Cloud Deployment

Team Id	NM2023TMID04410
Project Name	Project-Drug Traceability

Introduction:

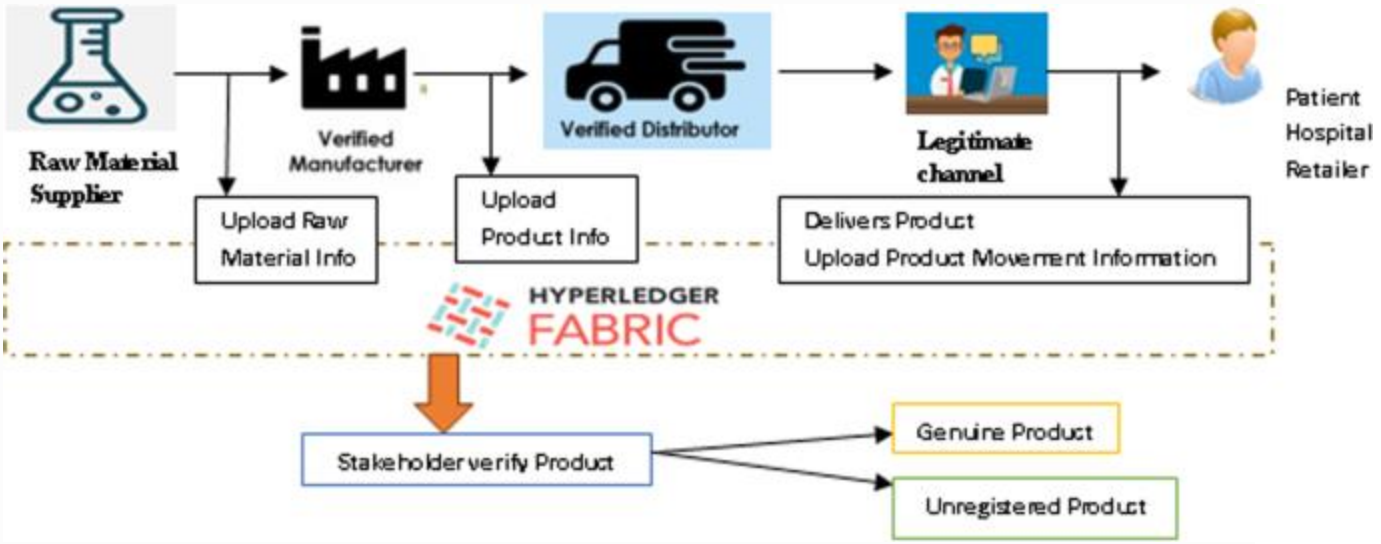
- Distribution points in the supply chain deliver pharmaceutical products to consumers in bulk.
- Drugs are transferred between parties during the supply chain life cycle— from manufacturers to intermediaries, pharmacy chains, hospitals, logistics companies, and the end users (patients).
- Pharma-manufacturers are responsible for manufacturing products according to certain specifications, and ensuring the correct packaging and labelling of the items with relevant information.
- Distribution companies package and transport pharmaceutical products to pharmacies that retail the products.
- Pharmacies, clinics, and hospitals across the city will be available for customers to purchase drugs. Several parties are involved in the supply chain of medical products, including manufacturers, suppliers, wholesalers, supermarkets, pharmacies, and patients.
- More diseases emerge every day, leading to the introduction of new drugs with different names in the market. In the United States, about $200 billion is lost each year in business as a result of drug counterfeiting.
- Many children are killed by fake drugs in developing countries, according to the World Health Organization (WHO) .
- Blockchain can help reduce operational costs, and improve transparency and traceability.
- Business and consumers are able to see more of the supply chain with greater transparency.
- For high-value goods such as diamonds and pharmaceutical drugs, blockchain can improve the supply chain transparency and reduce fraud. Blockchain delivers scalability, security, disaster recovery, and reliability to enterprises.

Blockchain Enabled Pharma Supply Chain:

- The proposed framework as shown in Fig. 3 will allow stakeholders to track the movement of products through the supply chain.
- Hyperledger fabric provides a secure and safe service for viewing the history of the drug supply chain. Our fabric utilizes the Byzantine-fault tolerant consensus to ensure safe and reliable communication in an untrusted environment.
- The components of Hyperledger fabric are certificate authority, peer nodes and ordering service. Certificate authority generates unique certificates for every stakeholder in the network.

- The transaction process is handled by peers in the blockchain. Transaction blocks are ordered through services before they are committed to the network.
- Our Hyperledger fabric provides privacy, efficient processing of transactions, and chaincode facilities to write access control rules using access control language.
- Over 3,500 transactions can be executed per second through the Hyperledger fabric .
- This system guides the consumer through the enlistment, shipping, and purchasing process. With complete visibility of the supply chain, customers can track their orders from start to finish.
- The application shows the order history, shipping history, and receipt history of a retailer and the customer who logged into the application.
- The regulator who oversees the proper implementation of the procedures has access to all orders in the system.

Fig. 3



- The proposed framework has two parts in terms of the development environment. The separate backend and frontend development environments were used in this system.
- Docker is used to implement the backend functionality for a pharma supply chain management system. Implementation and experiments were carried out using a Core i7-8765u processor and 8 GB of memory as shown in Table 1.
- Docker was used for the Docker running environment, as well as for configuring images and containers, and Docker-compose was used for creating virtual machines. A project of the Linux Foundation, Hyperledger Fabric 1.4v [29], was used for our research.
- A Cloud-based IPFS is used as the off-chain database. The Fabric SDK requires Java and Node as prerequisites for client development. REST APIs make it possible to visualize backend business logic, such as user requests, assets, search, and transaction APIs.
- Frontend development was carried out using HTML5, CSS3 and JavaScript. To make our web application more efficient and user-friendly, we use third-party frameworks such as Query and Bootstrap.

- Frontend programming is done with a database, and backend programming is done with REST API servers. Clients use web applications to perform actions that trigger HTTP methods such as POST, GET, PUT, and DELETE, which then cause the web service to respond to HTTP responses according to the requests made by the client.

Table 1 Environment configuration

Table 1 Environment configuration

Component	**Configuration**
System under Test	Hyperledger Fabric 1.4v
CPU & Memory	Core i7-8765u and 8 GB
On-chain database	Couch DB
Off-chain database	Cloud based IPFS
Test language	Node.js,Java

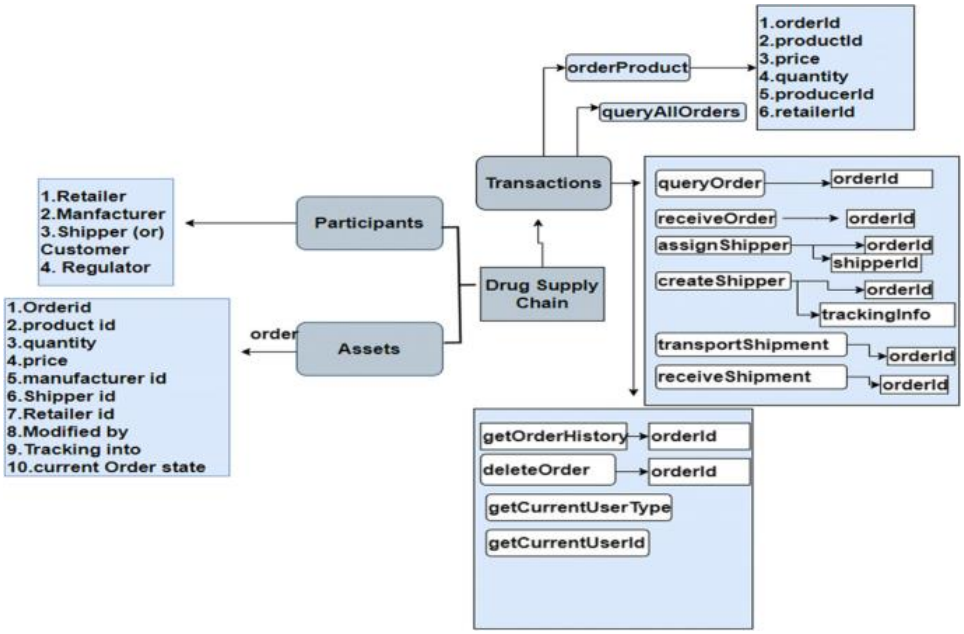
A business network's building component:

- Participants, assets, and transactions are the three major components of a business network built on Hyperledger fabric.

- Production, shipping, retail, and the customer are all stakeholders in the network. In addition to the drugs, prescriptions, orders, and history of products, the assets include the repository.

- Figure 7 shows the transactions set functionality which is designed in our framework.

Fig. 7



Business Network of the Proposed System

Algorithm of the proposed system:

- Our proposed system is implemented with four user types: Customer, Shipper, Manufacturer, and Regulator.
- New users must register through our admin interface. Once registered, users must enrol before logging in. There will only be one enrolment process as shown in Algorithm 1. Customers will be able to view the details and status of their orders using Algorithm 2 and the QR code.
- They must provide product ID, quantity, price, and producer ID to place an order. The producer will then receive the order. Manufacturers can view lists of their orders as well as their status.
- Each manufacturer has the option of accepting an order or rejecting it. The manufacturer must then assign a shipper to the order they have accepted. A shipper's details will be entered for the assignment.
- Products and their status can be seen by the shipper. As soon as the shipper accepts the shipment, it will be transported, and the status of the shipment will be changed from creation to transit.
- Once the shipment has been received by the retailer, the status will be changed to ‘delivered’. Each step will result in an update to the product status. When a customer enters a product ID, they can search for that particular item and access its details. Any product can be tracked by the regulator.

Algorithm 1:

1. Procedure Register_Participant()
2. Inputs:ID,Participant_profile,Participant license
3. Output: Boolean
4. If(Participant profile==Manufacturer)
5. If(Participant license==government licensed) then
6. IPFS←Encrypt(Enroll[id])
7. Hash value → Block
8. Else if(Participant profile==Distributor)
9. If(Participant license==government licensed) then
10. IPFS←Encrypt(Enroll[id])
11. Hash value → Block
12. Elseif(Participant_profile==Retailer|Pharmacist|Hospital|Patient)
13. IPFS←Encrypt(Enroll[id])
14. Hash value → Blocks
15. Else
16. Return False
17. End IF
18. End Procedure

Registration

Algorithm 2:

1. Procedure: Tracing Pharmaceutical Products

2. Input: QR code has Hash value of Product

3. Output: Track product lifecycle

4. Read → QR code

5. Extract Hash value

6. Transaction_Id←Hash values matches the IPFS

7. Decrypt(Transaction_Id[1..n])

8. Loop

9. If Transaction_Id [0] == source_id
10. Data Time ← Transaction_Id [1] to [6]
11. Name ← Transaction _Id [7] to [10]
12. Manufacture details← Transaction _Id [11] to [18]
13. Location← Transaction _Id [19] to [20]
14. Tracetothesource←Transaction_Id [21] to [30]
15. Else Transaction_Id="Product not Found"
16. End if

17. End loop

18. End Procedure

Traceability

Algorithm 3:

1. registerCompany ()

Data Required: Id, Location, Name and Company_Role

2. addDrug()

Data Required: Drug_Name, Serial_No, Manaufacture_date, Expiry_Date, Company_Details

3. transferDrug ()

generatePurchaseOrder ()

initiateShipment ()

updateShipment ()

drugTransaction ()

Data Required: Drug_Name, Purchaser Details, Company_Role, Serial_No, Transporter Details

4. drugHistoryView ()

Data Required: Drug_Name, Serial_No

5. drugState ()

Data Required: Drug_Name, Serial_No

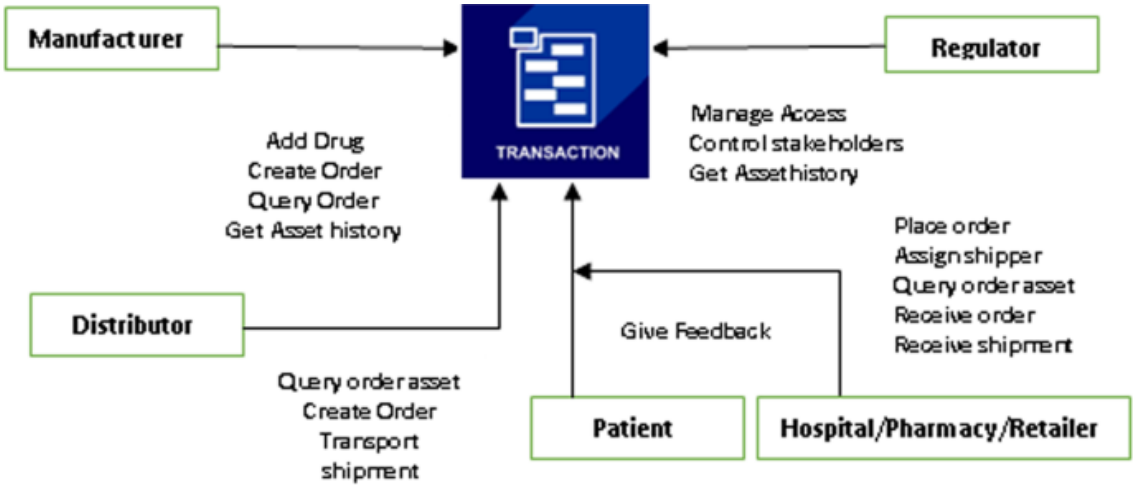
Implemented Functionalities:

Pharma supply chain-smart contract architecture:

Chaincodes are written and deployed into the fabric for the stakeholders to ensure access control, privacy and transparency. For the proposed system, the three different modules of chaincodes are created (Fig. 8).

They are:
• Registration of drugs by the provider
• Management of Assets
• Transaction update

Fig. 8



The Proposed Smart Contract Architecture

Pharma supply chain management in distributed ledger structure

- A blockchain file system and a world state are the two core components of Hyperledger fabric's ledger technology. Databases that keep a cache of the ledger state's current state are known as world state databases.
- The world state contains key-value pairs that store the data. State data is stored in a key–value pair in the default state database used by smart contracts.
- A default database is embedded within the same process as an operating system at each node of the network.
- Smart contracts can be formatted into JSON documents using Couch DB. REST APIs are supported by Couch DB by enabling all kinds of requests to access data which can store all the transactions of the blockchain.

Implementation results:

This proposed framework initially considers 1 organization which consists of peer 1, orderer and couch DB for implementation, and benchmarked using the system under a test. The Hyperledger fabric deployment has few steps as mentioned below:

- Start Hyperledger Fabric
- Initiate Smart Contract
- Create the Connection in the network
- Build and run the coded pharma supply chain applications
- Initiate the transactions

After the successful implementation, the assessment is made based on the following case studies.

Efficient drug transaction with attribute-based visibility:

- The framework used attribute-based access control that allows the visibility of product history based on the role of the stakeholders. The smart contract designed helps to monitor the visibility using a regulator.
- We can strengthen Quality Assurance and Pharmacovigilance by tracing products from the plant to the patient. Inventory visibility minimizes costs and waste. We can reduce revenue losses by identifying 'grey trade' incidents and strengthening brand trust.

Enhanced privacy and transparency:

- The framework is implemented in the permissioned Hyperledger fabric.
- The privacy and transparency of the supply chain is monitored using the Chaincode and distributed ledger which allows for the transparency of transactions to everyone in the network with attribute-based access control.
- The membership service providers maintain the identity of each stakeholder by providing the certificate using public key infrastructure.

Enhanced security:

- The technique of attribute-based access control allows the asset visibility to only allocated stakeholders.
- It has validated the Chaincode smart contract and certificate authority. Hence, information is confidential at each and every stage.

Enhanced scalability:

- These tests verify that our proposed system is able to handle large data with little increase in latency. This has been verified using the Hyperledger caliper benchmark.

Patient safety and quality assurance:

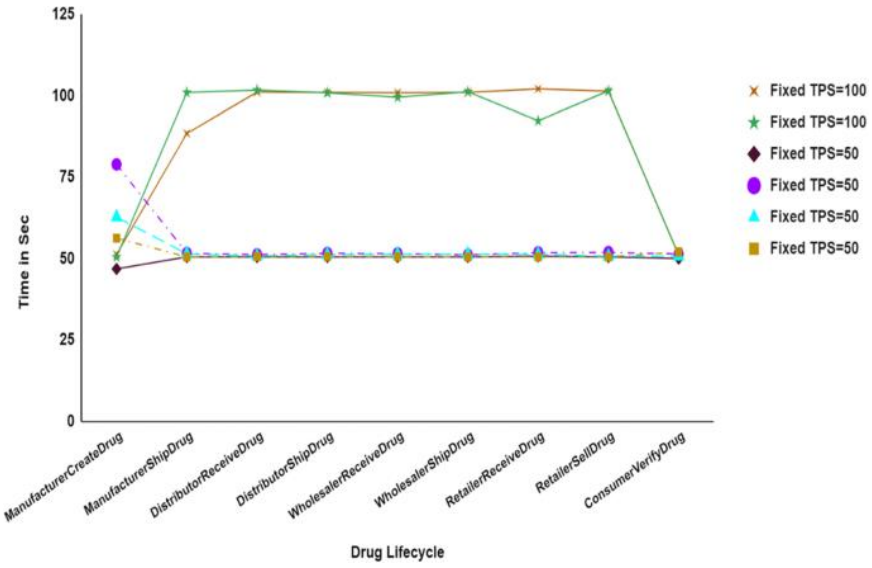
- It provides patients with information about the quality of their medications, and creates a feeling of security by making sure the medications are genuine and of high quality.
- It also facilitates the reporting of adverse events and recall of products more efficiently.

Performance Evaluation:

- Hyperledger Caliper was used to benchmark a blockchain-based application [31]. Caliper is designed to benchmark the performance of Hyperledger using different metrics such as throughput, latency, and success rate (average, minimum, maximum, and percentile). Furthermore, it indicates how resources such as CPU and memory will be allocated to the system. The result is calculated using the following metrics:
  - Success and Failure rate
  - Transaction/Read throughput
  - Transaction/Read latency (minimum, maximum, average, percentile)

- Resource consumption (CPU, Memory, Network (Traffic in and Traffic out and Data read and write)
- The system was initially tested for 1 organization and 1 peer node with various users such as Producer, Distributer, Retailer and Customer.
- The performance benchmarks are tested for manufacturerCreateDrug, manufacturerShipDrug, distributorReceiveDrug, distributorShipDrug, wholesalerReceiveDrug, wholesalerShipDrug, retailerReceiveDrug, retailerSellDrug, and consumerVerifyDrug.
- A simulation was performed for a 100-ms period.

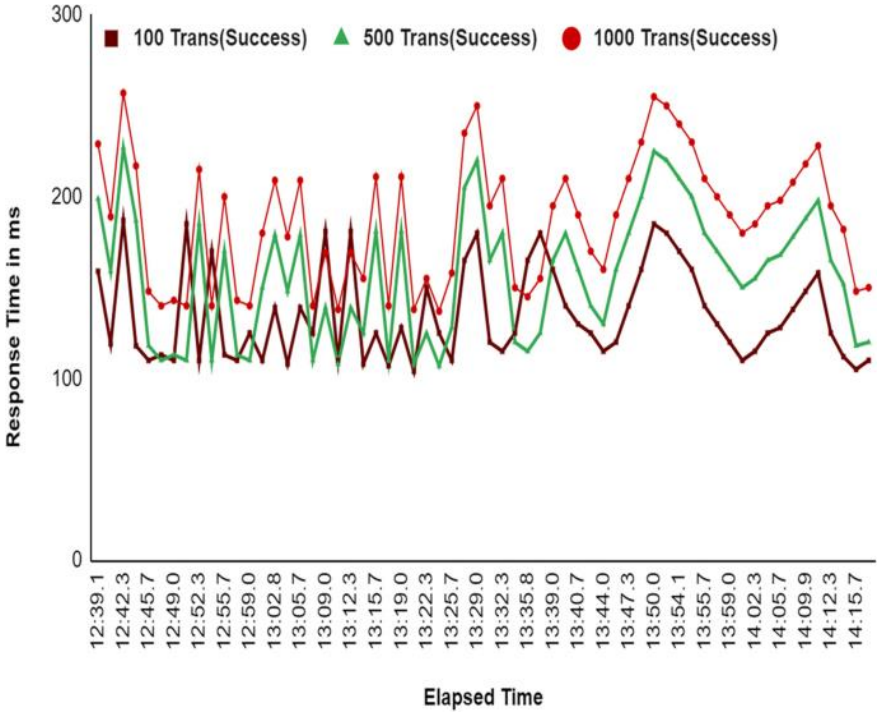
Fig. 9



Comparative Analysis of Response Time with Different Fixed Transaction Sets

We divided users into three groups based on transaction numbers: 100 transactions for round one, 500 transactions for round two, and 1000 transactions for round three. In Fig. 10, we see that the system responds almost the same way to the first two groups, but when we increase to 1000 transactions, the response time increases by only 30 ms. This means that even when the number of transactions increase, the system's response time stays the same.

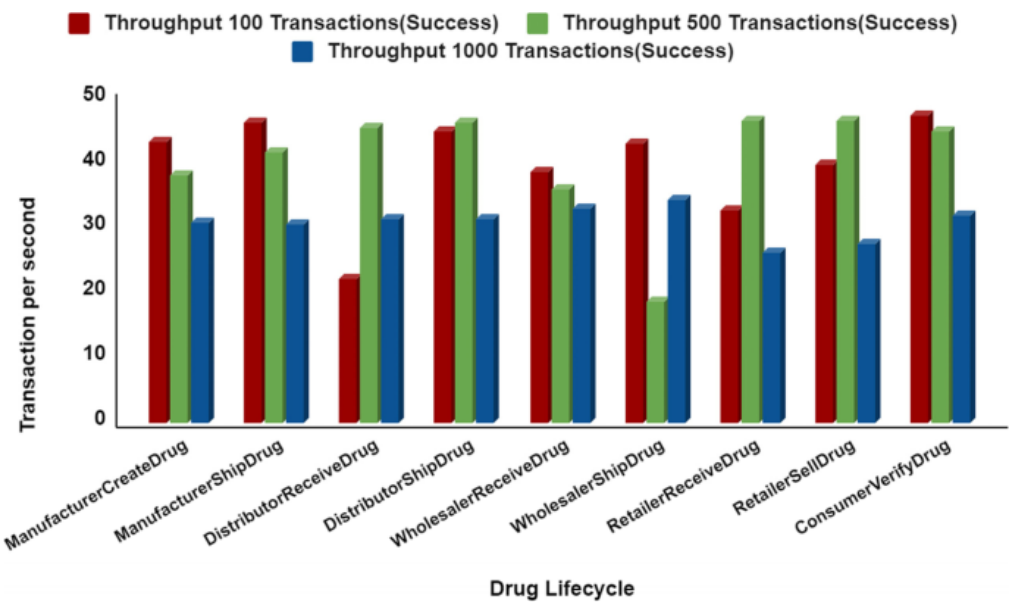
Fig. 10



Comparative Analysis of Response Time with Three Different Sets of Transaction

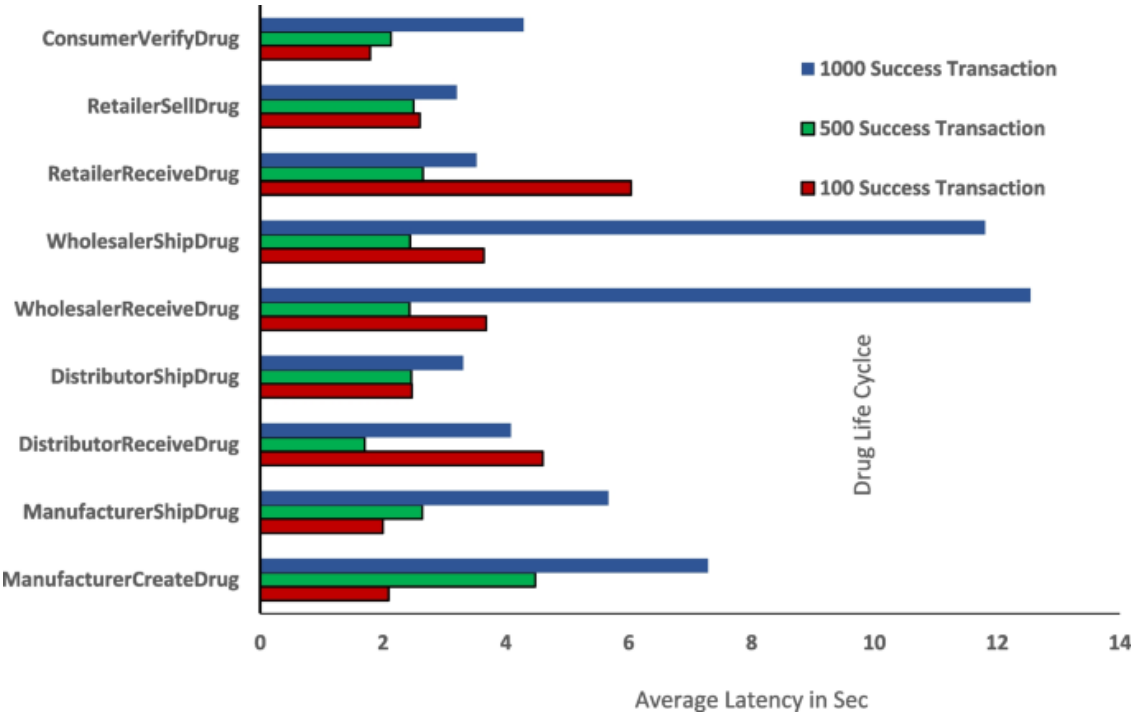
- In Fig. 11, we see that the number of transaction groups increase. The average successful transaction for 100 is 47, the average successful transaction for 500 is 46, and the average successful transaction for 1000 is 35.
- Hence, the average successful transaction has increased by only 25% when we increase the number of transactions from 100 to 1000. Figure 12 shows the average latency of 3 groups of successful transactions.
- For the 100 successful transactions, the average latency is 2.60 s. For the 500 successful transactions, the average latency is 3.21, and for the 1000 successful transactions, the average latency is 6.18.
- Thus, the average latency of successful transaction increases with increase in number of transaction group.
- Even though there is a little increase in latency, no failure or crash of network is recorded. Hence, the scalability of the proposed system is sustained when there is an increase in the number of transaction or users group.

Fig. 11



A Comparative Analysis of Throughput of Three Different Successful Transaction Sets

Fig. 12



Conclusion:

- Historically, traditional supply chains have been insecure, manual, anonymous, ineffective, opaque, centralized, and inaccessible. However, blockchain has proven its ability to transform the industry. Hacks and cyberattacks cannot occur on a blockchain.
- Blockchain technology is being used to digitize the drug supply chain, which will ensure security and transparency among users involved in it.
- To implement this proposed system, Hyperledger Fabric is considered as it is suitable for a permissioned network setup since the stakeholders will all have unique identifiers.
- Hyperledger fabric enables a user to have access controlled according to his or her role, therefore increasing the security and confidentiality of data. The system is reliable and always available since it is implemented in a distributed environment.
- The proposed system is transparent, secure, and private due to the use of smart contracts in conjunction with permissioned blockchain technology. Hence, drug trade is made more transparent and secure with blockchain technology.
- Utilizing blockchain technology, we have found that the proposed system achieves higher throughput, and lowers latency with minimal use of resources. Our current drug supply chain has loopholes that can be closed with the proposed model.
- In addition, Cloud-based blockchain frameworks can be used for scheduling real-time delivery, as well as detecting counterfeit drugs.
- There are a number of issues that can be addressed in future research, such as the development of a new supply chain framework for drugs that require extra handling. Blockchain technology is used in COVID-19 vaccine anti-counterfeiting frameworks.