Priority-Based Disk Space Sharing Using Blockchain

Husain Rehmanji
Dept. of Computer Engineering
D.J. Sanghvi College of Engineering
Mumbai, India
husainrehmanji@gmail.com

Arya Shirgaonkar
Dept. of Computer Engineering
D.J. Sanghvi College of Engineering
Mumbai, India
aryashirgaonkar42@gmail.com

Jay Mistry
Dept. of Computer Engineering
D.J. Sanghvi College of Engineering
Mumbai, India
jaymistry700@gmail.com

Aesha Shah
Dept. of Computer Engineering
D.J. Sanghvi College of Engineering
Mumbai, India
aeshashah2634@gmail.com

Kiran Bhowmick
Dept. of Computer Engineering
D.J. Sanghvi College of Engineering
Mumbai, India
kiran.bhowmick@djsce.ac.in

Nilesh Patil
Dept. of Computer Engineering
D.J. Sanghvi College of Engineering
Mumbai, India
nilesh.p@djsce.ac.in

***Abstract*—Decentralized storage systems make use of blockchain technology in offering security, transparency, and efficiency in sharing disk space. This study introduces a concept of prioritization of users based on considerations such as past efficiency in usage, timely release of space and rule adherence performed by the user. Utilization of Blockchain based smart contracts automates resource assignment and also provides transparency during storage allocation. Smart contracts monitor storage requests and assignments and track the user reputation progressively resulting in fair access. The outcome of integrating with blockchain is that security is enhanced, authenticity is assured and a scalable solution is obtained. The objective of this study is to emphasize the potential benefit when utilizing a blockchain based solution for a space sharing application.**

***Index Terms*—Blockchain, Decentralized Disk Space Sharing, Reputation based Resource Allocation, IPFS (InterPlanetary File System), Local Area Network (LAN) Storage Management, Peer-to-Peer Storage Systems**

I. INTRODUCTION

Effective allocation and distribution of disk space in contemporary computer systems is a matter of significance in terms of resource management, especially in the configuration of the local area network (LAN) of the average small office, school, and research institution. Despite the prevalence of storage space with high capacities, organizational resource imbalance owing to storage inadequacy still exists; systems might waste space while others would be out of space to save their files. The presence of such disparity causes inefficient reallocation of resources in the network, an aspect that is not productive and raises operational costs.

The traditional solution has the tendency to bring in extra hardware or utilize the facilities of cloud storage. Both, however, have serious disadvantages to smaller networks. The purchase of hardware is extremely expensive compared to performance and leads to idle assets that are not utilized optimally. Alternatively, cloud storage facilities are accompanied by the constraints of subscription charges and bandwidth constraints that slow down data access. In addition, most cloud providers use centralized infrastructures, leading to reliance on third-party vendors and eroding data sovereignty. Such constraints are generally realized in situations with uneven storage needs, such as in educational institutions where project needs vary with each learning term or in organizations with sporadic data processing operations. In such situations, the ability to dynamically provision and manage the available disk space resources within a local area network is the equivalent of an incredibly sought-after solution that enhances operational efficiency at the same time as lowering costs.

The system allows users to enroll disk space for providing or acquiring additional capacity for storage with ease, with transaction being automatically triggered and documented using the transparent and secure ledger of blockchain technology. Using a peer-to-peer, decentralized platform for disk space sharing specifically designed for use in LAN environments, this work is capable of effectively addressing the aforementioned issues. Smart contracts and blockchain technology facilitate trustless sharing of disk space, without the requirement of a centralized authority or secondary security infrastructure. Data confidentiality is supported using encryption mechanisms built-in that take advantage of instant access to LAN environments of files on peer machines. Data is encrypted when being transferred, safely stored on the provider machine for the duration of time previously agreed upon, and accessed and utilized by the owner on an as-needed basis. The approach bypasses the latency and bandwidth constraints of cloud storage and ensures data security.

One of the unique features of the method is the use of a reputation-based priority system. The system significantly improves the fairness of the system with respect to resource allocation and consistency. Each user of the method has a reputation score that is computed based on their own behavior within the network in the form of timely storage capacity provisioning, allocation of resources, and compliance with past

commitments. Higher reputation users scores have priority in resource allocation when making orders for disk space, thus creating a self-sustaining system that by design promotes honest players and deters opportunistic players.
The reputation system has multiple functions. It is a system of trust within a community where there is no trust by default; it encourages responsible sharing of resources, and it is the basis for intelligent allocation decisions. As users build more stable reputations through repeated good conduct, the integrity of the system as a whole is enhanced, thus establishing a stable sharing economy within the local area network environment. By combining blockchain's security with a reputation-based priority system, our project is a good solution for disk space management within small networks, decreasing dependency regarding external suppliers, reducing expenses, improving access velocities, and enabling fair allocation of resources.

II. LITERATURE REVIEW

Blockchain has been widely used to form secure networks and conduct transactions while maintaining security and integrity. It has also been useful in eliminating single point of failure. It has also been used in data sharing mechanisms using InterPlanetary File System (IPFS) [1]. While some systems use additional encryption to secure data, some propose to distribute the file among various nodes. Some also use the Lightning Network [2] to process transactions in an efficient way.

Naz. et al. [3] work in their paper to create a data sharing platform by leveraging IPFS. The party selling the data uploads the file on the IPFS network, which returns hashes for the data. SSS algorithm is used to divide the IPFS hash into k number of shares, and n number of random keys are then assigned for encryption of these hashes. The encrypted hashes are stored in the blockchain. The customer can see the reviews for data that they want to buy, and request for a file. Once they pay the deposit, they can access the complete data using smart contracts. They can also receive 10% of deposit back if they submit valid reviews for the data.

Data sharing in the medical domain can also integrate blockchain for maintaining privacy and security. Hasan et al. [4] worked on Software-Defined Wireless Body Area Networks (SDWBANs), and used smart contracts to create an access control policy which makes sure that data owners have full control over their health data.

Khalid et al. [5] explore the centralized and decentralized manner of data storage, and specifically, the way in which blockchain has been used for the purpose. The problem, they saw, was that the blockchain storage networks cannot practically store entire files among the users. However, smart contracts-based solutions offered to overcome this issue, but they only seemed to work for IPFS, which was implemented by Storj [6]. All consensus mechanisms were created to achieve a specific goal, for example, speed, which could be altered depending on the use-case. The drawbacks that they found were access issues, data analysis, and scalability issues.

In their research paper [7], Pradhan et al. propose a decentralized storage rental system that uses blockchain's transparency and distributed consensus to achieve data integrity and resilience. Its decentralized architecture makes sure that there are no single points of failure, ensuring that the data storage is robust. They use smart contracts to automate transactions to reduce reliance on intermediaries to increase efficiency.

Sindhu et al. [8] proposed a system to create a decentralized peer-to-peer system for users to rent space on the disk of peer computers. They created a user interface where the data owners can register themselves and perform the payment after choosing the size of disk space wanted. They use AES encryption algorithm with fourteen rounds and a 256-bit key to ensure the safety and integrity of data being shared by IPFS. They use consensus mechanisms, like Proof-of-Work (PoW), or Proof-of-Stake (PoS), to ensure the security of transactions. They employ a token-based reward system, where the users get tokens for giving up their disk space for sharing. They can then use these tokens either within the network or to exchange digital assets.

Another solution for a disk space rental system is provided by Kamble et al. in their research paper [9], where they use IPFS to distribute data to several nodes in the network. They use redundancy and fault tolerance to ensure that the shared data is always available even if any individual node fails. The metadata of the file, including the file hash, and other retails about the transaction are stored in the blockchain. Automated smart contracts are deployed, which store the agreements between the owners of data and owners of disk space. They use the Lightning Network to perform the transactions between users, which makes the process faster and reduces costs.

While Mpyana and Hoh focus on data sovereignty among decentralized storage systems in their research paper [10], they distinguish between public blockchain, private blockchain and consortium blockchain. They show the distinct advantages and drawbacks of all and compare various use cases for each type, and elaborate about the trade-offs made between transparency, scalability and complexity.

Punia et al. [11] conduct a review on various access based control systems used in the cloud environment. Various blockchain algorithms used in the decentralized cloud environment work differently with different trade-offs. Each method has different goals and differs in the functionalities that it provides. While some have no single point of failure, some have limited flexibility, others more secure and resource-intensive. Each method has advantages and disadvantages which are unique to each use-case.

Gavankar et al. [12] propose a disk rental system where users connect to the network via a digital wallet. The disk space owners need to have a stable internet as they interact with the system using a React JS application for front-end and PHP to access third-party blockchain network. The renter can search for plans which suits their storage requirements. They can check their status and see how much storage they have used or add up to their existing rental space if they need more space. They perform transactions using the lightning network and can cancel the transactions whenever they want.

III. METHODOLOGY

The integration of the blockchain and IPFS (InterPlanetary File System) ensures safe, decentralized file storage for LAN environment and creates a priority-based sharing system for available disk space. The solution described allows users to rent out unused disk space while maintaining equal sharing with a reputation-based mechanism. The local blockchain (Ganache) and smart contracts (Truffle framework) take care of managing the allocated disk space while IPFS manages the underlying file storage. The system workflow is represented in several stages, starting from smart contract creation, integration of blockchain and IPFS, a reputation system and a python-based user-interaction system. The following sections elaborate on the above step-by-step process.

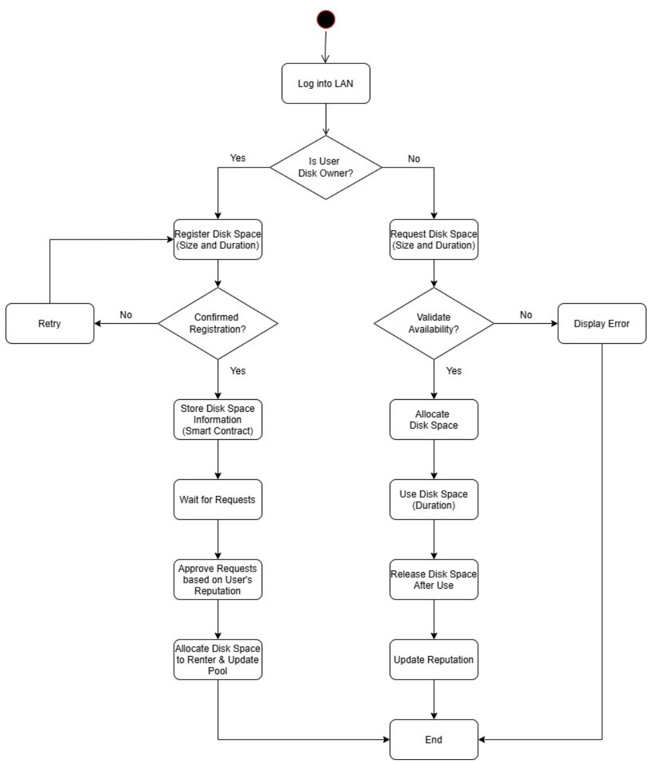

Fig. 1. User Activity Diagram

A. Smart Contract Deployment on Local Blockchain

The logic required for this reputation-based disk space allocation system is written in a solidity-based smart contract code. The smart contract is deployed on a local Ganache blockchain using Truffle framework. It keeps track of all transactions that occur within the system, including user requests, metadata of stored files, and the maintenance of the user reputations. It also defines methods for efficient disk space allocation, storage of hashes of uploaded files and penalties for late release using the logic as mentioned in the implementation of the reputation system.

The Truffle framework is used for the compilation, deployment and interaction with the smart contract. Truffle is an ideal environment for development as it simplifies this process by providing a well-structured framework for writing and testing Solidity based smart contracts. The smart contract maintains the IPFS hashes of the files that the user uploads and links that data to the respective user. It also creates a block for the same, thus making it verifiable and immutable. The contract gives priority to users with higher reputation scores, hence, when a user requests disk space, the system responds accordingly taking into account the requester's reputation score and the requested space.

The contract expects users to free up the allotted space after the rental period which is asked to the renter before space is allotted. The reputation system accordingly assigns reputation scores to the users after each such event. So if the renter frees the space within the requested time they may be awarded points or would maintain their previous reputation score, but if the space is not released within the stipulated time the renter will be penalized accordingly. Future requests from such users will be entertained taking into consideration their previous reputation scores, hence overtime the system can accurately monitor user behaviour and better decisions can be taken eventually.

B. Implementation of the Reputation System

The reputation scoring system is based solely on the idea of rewarding users who respect the stipulated storage time and penalizing users who do not. It uses the following mathematical formula to calculate the reputation score using the difference between actual and expected release times. The reputation score formula (RS) is given as:

$$\text{RS} = 100 - \beta \cdot (T_{\text{actual}} - T_{\text{expected}}) - \delta \cdot L$$

Where:

- T_{expected} is the agreed rental time.
- T_{actual} is the actual release time.
- β is the deviation penalty factor applied for early or late releases. A higher value of β means even small deviations significantly reduce reputation and vice versa.
- δ is an additional penalty per late release minute and comes into effect only after the late threshold. A higher value of δ punishes late releases more aggressively.
- L represents the extra time the storage was held beyond the expected period.

The two variables β and δ are introduced here to control the strictness of the reputation system and play a critical role in the Reputation Score formula. An optimal range for a combination of these values can be determined by exploring various values for these variables via trial and error method. Users can set these values according to the severity or strictness of the reward/penalty system. A range of values is listed below denoting the preferred choice of values for these constants to either evaluate the Reputation Score leniently or strictly, as preferred by the users.

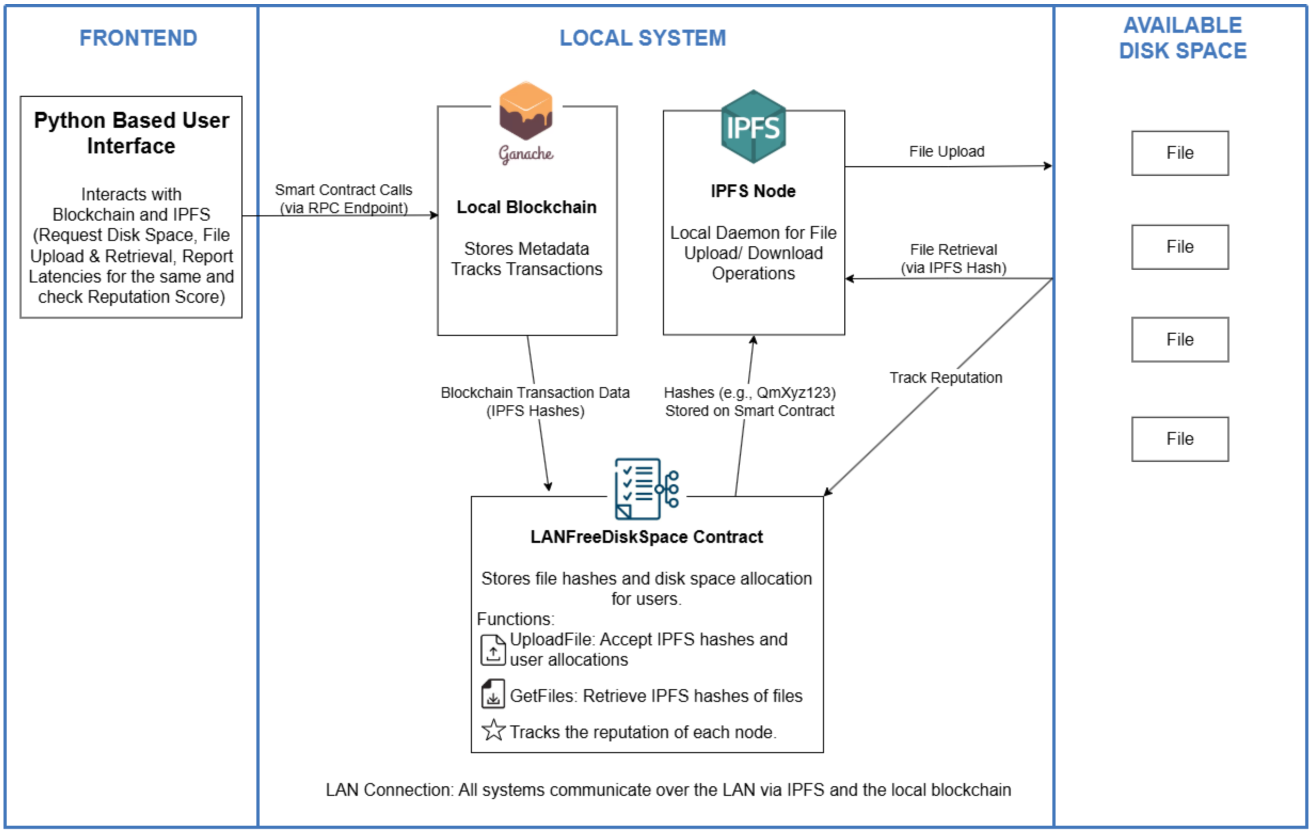

Fig. 2. System Architecture

TABLE I
OPTIMAL VALUES FOR β AND δ BASED ON FLEXIBILITY

Scenario	β values	δ values	Justification
Strict Timing	3–5	4–5	Minimal disruption
Moderate Strictness	2–2.5	2.5–3	Balanced Scores
Flexible System	1–2	1–2	Leeway for minor delays

Late-release users have their reputation score decrease step by step, while users releasing storage on time or even ahead of schedule have a positive reputation score. These users with a good record of releasing in time become preferable to allocate future storage requests. The reputation score is stored on the blockchain ledger so that the reputation score of any user cannot be altered.

C. Truffle and Ganache for Local Blockchain Development

Deploying and testing a smart contract is accomplished using the Truffle framework since it includes compilation, automated testing, and deployment tools for various networks. For testing, Ganache emulates an Ethereum blockchain while providing a closed system for developers to create and run contracts, transactions.

In order to deploy the smart contract to the Ganache blockchain, a migration script needs to be created in JavaScript using the Truffle migration framework which does the actual deployment. After the smart contract has been deployed on Ganache, applications can externally access it through the GUI. Ganache is in itself equipped with an impressive GUI that can effectively track the flow of the transactions, events in the smart contract, and the amount of gas used.

D. Integrating Blockchain and IPFS

Being unsuitable for the storage of large files, the Inter-Planetary File System (IPFS) is integrated as a framework in the file to hold actual files; IPFS is native to a peer-to-peer protocol to store files across multiple nodes, ensuring redundancy and integrity of data. When a user uploads a file its contents are wrapped with a cryptographic hash, nicknamed a Content Identifier (CID), creating a unique hash. This hash is then written on a blockchain using a smart contract, which grants users access to the file later at some place in time. In order to retrieve the file, the user queries the smart contract to obtain IPFS hash which can be used to access the file from IPFS Network.

As blockchain and IPFS are coupled and used as the main technologies for this study, it ensures a completely tamper resistant and decentralized system. The smart contract is used here as a logbook or registry to keep track of each transaction and user, thereby making sure that all the activities within the system are recorded and stored in the form of immutable records. While IPFS is used as the media for sharing files, the smart contract deployed on the local blockchain network keeps track of the hashes for the same and also records the timestamps for each file upload or retrieval, which is used to finally calculate the reputation score. The upload and retrieval latencies are also recorded to give an idea of the efficiency of this system. A simple python based user interface is used to interact with the system wherein the user sees a switch case based program to upload or retrieve files.

E. Python Based Frontend for User Interaction

Users can interact with this system easily using a simple python based interface. This python built interface works with the Ganache local blockchain and the IPFS node so as to provide a seemingly uncomplicated way of navigating through the system. Users will see a switch case driven program featuring options like requesting disk space, uploading of files, downloading of files, and checking their reputation. It also shows metrics like file upload and download latencies after each respective operation is performed.

IV. RESULTS

A college laboratory setting with 10 computers was utilized to emulate a typical LAN environment to evaluate the proposed system. It demonstrated reliable performance in sharing files securely, peer-to-peer space allocation and reputation based resource management. In the testing phase, renter PCs uploaded multiple files ranging from 5 KB to 500 MB to multiple rentee PCs, with AES-256 encryption to ensure data confidentiality. Efficient handling of data was demonstrated as the observed average round-trip time for a 20 MB file transfer over a 1 Gbps LAN was 0.8 seconds. The reputation-based prioritization system encouraged timely use of resources, with users being rewarded points on compliance with agreed rental times. For instance, a 15-minute rented file returned within 7 minutes rewarded 108 points to the renter, maintaining fairness and coordination. Blockchain-based logging of transactions guaranteed secure and transparent operations independent of threats of single points of failure. The efficiency of the system's resources was well demonstrated with an initial rented disk size of 10 GB, of which 1.2 GB was dynamically assigned to the 10 PCs during the test. The reputation rating of the renters went up from an initial rating of 100 points to an average rating of 162 points after five transactions per PC, indicating a constant adherence to the rental agreements. The system could process simultaneous file transfers between multiple PCs without a loss of performance, validating its efficiency in small-to-medium LANs. The findings highlight the system's capability to offset storage disparities in educational environments, offering a cost-effective and secure solution for hardware upgrade or cloud storage. Although the test was performed in a laboratory environment, the findings demonstrate immense potential for scaling in larger institutional networks, which can be further investigated.

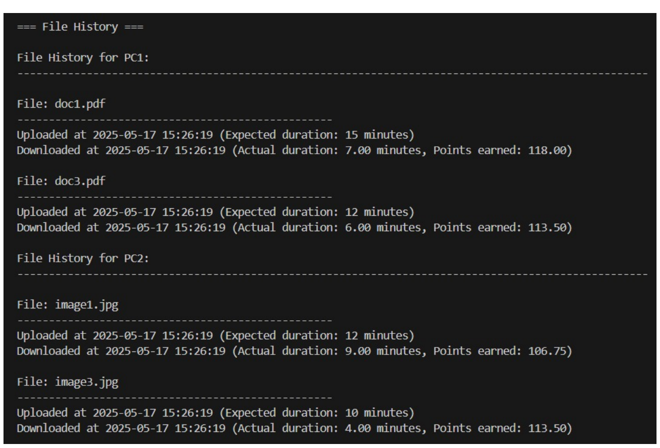

Fig. 3. Sample Output with β equals 2.25

V. CONCLUSION

The suggested system that is designed especially for Local Area Networks such as a small academic or business environment sets an important step in the field of decentralized storage management systems. It combines blockchain technology with IPFS for distributing a robust system for solving conventional problems pertaining to storage allocation. Furthermore, responsible behaviour is promoted among the participants owing to the reputation based priority system that ensures balanced and effective use of resources. Common problems in storage management systems such as imbalance of resources, over-spending on hardware, and dependence on centralized cloud services are easily addressed by the proposed system. It thereby offers a potentially scalable and immensely economically viable replacement for traditional storage management products by allowing direct peer-to-peer trustless sharing with transparency. Using smart contracts makes sure that the transactions are automated and trustless and the reputation system enforces accountability and best utilization of resources. Future directions for such a framework can incorporate additional enhancement of the reputation algorithm, or to accommodate various network topologies and enhancing efficiency by optimizing in various computational environments.

REFERENCES

[1] Benet, Juan. "Ipfs-content addressed, versioned, p2p file system." arXiv preprint arXiv:1407.3561 (2014).

[2] Poon, Joseph, and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant payments. 2016.

[3] Naz, Muqaddas, Fahad A. Al-zahrani, Rabiya Khalid, Nadeem Javaid, Ali Mustafa Qamar, Muhammad Khalil Afzal, and Muhammad Shafiq. "A secure data sharing platform using blockchain and interplanetary file system." Sustainability 11, no. 24 (2019): 7054.

[4] Hasan, Khalid, Mohammad Jabed Morshed Chowdhury, Kamanashis Biswas, Khandakar Ahmed, Md Saiful Islam, and Muhammad Usman. "A blockchain-based secure data-sharing framework for software defined wireless body area networks." Computer Networks 211 (2022): 109004.

[5] Khalid, Muhammad Irfan, Ibtisam Ehsan, Ayman Khallel Al-Ani, Jawaid Iqbal, Saddam Hussain, and Syed Sajid Ullah. "A comprehensive survey on blockchain-based decentralized storage networks." IEEE Access 11 (2023): 10995-11015.

[6] Wilkinson, Shawn, Tome Boshevski, Josh Brandoff, and Vitalik Buterin. Storj a peer-to-peer cloud storage network. 2014.

[7] Pradhan, Dr.Madhavi, Akanksha Deshpande, Yash Dhumal, Kunal Jadhav, and Gaurav Bagal. ”Decentralized Storage Rental System.” International Journal for Research in Applied Science and Engineering Technology (2023). 11. 1475-1479. 10.22214/ijraset.2023.56790.

[8] MP, Sindhu, M. Sivakrishnan, Pranave M, Milan Joseph, and Thamanna Fairooz K.A. ”Blockchain Disk-Space Rental System.” 2024.

[9] Kamble, Ashwini, Manoj Pise, Ajit Waman, Rameshwar Vishwakarma, Suraj Suryawanshi, and Shankar Amalraj. “DECENTRALIZED BLOCKCHAIN-BASED DISK RENTAL SYSTEM WITH ENHANCED SECURITY AND EFFICIENCY.” International Journal of Data Science and Artificial Intelligence, vol. 02, no.06, pp. 190-204, 2024.

[10] Merlec, Mpyana Mwamba, and Hoh Peter In. ”Blockchain-based decentralized storage systems for sustainable data self-sovereignty: A comparative study.” Sustainability 16, no. 17 (2024): 7671.

[11] Punia, Aarti, Preeti Gulia, Nasib Singh Gill, Ebuka Ibeke, Celestine Iwendi, and Piyush Kumar Shukla. ”A systematic review on blockchain-based access control systems in cloud environment.” Journal of Cloud Computing 13, no. 1 (2024): 146.

[12] Gavankar, Dhanshree, Vaishnavi Khairnar, and Prajwal Pahilwan. ”Disk Space Rental System.” JOURNAL OF TECHNICAL EDUCATION: 164.