

Decentralized Chat Application Using Blockchain

Venkata Narendra Vudatha, B.Tech, Computer Science VIT, Vellore Tamilnadu, India.

venkata.narendra2018@vitstudent.ac.in

Sri Sai Manikanta Vishnu Shasank Mallela, B.Tech, Computer Science VIT, Vellore, Tamilnadu,

India. srisai.manikanta2018@vitstudent.ac.in

Dr.Jagalingam P, VIT University, India. jagalingam.p@vit.ac.in

Abstract—Blockchain ensures that communication is carried out in a decentralized manner. Decentralized network runs via P2P(peer-to-peer) protocols. 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 server/node failure, data manipulation, insecure message channelling, third party intervention etc. Decentralized Chat application allows you to send encrypted messages via smart contract. Messages are stored and synchronized among all the participants in the network in real-time

Keywords— Solidity, Decentralization, Immutable Ledger, Smart Contract, Digital Signature.

I.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 predefined 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.

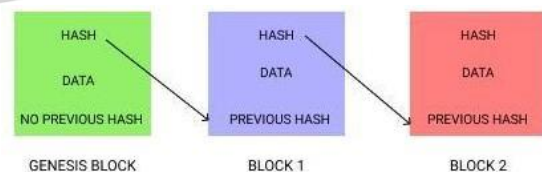


Fig 0: Block link

II. LITERATURE SURVEY

1. Kesavan, S., Charles, R., Rajavel, V., & Kiruba, B.” P2DB-chat”[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. Singh, R., Chauhan, A. N. S., & Tewari, H.” Blockchain-Enabled End-to-End Encryption for Instant Messaging Applications” [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. Cachin, C.” Architecture of the hyperledger blockchain fabric”.[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. Li, Y. (2019). “Emerging blockchain-based applications and techniques”.[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. Islam, I., Munim, K. M., Oishwee, S. J., Islam, A. N., & Islam, M. N. “A critical review of concepts, benefits, and pitfalls of blockchain technology using concept map.” [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.

III. BLOCK CHAIN WORKING

Blockchain is an immutable publicly shared ledger which is maintained among P2P connected computers (Nodes) across network. Nodes form the infrastructure of

Blockchain.P2P (peer-to-peer) is a decentralized network where there is no server or central authority over the network, where all the participants in network have equal power. If one has to participate in the network, one has run his own Node or use a Node provider to participate in the network.

Running a node requires a wallet which corresponds to location in the blockchain which contains node address, public key and private key. Wallet generates Private key from the public key using key generating algorithms like RSA, ECDSA etc. One key is used for decrypting (public key) and another is used for encrypting (private key). Private key is a 256-bit unique combination of strings and numbers. A public key is generated from private key, but generating private from public which is not possible.

When a participant in a blockchain initiates a transaction, transaction data is hashed using sender's private key (Digital Signature) and it is broadcasted into the blockchain network. Once transaction is sent it will be added to "Mempool" which is a storage of unverified transactions in blockchain.

Miner are people who verify transactions first they check digital signature by decrypting it using senders public key if the hash matches, they are ready for finalization. Miners Verify these transactions and sent them to other participants to store the transaction data in their respective nodes. Every node in the blockchain network has an option to become a mining node. Every Blockchain needs 3-6 verifications to be added to blockchain depending on the blockchain.

A miner can accept or reject transaction and the accepted set of verified transactions is combined together to form a block. Some miners cannot take transaction due to power shortage, offline, memory outage etc. the miner who takes and verifies the transaction will later broadcast transaction to all nodes when they are online.

If two miners solve the hash at precisely the same time then the miner with more network connections to the bitcoin network or other criteria will get accepted by the network first. Miners verify transaction and make a hash out of it taking all transaction data as input and including a random string called 'nonce' to get a desired no of zeros before hash. having those number of zeroes before hash is must and rule of blockchain to add block to blockchain. once required hash is done using miners' computational power its ready to get added to blockchain and it will be added.

Data at the nodes end is stored in the form of blockchain once verification is done by miners. The following block will be hashed by adding the hash of previous block. hence, forming a chain-link. Once added others will be broadcasted transaction, they will check the hash condition and they will also add block to their chain. Then, update is distributed across the ledger.

IV. METHODOLOGY

A. 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.

B. Objectives: These days the popularity and the usage of block chain has been increasing so rapidly, so making chat application using block chain is helpful for the users who prefer security.

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

C. Implementation: Methodology of this web3 application goes as follows.

User Authentication: User needs an Ethereum wallet like metamask to authenticate themselves with the app.

Here user Digitally authenticate himself with digitally signing to the polygon network using his metamask wallet. We used Moralis web3 backend provider for authenticating and managing the sessions of user. Code is written in a such a way that it notifies user to change to Polygon Network. Metamask takes care of connecting with Polygon Network using its RPC Node URL. Polygon is preferred in building this project because of its low gas fees compared to others.

Digital Signature: Digital Signature is the process of generating hash from signing data and encrypting it with sender's private key and it is verified by every other node in the chain by decrypting it using sender's public key, hence making the signature is legit and data comes from a trusted source. While verifying only if the hash of encryption and

decryption matches it is assurance that messages are not through any modifications in transit.

User room Creation / Joining: Once, authenticated user is assigned a new User-ID automatically by the system and also Dashboard displays the user's Public-key and User's ID. Here for creating the rooms we used Moralis Database which just stores data of Room ID and Users in the Room but not any Chat Data (all the chat data with keyword: room-id is stored in blockchain). User is allowed to join the rooms he is part of or he can create a one for himself and add others.

Message Sending and Receiving in Chat: User is allowed to send and receive messages in Realtime in a decentralized manner in the chat box once he joined the group. Once a user sends the message a function connecting the smart contract deployed in the Polygon network using hardhat is invoked. This function initiates a transaction with "0" payable amount, with estimated gas fees, once payment is done message gets added to blockchain. Smart contract is programmed in a way that it stores the data of Sender ID, Receiver, amount, keyword-room-ID, timestamp. And all also we programmed a public viewable array which stores the object containing all these details whenever data is added to blockchain. We are fetching the chat details from this public viewable array which resides in smart contract and filtering them according to room-ID and showing them in Realtime in the chat box. Here we used Ether.js to communicate with smart contract using its contract address and ABI.

In this section users are allowed to add other users to the group using the User-ID assigned to them initially when they were introduced to application.

D. Figures and tables:

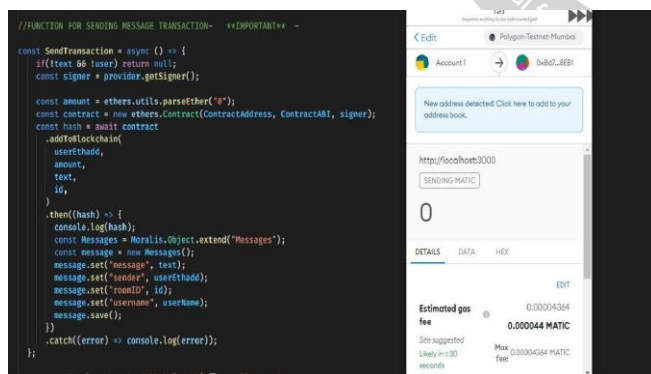


Fig 1: Output

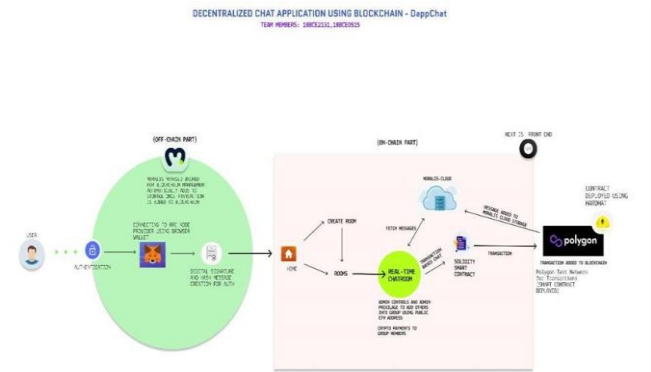


Fig 2: Architecture diagram

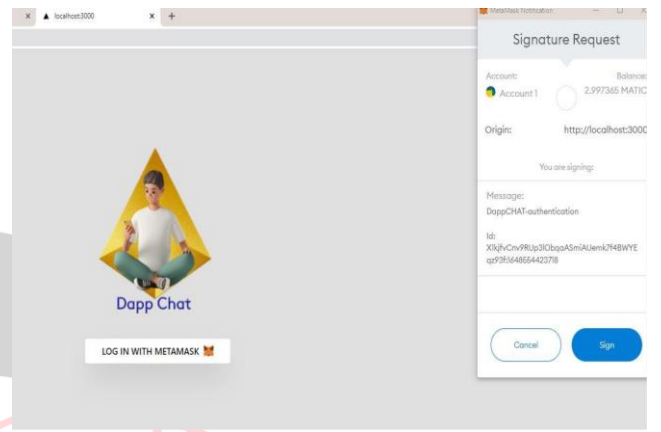


Fig 3: Login page

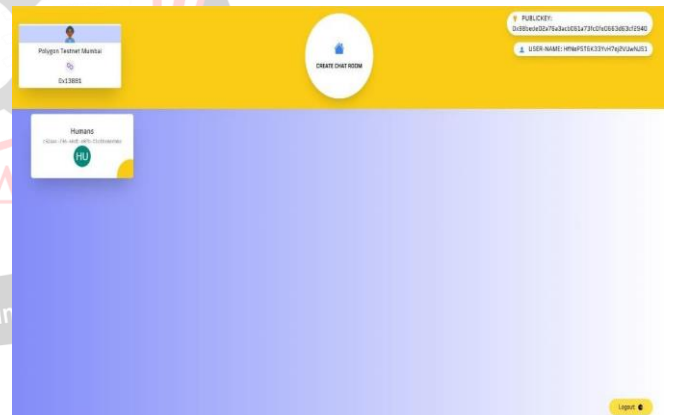


Fig 4: Dash board

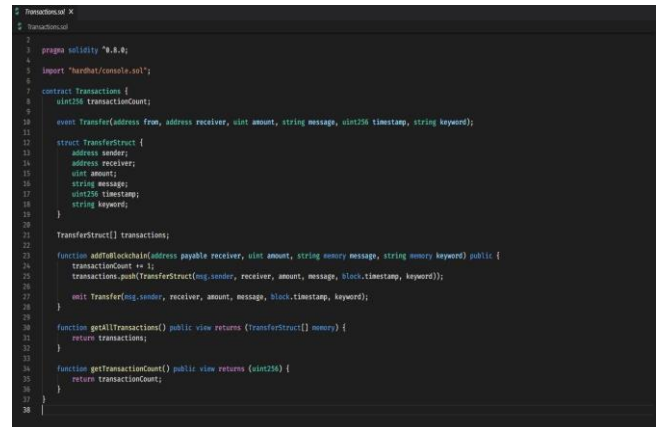


Fig 5: Solidity code

ACKNOWLEDGEMENT

This paper is completed with the help of professor Jagalingam.P without him this work could not have completed this successfully. We are also very thankful to the help, support and the freedom given by the Vellore Institute of Technology, Vellore management who had kept faith in our idea and have guided us in the right way continuously for this achievement.

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.

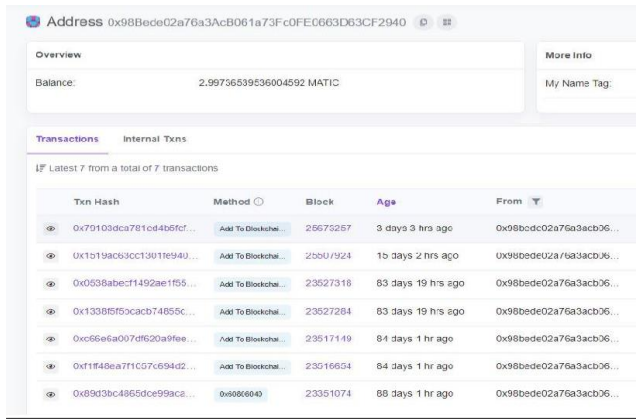
[5] Islam, I., Munim, K. M., Oishwee, S. J., Islam, A. N., & Islam, M. N. (2020). A critical review of concepts, benefits, and pitfalls of blockchain technology using concept map. IEEE Access, 8, 68333-68341.

[6] Peng, L., Feng, W., Yan, Z., Li, Y., Zhou, X., & Shimizu, S. (2021). Privacy preservation in permissionless 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).



Txn Hash	Method	Block	Age	From
0x7d103dca781c64b6f...	Add To Blockcha...	26675257	3 days 3 hrs ago	0x98bede02a76a3acb06...
0x1b19ac63cc33011e940...	Add To Blockcha...	25511924	15 days 2 hrs ago	0x98bede02a76a3acb06...
0x0538abec14592ae1f55...	Add To Blockcha...	23527318	83 days 19 h's ago	0x98bede02a76a3acb06...
0x1338f5f5cabc74855c...	Add To Blockcha...	23527284	83 days 19 h's ago	0x98bede02a76a3acb06...
0xc66e6a007df020a9fee...	Add To Blockcha...	23517149	84 days 1 hr ago	0x98bede02a76a3acb06...
0xf148ea7f1c57c594d2...	Add To Blockcha...	23316654	84 days 1 hr ago	0x98bede02a76a3acb06...
0x89d3bc4865dce99ac...	0x8006040	23351074	88 days 1 hr ago	0x98bede02a76a3acb06...

Fig 6: Polygon scan results



objectid	block_hash	gas_price	block_timestamp	receipt_cumula	ACL	receipt_gas_used	input	receipt_contract	hash	updatedAt
1	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	(underflow)	0x12f5a00000...	(underflow)	0x720ba312e...	24 Mar 2022
2	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
3	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
4	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
5	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
6	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
7	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
8	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022
9	0x101541a1a1...	439000000	24 Mar 2022 11:...	(underflow)	Public feed + k.	22048	0x12f5a00000...	(underflow)	0x1170a312e...	24 Mar 2022

Fig 7: Moralis server database

V. PERSPECTIVE CHALLENGES

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.

VI. CONCLUSION

We had designed a chat application using blockchain which is fully decentralized and having options like adding people to group without exposing their details but just by their public key and also not depending upon the central server which is owned by third-party service provider makes our project unique. 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.