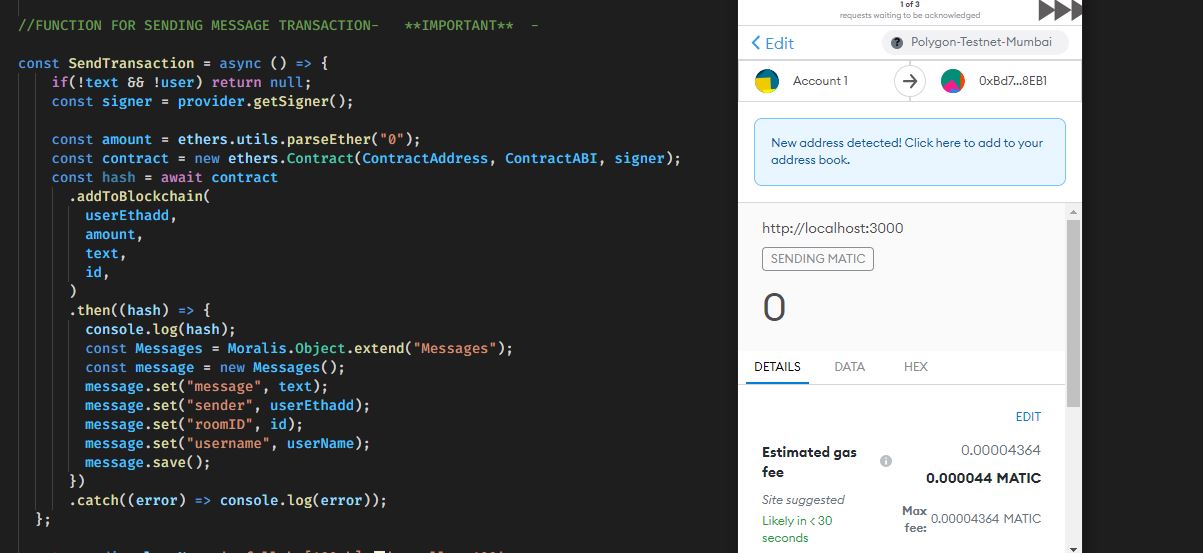
Blockchain helps to make our transactions safe as modifications can’t be done unless you own at least 50% of the blockchain which is practically impossible for any intruder. The rectification of privacy issues is our problem statement.



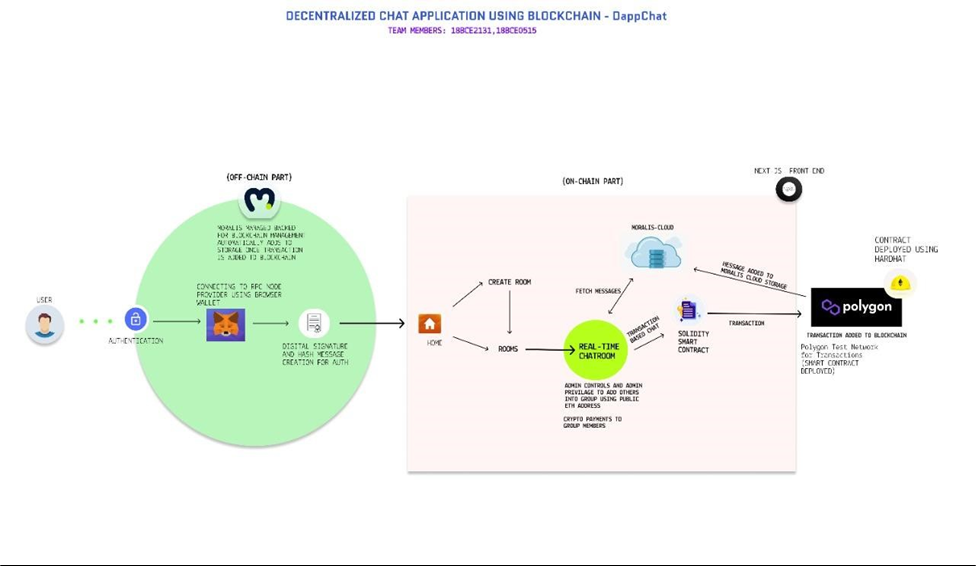
**METHODOLOGY:**

In our project we are using meta mask as our wallet. Our wallet is assigned a unique Public key and Secret key. user has to login to our Decentrailsed app using his metamask wallet by digitally signing. Once digitally signed user will be taken Dashboard where he can create and join Rooms. Dashboard contains all the information about the user displayed and logout functionality.

We can join the room and Room details stored in the Polygon blockchain will be synced from smart contract and Chat page will be showed and after that for every message you send smart contract through RPC-JSON protocol will invoke payment function where pays some gas fee to store their data in blockchain. Sender name, Room-ID, Message, Timestamp, will be stored in the blockchain. We also created a special function in the smart contract to store these all details in a publicly readable array using which we fetch the message details and fetch in the respective room pages respectively. And also, sometimes transaction takes few seconds to complete so to make the process much faster we are using secondary storage moralis for fast posting and fetching of messages in real-time.

****

* **ARCHITECTURE:**



**CHALLENGING ISSUES:**

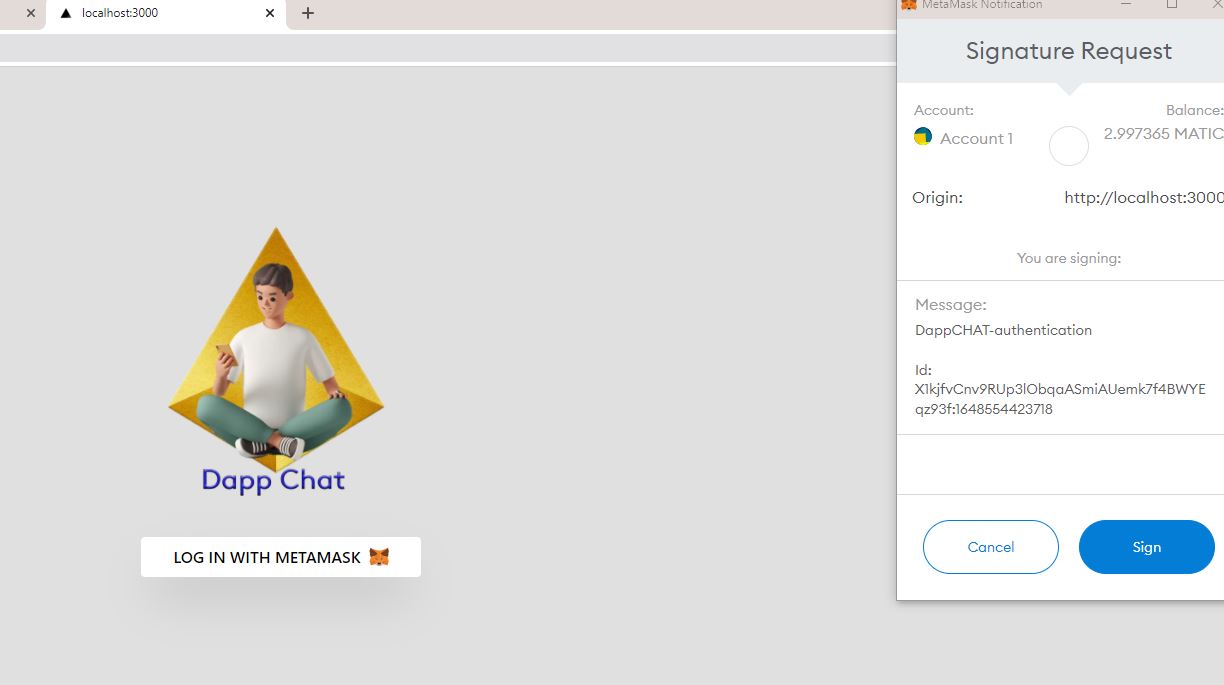
While implementing this project we have found few issues like: -

* During the literature survey part, we didn’t find many papers in order to have a better understanding of how the previous models worked. This problem is being addressed by learning ourselves.as this is a completely new and first of its kind implementation.
* As the blockchain is an emerging technology it has been challenging to do the project but trying new things and making modifications helped us to rectify them on our own.
* Storing this much of data in the blockchain is quite computationally expensive, since this is the utmost challenging issue in this project. But there is future scope for improvement.
* The comparison between the project we have done and the ones which has been previously implemented is there is no feature of group chat in real-time. There are only previous works on chatbots.

**EXPECTED RESULTS:**

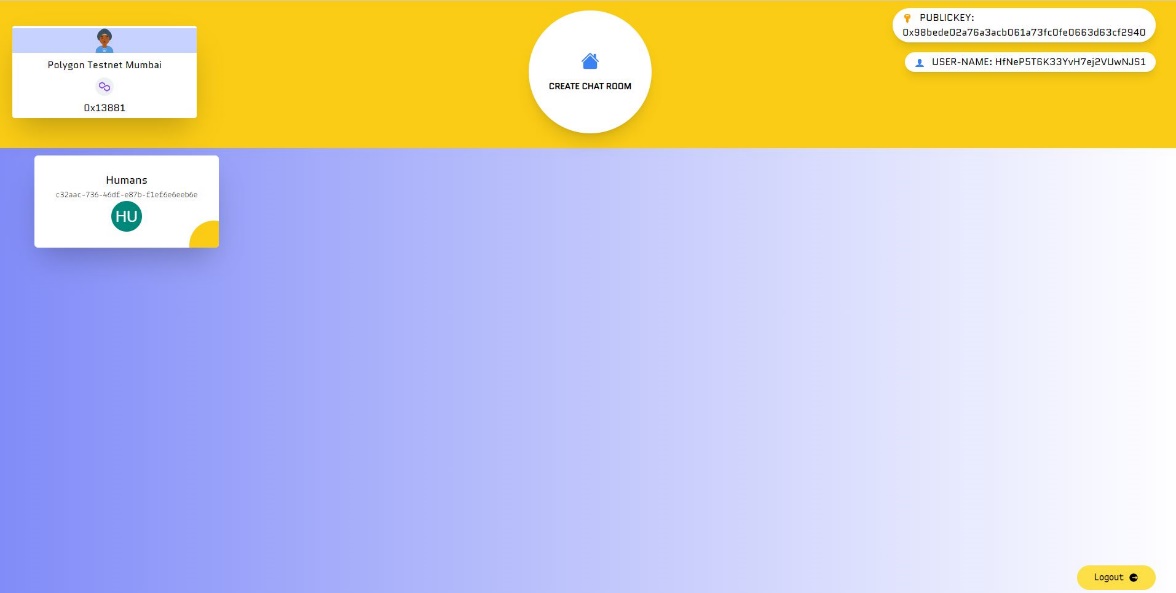
* **LOGIN PAGE: (metamask login feature)**

Here we are digitally signing with our wallet to log-in website.

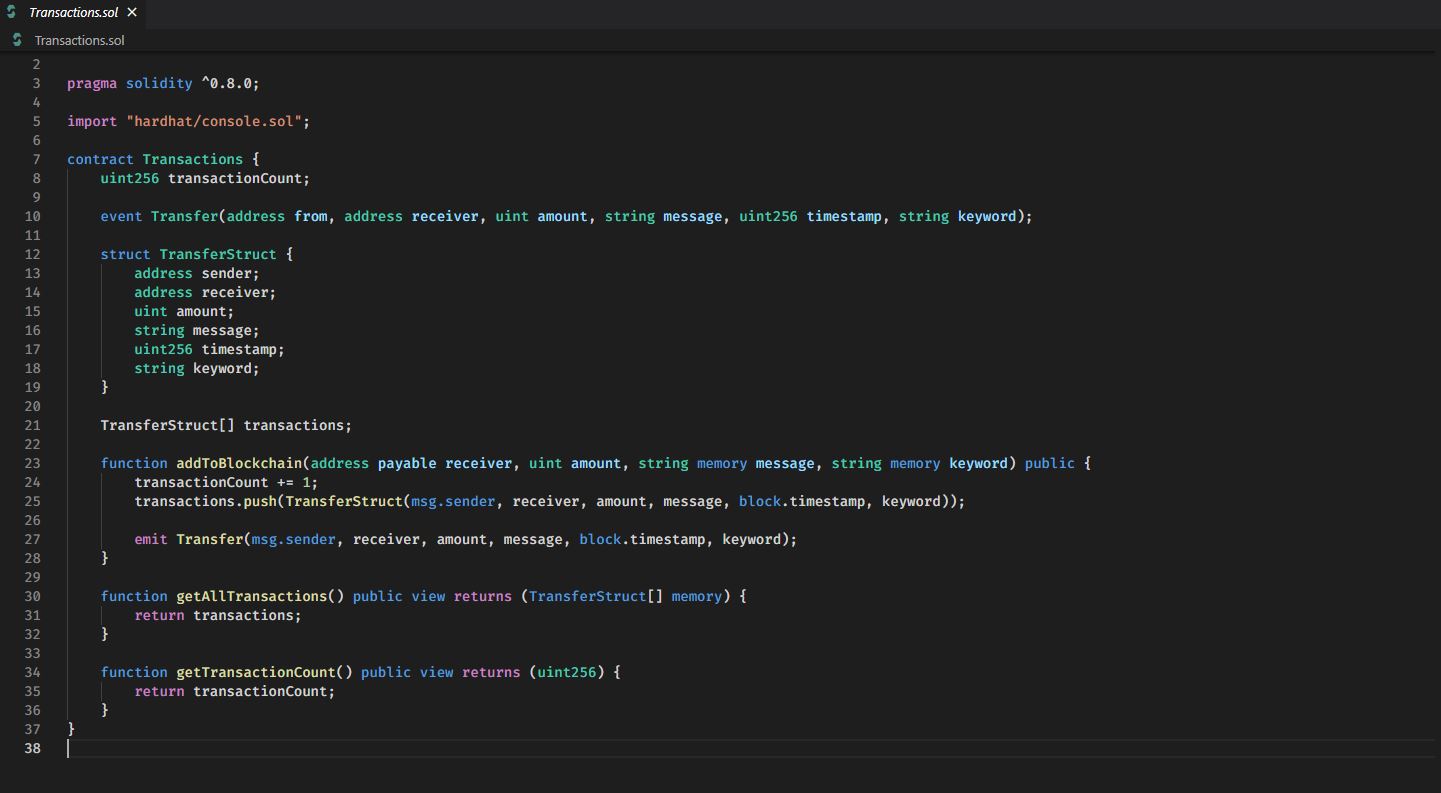
****

* **DASH BOARD:**

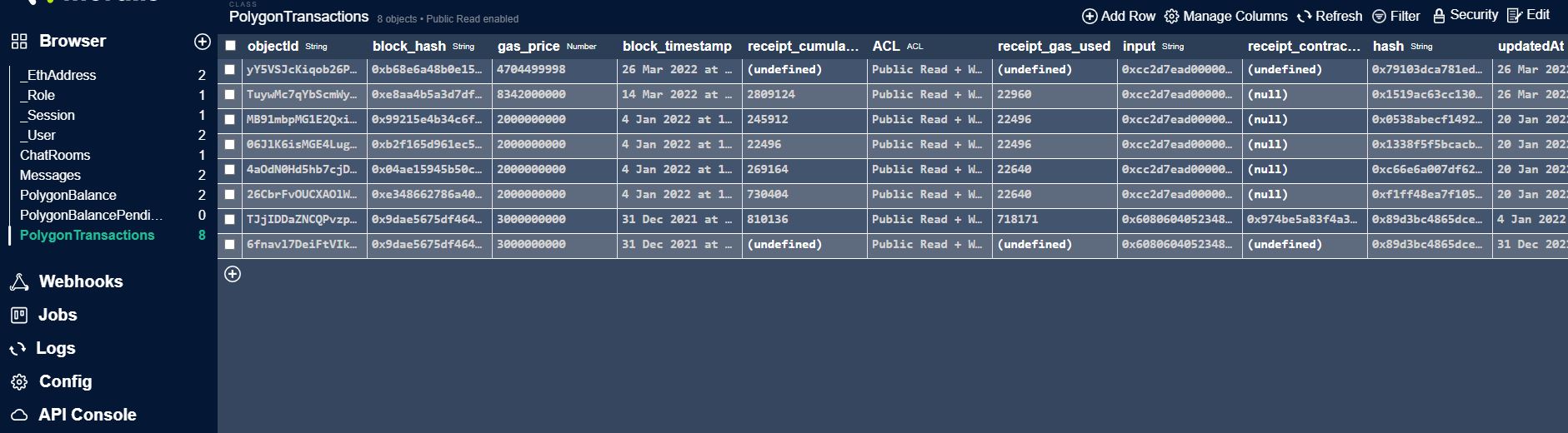
We can create and join chat rooms in the dashboard and all the user wallet details is mentioned in the user-interface here.



* **SOLIDITY CODE (SMART CONTRACT):**

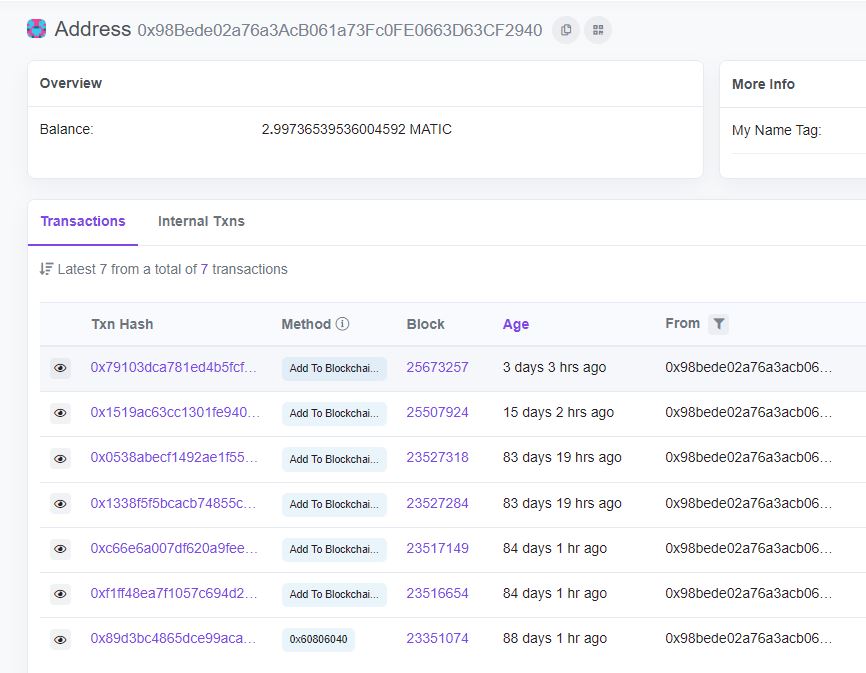
****

* **MORALIS SERVER (OPTIONAL DATABASE):**

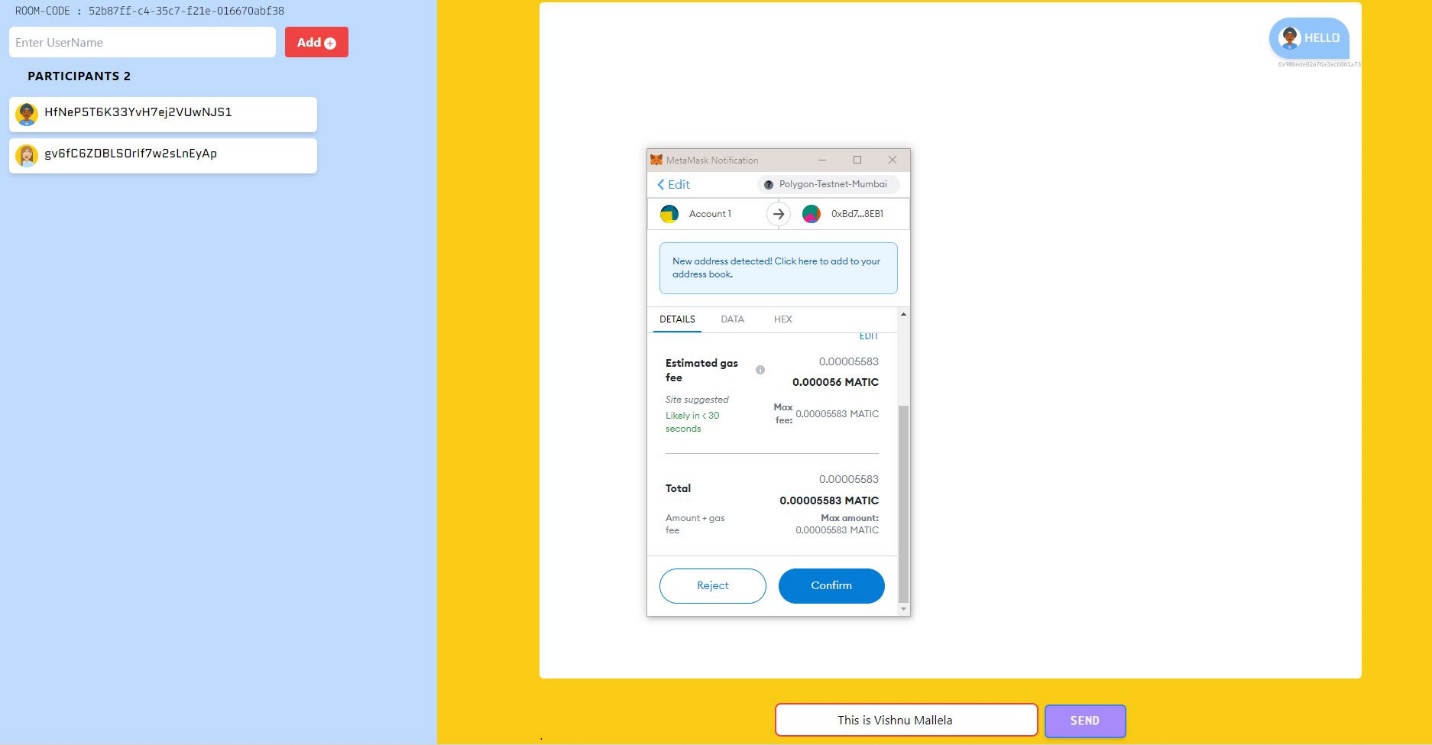
****

* **POLY SCAN RESULTS:**

It is to check the successful transaction on blockchain. Proof the details are added to blockchain.



* **CHAT ROOM:**

****

**DETAILS OF SOFTWARE:**

**FRONT-END:**

**NEXT.JS:**



* It is used build the front-end part of application. Next.js is a open source web-framework built on top of Node.js and React. It comes with the in-built server-side rendering.

**META MASK WALLET:**

****

* Metamask is a software cryptocurrency wallet used to interact with the Ethereum blockchain. It allows users to access their Ethereum wallet through a browser extension or mobile app, which can then be used to interact with decentralized applications.

**SOLIDITY:**

****

* Solidity is an object-oriented, high-level language for implementing smart contracts. Smart contracts are programs which govern the behavior of accounts within the Ethereum state.

**HARDHAT:**

****

* Hardhat is **an environment developer use to test, compile, deploy and debug dApps based on the Ethereum blockchain**. As such, it helps coders and developers to manage many of the tasks that are inherent to developing dApps and smart contracts.

**ETHER.JS**

****

* The ethers.js library aims to be a complete and compact library for interacting with the Ethereum Blockchain and its ecosystem.

**BACK-END:**

**MORALIS:**

****

* Moralis is the fastest path from idea to mass-scale dApp, letting you focus on great experiences rather than heavy lifting.
* Moralis as an optional database. Though backend is built totally on blockchain moralis usage helps us to auto sync login sessions details and users’ balances, messages and all user related data including Room details. It is used as secondary storage in this project.

**BLOCK CHAIN:**

**POLYGON:**



**Polygon** solves common **blockchain** pain points, offering low gas fees and high speeds without sacrificing security. Over 6000 dApps are already onboard.

**Literature Survey:**

1. In this paper the author has used libp2p model in order to get rid of the centralization model and could obtain decentralized one. Due to this feature the data is not stored in a single location thus increases security. Here there is a login page with a basic validation then a chat page will be appeared where we will be having two channels firstly where anyone can join and chat, second one is a bit private only the persons with a URL can only join. There are other options like file uploading. The drawbacks of this model are, there are very less features and more over while sending the file in the group raises a lot of duplicate copies which increase the garbage.
2. In this paper the author feels that present messaging apps like what's app and signal are not that safe for the users as hackers could enter and modify as all the private keys are stored in a un trusted third party location the hackers can trick these applications by sending a decryption key to them.So, the author use blockchain enables E2EEE framework which uses the concept of the PKI public key infrastructure and can provide end to end encryption for communications that are made online. Here we have four phases 1. Certificate generation where the registered user gets a public key which is stored in mobile network operator and instant messaging server. All the private keys for the generated public keys are stored in their trusted third-party apps where we need a digital signature to access them. 2.While sending a message the sender will get the public key from the MNO and using his secret key, he will generate a chain key using all these a message key is also created, for every message a chain key and message key will be generated and updated so even if a key gets lost no problem will be raised .3. While receiving message the message key will be decrypted and chain key will be updated as here no common secret key the problem is removed .4. For backing up the data the author uses user's own google drive etc... and make free that they can create a key.

**[3]** In this paper the author talks about the architecture of a Hyperledger fabric. A Hyperledger is a basic platform for distributed ledgers which can perform various transactions taken. Hyperledger fabric is used for various implementations like smart contracts (used to store procedure), benefiting technologies this is an extension of Hyperledger.

The architecture of this includes namely three type 1.deploy transaction: a ready to be invoked chain code is installed on peer, 2. invoke transaction: this is a continuation of previous where the installed chain code is invoked, it executes and tells status as success of failed, 3. query transaction: the state will be given as output.

Every peer gets a certificate to be member. During a transaction a peer is connected to network using this and generates transaction certificate.

**[4]** In this paper the author has explained about the,Emerging blockchain based application: In the block chain the data is immutable even though if it is changed, we check the peers and update with the correct value. Bitcoin is another technology which makes a digital wallet without any bank account. It can also be used in medical fields where full patient access is not provided this leads to false suggestion of medicines. Smart contract is a useful concept in block chain where the process is predefined and without the need of anyone the transaction would take place. It is even helpful in educational system by making research papers open and decentralised so that everyone could read them and if possible, make modifications, global grading system can also be implemented so the view is unified. Similarly, there are so many uses of blockchain.

Advantages of blockchain: We can create smart contract which are operated themselves. A credit system can be implemented which is good. And we have four types in block chain namely public, private, consortium and hybrid we can use anything based on our usage. We just need blockchain, smart contracts, services and user interfaces to implement a blockchain.

**[5]** According to author, Blockchain consists of blocks which contains messages, proof of work, reference and maintains historical record, transparency and includes shared database, transaction, peer to peer network using these a block chain preserves irreversibility and uses pseudonymise and secures cryptocurrency made up of and bitcoins. Blockchain provides service perspective, supports logical inclusion and architectural characteristics. Service perspective has validation, distributed trust and offers scalability and multiple writers. Logical inclusion has transaction dependency which provides computational logic and offer transaction rules. Architectural characteristics have timestamp blocks these use encrypted data transmission and P2P transmission.

The advantages provided by the blockchain are independence, security, efficiency, robust, low cost, reliable and so on

The pitfalls mentioned by the author are complicated usability, legalization, latency, wasted resources, low throughput etc... thereby few advantages and few disadvantages are gained.

**Conclusion:**

We had designed a chat application using blockchain which has an improvement of creating groups and hiding the personal details of the user , and having an option like adding the persons whom we want to add in our group which makes our project different from the previously made chat application using blockchain . We will not let our idea stop here any betterment’s that can be made will be made by adding some more features based on the future needs.

**References:**

1. Kesavan, S., Charles, R., Rajavel, V., & Kiruba, B. (2019). P2DB-chat.
2. Singh, R., Chauhan, A. N. S., & Tewari, H. (2021). Blockchain-Enabled End-to-End Encryption for Instant Messaging Applications. arXiv preprint arXiv:2104.08494.
3. Cachin, C. (2016, July). Architecture of the Hyperledger blockchain fabric. In Workshop on distributed cryptocurrencies and consensus ledgers (Vol. 310, No. 4, pp. 1-4).
4. Li, Y. (2019). Emerging blockchain-based applications and techniques. *Service Oriented Computing and Applications*, *13*(4), 279-285.
5. Islam, I., Munim, K. M., Oishwee, S. J., Islam, A. N., & Islam, M. N. (2020). A critical review of concepts, benefits, and pitfalls of blockchain technology using concept map. IEEE Access, 8, 68333-68341.

Peng, L., Feng, W., Yan, Z., Li, Y., Zhou, X., & Shimizu, S. (2021). Privacy preservation in permission less blockchain: A survey. Digital Communications and Networks, 7(3), 295-307. [7] Singh, S., Hosen, A. S., & Yoon, B. (2021). Blockchain security attacks, challenges, and solutions for the future distributed iot network. IEEE Access, 9, 13938-13959. [8] Iyer, Shankar Subramanian, Arumugam Seetharaman, and Bhanu Ranjan. (2020)

"RESEARCHING BLOCKCHAIN TECHNOLOGY AND ITS USEFULNESS IN HIGHER

EDUCATION."

[9] Stearns, M. (2019). A Decentralized Approach to Messaging Using Blockchain Technology (Doctoral dissertation, California State University, Northridge).

---o---