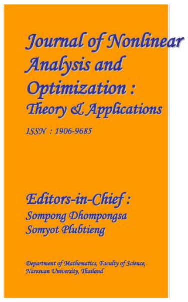
Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1, No.15 : 2024
ISSN : **1906-9685**



A

B
L
O

CKCHAIN BIGDATA SHARING SYSTEM

Dr. B. Sravan Kumar

Mr. K. Rajashekar

Dr. B. Sravan Kumar, Assistant Professor CSE(AI&ML), Vaagdevi College of Engineering (Autonomous), Bollikunta, Khila Warangal (Mandal), Warangal Urban – 506005(T.S)

Mr. K. Rajashekar, Assistant Professor CSE(AI&ML), Vaagdevi College of Engineering (Autonomous), Bollikunta, Khila Warangal (Mandal), Warangal Urban – 506005(T.S)

K. Abhiram (20641A6678), UG student CSE(AI&ML), Vaagdevi College of Engineering (Autonomous), Bollikunta, Khila Warangal (Mandal), Warangal Urban – 506005(T.S)

MD Faizan Safi (20641A66A0), UG student CSE(AI&ML), Vaagdevi College of Engineering (Autonomous), Bollikunta, Khila Warangal (Mandal), Warangal Urban – 506005(T.S)

P.Tarun Jay (20641A66A2), UG student CSE(AI&ML), Vaagdevi College of Engineering (Autonomous), Bollikunta, Khila Warangal (Mandal), Warangal Urban – 506005(T.S)

P.Mahesh (20641A66B5), UG student CSE(AI&ML), Vaagdevi College of Engineering (Autonomous), Bollikunta, Khila Warangal (Mandal), Warangal Urban – 506005(T.S)

ABSTRACT

Chain of custody is needed to document the sequence of custody of sensitive big data. In this paper, we design a blockchain big-data sharing system (BBS) based on Hyperledger Fabric. We denote the data stored outside of a ledger for sharing as "off-state" and "big data" (referring to extremely large data) is in this category. In our off-state sharing protocol, a sender registers a file with BBS for sharing. To acquire the file, an authenticated and authorized receiver has to use transactions and interacts with BBS in four phases, including the file transfer request, encrypted file transfer, key retrieval, and file decryption. The corresponding transactions are recorded in the ledger and serve as chain of custody to

document the trail of the data. Compared with related work, BBS can perform the four phases autonomously. It utilizes the permissioned blockchain, i.e. Hyperledger Fabric, for access control and can defeat dishonest receivers. We design and implement a prototype of BBS for big file sharing. Extensive experiments were performed to validate its feasibility and performance.

1. INTRODUCTION

A blockchain system can build trust in the data that it maintains without a centralized authority. Data in conventional blockchain systems is often stored in a ledger, which includes a world state and a blockchain. The world state stores the current system state such as the user cryptocurrency balance in Bitcoinand the blockchain saves all transaction history, which contains operations on the world state and/or the data used to update the world state. Smart contract controls operations on the world state. The ledger is synchronized across all blockchain nodes. In this paper, we use blockchain to share sensitive big data such as scientific and biomedical data freely and establish the chain of custody. Sharing such data with trusted parties without charge allows independent verification of published scientific results and enhances opportunities for new discoveries. With concerns of intellectual property (IP) theft and industrial espionage, we desire secure and trustworthy big data sharing systems that can record the chain of custody to document the trail of the data, e.g. who requests and owns what data. However, existing blockchain frameworks cannot be directly applied to big data sharing. In current blockchain frameworks, the data size and data type in ledgers is limited due to transaction fees, system performance and other concerns [3]– [6]. The ledger is conventionally designed to maintain the state data, such as cryptocurrency balance.

All blockchain nodes often maintain the same ledger. However, because of privacy and intellectual property (IP) concerns, owners may not want to share the big data across all nodes. Related work on big data sharing pertaining to Blockchain cannot be used for the application we target. FairSwap [7] is an off-chain based big file selling application. (i) FairSwap sells digital goods for money in Ethereum with cryptocurrency. Users need pay transaction fees to miners for smart contract execution. It is not designed for free data sharing for scientific discovery. (ii) In FairSwap, the file transfer and encryption/decryption operations are conducted off-chain by a sender and a receiver while the blockchain system conducts work such as cryptocurrency transfer and encryption key exchange. That is, FairSwap segregates file transfer from the blockchain system and is not designed for autonomous big data sharing. (iii) FairSwap works on a public blockchain. The encryption key is revealed to the public on the ledger. Once an encrypted file is leaked, for example, intercepted by malicious cyber players, the encrypted file can then be decrypted.

A token-based off-chain data sharing scheme is briefly discussed in [8] and there is no concrete protocol. Their blockchain system has no control over the actual data sharing process. Chain of custody cannot be properly established. We propose a blockchain big-data sharing system (BBS) based on the permissioned blockchain framework, the Hyperledger Fabric [9] (abbreviated as Fabric in the rest of the paper), so as to securely share sensitive big data with authenticated and authorized users and record the chain of custody within the ledger. Our major contributions can be summarized as follows. We introduce the concept "off-state", which is data, particularly big data, maintained at a separate storage space from the ledger at blockchain nodes. Off-states can be shared between parties of interest and do not need to be synchronized across all nodes.

Smart contracts can directly operate on the off-states such as sharing. To autonomously and securely establish the chain of custody of off-states, we propose a novel off-state sharing protocol utilizing the smart contract. Only an authorized receiver can obtain off-states through transactions, which document the chain of custody. Requested data is encrypted and transferred on demand. The receiver has to propose a transaction, which will be recorded into the ledger, to obtain the key and decrypt the file. The key is known only to the sender and receiver. Therefore, dishonest receivers who obtain the key cannot deny that they obtain the original data.

2. LITREATURE SURVEY

This paper proposes a blockchain-based solution for secure and efficient data sharing in IoT environments. It discusses the design and implementation of a BBS system with focus on access control mechanisms.

This work introduces a secure and efficient data sharing framework for IoT environments using blockchain and smart contracts. It addresses privacy and security concerns in data sharing and proposes mechanisms for access control and data provenance..

This work presents a blockchain-enabled secure data sharing solution for cloud computing environments. It discusses the integration of blockchain with cloud platforms and its impact on data sharing security, privacy, and auditability

The authors propose a federated learning approach with blockchain for privacy-preserving data sharing. They discuss the integration of blockchain with federated learning frameworks and its potential applications in collaborative data sharing while preserving data privacy.

1. Title : Blockchain Based Big Data Sharing and Access Control for IoT Author: S. Xu et al Abstract: This paper proposes a blockchain-based solution for secure and efficient data sharing in IoT

environments. It discusses the design and implementation of a BBS system with focus on access control mechanisms.

2. Title: Secure and Efficient Data Sharing Framework Based on Blockchain and Smart Contract for IoT Author: H. Zhu et al Abstract: This work introduces a secure and efficient data sharing framework for IoT environments using blockchain and smart contracts. It addresses privacy and security concerns in data sharing and proposes mechanisms for access control and data provenance..

3. Title: Blockchain-Enabled Secure Data Sharing in Cloud Computing Environments Author: G. Zhang et al Abstract: This work presents a blockchain-enabled secure data sharing solution for cloud computing environments. It discusses the integration of blockchain with cloud platforms and its impact on data sharing security, privacy, and auditability

4. Title: Federated Learning with Blockchain for Privacy-Preserving Data Sharing Author: R. Patel et al Abstract: The authors propose a federated learning approach with blockchain for privacy-preserving data sharing. They discuss the integration of blockchain with federated learning frameworks and its potential applications in collaborative data sharing while preserving data privacy.

3. PROBLEM STATEMENT

In traditional data sharing paradigms, the absence of a standardized and secure chain of custody for sensitive big data has been a notable challenge. Current systems often lack the necessary mechanisms to ensure transparent tracking of data access and transfers, especially when dealing with large datasets commonly referred to as "big data." The absence of a dedicated system for managing the off-state sharing of such data raises concerns about security and accountability.

Recognizing these shortcomings, this paper proposes the Blockchain Big-Data Sharing System (BBS) as an innovative solution to address the limitations of existing data sharing practices. The inadequacy of conventional systems in providing a secure and traceable chain of custody for big data is a primary motivation for the development of BBS. In contrast to traditional methods, which may lack automation and rigorous access control, BBS leverages the capabilities of Hyperledger Fabric, a permissioned blockchain framework, to introduce a systematic and secure approach to big data sharing. The shortcomings of existing systems become particularly apparent when faced with the need for a comprehensive and autonomous four-phase process, including file transfer request, encrypted file transfer, key retrieval, and file decryption, all of which are seamlessly recorded in the ledger to establish an indisputable chain of custody.

3.1 LIMITATIONS

Scalability Concerns: One limitation of the Blockchain Big-Data Sharing System (BBS) lies in its potential scalability challenges. As the size of shared data and the user base increases, the performance of the system may be impacted, potentially leading to slower transaction processing times and increased resource requirements.

Resource Intensiveness: The resource requirements for maintaining a permissioned blockchain, such as Hyperledger Fabric, can be substantial. This could pose a limitation for organizations with limited computational resources, potentially hindering the widespread adoption of BBS, especially in resource-constrained environments.

Learning Curve and Implementation Complexity: Implementing and managing a Hyperledger Fabric-based system like BBS may involve a steep learning curve for administrators and users unfamiliar with blockchain technology. The complexity of configuring and maintaining the system could be a limiting factor, potentially slowing down the adoption process.

Dependency on Hyperledger Fabric: BBS's reliance on Hyperledger Fabric as the underlying blockchain framework introduces a dependency on the development and maintenance of the Hyperledger ecosystem. Changes or vulnerabilities in Hyperledger Fabric could directly impact the functionality and security of BBS, making it susceptible to external factors beyond its immediate control.

Limited Interoperability: Interoperability with other existing systems and non-blockchain databases might be a challenge. BBS may face limitations in seamlessly integrating with diverse data storage and management solutions commonly used in various organizations, potentially restricting its applicability in heterogeneous computing environments.

4. PROPOSED SYSTEM

The proposed Blockchain Big-Data Sharing System (BBS) introduces a comprehensive solution to overcome the limitations of existing data sharing paradigms. Building upon the foundations of Hyperledger Fabric, the proposed system is designed to address scalability concerns by implementing optimization strategies that enhance transaction processing speed and minimize resource overhead. Through careful architectural considerations, the BBS aims to strike a balance between the robustness of a permissioned blockchain and the need for efficient scalability, ensuring that the system remains performant even as the volume of shared data and user interactions grows.

To mitigate the resource intensiveness associated with maintaining a blockchain infrastructure, the proposed system incorporates mechanisms for resource optimization, exploring avenues such as data pruning and storage efficiency. This approach seeks to make BBS more accessible to organizations with varying computational resources, promoting wider adoption across diverse environments.

Recognizing the potential learning curve associated with blockchain technologies, the proposed system places emphasis on user-friendly interfaces and comprehensive documentation. This approach aims to reduce the complexity of system implementation and administration, making BBS more approachable for users unfamiliar with blockchain concepts.

4.1 ADVANTAGES

Secure Chain of Custody: The Blockchain Big-Data Sharing System (BBS) offers a secure and transparent chain of custody for sensitive big data. Through the use of Hyperledger Fabric, every interaction and transaction related to data sharing is recorded in the immutable blockchain ledger, ensuring a tamper-resistant and auditable trail of the data's journey. This feature enhances data integrity and accountability.

Autonomous Four-Phase Process: BBS distinguishes itself by autonomously executing the four phases of the data sharing process—file transfer request, encrypted file transfer, key retrieval, and file decryption. This autonomy streamlines the sharing process, reducing the need for manual interventions and enhancing overall operational efficiency for both senders and receivers.

Permissioned Blockchain for Access Control: Leveraging Hyperledger Fabric as a permissioned blockchain ensures robust access control in BBS. Only authenticated and authorized users have the privilege to engage in the data sharing process. This feature enhances the system's security, preventing unauthorized access and mitigating the risk of data breaches.

Defeat of Dishonest Receivers:BBS incorporates mechanisms to defeat dishonest receivers, adding an extra layer of security to the data sharing process. Through the permissioned blockchain's access controls and cryptographic protocols, the system is designed to identify and prevent malicious actions by receivers who may attempt unauthorized or fraudulent activities.

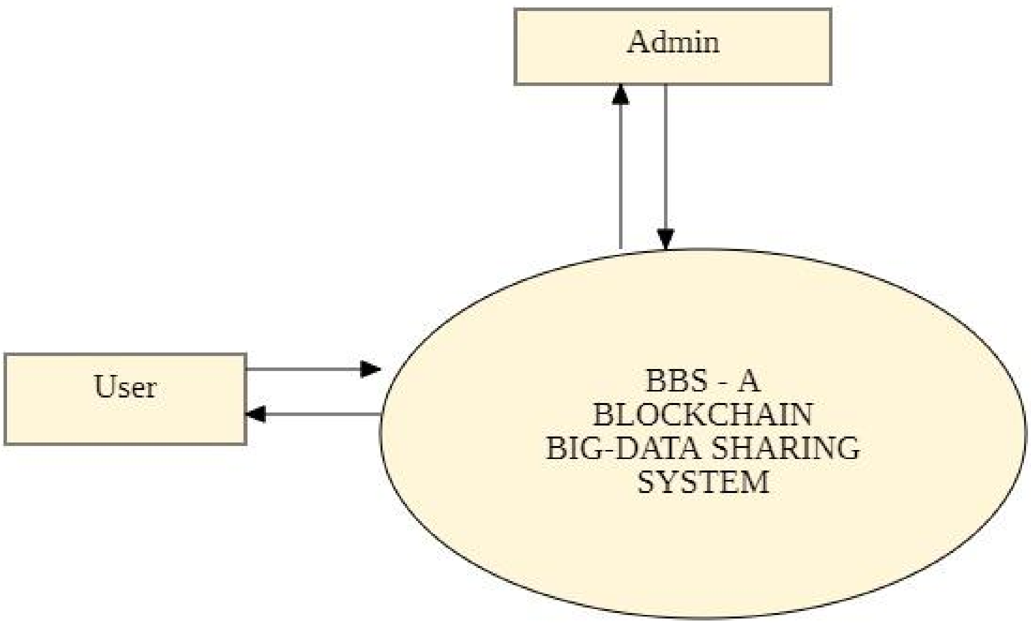
Feasibility and Performance Validation: The advantages of BBS are substantiated through the design and implementation of a prototype, backed by extensive experiments that validate its feasibility and performance. The system has been tested under various scenarios, demonstrating its capability to handle large file sizes securely and efficiently. This empirical validation enhances the reliability and credibility of BBS as a practical solution for big data sharing.

5. SYSTEM ARCHITECTURE

DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail



5.1 UML DIAGRAMS:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects

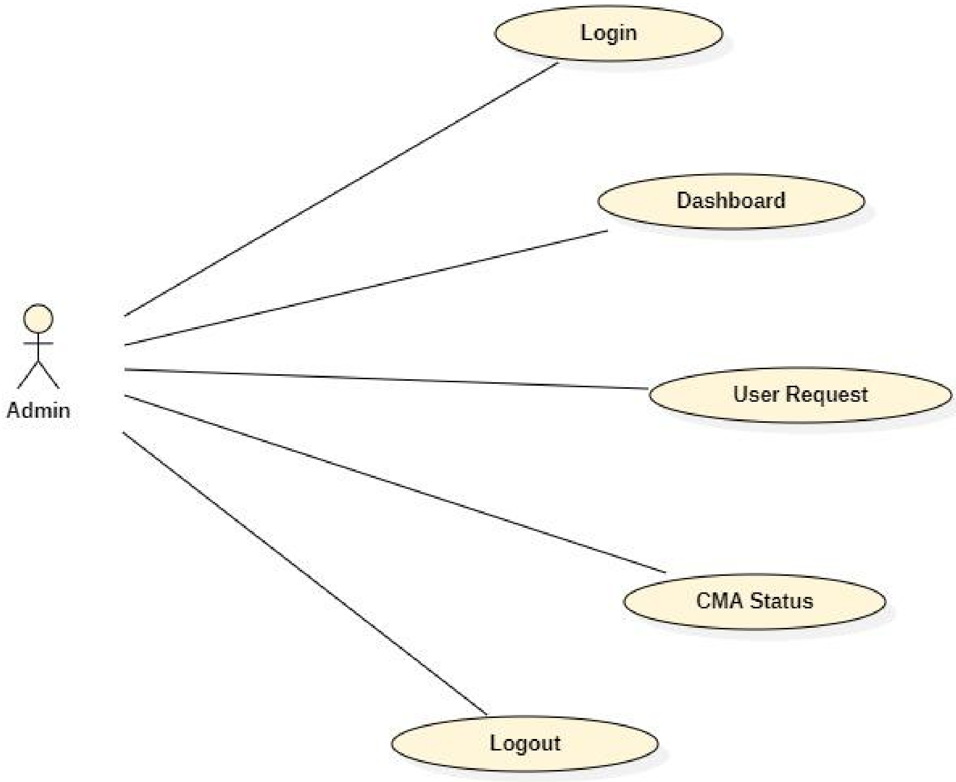
GOALS:

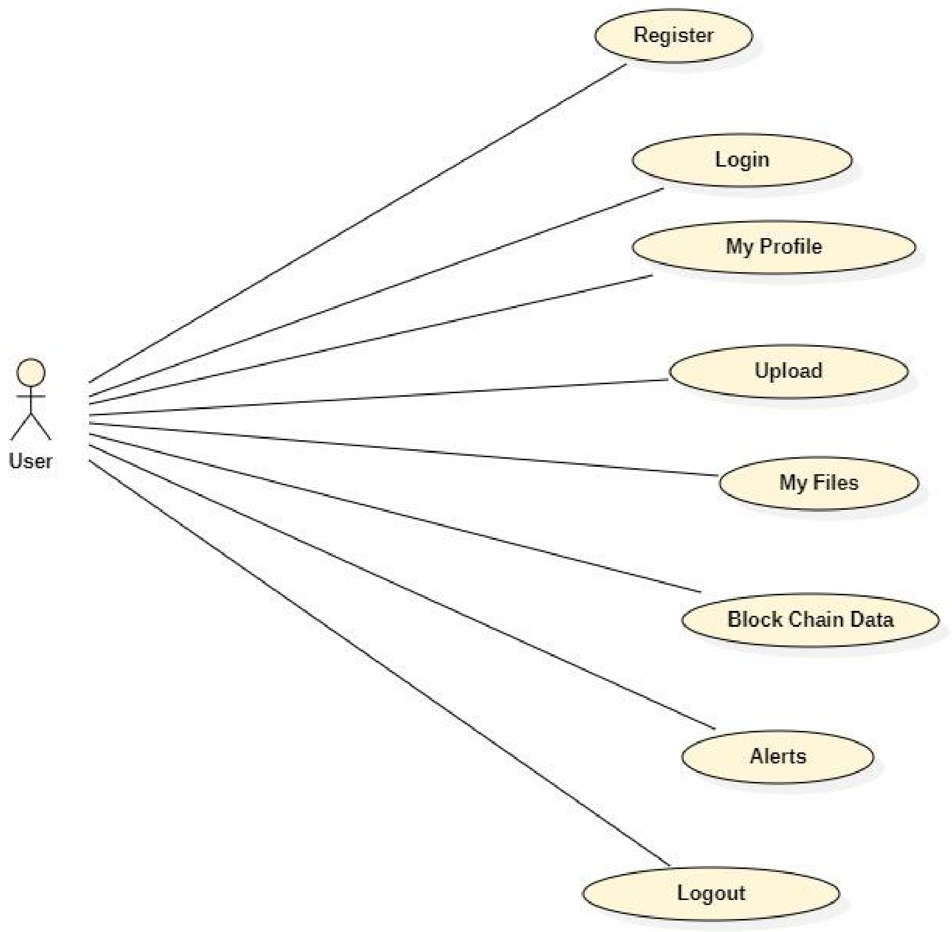
The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices

5.1.1 USE CASE DIAGRAM:

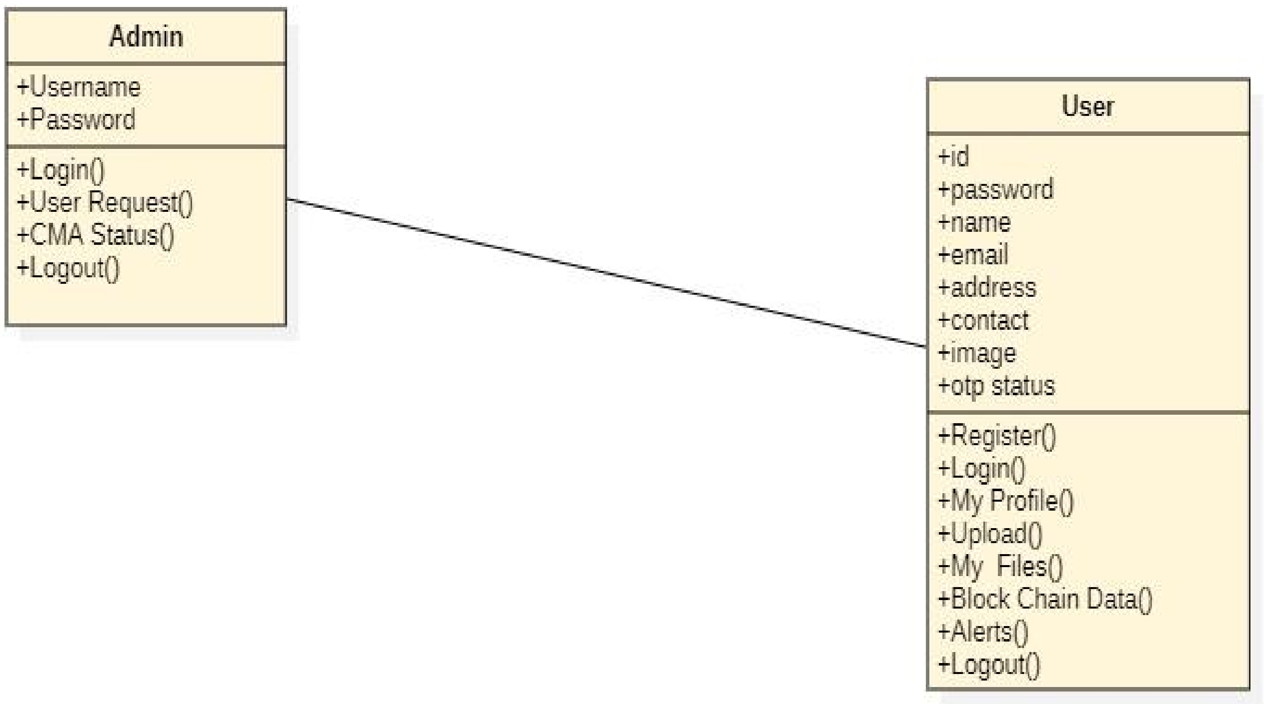
A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted


Login
Dashboard
Admin
User Request
CMA Status
Logout


Register
Login
My Profile
Upload
My Files
Block Chain Data
Alerts
Logout
User

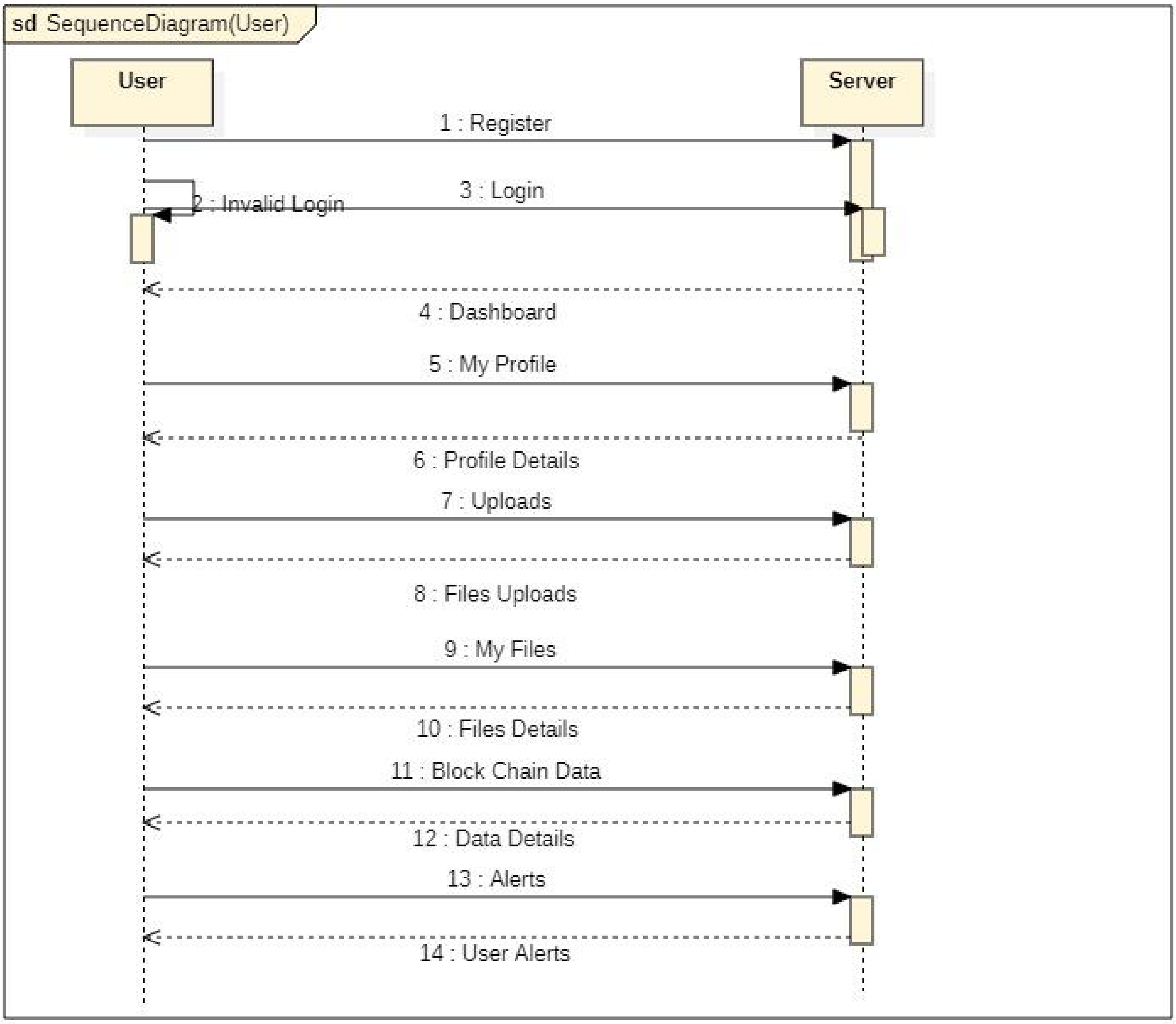
5.1.2 CLASS DIAGRAM:

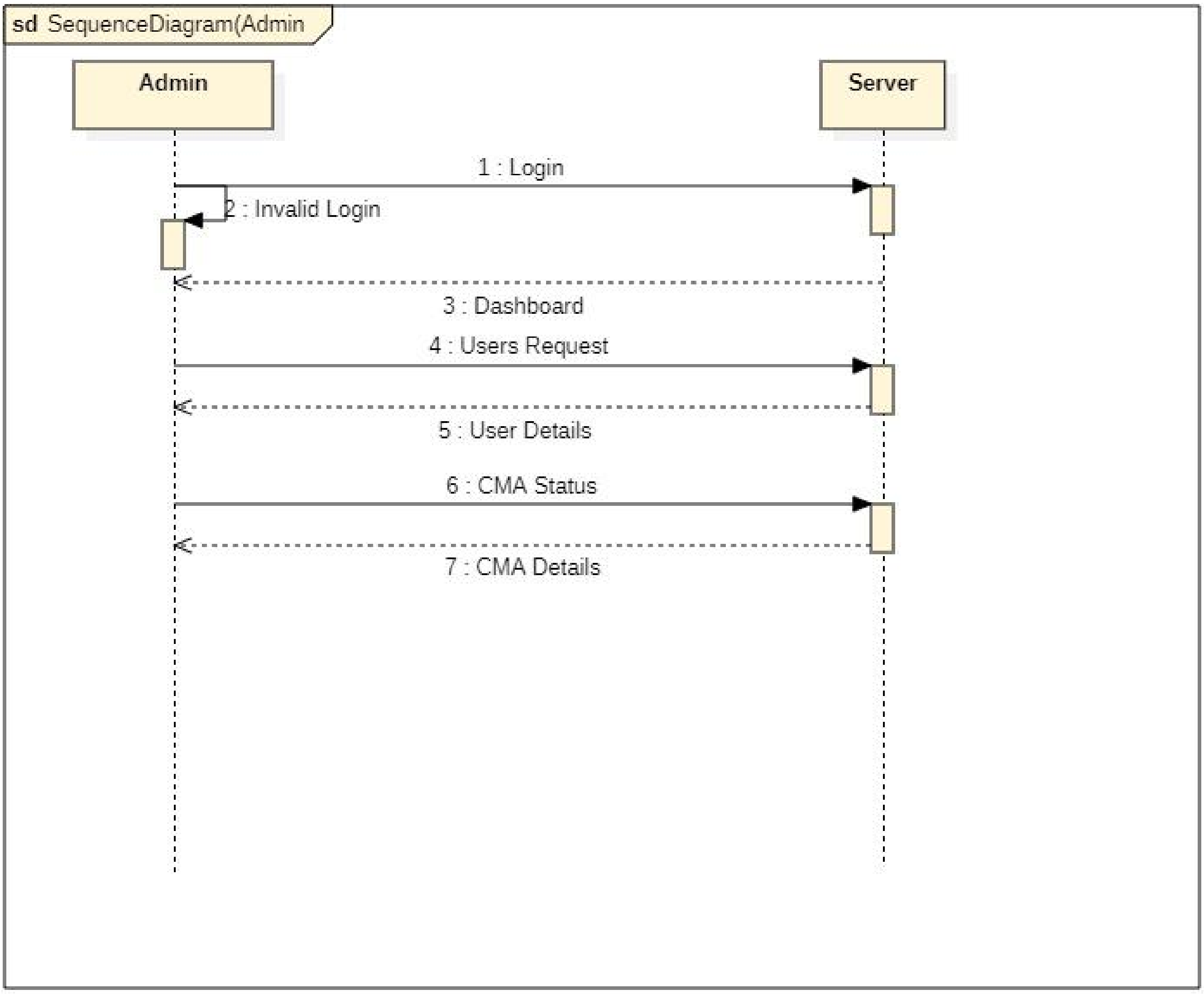
In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information



5.1.3 SEQUENCE DIAGRAM:

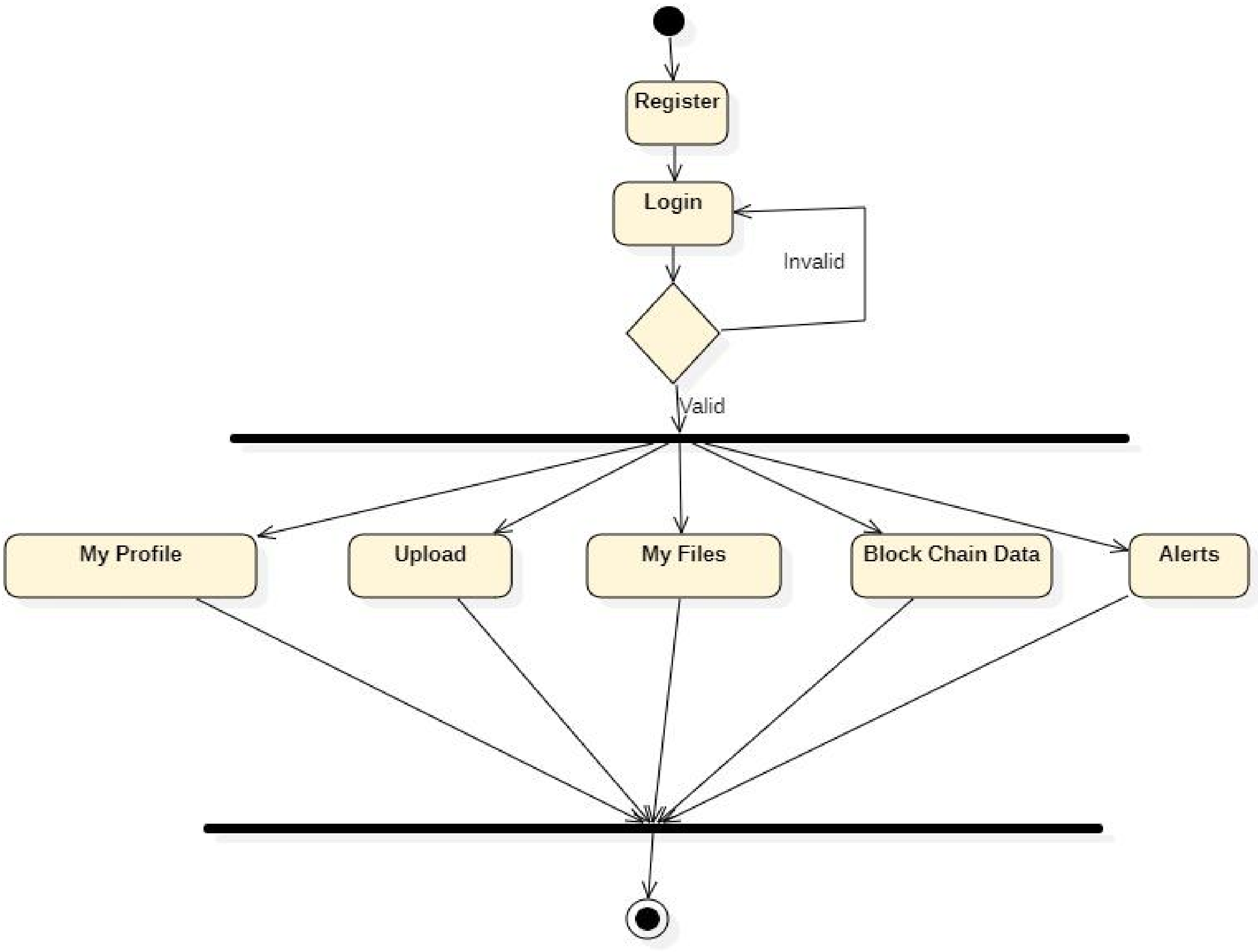
A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams

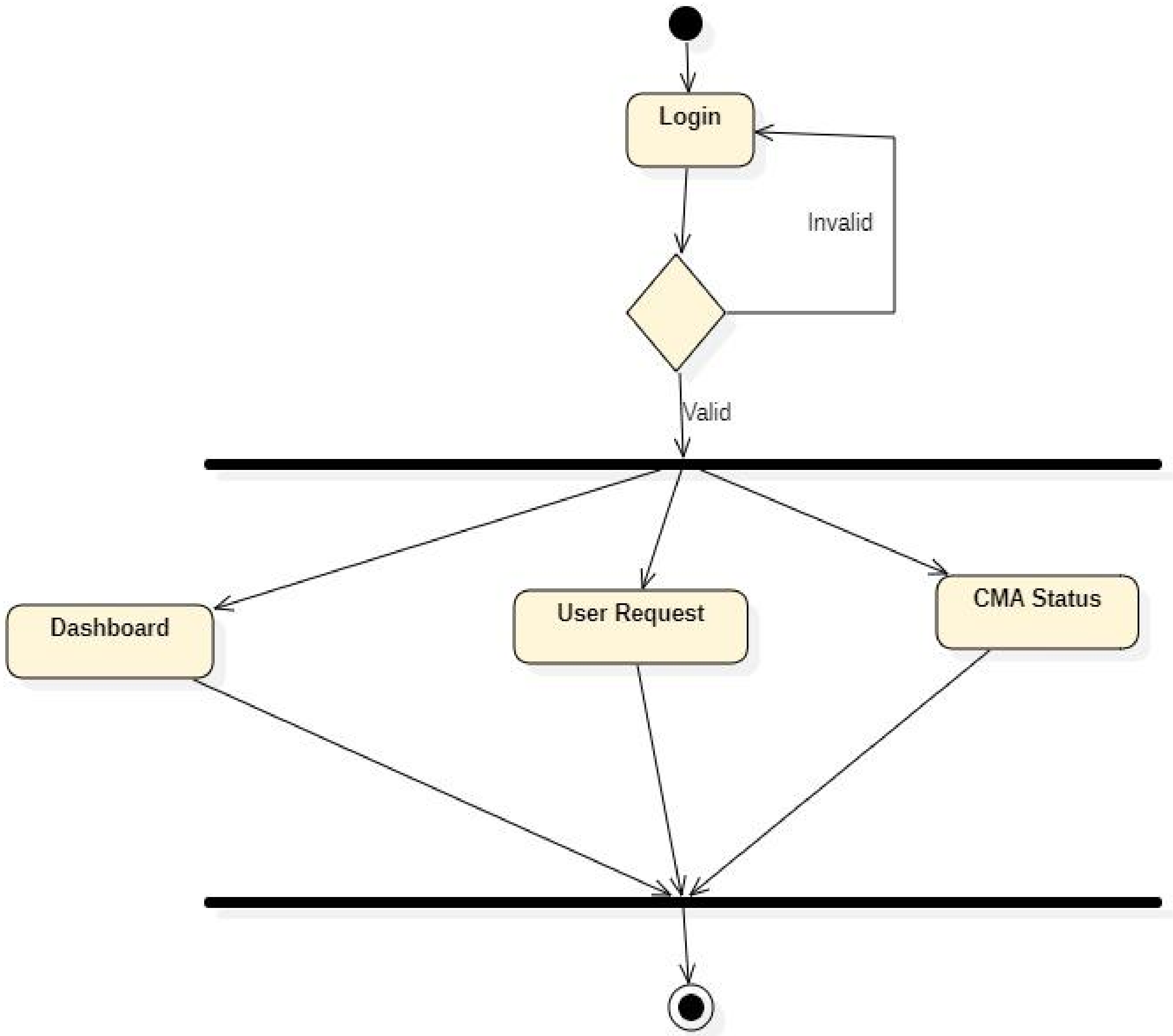

sd SequenceDiagram(User)
User
Server
1 : Register
2 : Invalid Login
3 : Login
4 : Dashboard
5 : My Profile
6 : Profile Details
7 : Uploads
8 : Files Uploads
9 : My Files
10 : Files Details
11 : Block Chain Data
12 : Data Details
13 : Alerts
14 : User Alerts



5.1.4 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control


Register
Login
Invalid
Valid
My Profile
Upload
My Files
Block Chain Data
Alerts


Login
Invalid
Valid
Dashboard
User Request
CMA Status

5.1.5 DEPLOYMENT DIAGRAM:

Deployment Diagram is a type of diagram that specifies the physical hardware on which the software system will execute. It also determines how the software is deployed on the underlying hardware. It maps software pieces of a system to the device that are going to execute it.

The deployment diagram maps the software architecture created in design to the physical system architecture that executes it. In distributed systems, it models the distribution of the software across the physical nodes.

The software systems are manifested using various artifacts, and then they are mapped to the execution environment that is going to execute the software such as nodes. Many nodes are involved in the deployment diagram; hence, the relation between them is represented using communication paths.

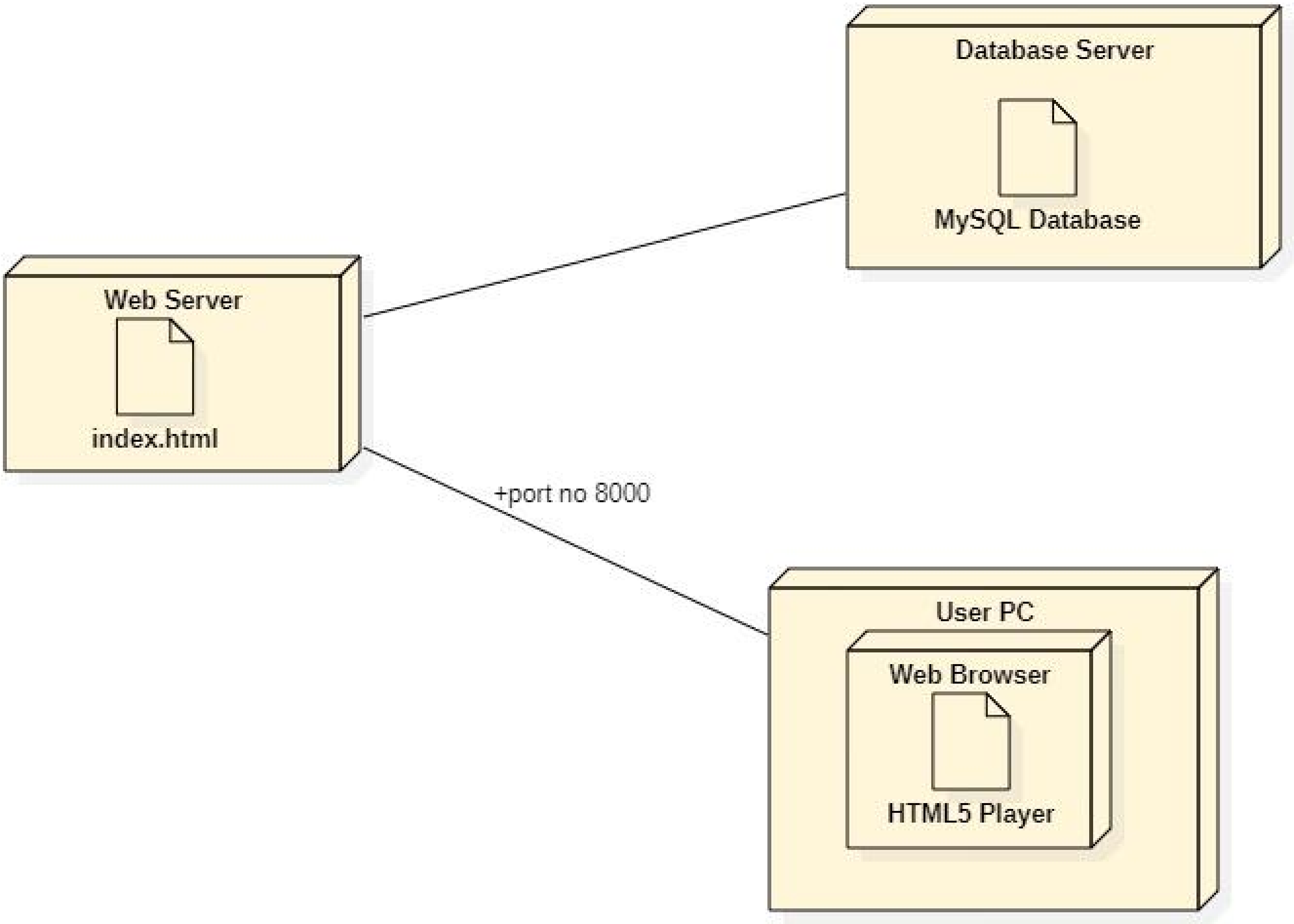
There are two forms of a deployment diagram.

- Descriptor form
  - It contains nodes, the relationship between nodes and artifacts.
- Instance form
  - It contains node instance, the relationship between node instances and artifact instance.
  - An underlined name represents node instances.

Purpose of a deployment diagram

Deployment diagrams are used with the sole purpose of describing how software is deployed into the hardware system. It visualizes how software interacts with the hardware to execute the complete functionality. It is used to describe software to hardware interaction and vice versa.

Deployment Diagram Symbol and notations



6. IMPLEMENTATION

Registration Module: The Registration Module facilitates the initiation of the data sharing process. In this module, a sender registers a file with the Blockchain Big-Data Sharing System (BBS) for sharing. This involves providing necessary metadata, access permissions, and other relevant details. The module ensures proper authentication of the sender and logs the initiation of the data sharing process in the Hyperledger Fabric ledger.

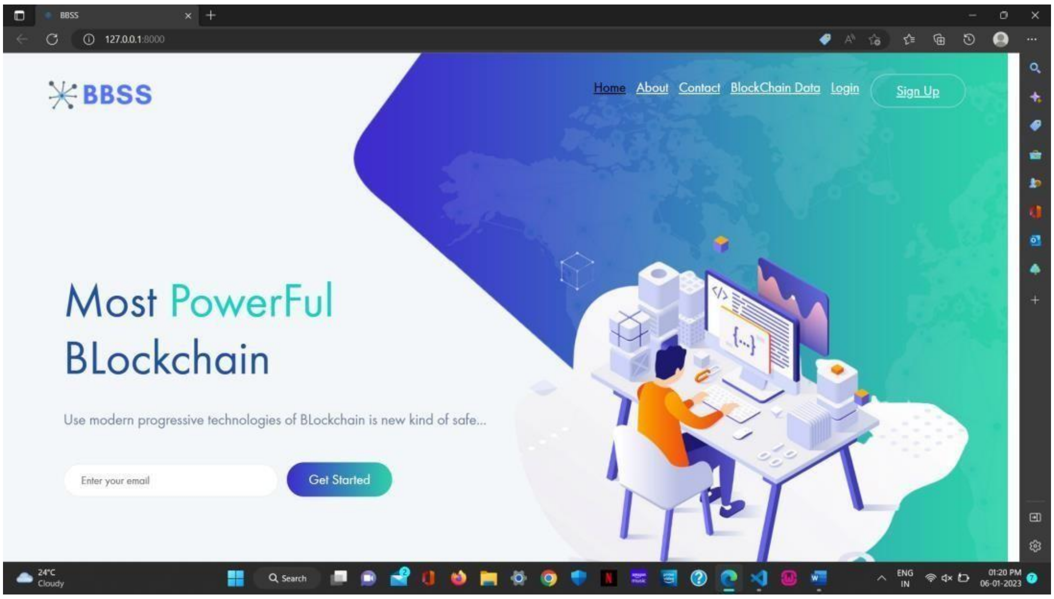
File Transfer Module: The File Transfer Module is responsible for managing the actual transfer of the encrypted file between the sender and the authenticated receiver. This module employs secure encryption techniques to protect the confidentiality of the data during transit. It coordinates the exchange of the encrypted file, ensuring data integrity and secure communication.

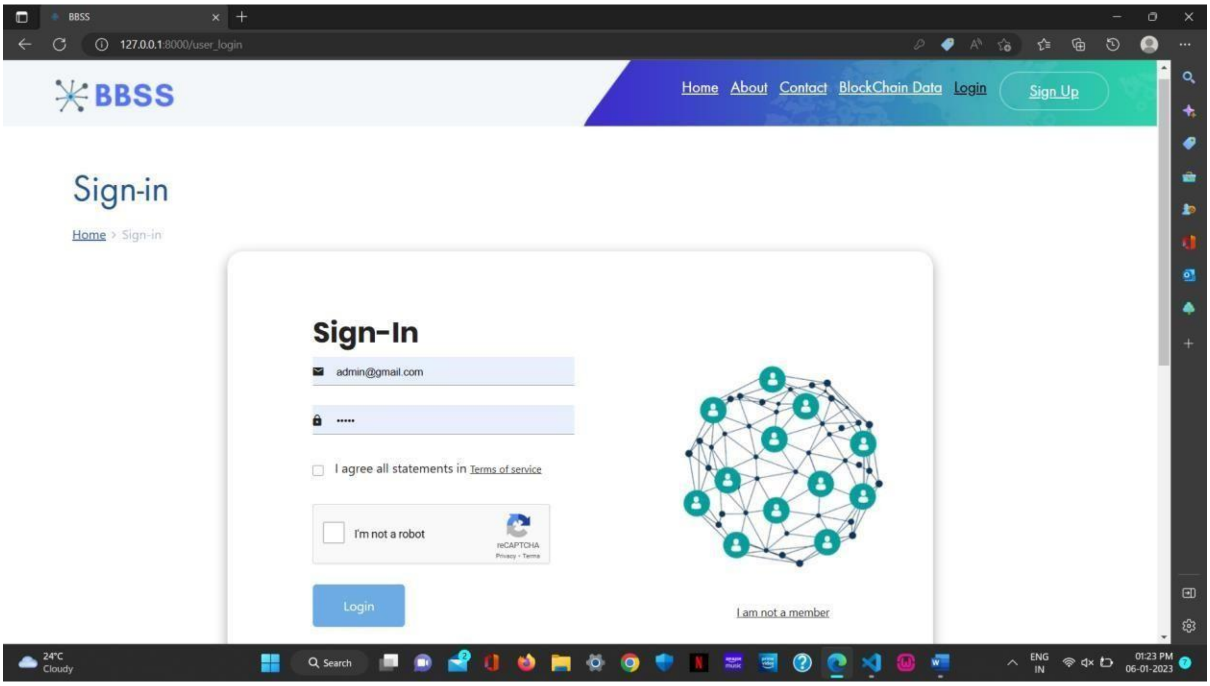
Key Management Module: The Key Management Module plays a crucial role in securely managing encryption keys associated with the shared files. It includes functionalities for key generation, storage, and retrieval. This module ensures that only authorized users can access the necessary keys for decrypting the shared files, enhancing the overall security of the data sharing process

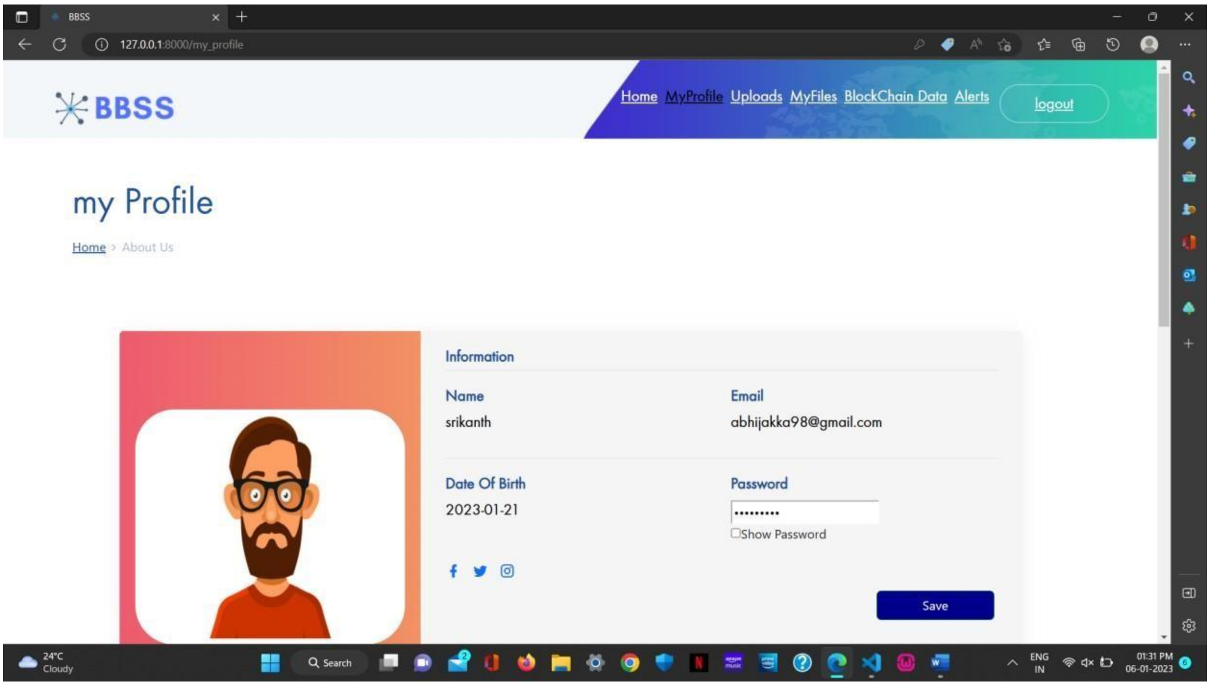
Decryption Module: The Decryption Module is responsible for the final phase of the data sharing process. Authenticated and authorized receivers interact with this module to decrypt the received files using the appropriate encryption keys. The module ensures a secure and controlled environment for file decryption, preventing unauthorized access to sensitive data.

Ledger and Chain of Custody Module: Leveraging blockchain, BBS tracks data access, building a tamper-proof record that reveals who accessed what data. This fosters trust in shared research by guaranteeing traceability and accountability.

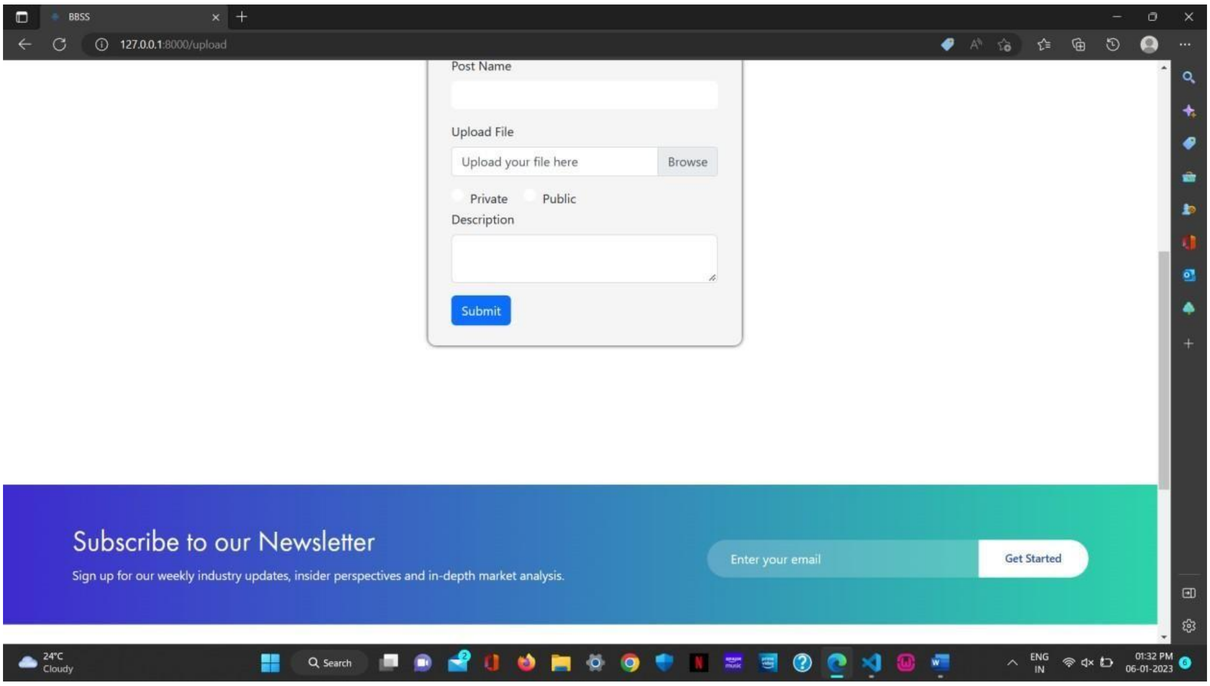
7. OUTPUT RESULTS

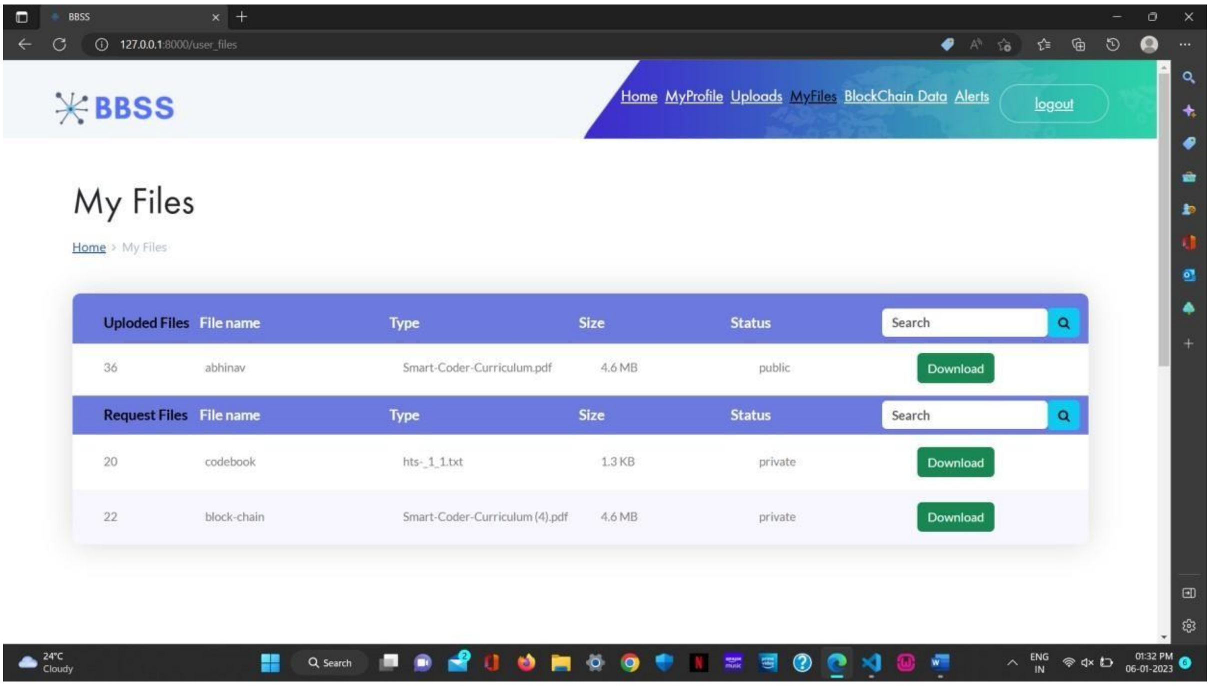


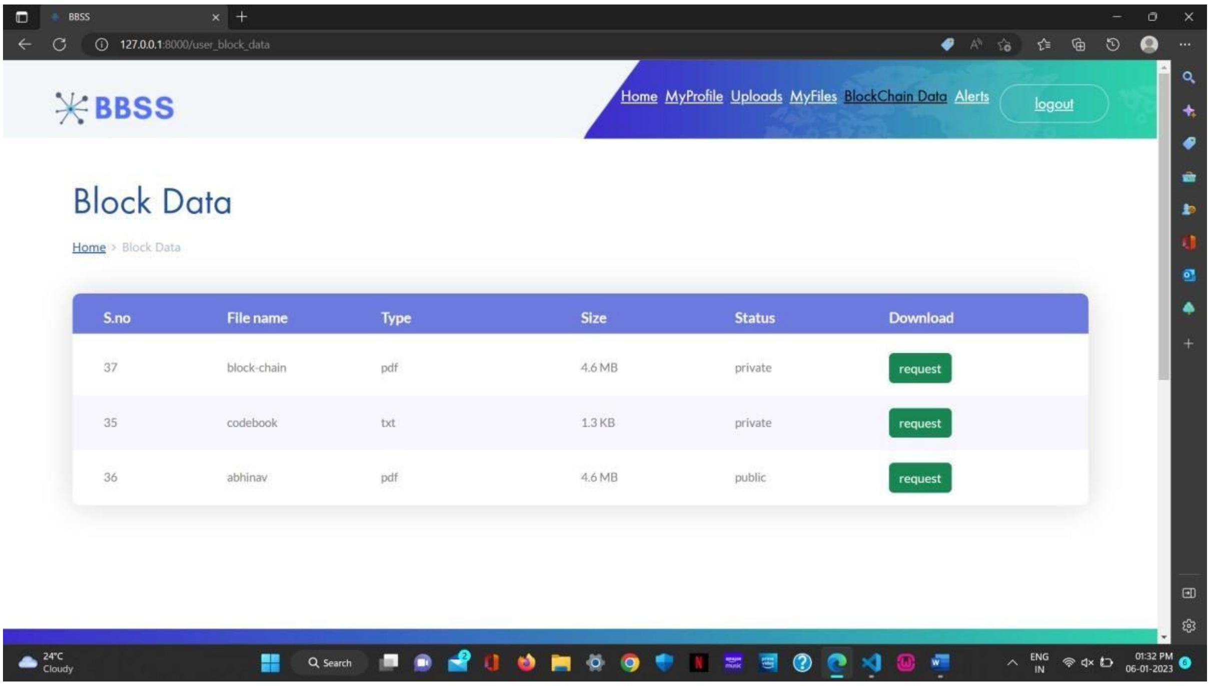


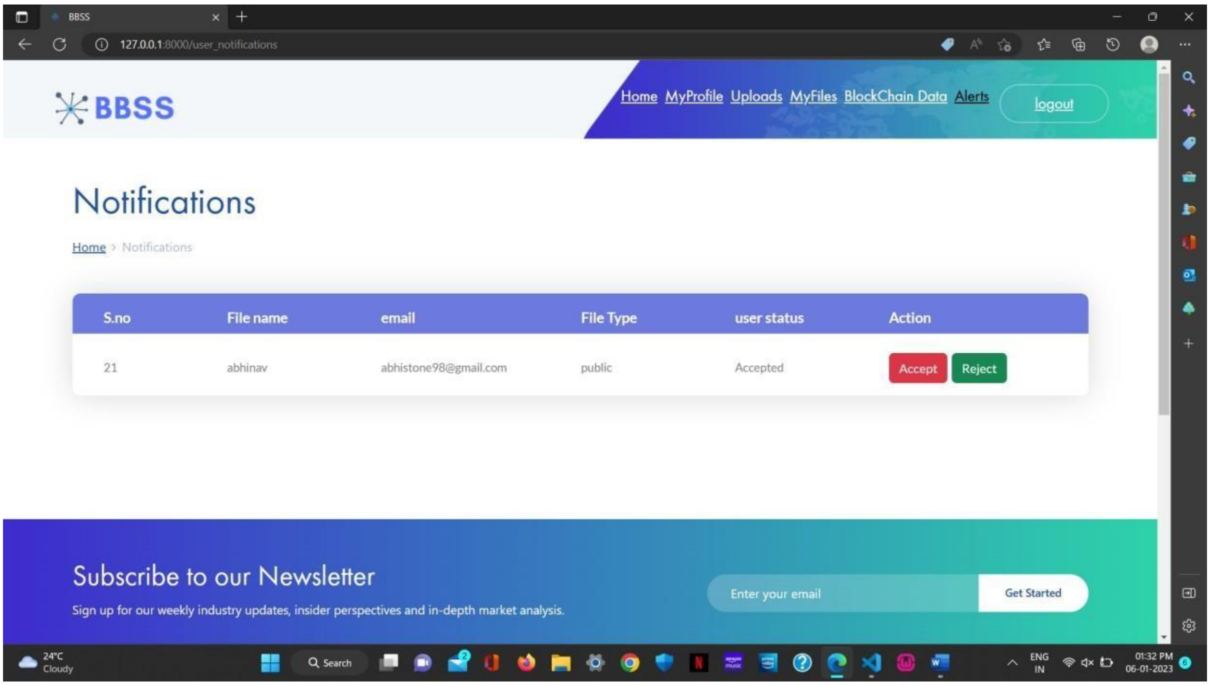


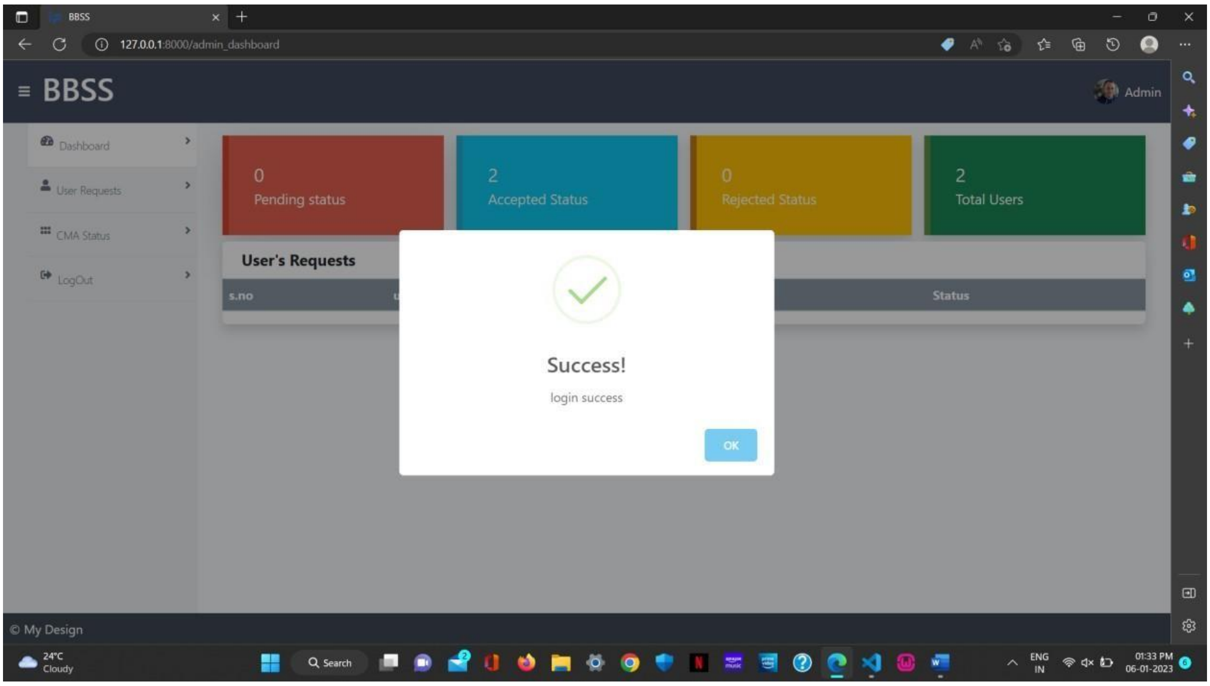
BBSS
127.0.0.1:8000/my_profile
BBSS
Home MyProfile Uploads MyFiles BlockChain Data Alerts
logout
my Profile
Home > About Us
Information
Name
srikanth
Email
abhijakka98@gmail.com
Date Of Birth
2023-01-21
Password
Show Password
Save

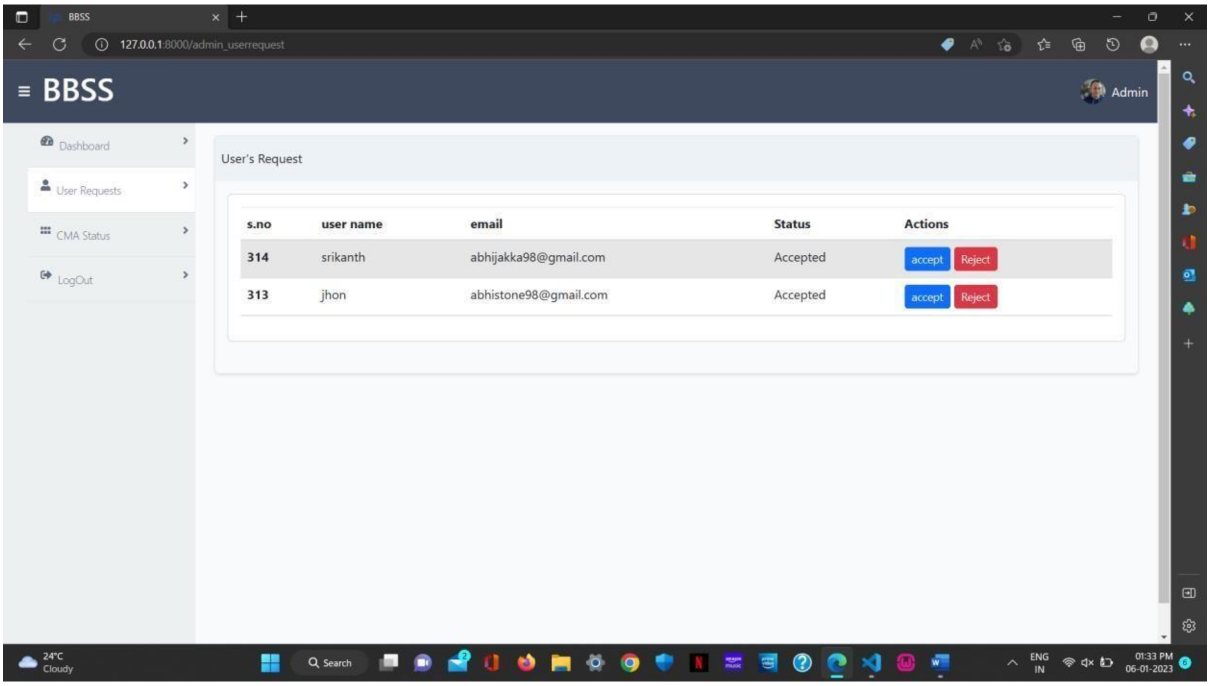

BBSS
127.0.0.1:8000/upload
Post Name
Upload File
Upload your file here
Browse
Private
Public
Description
Submit
Subscribe to our Newsletter
Sign up for our weekly industry updates, insider perspectives and in-depth market analysis.
Enter your email
Get Started

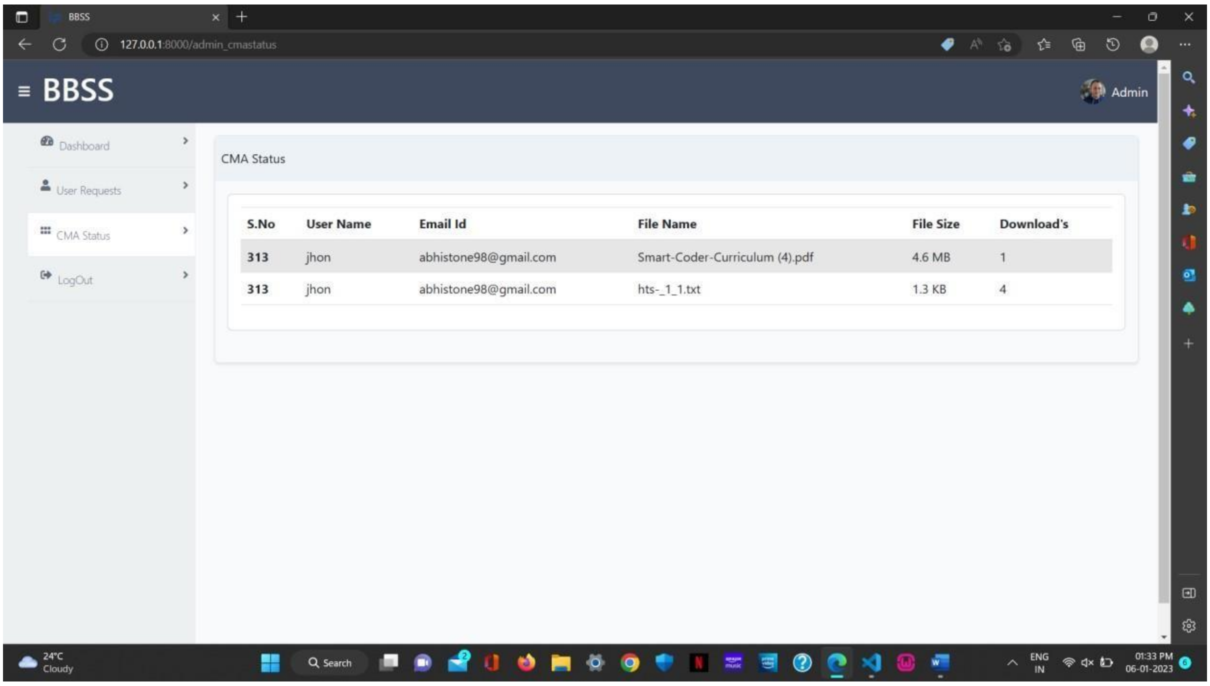

BBSS
127.0.0.1:8000/user_files
BBSS
Home MyProfile Uploads MyFiles BlockChain Data Alerts
logout
My Files
Home > My Files
Uploded Files File name Type Size Status
Search
36 abhinav Smart-Coder-Curriculum.pdf 4.6 MB public
Download
Request Files File name Type Size Status
Search
20 codebook hts-_1_1.txt 1.3 KB private
Download
22 block-chain Smart-Coder-Curriculum (4).pdf 4.6 MB private
Download


BBSS
127.0.0.1:8000/user_block_data
BBSS
Home MyProfile Uploads MyFiles BlockChain Data Alerts
logout
Block Data
Home > Block Data
S.no File name Type Size Status Download
37 block-chain pdf 4.6 MB private request
35 codebook txt 1.3 KB private request
36 abhinav pdf 4.6 MB public request
24°C Cloudy
Search
ENG IN
01:32 PM 06-01-2023


BBSS
127.0.0.1:8000/user_notifications
BBSS
Home MyProfile Uploads MyFiles BlockChain Data Alerts
logout
Notifications
Home > Notifications
S.no File name email File Type user status Action
21 abhinav abhistone98@gmail.com public Accepted Accept Reject
Subscribe to our Newsletter
Sign up for our weekly industry updates, insider perspectives and in-depth market analysis.
Enter your email
Get Started
24°C Cloudy
Search
ENG IN
01:32 PM 06-01-2023


BBSS
127.0.0.1:8000/admin_dashboard
BBSS
Admin
Dashboard
User Requests
CMA Status
LogOut
0
Pending status
2
Accepted Status
0
Rejected Status
2
Total Users
User's Requests
s.no
Status
Success!
login success
OK
© My Design


BBSS
127.0.0.1:8000/admin_userrequest
BBSS
Admin
Dashboard
User Requests
CMA Status
LogOut
User's Request
s.no
user name
email
Status
Actions
314
srikanth
abhijakka98@gmail.com
Accepted
accept
Reject
313
jhon
abhistone98@gmail.com
Accepted
accept
Reject



8.CONCLUSION

In this paper, we solve a novel problem—how to share sensitive big data within a blockchain system autonomously and with no charge, and establish the chain of custody of the shared data securely. Such a data sharing application is critical for protecting intellectual property (IP) and fighting industrial espionage in fields including biomedical research. Three challenges including storage space limitation, privacy requirement and security requirement are identified in implementing the blockchain big-data sharing system (BBS). We denote data such as a big file stored at a blockchain node but outside of the ledger as off-state. We carefully present our off-state sharing protocol. The transactions generated by our protocol will serve as auditing evidences for the chain of custody. Weimplement BBS over Hyperledger Fabric and conduct extensive experiments to evaluate its feasibility and performance.

9. FUTURE SCOPE

The future scope for the project "Big Data Sharing Using Blockchain" is promising, with potential applications across industries such as healthcare, finance, and supply chain management. By leveraging blockchain's immutable and decentralized nature, the project can ensure secure and transparent sharing of vast amounts of data while maintaining privacy and integrity. Enhanced data interoperability, streamlined processes, and reduced costs are anticipated benefits. Furthermore, integrating advanced analytics and AI algorithms can unlock valuable insights from shared data, fostering innovation and informed decision-making. Continuous refinement and adaptation to evolving technological landscapes will be crucial for maximizing the project's impact and scalability.

10.REFERENCES

[1] A. M. Antonopoulos, Mastering Bitcoin: unlocking digital cryptocurrencies. O'Reilly Media, Inc., 2014.

[2] K. Budd, "Foreign data theft: What academic institutions can do to protect themselves," November 2019.

[3] B. Wiki, "Weaknesses," 2021, [Online]. (Accessed 26 March 2021). [Online]. Available: https://en.bitcoin.it/wiki/Weaknesses

[4] B. Forum, "New bitcoin vulnerability: A transaction that takes at least 3 minutes to be verified by a peer," 2021, [Online]. (Accessed 26 March 2021). [Online]. Available: https://bitcointalk.org/index.php?to pic=140078.0

[5] Etherscan, "Ethereum average gas limit chart," 2021, [Online]. (Accessed 26 March 2021). [Online]. Available: https://etherscan.io/c hart/gaslimit

[6] C. Gorenflo, S. Lee, L. Golab, and S. Keshav, "Fastfabric: Scaling hyperledger fabric to 20 000 transactions per second," International Journal of Network Management, vol. 30, no. 5, p. e2099, 2020.

[7] S. Dziembowski, L. Eckey, and S. Faust, "Fairswap: How to fairly exchange digital goods," in Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, 2018, pp. 967– 984.

[8] P. Zhang, J. White, D. C. Schmidt, G. Lenz, and S. T. Rosenbloom, "Fhirchain: applying blockchain to securely and scalably share clinical data," Computational and structural biotechnology journal, vol. 16, pp. 267–278, 2018.

[9] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich et al., "Hyperledger fabric: a distributed operating system for permissioned blockchains," in Proceedings of the thirteenth EuroSys conference, 2018, pp. 1–15.