International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 06 | June - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

DeChat : A Blockchain Based Chat Application

Saurav Kumar[1], Rajat Kokane[2], Nitin Kadam[3]

BE Students, Department of Computer Engineering, Dr. D. Y. Patil College of Engineering, Ambi, Pune, Maharashtra, India

---

Abstract - DeChat is a decentralized, Peer-to-Peer chat application that works on the Ethereum Blockchain. Unlike other chatting applications or websites, DeChat does not store user data on any centralized server. It works in a trustless environment to protect user's data. Data stored in a centralized server can be hacked by anybody or be used maliciously against the user. Making chatting P2P will ensure that data will not be shared with a third party at any point, and that it can be deleted whenever the user wants to, protecting privacy.
The need for decentralization is at an all-time high due to corporations violating user privacy, authoritative and oppressive governments using data to spy on citizens, and increasing cyber-attacks that leak private data of users. There is no trust on the internet. Software that doesn't rely on trust are needed to combat privacy grievances. This project hopes to aid in this effort.

***Key Words*:** Blockchain Technology, Ethereum network, Peer to Peer networking, Decentralized systems, Chat applications

1. INTRODUCTION

Blockchain technology is getting more and more popular in recent times. A shared ledger ensures that data is correct and unaltered. Most people, when asked about Blockchain, think of cryptocurrencies, the most popular being Bitcoin. However, Blockchain technology is much more than that. It has found various applications, such as storing lists, ensuring data integrity, IOT, real life applications such as E-Voting, banking, online marketplaces, digital goods, the list goes on.

Along with so many applications, Blockchain technology also stands for anonymity and privacy. As data is not stored in any centralized servers, users retain full control of their data.
Privacy focused applications are gaining popularity in recent years as people are becoming more and more aware about how their data is being used. One does not wish for their private chats to be leaked. Due to this, Blockchain technology can be used to create a privacy focused application.

1.1 LITERATURE SURVEY

[1] Blockchain Technology has a number of conceivable uses and applications along with decentralization. This article also has a full analysis about the benefits this technology offers and environments where it will improve computation compared to traditional approaches. The Blockchain has emerged as one of the most promising infrastructure technologies within next generation of online-based programmes, such as social services, the internet of Things (IoT), name systems, and security due to the benefits of power allocation, consistency, security, and transparency.
[2] Solidity is an object-oriented, high-level language for implementing smart contracts. Smart Contracts are programs which govern the behaviour of accounts within the Ethereum state.
It's an agreement or set of rules that govern a business transaction. It's stored on the blockchain and is executed automatically as part of a transaction. It allows transaction to be carried out without the need for a governance, legal system, central authority or external enforcement mechanism.
[3] Ethereum now uses a proof-of-stake algorithm for mining, which consumes less energy than proof-of-work algorithm. This removes the need for expensive hardware and reduces gas fees for transactions.
[4] Decentralized applications (DApps) are digital applications or programs that exist and run on a blockchain or peer-to-peer (P2P) network of computers instead of a single computer. DApps are outside the purview and control of a single authority. DApps—which are often built on the Ethereum platform—can be developed for a variety of purposes including finance, management, data storage, gaming, social media, etc.

1.2 RESOURCES REQUIRED

HARDWARE REQUIREMENTS

Sr. No.	Parameter	Minimum Requirement	Justification
1.	CPU	Pentium 4	Minimum required specification for a modern web browser.
2.	RAM	4 GB	To ensure the 64-bit version of the browser works and for overall smoothness.
3.	Storage	200 MB	Size of a modern web browser.

SOFTWARE REQUIREMENTS

1. **Operating System:** Windows 7 (64 bit)
2. **IDE:** Microsoft Visual Studio Code, Remix IDE

3. **Programming Languages:** Solidity, JavaScript, React.js, CSS, HTML

TOOLS

1. **Software:** Node.js, ethers.js, crypto.js, Ganache, Hardhat
2. **Services:** MetaMask, Etherscan
3. **Cryptocurrency:** Sepolia ETH, local testing ETH

2.METHODLOGY

The aim of this project was to create a chatting application which eliminated the privacy woes of currently popular applications that use the client-server model. There was a need for a technology which eliminated the need for a server, and which will connect the users in a peer-to-peer. As Blockchain technology fulfilled these requirements, it was decided that to achieve our goal, Blockchain technology would be used.

Ethereum network is selected for our project. The most important reason for this was that the Ethereum network supports smart contracts. Smart contracts are digital contracts that can be used to provide functionalities to decentralized applications and require no maintenance. A smart contract will carry the entire logic of our project.

The aim of our project was to create a platform where users will be able to chat without any intermediary. To do so, we wrote a smart contract which allowed exchange of encrypted messages between two users. As the smart contract is on the Ethereum network, user data won't be stored anywhere but their own devices,

After the smart contract was created, the UI of the app was built using React.js, CSS, HTML, and JavaScript. An effort was made to keep the user experience as simple as possible. Even for signing up on the application, all users have to do is provide their Ethereum account numbers, which were directly retrieved from their MetaMask wallets. This made logging in simpler, and improved privacy.

For testing purposes, Ganache and Hardhat were used. Test ETH wallets were provided by these software to test the smart contract without incurring any costs. After ensuring that the smart contract was working as intended, the application was put in production by deploying it on a web domain.

Objective

Our objectives for this project are as follows:

1. Create a smart contract which contains the logic of our application.
2. Obtain enough Sepolia Ethereum required for making our application work.
3. Create a UI/UX that is user friendly.
4. Use Hardhat and Ganache to test the application.
5. Ensure that chats will stay between the involved users only.
6. Implement encryption to ensure that the chats are secured.
7. Create a website to host our application.

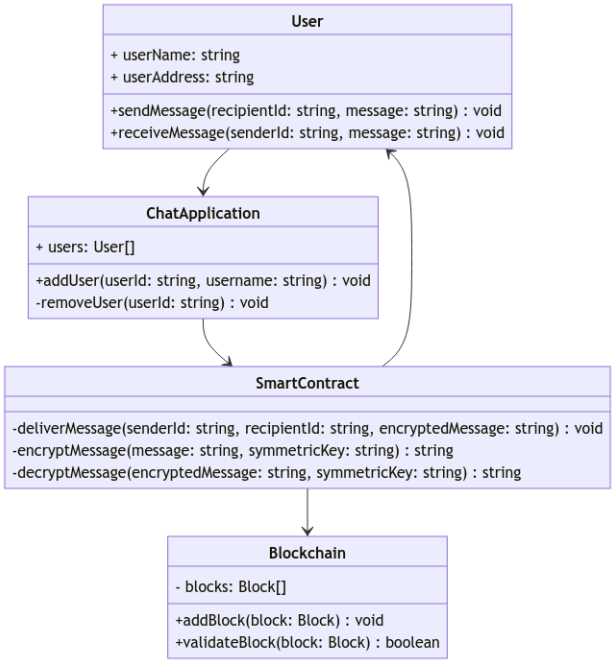
3.STATEMENT OF SCOPE

The project is a chatting application where text messages can be sent and received. It is a peer-to-peer, decentralized application. It means that no centralized server or a third party is involved when two users are chatting. To use the application, the users are required to have the MetaMask browser extension installed. Their MetaMask must also have at least 0.5 Sepolia ETH in order to use the application.

This application only supports text messages in its current form, and not media or file transfer or voice or video calling. While these features can be implemented in the application, it's beyond the scope of the project.

As this application stores user data locally, and no centralized servers are involved, this application is suited for users who prioritize their privacy above everything else.

SYSTEM ARCHITECTURE



Class Diagram of DeChat

The application is a web application that can be accessed through a browser. When opening the website, the user will see a homepage that asks for a username and Ethereum account address, which is public data. After this, the smart contract will be executed to allow the user to use the application freely. To do this, the user must pay a token amount of Ethereum from their entered address, which they can do so from MetaMask.

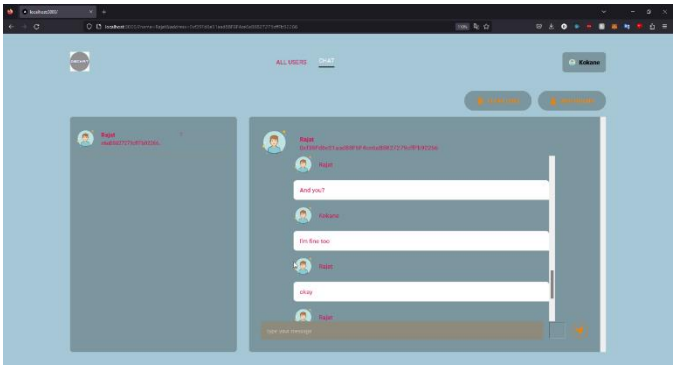
After the transaction is completed, the user will be taken to the application, where they can see a chat screen, a Add Friend button, and all their chats listed. To chat with a new friend for the first time, the user will have to click on the Add Friend button. Here, the user will be prompted to enter their friend's username and Ethereum account address. Upon doing

so, a smart contract will have to be executed, for which the user must pay a token amount. After the transaction has been completed, both users will be able to chat with each other freely.

The smart contract will encrypt the chats and carry them from the sender to the recipient though the blockchain. For encryption, AES-256 algorithm will be used. The user data will be stored on the user's devices locally.

4.RESULTS AND PERFORMANCE ANALYSIS

When a user sends a message to another user, it is encrypted and then carried by a smart contract through the blockchain network and sent directly to them, without any third-party interference. For sending messages, the smart contract requires a payment to execute it. Users can easily complete the payment as they receive a payment request on their MetaMask wallets. User chats are stored locally and can be deleted anytime.



Chat Screen

As the application utilizes decentralized technology, it requires no verification from users such as phone numbers, email IDs, or any official documentation. It also has zero downtime, unlike an application that uses a client-server system. Even if a single node is down, other users are unaffected as they're connected in a P2P network.

To achieve such a high level of privacy, some functionalities found in typical chat applications had to be given up. Free messaging is not possible on this application, as the smart contract requires ETH to execute. As user data is stored locally, a user cannot access their chats on multiple devices. The application is also slow when compared to conventional chat applications due to its decentralized nature.

There is also a risk of the application being used for distribution of illegal content, as users will easily be able to cover up their tracks.

5.CONCLUSION

In this way, we can use Smart Contracts and the Ethereum Blockchain to create a Decentralized chat application. As Ethereum works on the Proof of Stake algorithm, there isn't a need for powerful hardware or high computational power to make our program work efficiently. Conventional centralized technologies as of now are still faster and cheaper for chatting purposes but they have serious privacy concerns too. As further developments occur on the Ethereum network, the time taken to execute Smart Contracts will reduce too. This method can also be applied to other blockchains such as Solana and Polygon to reduce operational costs.

Decentralization will be essential in the near future as censorship and active monitoring by advertising firms and authoritative governments become more and more severe. People must have a right over their own data and this technology will help them to protect it.

6.REFERENCES

[1] Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends". 2017 IEEE International Congress on Big Data (BigData Congress) (2017).

[2] Sourabh, Deepanker Rawat, Karan Kapkoti, Sourabh Aggarwal, Anshul Khanna "bChat: A Decentralized Chat Application". International Research Journal of Engineering and Technology (IRJET) (2020).

[3] Proof of Stake: https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/

[4] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng and V. C. M. Leung, "Decentralized Applications: The Blockchain-Empowered Software System," in IEEE Access, vol. 6, pp. 53019-53033, 2018, doi: 10.1109/ACCESS.2018.2870644.

[5] Secure Peer-to-Peer communication based on Blockchain Kahina Khacef, Guy Pu-jolle.

[6] Buterin V et al (2014) A next-generation smart contract and decentralized applica-tion platform.

[7] Wu, K, Ma, Y, Huang, G, Liu, X. A first look at blockchain-based decentralized ap-plications. Softw: Pract Exper. 2021; 51: 2033– 2050. https://doi.org/10.1002/spe.2751

[8] Parameswaran, Manoj & Susarla, Anjana & Whinston, Andrew. (2001). P2P net-working: An information-sharing alternative. Computer. 34. 31 - 38. 10.1109/2.933501.

[9] Nicolas Six, Nicolas Herbaut, Camille Salinesi, Blockchain software patterns for the design of decentralized applications: A systematic literature review, Block-chain: Research and Applications, Volume 3, Issue 2, 2022, 100061, ISSN 2096-7209, https://doi.org/10.1016/j.bcra.2022.100061.

[10] Nallapaneni Manoj Kumar, Pradeep Kumar Mallick, Blockchain technology for se-curity issues and challenges in IoT, Procedia Computer Science, Volume 132, 2018, Pages 1815-1823, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2018.05.140.

[11] Xuemin Shen, Heather Yu, John Buford, and Mursalin Akon. 2009. Handbook of Peer-to-Peer Networking (1st. ed.). Springer Publishing Company, Incorporated.

[12] Zupeng Li, Daoying Huang, Zinrang Liu and Jianhua Huang, "Research of peer-to-peer network architecture," International Conference on Communication Technology Proceedings, 2003. ICCT 2003., Beijing, China, 2003, pp. 312-315 vol.1, doi: 10.1109/ICCT.2003.1209091.
[13] N. Chaudhry and M. M. Yousaf, "Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities," 2018 12th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, 2018, pp. 54-63, doi: 10.1109/ICOSST.2018.8632190.
[14] Xiong, H.; Chen, M.; Wu, C.; Zhao, Y.; Yi, W. Research on Progress of Block-chain Consensus Algorithm: A Review on Recent Progress of Blockchain Consensus Algorithms. Future Internet 2022, 14, 47. https://doi.org/10.3390/fi14020047