Smart Computing and Systems Engineering, 2021
Department of Industrial Management, Faculty of Science, University of Kelaniya, Sri Lanka

Paper No: SE-18 **Systems Engineering**

A decentralized social network architecture

Tharuka Sarathchandra*
Department of Software Engineering
University of Kelaniya, Sri Lanka
tharukas@kln.ac.lk

Damith Jayawikrama
H&D Wireless SL, Sri Lanka
damith@damith.com

***Abstract* - Billions of people use social networks, and they play a significant role in people's lifestyles in the current world. At the same time, due to globalization and other factors, the use of these social platforms is expanding daily, and a variety of activities take place inside these platforms. These networks are centralized, allowing social network-owned companies to track and observe the activities of their users. Therefore, this has been challenged to the privacy of the data of users. Also, these companies tend to sell them to third parties keeping huge profits without users' permission. Since data is the most valuable asset in today's and tomorrow's world, many have pointed out this issue. Even though decentralized, community-driven applications have come to play as a solution to this problem, there is still no successful application that competes with centralized social network platforms. Therefore, this study attempted to develop a decentralized social network architecture with the basic functionalities of a social media platform to assure the privacy of the users' data.**

Keywords - blockchain, ethereum, decentralized web, ipfs, web3.0

I. INTRODUCTION

Nowadays, almost every person uses social media, and social networks play a crucial part in lifestyles. Most people around the world are connected with social networks. The first recognizable social network, "Six Degrees," was launched in 1997, allowing users to create a profile and become friends with other users [1]. The world has come a long way where now people are using many social networks like Facebook, WhatsApp, Instagram, Twitter, etc. The usage of these social networks is increasing day by day [2].

It is a must to discuss the reasons for the increment of the usage of these social networks. The most common reason is that the users want to stay in touch with others and stay updated on what is going on around the world. With the busy schedules and workload, people miss the chance of meeting people physically. Therefore, they spend time virtually, which is more beneficial. On the other hand, People use social networks to share photos and videos for entertainment and share opinions and ideas. Besides that, people use social media platforms to research new products to buy. With these facts, businesses do more and more online marketing focusing on the target audience and try to grab the customer. However, these social networks are centralized, and there are several problems with those social networks. Recently, those problems became hot topics.

With the popularity of web 3.0, people tend to find a solution within the web 3.0 technology stack for the problems they face with current web 2.0 technologies. As a result of that, decentralization came to play. With the evolution of cryptocurrency, mainly Bitcoin, this decentralized culture came into practice.

Then people proposed decentralized solutions to develop applications apart from using decentralization only in cryptocurrencies. Nowadays, many organizations have developed applications for web, mobile and desktop computers with decentralized technologies. Those applications are called as DAPPs, and they have been able to solve many real-world problems.

II. DECENTRALIZATION

After pouring the cryptocurrencies, the word decentralization [3] became a bus word because this is used most in the crypto-economics space. Many people and companies started to do researches in this area, and thousands of hours of research, and billions of dollars of cash power, have been spent for the sole purpose of attempting to achieve decentralization. There are many misunderstandings about decentralization. Therefore, it is necessary to understand the differences between Centralized Networks, Decentralized Networks, and Distributed Networks.

A centralized system means a central location provides all services, and the network resources are placed and managed from the central location. Distributed means the network resources and the services are distributed through the network, and it might also be geographically distributed over the internet. However, the network and the resources are handled by a central authority, and they have the administrator power to do everything in the system. Google, Facebook, AWS, like almost every big company, use this distributed model in their systems. Decentralized means there is no central place or no one has administrator powers to govern the system. The system is distributed through the users of the system. Users are the people who govern the system. However, there is a critical point to consider. That is who maintains the system and who develops the new versions of the system. The answer is that almost all decentralized projects are free and open-source projects. Therefore, anyone interested in a particular area can develop the project, and many of them run on funds. However, some decentralized system has their business models, and a small amount of money is charged from the users when using the system for maintenance and other infrastructure developments. Nevertheless, by the architecture of the decentralized system, the developers of a particular system do not have a central authority to handle the system.

A. Smart contract

A smart contract [3] is older than bitcoin, and it is a computer protocol to digitally facilitate, verify, or enforce the negotiation or performance of a contract. In other words, Smart contracts are computer programs that a network of mutually distrusting nodes can correctly execute without the need of an external trusted authority. With the development of cryptocurrencies and Blockchain, Smart Contracts got attraction due to their architecture. If a bitcoin-like cryptocurrency saves the transaction in a blockchain, it is just some kind of data. However, these Smart Contracts open a way to store some executable code inside a blockchain, and it is an immutable program. It can be partially or fully self-

executing, self-enforcing, or both. Most cryptocurrencies have the facility to implement these smart contracts. Smart contracts can be used to do complex transactions between two anonymous parties. Moreover, it does not require a central authority, enforcement system, or legal guidance because it can be self-executed by itself. Therefore, Smart Contracts can be programmed to enable a wide variety of actions.

B. Cryptography

Cryptography or cryptology is the domain and study area of ways and technologies to secure communication between two or more parties from external parties. In general terms, cryptography is about constructing and analyzing protocols that prevent third parties or the public from reading private data. Various aspects of information security, such as data confidentiality, authentication, non-repudiation, and data integrity, belong to modern cryptography. Cryptography mechanisms are based on mathematics and computer science, electrical engineering, communication science, and physics disciplines.

When discussing decentralized systems, the privacy of the data is a huge issue. As data is shared publicly, different cryptographic mechanisms are used in decentralized networks to resolve this problem. Cryptography has two categories as symmetric-key cryptography and asymmetric key cryptography

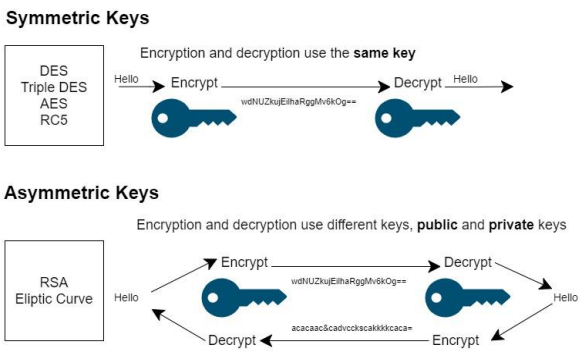

Fig. 1. Types of encryptions

In symmetric-key cryptography, it is used only one key in both encryption and decryption processes. However, asymmetric key cryptography uses two different keys in encryption and decryption processes.

C. Blockchain

A blockchain [4] is a data structure that enables identifying and tracking transactions digitally and sharing this information across a distributed network of computers, creating a distributed trust network. The data structure of a blockchain is a linked list, and the speciality is it being an immutable linked list.

D. Features of Blockchain

- **Distributed and Decentralized** - Data are replicated on every node in a distributed P2P network. Furthermore, each copy is identical to others. It can also be decentralized with some lighter nodes not having whole data storage with limited connection.
- **Consensus mechanism** - All users in the blockchain network can come to a predetermined programmable agreement on the validation method and can be by consensus. There are several consensus algorithms like Proof of Work, Proof of Stake, Delayed Proof-of-Work, Proof of Importance, Delayed Proof-of-Stake, etc. Most decentralized applications use POW and POS[5].
- **Irreversibility and crypto security** - One would need to command at least 51% of the computing power (or nodes or sake) to take control of the bitcoin blockchain (or other) [6].

E. Bitcoin and cryptocurrencies

Bitcoin [7] is the world's first known public cryptocurrency invented by an anonymous man known as Satoshi Nakamoto. He published the research paper Bitcoin: A Peer-to-Peer Electronic Cash System in 2008. That was the point that the whole world gives attention to cryptocurrencies. In this paper, the author has resolved the problem of public transaction verification. The concept of proof–of–work [8] is intruded in this research. It uses a blockchain as its underlying data structure and the public ledger.

After introducing Bitcoin in 2009, hundreds of cryptocurrencies were introduced to the world, and most of them have used blockchain, and some of them have their own data structures like Directed Acyclic Graph. However, the concept was almost the same. That means all data store in a public ledger. After the invention of proof-of-work, people have invented new concession algorithms which are different from proof-of-work. Proof of Stake, Proof of Elapsed Time, Proof of Authority, Proof of Capacity, Proof of Activity, and Proof of Burn are examples of those algorithms used in different cryptocurrencies and decentralized application developments.

F. Ethereum ecosystem

With various types of cryptocurrencies, the Ethereum [9] research was published in 2014. It is not just a cryptocurrency. Ethereum saw this use case with a different and broad view. It has proposed a platform to develop any decentralized application. Also, it has its cryptocurrency called Eth.

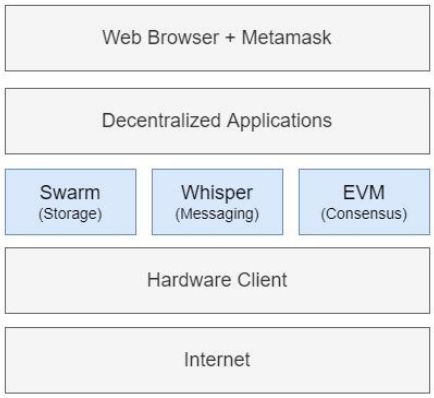

Fig. 2. Ethereum Ecosystem

Now there are hundreds of decentralized applications and crypto tokens that have been built on top of the Ethereum platform [10] UPort, Brave, Toshi, Auger, CryptoKitties, GitCoin, Minds, and Akasha are a few more popular example applications based on the Ethereum platform.

G. Ethereum Virtual Machine

Ethereum is somewhat a predecessor to Bitcoin. The Ethereum Virtual Machine (EVM)[11] is one of the primary core reasons for Ethereum to be created. While Bitcoin does contain a programmable scripting language, the Bitcoin scripting language is limited to performing token transfers. EVM is designed with 20-byte addresses. Furthermore, each address space has a counter, a balance value, contract code, and persistent storage [11]. Overall, Bitcoin scripting offers a limited feature set compared to EVM as Ethereums' primary goal is to create a globally distributed computing platform.

H. Decentralized chatting

Chatting is a significant component of social media. When people are using the current social media, almost all of them have chatting functionality. Video calls, voice calls are also some categories of chats.

Before exploring more information about current projects, researchers had to answer the problem; what is the purpose of using this decentralized nature for the chat protocols? The answer is that these chat service providers can listen to what the users are chatting about and all these data routes through their servers. Therefore, privacy cannot be ensured

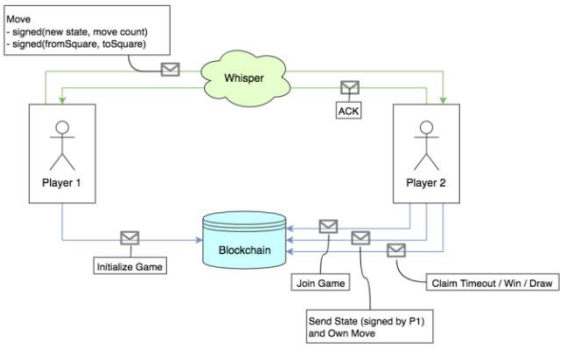

Fig. 3. Blockchain and Whisper

Whisper [12] is a decentralized communication protocol to communicate with each other. Darkness is an important feature of this protocol. Therefore, no one can trace the message senders and receivers. The protocol sends the message to everyone in the network, and it behaves like a gossip protocol to achieve that darkness. It sends the message to all connected nodes; the origin node sends messages to the connected node. Likewise, the message is communicating to everyone. However, only the relevant node has the private key to decrepit the message because it uses asymmetric key cryptography. Therefore, the cost of the protocol is relatively high[13].

I. Decentralized storages

There is a major problem when considering the storage mechanism for a decentralized application. Most people lack knowledge about blockchain technology and think it is possible to use a blockchain for a complete solution for a storage problem in a decentralized application. However, it is impossible to store a large amount of data in a blockchain. The best solution for the storage problem in decentralized applications is the InterPlanetary File System (IPFS), and blockchain can be used to reference the IPFS platform.

IPFS gives a unique cryptographic fingerprint to every content which is published in the file system. It removes the duplication of the file across the system, and it delivers the files from the nearest node to which the file system hosts a file. Each network node stores only the content that it is interested in, and some indexing information helps figure out who is storing and what is also stored in the nodes. When looking up files, the user asks the network to find nodes storing the content behind a unique hash.

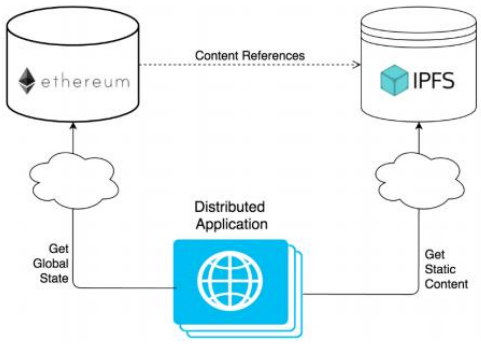

Fig. 4. Decentralized storage - IPFS

IPFS is called a permeate web because the file system behaves quite like the Git version controlling the file system. Every version of the file will be stored. It uses local storage to store files and distribute that file within other nodes [14].

When someone uploads something, the file is chunked by IPFS and stored in his cache folder (ipfs). Suppose a user tries to see the file on another peer of the network (say the main gateway, ipfs.io) that peer requests the file to you and caches it too. If he switches off his daemon, he can still see the file on the gateway, probably because the gateway or some other peer on the web still has it cached. When a peer wants to download a file, but it is out of memory (it can be no longer cached), the oldest used files get forgotten to free the space. That is a simple explanation for IPFS, but it is more complicated than this.

InterPlanetary Naming System (IPNS) behaves like Domain Name Service (DNS) in the decentralized IPFS ecosystem. In IPFS, an uploaded content is identified using its fingerprint hash. However, it is difficult to remember that hash. Therefore, this IPNS is providing a service to have a unique human-readable identity to each hash.

There are several other ongoing research projects to solve the same storage problem. FileCoin, Storj, MaidSafe, SWARM are few examples.

FileCoin [15] is a cryptocurrency like bitcoin, but miners must share their computer storage with the network users. Bitcoin uses proof of work as the consensus algorithm, introducing a novel consensus way of mining FileCoin called Proofs of-storage. It is based on two consensus algorithms Proof-of-Replication and Proof-of-Spacetime. Proof-of-Replication: allows storage providers to prove that data has been replicated to its own uniquely dedicated physical

storage, and Proof-of-Spacetime: allows storage providers to prove they have stored some data throughout a specified amount of time. Then miners can earn coins by providing their storage. Therefore, this network has massive, decentralized storage. The important point is that this FileCoin network works as an incentive layer on top of IPFS. Therefore, IPFS seems like a good storage solution for a decentralized ecosystem.

STORJ [16] is another ongoing decentralized storage research project, and it is about a decentralized cloud storage network framework. In this framework, when a client saves a file, it will be encrypted first on the clientside. Then it is chunked into small pieces, and those pieces are sent to storage nodes, and storage nodes are storing those chunked data pieces. When chunking data, a central server keeps tracking which parts are relevant to the chucked file. When constructing the file again that metadata is used. The chucked pieces are replicated through the storage nodes based on a threshold value. Storage nodes are selected using several factors like ping time, latency, throughput, bandwidth caps, client disk space, geographic location, uptime, history of responding accurately to audits, etc. The speciality of this system is that this is an S3 capable platform.

SWARM [17] is also another solution for the distributed storage. It provides a content distribution service, a native base layer of the Ethereum Web3 stack. SWARM has been decentralized to serve as a redundant store of Ethereum's public records to store and distribute Smart Contracts. This platform is also a peer-to-peer storage platform maintained by its peers who contribute their storage and bandwidth resources. Being a peer-to-peer system, this has no single point of failure, and it is resistant to failures and Distributed Denial-of-Service (DDoS) attacks.

J. Other decentralized social networks

There are few ongoing projects on decentralized social networks. Furthermore, they are focused on different kinds of areas in the decentralized social network domain. Seemit, Sola, Memo, VeganNation, and Indorse are few examples that focus on different aspects. However, none of them could give a better solution to overcome social networks such as Facebook and Linked-In. There are many reasons behind that. The main reasons are the lack of awareness of ordinary people about the current social media ecosystem's problems and their need to do everything quickly. As this decentralized world is still in its early stages, there are many usability issues. As a result, the decentralized world remains popular among those who are familiar with the underlying technology [18].

III. TECHNOLOGY ADAPTED

A. *Web 3.0 stack*

There are few new different categories of technologies that are included in the web 3.0 stack. Web browser, Web application, Web Protocols, Network architecture, Data storage, Application deployment are a few. It is needed to find a solution for each category replacement in the web 3.0 stack to replace the web 2.0 stack. However, Web 3.0 is still in the research stage and not mature as the Web 2.0 technology stack. Therefore, it is hard to find a complete solution only using the web 3.0 stack to achieve the requirements. Therefore, it will have to use a hybrid but more into Web 3.0 approach when developing a solution.

B. *MetaMask*

MetaMask is an application that helps the decentralized application to perform its transaction in the Ethereum network. It can be added to the web browser as plugging, and then it will be automatically triggered whenever the user is going to do a transaction in the blockchain network. It is a bridge between the decentralized web application and the blockchain network. Using MetaMask, connecting to the Main Ethereum networks or any other custom Ethereum network is possible. It provides an Ethereum wallet management facility and account management facility as well. So, it is straightforward to keep several accounts in different or the same blockchain networks. Also, it provides an account recovery facility too.

C. *Oracle*

One significant restriction of smart contracts is that they cannot directly access other data sources such as APIs, databases, and IoT sensor data. Because the data access for outside can be changed with time and the Smart Contract execution is fully deterministic. Since the external sources on the internet are non-deterministic, it is impossible to get the same state after replaying the changes to the blockchain over time. Nodes of the network come to a consensus with this determinism of the network. That is the place that the oracles come to play. They give the flexibility to the Smart Contracts to interact with off-chain data sources. Oracles themselves are not data sources. They work as an extra layer between smart contracts and off-chain data sources. Mainly, there are several types of oracles such as Software Oracles, Hardware Oracles, Human Oracles, Contract-specific Oracles, etc. BandChain, Oracalize, Chainlink, Tellor, and Provable are few oracle services that enable external data access to the Ethereum blockchain [19].

D. *Infura*

Infura is a platform that helps to develop decentralized applications easily. It provides an infrastructure to interact with Ethereum and IPFS gateways. It provides secure, reliable, and scalable access to those gateways.

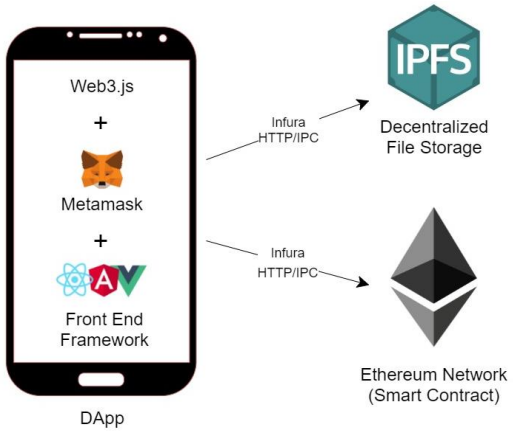

Fig. 5. Connection between IPFS, Smart Contract and Infura

As the figure shows, using Infura, it is possible to interact with remote IPFS and Ethereum networks directly. Otherwise, it is needed to host a local Ethereum or IPFS client.

IV. Proposed Architecture

A. *Decentralized social network high-level architecture*

As shown in Fig.6, DAPP will be a client-side application that users can access through their web browsers. It will base on JavaScript, HTML, and CSS. On top of these traditional web technologies, there will be a Web3.js layer as the bridge between the client application and the back end.

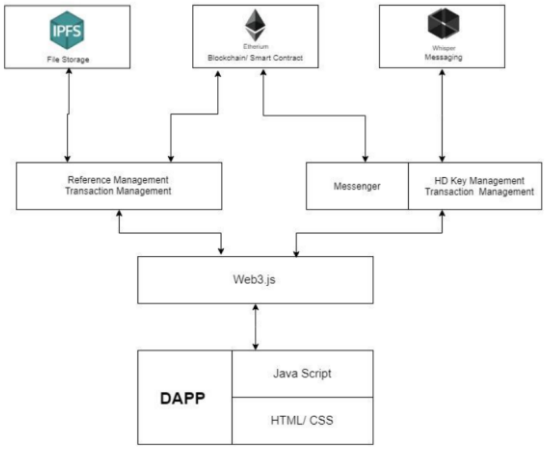

Fig. 6. Decentralized Social Network High-Level Architecture

In this case, the backend will be handled using the Ethereum blockchain network using the smart contracts deployed inside the Ethereum blockchain. Interplanetary file system (IPFS) will be working as the data storage layer in the system. Whisper will work as the messaging platform of the system.

B. *Front-end architecture*

Fig. 7. shows the client-side architecture of the system. Here the application is designed using component-based architecture. All external calls such as API calls, JSON RPC are handle by the services. It is the interface of the client-side applicant to external entities.

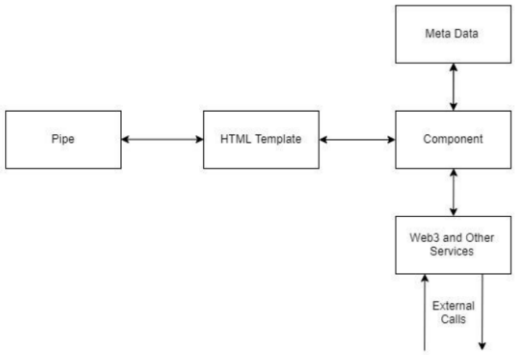

Fig. 7. Frontend architecture

C. *Backend external API support*

Nowadays, almost every web service can communicate with external web services. However, in this decentralized scenario, Smart contracts are living in the blockchain. Therefore, they can interact with data living in the same blockchain network. However, the limitation is that they cannot interact with the outside blockchain, such as web API. Nevertheless, for modern applications, it is a must to interact with external APIs. Here is the design for support for the external APIs.

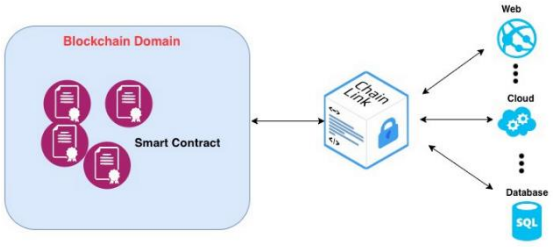

Fig. 8. Connection between Smart Contract and external datastores through Oracle

Here, ChainLink will work as a middle platform. Smart-Contract can interact with ChainLink. ChainLink has Smart Contract to support that purpose. Then ChainLink will call the external web APIs or other external off-chain services. Then after the result comes to the chainlink, its callback to the called Smart Contract.

D. *Interact with IPFS*

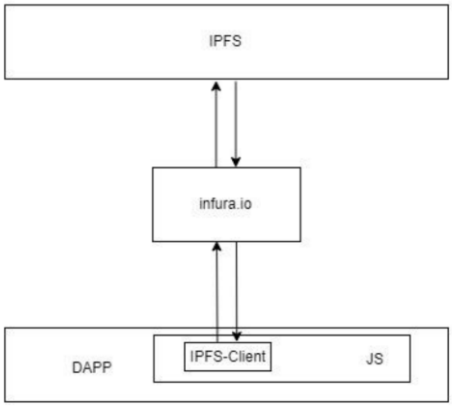

Fig. 9. DApp Interaction with IPFS

This diagram shows how a DAPP interacts with the InterPlanetary File System. IPFS-Client library is the client-side interface of the interaction, and It creates a connection with the Infura platform and provides the facility to communicate with the IPFS network. With this design, there is no need to run a local IPFS-Client.

V. Implementation

Truffle was used as the Smart Contract development framework in application development, and the programming language was Solidity [20]. It is one of the mature solutions in developing Ethereum based decentralized applications.

A. *Solidity*

Solidity is the programming language used to program the smart contract for the system. It is a contract-oriented

programming language, and it is Turing complete programming language. Solidity codes are compiled into bytecode using Remix [21] compiler.

B. *Truffle*

Truffle makes the Smart Contract development process easy. It handles the Smart Contract compilation, bytecode management, linking, and deploying the smart contract in the given Ethereum network. It gives a command-line interface, and it is instrumental in development.

C. *Mocha*

Mocha was used together with Truffle as the testing framework. It supports Smart Contract testing.

D. *Web3 JS*

Web3 JS is a library that works as a bridge between client-side applications and the Ethereum blockchain. It has several implementations in several languages. Web3.js is a major implementation of Web3, which is used to work with web applications

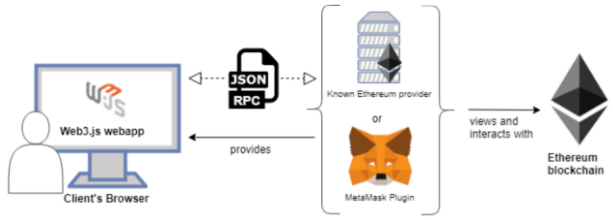

Fig. 10. Interaction between Web3js and Ethereum

Web3.js is the bridge between Ethereum blockchain and the web browsers (client-side). It is a JavaScript API that is compatible with the Ethereum blockchain. It uses generic JSON RPC to work with the client-side. To communicate with the blockchain, it uses an application binary interface (ABI) provided by Smart contracts.

E. *IPFS-API*

IPFS API is a JavaScript library that was used to interact with the InterPlanetary file system. This library can be configured with Infura.io. Then it is possible to communicate remote IPFS gateways easily. Whole data sending and receiving processes are passing through this IPFS-API library

F. *Crypto JS*

CryptoJS is a collection of secure and standard cryptographic algorithms implemented using JavaScript with best-practice patterns and practices. They are fast, and they have a consistent and simple interface.

G. *Implementation of the client-side*

As a unit testing formwork, Mocha was used. As libraries, Web3.js, Angular was used in developing the client-side of the application (Front end). The system is designed to use a component-based architecture. Web3 JS, ipfs-client are the most impairment libraries, and they were connected to achieve the desired decentralization. User can store data and retrieve using ipfs-client. Web3 is used to create a new user account and store user IPFS hash to reference user detail in the IPFS.

H. *Custom ethereum network*

Ganache Blockchain is a perfect development solution, debugging, and test because it provides many features [22]. However, for the actual implementation, there are two solutions. The first is to deploy the decentralized application in Ethereum's main network or public Ethereum test networks like Ropsten or Rinkeby. The second option is to develop a private Ethereum network available only for social network users [23]. In this research, the second option has been chosen. Because this works as a separate platform and, it cannot be dependent on another Ethereum network. Suppose it depends on another Ethereum network. If the app uses an Ethereum public network, there are several problems: the gas limit, coin base, difficulty, etc. If it happens, it is hard to achieve the intended purpose of the application. In this case, it is needed to develop a custom Ethereum network, and it can be easily done by the go-Ethereum client software (Geth). The first thing that needs to do is to create a genesis block of the network. It is the first block of the network. It can be defended as a JSON file.

VI. Results

For testing purposes, this research used an intel core i7 laptop with 16GB memory, and the operating system was Windows 10 Student Edition. Both blockchain and the Angular front-end application were deployed in the same machine.

Using the proposed system architecture and technologies, it could develop a prototype of this decentralized system that has features to create and update user profiles, search user profiles, add friends, chat with friends, post text and photos on the user's wall, and comments on the post. The system was tested with only ten concurrent users, and the response time for creating/updating users, sending friends requests, and adding friends were less than 8 seconds. For the post-sharing functionality, the response time depends on the size of the content. Generally, IPFS takes 16s to upload 1GB of data [24], and then after uploading the multimedia file, it takes up to 8 seconds to process inside the developed prototype.

VII. Conclusion

With the advances of technology, Web 3.0 is expected to be the future of the web. However, people doubt whether the web 2.0 centralized web architecture can be replaced by decentralized web 3.0 architecture. This research focused on developing a decentralized social network architecture that can provide more privacy, data ownership, and community-driven facilities mainly based on the Ethereum platform. However, the platform can be changed to achieve efficiency in the future as there are commonly known limitations in Ethereum blockchain and other blockchains. Giant organizations such as Facebook, Google, and Microsoft are also developing and exploring these technologies, which look promising about decentralized computing.

VIII. Future work

The research is proposed with a whole system architecture to develop the decentralized applications. However, the implementation of such an application is massive work. Therefore, the implementation of the research is just a proof of concept. In the future, the application will be fully implemented with the proposed concept.

Furthermore, it will be available for the public to use. When considering the security of the data, the application must have more consideration. The system design can currently set data visibility to only me or the public, handled by basic encryption and decryption mechanisms. However, more focus should be on authentication and authorization with different access levels for different data types.

As the initial step, the application is developed based on a public IPFS network. Nevertheless, in the future, with the improvement of the system's user base, it can be developed into a custom IPFS network. Then it can be dedicated to this decentralized social network. By developing such a network, the efficiency of the system can be improved.

REFERENCES

[1] "The History and Evolution of Social Media," Webdesigner Depot, Oct. 07, 2009. https://www.webdesignerdepot.com/2009/10/the-history-and-evolution-of-social-media/ (accessed Feb. 09, 2019).

[2] "Number of social media users worldwide 2010-2021," Statista. https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/ (accessed Feb. 09, 2019).

[3] "What Are Smart Contracts? A Beginner's Guide to Smart Contracts," Blockgeeks. https://blockgeeks.com/guides/smart-contracts/ (accessed Dec. 03, 2018).

[4] Hartikka, "A blockchain in 200 lines of code," Lauri Hartikka, Mar. 04, 2017. https://medium.com/@lhartikk/a-blockchain-in-200-lines-of-code-963cc1cc0e54 (accessed Dec. 02, 2018).

[5] M. Bach, B. Mihaljevic, and M. Zagar, "Comparative analysis of blockchain consensus algorithms," in 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, May 2018, pp. 1545–1550. doi: 10.23919/MIPRO.2018.8400278.

[6] Ye, G. Li, H. Cai, Y. Gu, and A. Fukuda, "Analysis of Security in Blockchain: Case Study in 51%-Attack Detecting," in 2018 5th International Conference on Dependable Systems and Their Applications (DSA), Dalian, China, Sep. 2018, pp. 15–24. doi: 10.1109/DSA.2018.00015.

[7] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," p. 9.

[8] "Proof of Work vs Proof of Stake: Basic Mining Guide," *Blockgeeks*. https://blockgeeks.com/guides/proof-of-work-vs-proof-of-stake/ (accessed Dec. 02, 2018).

[9] "What is Ethereum? — Ethereum Homestead 0.1 documentation." http://ethdocs.org/en/latest/introduction/what-is-ethereum.html (accessed Dec. 02, 2018).

[10] "Ethereum Project." https://www.ethereum.org/ (accessed Dec. 02, 2018).

[11] "r/ethereum - Can someone possibly explain the concept of GasPrice?" *reddit*. https://www.reddit.com/r/ethereum/comments/3fnpr1/can_someone_possibly_explain_the_concept_of/ (accessed Dec. 02, 2018).

[12] "whisper-overview," Ethereum Wiki. https://eth.wiki/concepts/whisper/whisper-overview (accessed Jun. 12, 2021).

[13] L. Zhang, Z. Zhang, Z. Jin, Y. Su, and Z. Wang, "An approach of covert communication based on the Ethereum whisper protocol in blockchain," Int. J. Intell. Syst., vol. 36, no. 2, pp. 962–996, Feb. 2021, doi: 10.1002/int.22327.

[14] J. Benet, "IPFS - Content Addressed, Versioned, P2P File System," ArXiv14073561 Cs, Jul. 2014, Accessed: Jun. 12, 2021. [Online]. Available: http://arxiv.org/abs/1407.3561

[15] Y. Psaras and D. Dias, "The InterPlanetary File System and the Filecoin Network," in 2020 50th Annual IEEE-IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S), Valencia, Spain, Jun. 2020, pp. 80–80. doi: 10.1109/DSN-S50200.2020.00043.

[16] "storj.pdf." Accessed: Jul. 12, 2021. [Online]. Available: https://www.storj.io/storj.pdf

[17] "swarm-whitepaper-eng.pdf." Accessed: Dec. 02, 2018. [Online]. Available: https://docs.swarm.fund/swarm-whitepaper-eng.pdf

[18] Building Blockchain Projects. Accessed: Jul. 14, 2021. [Online]. Available: https://learning.oreilly.com/library/view/building-blockchain-projects/9781787122147/

[19] A. Beniiche, "A Study of Blockchain Oracles," ArXiv200407140 Cs, Jul. 2020, Accessed: Jul. 13, 2021. [Online]. Available: http://arxiv.org/abs/2004.07140

[20] D. Mohanty, Ethereum for Architects and Developers: With Case Studies and Code Samples in Solidity. Berkeley, CA: Apress, 2018. doi: 10.1007/978-1-4842-4075-5.

[21] "Remix - Ethereum IDE." https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null (accessed Jul. 12, 2021).

[22] "Ganache | Overview | Documentation," Truffle Suite. https://trufflesuite.com/docs/ganache/overview (accessed Jul. 14, 2021).

[23] "Enterprise on Ethereum mainnet," ethereum.org. https://ethereum.org (accessed Jul. 14, 2021).

[24] T. Li et al., "FAPS: A fair, autonomous and privacy-preserving scheme for big data exchange based on oblivious transfer, Ether cheque and smart contracts," Inf. Sci., vol. 544, pp. 469–484, Jan. 2021, doi: 10.1016/j.ins.2020.08.116.