Blockchain based Information Architecture for Medical Product Supply Chain

Dr. M. Sangeetha
Professor/IT Department,
V.S.B. Engineering College,
Karur-639111, Tamil Nadu.

Mr. G. Mahalingam,
IV-Year (B.Tech-Information Technology),
V.S.B Engineering College,
Karur-639111, Tamil Nadu.

Mr. S. Manisankar,
IV-Year(B.Tech-Information Technology),
V.S.B Engineering College,
Karur-639111, Tamil Nadu.

Mr. S. Mythreyan,
V.S.B Engineering College,
IV-Year(B.Tech-Information Technology)
Karur-639111,Tamil Nadu.

Abstract:- **The medical product supply chain is the most complex and fragmented of all supply chains. The production is found all over the world. A lot of manufacturer and retailers are difficult to identify and track. For all the participants in the product supply chain this creates uncertainty and risk. Mitigating this uncertainty comes at a quality, and the outcome may still be insufficient. Examples of problems that have been difficult or impossible to solve with current technologies include establishing reliable provenance and preventing fraud and counterfeiting. These issues can have knock-on effects on public health and the environment, and reduce financial costs of unnecessary recalls of Medical products. To overcome the above challenges, a blockchain based Medical Product traceability system is proposed in this study, to achieve the following: To integrate blockchain technology for effective and efficient traceability, and to support shelf life adjustment and quality decay evaluation for improving quality. For the sake of better computational load, the blockchain is modified as a lightweight blockchain to be associated with cloud computing to support monitoring, and can be analysis after the whole life cycle of traceability to release computational resources of the system. By using a collection of reliable data, the decision support in product quality can be made by using fuzzy logic to determine adjustment of shelf life, rate, and order of quality decay, according to different situations for each batch of perishable products at processing sites. Therefore, the proposed traceability model is extended to the modern Medical Product supply chain environment, resulting in reliable and intelligent monitoring, product tracking, and quality assurance.**

Key terms:- Blockchain, Ethereum, smart contracts, traceability, Medicine, medicine supply chain, medicine safety.

INTRODUCTION

Blockchain has huge potential to impact global Medical Product supply chain (MPSC) by increasing productivity in terms of supply chain performance. Among many challenges the United States Center for Diseases Control (CDC) estimates that 48 million people get sick from expired medical product usage , 128,000 are seriously hospitalized, and 3,000 die each year in the U.S. alone. Apart from illness, economically and criminally motivated Medical Product adulteration is also a growing concern due to globalization and wide growing supply chain networks. Real-time monitoring of the medical product quality and visibility of that quality index would prevent outbreak of food-borne illnesses, economically motivated adulteration, contamination, food wastage due to misconception of the labeled expiry dates, and losses due to spoilage, which have broad impacts on the medical product security.

In order to improve safety and prevent wastage, modern blockchain based technologies are required to monitor the Medical product quality and increase the visibility level of the monitored data. There are a number of Block Chain based tracking and tracing infrastructures such as Electronic Article Surveillance (EAS), Radio Frequency Identification (RFID), and QR codes which are primarily targeted for automatic package level tracking. However, the role of these technologies is limited in identifying the medical product package and does not provide any information pertaining to the state of the Medical product quality. This limitation prevents quick removal of a defective product from reaching higher levels of the MPSC. For example, when a quality control lapse is identified along the MPSC, the company is forced to recall all the Medical products within a certain time frame leading to a huge economic loss, which can be mitigated with the availability of individual Medical Product package quality information resulting in targeted recalls. In literature, a number of sensing techniques compatible with existing tracking and tracing infrastructure are proposed for monitoring Medical products.

These can be invasive or non-invasive in monitoring the physical or chemical properties of medical products such as pH, conductivity, and permittivity or the packaging environment such as temperature, humidity,

moisture or aroma. In general, these are aimed to prevent defective products from reaching the consumers. Furthermore, these sensors help in identifying key bottlenecks in the MPSC to improve the overall efficiency. Currently, little work has been done in integrating these to the tracking and tracing infrastructures. Moreover, the collected tracking as well as sensing data is more centralized and selectively used by specific entities of the MPSC. The consumers have to trust the quality of the product based on the printed expiry date without any additional knowledge of its current quality. To move beyond a "traceability-centric" or "income-centric" to a "value-centric" supply chain, a more decentralized approach is needed in terms of data sharing. However, a trade off exists between providing sufficient information to the consumer about an individual product and at the same time safe guarding the operational privacy of the MPSC.

Blockchain has emerged as a decentralized public consensus system that maintains and records transactions of events that are immutable and cannot be falsified. Blockchain technology has attracted attention beyond crypto currency due to its ability to provide transparent, secure, and trustworthy data in both private and public domains. The technology is based on a distributed ledger, which is not owned or controlled by a single entity. Data in the public ledger is visible publicly and any authorized entities can submit a transaction, which is added to the Blockchain upon validation. The advantage of Blockchain technology can be applied in MPSC to improve the digital data integrity which is obtained as the product passes through different entities of the MPSC. The complete Medical product visibility across different entities of the supply chain can become a reality with the integration of sensor based Blockchain technology data management systems. The key benefits of applying Blockchain technology in MPSC are: real time tracking and sensing of Medical products throughout the MPSC, and allowing identification of key bottlenecks; Discouraging adulteration of Medical products, and identifying weak links on occurance; determining the shelf life of Medical products leading to reduced waste; providing end to end information to the consumer; and allowing specific and targeted recalls. A test prototype of the Unique ID is integrated are demonstrated experimentally in this work. The Unique ID integrated can be attached to a food package to extract information regarding the package along MPSC.

DOMAIN INTRODUCTION

A blockchain, is a increasing list of records, denoted at blocks, that are linked using cryptography techniques. Here, each block contains a previous block of information, a timestamp, and transaction datas. The design, of a blockchain is resistance of the modified data. It is works with blocks, where as spreadsheet works with "rows" and "columns". A block in collection of data.Blockchain is a distributed ledger, A block in collection of distributed data collection database, which simply means that a record is spread across the network among all peers in the network, and each peer holds a copy of the complete registry of distributed records.

Blockchain is an increasing list of database. It is denoted as blocks, that are linked using cryptography. Each block contains a previous block records, a timestamp, and transaction data. The design, of a blockchain to acting like resistant of the modified data. Once the data is registered, cannot be altered retroactively without alteration of all subsequent blocks, which requires consensus of the network majority. The blockchain technique contain permanent records, and it may be considered secure by design and exemplify a distributed computing system with high Byzantine fault tolerance. Decentralized consensus has therefore claimed with a block chain technique.

Block chain technique was invented by Satoshi Minamoto in 2008. It serves as the public transaction record of the cryptocurrency bitcoin. The identity of Satoshi Minamoto is unknown. The block chain is a distributed ledger that enables peer-to-peer transaction in one of the safest environments.. Block chain is considered a type of payment rail. Private blockchains have been proposed for business use.

STRUCTURE

The block chain is, an immutable timestamp series registry of data that is distributed and managed by cluster of computers. Here, the block chain technique allows the user to verify and audit transactions independently and relatively inexpensively. Block chain technique is a shared and immutable ledger, the information in it is open for anyone and everyone to see. Lets, they are authenticated by mass collaboration powered by collective self-interests. Here, the design facilitates robust workflow and where users' uncertainty regarding data security. The use of a block chain removes the characteristic of infinite reproducibility from a digital asset. They confirms that each unit of value was transferred only once, solving the long-standing problem of double spending. A block chain has been described as a value-exchange protocol.A block chain can maintain title rights because, when properly set up to detail the exchange agreement, it provides a record that compels offer and acceptance.

BLOCKS

The blocks hold batches of valid transactions. Here, they are hashed and encoded into a Merkle tree. Each block includes the cryptographic hash of the prior block in the block chain. The linked blocks form a chain. Here, that type of iterative process confirms the integrity of the previous block. In these techniques sometimes specific blocks can be produced concurrently, creating a temporary fork. In addition to a secure hash-based history. Block chain has a specified algorithm for scoring different versions of the records. In that place one with a higher score can be selected over others.

Here, blocks are not selected for inclusion in the chain are called orphan blocks. User supporting the database have different versions of the history from time to time. Here, it can keep only the highest-scoring version of the database known to them. Whenever a user receives a higher-scoring version they extend or overwrite their database and re-transmit the improvement to their users. Here, is never an absolute guarantee that any particular entry will remain in the best version of the history forever. The block chains technique main objective to add the new blocks onto old blocks and are given incentives to extend with new blocks rather than overwrite old blocks. Therefore, the probability of an entry becoming decreases exponentially. The best example for bitcoin. It uses a proof-of-work system, where the chain with the most cumulative proof-of-work is considered the valid one by the network. Here, there are a number of methods can be used to demonstrate a sufficient level of computation.

BLOCK TIME

The block time denoted at the average time is taken for the network to generate one extra block,it is called block time. Here, some blocks chain technique to create a new block as frequently as every five seconds. At the time of the block completion, the included data go to verifiable state. Here, the shorter block time denoted at faster transactions. The Fastest transaction achieved by using cryptocurrency technique. The block time of Ethereum is 14 to 15 seconds, while for bitcoin it is on average 10 minutes.

HARD FORKS

The hard fork is a set of rules about software validation. Here, all node works mainly base on new rules. This type of new rules used to upgrade their software specification. In hard forks method one group of nodes continues to use the old software and other nodes use the new software, a permanent split can occur. Here, the best example for Ethereum. The hard-fork to makes whole the investors in the DAO, which had been hacked by exploiting a vulnerability in its code. In this type of scenario, the fork resulted in a split creating Ethereum and Ethereum Classic chains. In 2014 the NXT community was asked to consider a hard fork is a rollback of the block chain records to mitigate the effects of a theft of 50 million NXT from a major cryptocurrency exchange. here, the hard fork proposal was rejected, and some of the funds were recovered after negotiations and ransom payment.

DECENTRALIZATION

The decentralized network data can be stored in P2P network, here the blockchain technique to eliminates a number of hazard that come with the data being held centrally. The decentralized blockchain technique may use adhoc message in passing and distributed networking.

Peer-to-Peer blockchain networks technique have a lack of centralized point of unprotected computer crackers can exploit; likewise, it doesn't contain central point of failure. Blockchain security technique include the use of public-key cryptography. A public key is an address on the blockchain technique. Here, value tokens sent across the network are recorded as belonging to that address. A private key is a password. It can be provided by the product owner access to their digital assets or the means to otherwise interact with various capabilities that blockchains now support. All processed data are stored on the blockchain is generally considered undestroyable.

Each and every node in a decentralized system has a copy of the blockchain technique. Here, the data quality is maintained by massive database replication, and computational trust. Transactions are broadcast to the network using software. Messages are delivered on a best-effort basis. The mining nodes to validate the transactions and add them to the block. The blockchain technique use various time-stamping methods, such as proof-of-work, to serialize changes. Alternative consensus methods include proof-of-stake. Here, the growth of a decentralized blockchain is accompanied by the harm of centralization because the computer resources required to process larger amounts of data become more expensive.

OPENNESS

The open blockchain technique are more accommodating than some classic ownership records, which, while open to the public, still require physical access to view. Because all early blockchain technique were permission less, argument has arisen over the blockchain technique definition. The idea of the ongoing controversy is whether a private system with validates task and permission by a decentralized authority should be premeditated a blockchain technique. The proponents of authenticated or private chain action that the term "blockchain" may be appeal to any data structure that batches data into time-stamped blocks. Blockchain technique to serve as a distributed version of multi version concurrency control (MVCC) in databases. Here, MVCC avert two transactions from contemporaneous modifying a single object in a database, blockchain technique to prevent two transactions from spending the same single output in a blockchain. Challenger says that permission systems resemble traditional corporate databases, not supporting decentralized data verification, and that such systems are not hardened against operator tampering and revision.

PERMISSION LESS

The great feature to an open, permission less, or public, blockchain network is that oversee against unpleasant actors are not need, and no access control is needed. This means that applications can be joined to the network without the approval or trust of others, using the blockchain technique as a transport layer. The bitcoin and other cryptocurrencies secure right now their blockchain technique by requiring new arrival to include a proof of work. To extend the blockchain technique, bitcoin uses Hash cash puzzles.

DISADVANTAGES OF PRIVATE BLOCKCHAINE

The private blockchain technique already controls 100 percent of all block establishment resources. Here, if you could damage the blockchain creation tools on a individual corporate server, you could successfully control 100 percent of their network and modify transactions however you wished. This has a set of specifically intelligent adverse intimation during a financial crisis or debt crisis like the financial crisis of 2007–08, where politically powerful actors may make decisions that approval some groups at the consumption of others, and "the bitcoin blockchain technique is protected by the massive group mining effort. It's unlikely that any private blockchain technique will try to save records using gigawatts of computing power — it's time ingest and expensive. This means that many in-house blockchain technique solutions will be nothing more than cumbersome databases."

USES

The blockchain technology can be accommodated into multiple areas. The fundamental use of blockchains today is as a distributed registry for cryptocurrencies, most notably bitcoin. There are a few valuable products maturing from proof of concept by late 2016. The businesses have been thus far reluctant to place blockchain at the core of the business structure.

CRYPTOCURRENCIES

Many of the cryptocurrencies technique use a blockchain technology to save transactions. The best example, the bitcoin network and Ethereum network are both based on blockchain technique. On May 8, 2018 Facebook confirmed that it would open a new blockchain technique group which would be headed by David Marcus, who previously was in charge of Messenger. The Facebook's planned cryptocurrency platform, Libra, was formally announced on June 18, 2019.

SMART CONTRACTS

The blockchain-based smart contracts are suggested contracts that can be been moderately or fully executed or enforced without human intercommunication. One of the main intentions of a smart contract is automated covenant. An IMF staff conversation reported that smart contracts based on blockchain technology might reduce moral danger and optimize the use of contracts in accustomed. Here, “no viable smart contract systems have yet emerged.” Due to the lack of general use their legal status is unclear.

SUPPLY CHAIN

There are a number of attempt and industry organizations working to employ blockchain technique in supply chain logistics and supply chain management. The Blockchain in Transport Alliance (BiTA) works to establish open standards for supply chains.

TYPES

Immediately, there are at least four types of blockchain networks — public blockchains technique, private blockchain technique, consortium blockchains technique and hybrid blockchain technique.

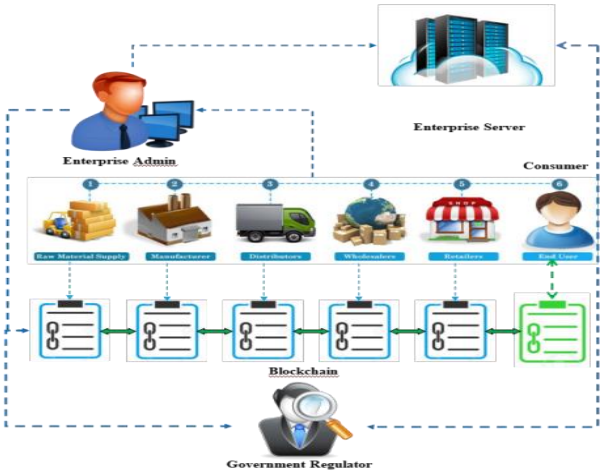
PROPOSED SYSTEM

Data gathering and distillation node, that scans a secret code is called as a ‘terminal.” The transaction is confirmation based on the consensus of cooperate terminals, the transaction is converted into a ‘block’ and included in the blockchain technique. The manager is responsible for policy making and processing requests based on consensus with other nodes. Finally, there exists a third type of node, called, ‘agent’, that requests information about a secret ID from the blockchain technique by providing a proper cyber address. ‘Address collision’ is referred to the existence of a minimum of two identical cyber, or physical addresses. A typical Medical product based supply chain is each packaged food product with an embedded secret ID travels through multiple stages of transactions at different terminals starting from packaging through shipment, cache and finally to a consumer for purchase. A data block be created containing the information about the package at each valid transaction. Here, once the transaction is verified, the transaction of the confidential ID is converted into a block of information and appended to its preexisting data blocks, thus, forming a chain of information blocks and thus a blockchain technique

Advantage

- Real time tracking and sensing of food products throughout the MPSC, and allowing identification of key bottlenecks.
- Discouraging adulteration of Medical products, and identifying weak links on occurrence.
- Determining the shelf life of Medical products leading to reduced waste.
- Providing end to end information accurately.
- Allowing specific and targeted recalls.

Architecture Diagram



Modules

1. Enterprise: The main needs of enterprises in the food data shared supply chain are:

1)the specific accessibility of their on the blockchain must be assured to prevent the leakage of sensitive information and to provide confidentiality.

2) The maintenance cost of blockchain system should be appropriately controlled. Only by satisfying the above needs will this system truly benefits enterprises.

1. Consumer: For consumers, the most basic and essential requirement of the system is to provide traceability for the product, they purchased. The characteristic of data according to the demand of consumer ought to be tamper-proof as well as confidential. Additionally, the system needs to be available for the public by the concise and low-cost design.

2. Government Regulator: As for the demand of government regulators, we should provide the highest accessibility to them to monitor all data on the traceability system in order that they can pinpoint the culpable sector as soon as possible once the food safety event occurs. Also, they should have capability to ensure that all data uploaded by the enterprise is legal and verified.

1. Enterprise-user server

1.1. ***Traceability Information Capture Module:*** This module is designed to collect key traceability information brought forth by the process of production, storage, circulation of food. It can work automatically and manually to identify and create detailed event information from the circulation of food in the supply chain.

1.2. ***Event Information Database:*** This database is mainly used for the preservation and management of all food information from the capture module.

1.3. ***Information Extraction Module:*** This module is primarily devised for extracting information that needs to be uploaded on blockchain from the traceability information database as well as preparing the data for the uploading.

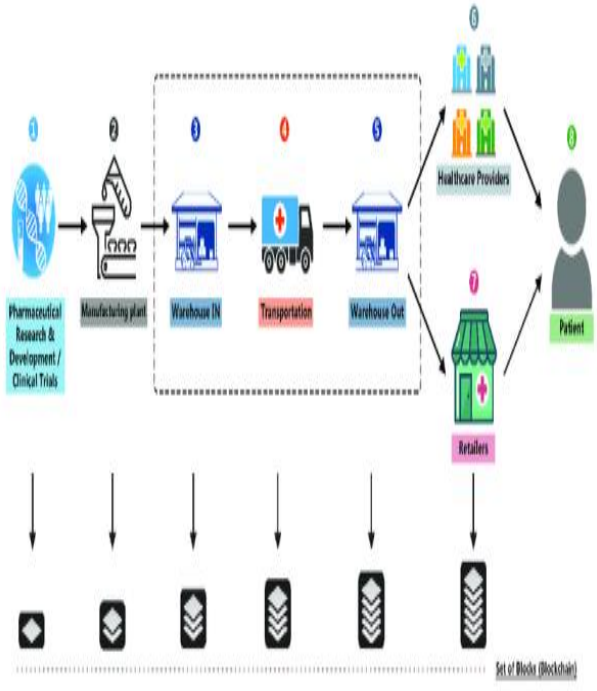
1.4. ***Blockchain Module:*** Blockchain module has two functions. One is the data interaction including the upload of key traceability information on blockchain, the request of on-chain information and the verification of event information. The other is to provide options for users to be the full blockchain node or the light-weight blockchain node i.e. to decide whether or not to participate in the maintenance of the blockchain.

1.5. ***Interaction Authority Management Module:*** This module is in charge of the verification of enterprise identity when there is any event information interaction i.e. to determine whether the requester who initiates the request for event information is in this supply chain.

2. Consumer Traceability Client

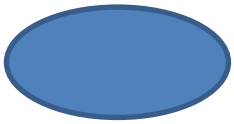
2.1. ***Blockchain Module:*** This module is designed for the link between the client and system, through which it can request information on the blockchain and verify the legitimacy of the information. A light node is chosen for this module to lower user's maintenance cost.

2.2. ***Information Cache Database:*** This cache database is built to cache the corresponding food traceability data requested by users.



Data Flow Diagram

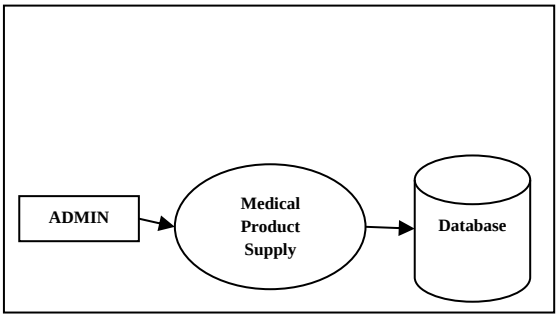
A data-flow diagram is a way of representing a flow of a data of a process or a system. The Data Flow Diagram(DFD) also provides information about the outputs, and inputs of each entity and the process itself. The data flow diagram has no control flow, there are no decisioning rules, and no loops.

Symbol	Description
	An **entity**. A source of data or a destination for data.
	A **process** or task that is performed by the system.
	A **data store**, a place where data is held between processes.
	A **data flow**.

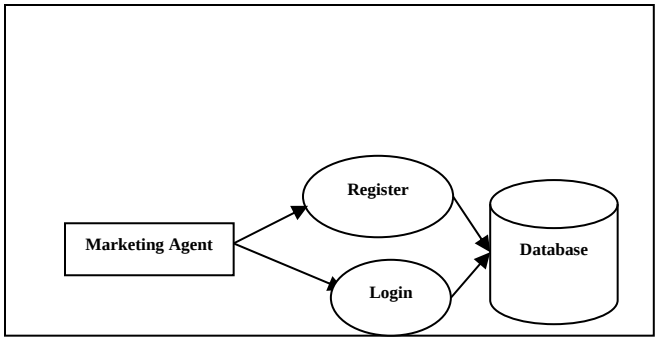
Visual depiction makes it a good contact tool between User and System designer. structure of data flow diagram allows starting from a wide overview and expand it to a hierarchy of detailed diagrams. DFD has often been used due to the following reasons: Determination of physical system construction requirements.

Data flow Symbols

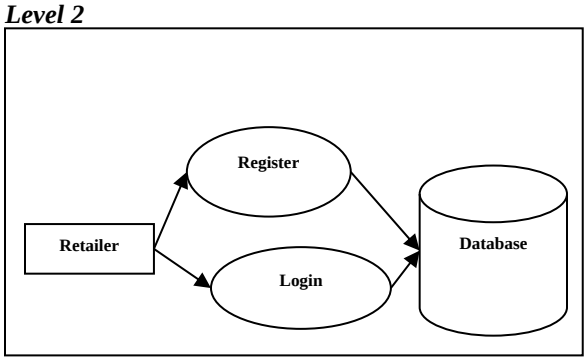
Level 0



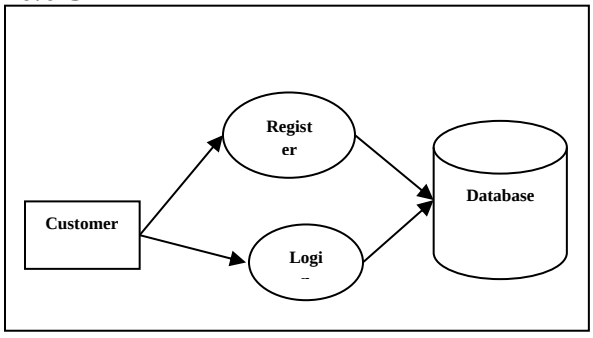
Level 1



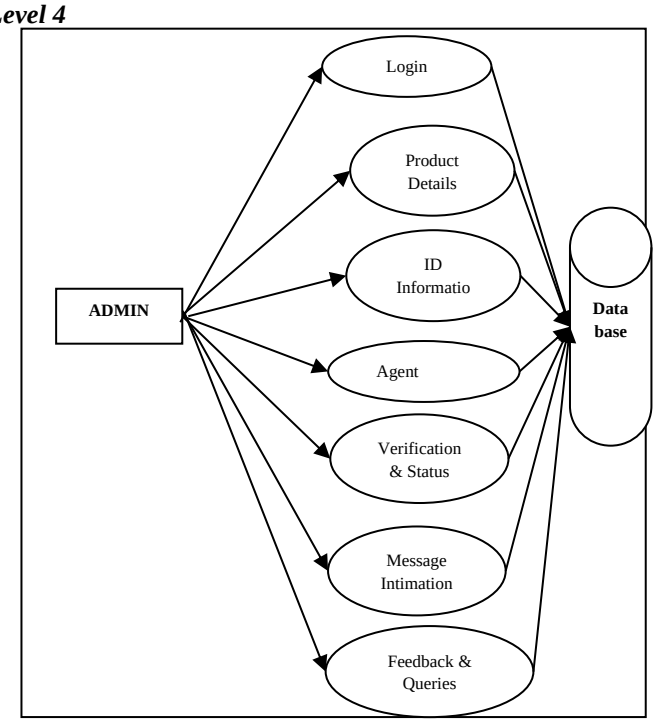
Level 2



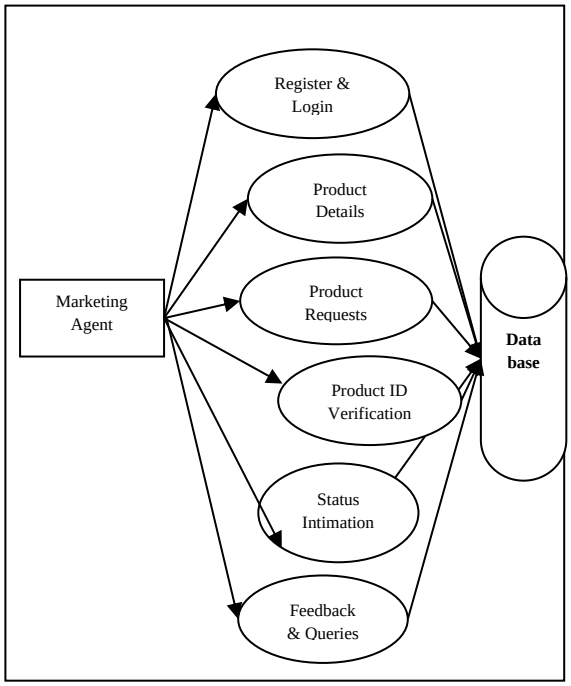
Level 3



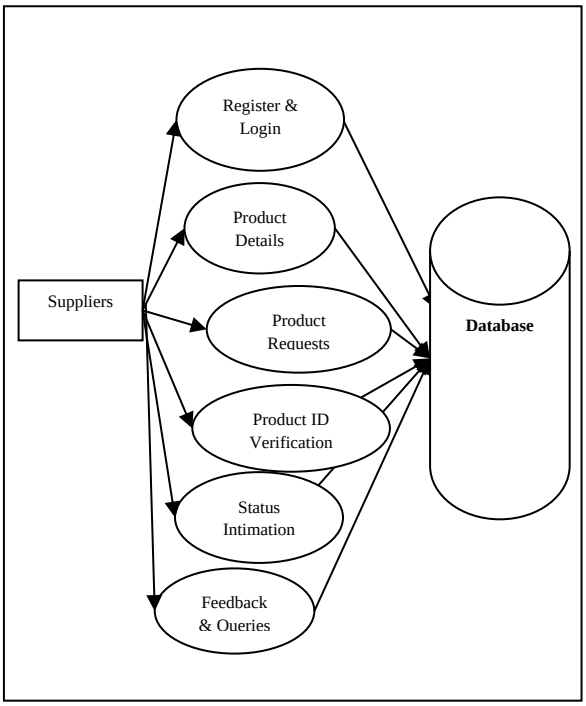
Level 4



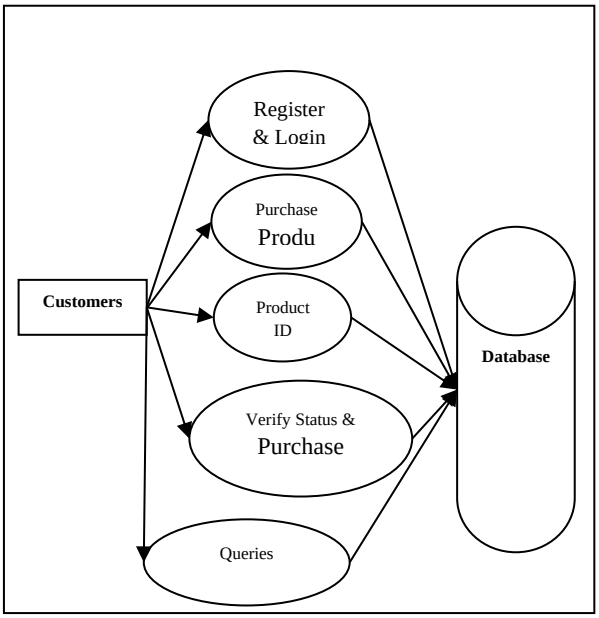
Level 5



Level 6



Level7:

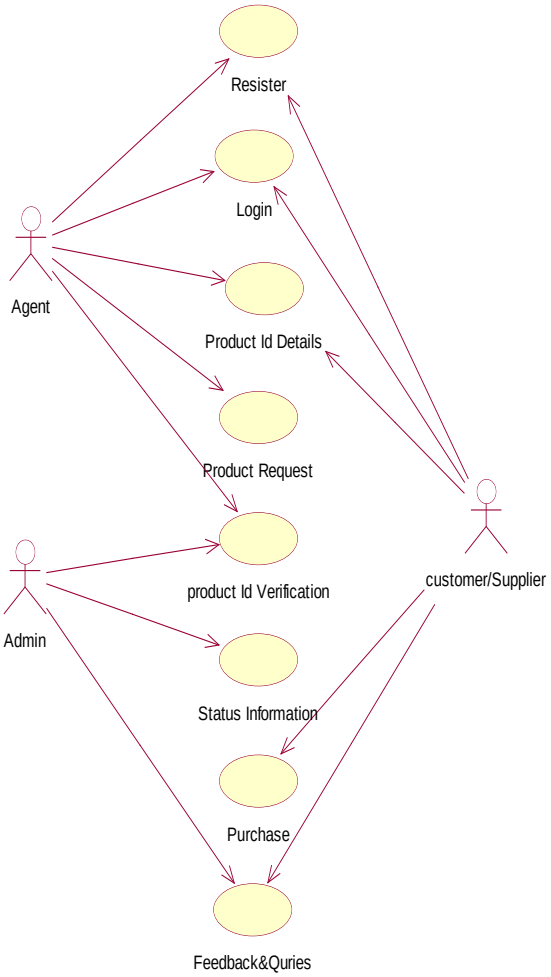


Use Case Diagram

The use case diagrams are usually referred to as behavior diagrams used to report a set of actions (use cases) that some system or subject should or can perform in collaboration with one or more outer users of the system.

The use case diagram at its easy is a representation of a user's interaction with the system that shows the correspondence

between the user, and the different use cases in which the user is involved.



Class Diagram

The main building block of object-oriented modelling are class diagram. It is used for accustomed conceptual modeling of the systematic of the application, and for comprehensive modeling translating the models into programming code. The class diagrams can also be accustomed to data modelling.

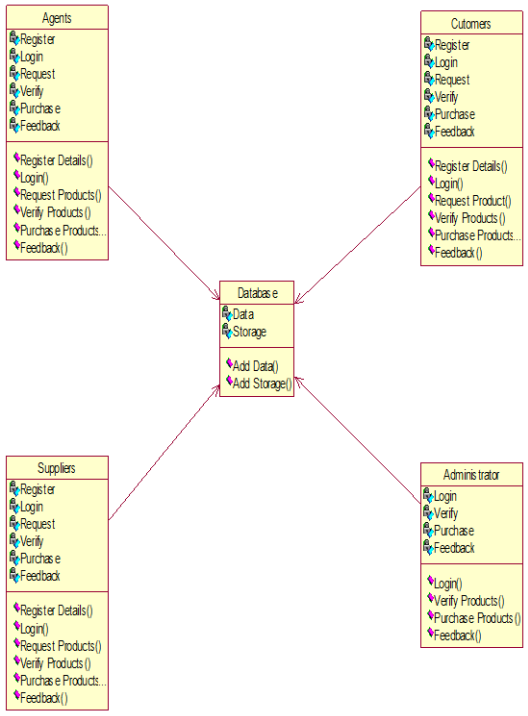
1. In the diagram, classes are represented with boxes that contain three compartments:

2. The top compartment contains the name of the class.

3. The middle compartment contains the attributes of the class.

4. The bottom compartment contains the operations the class can execute.

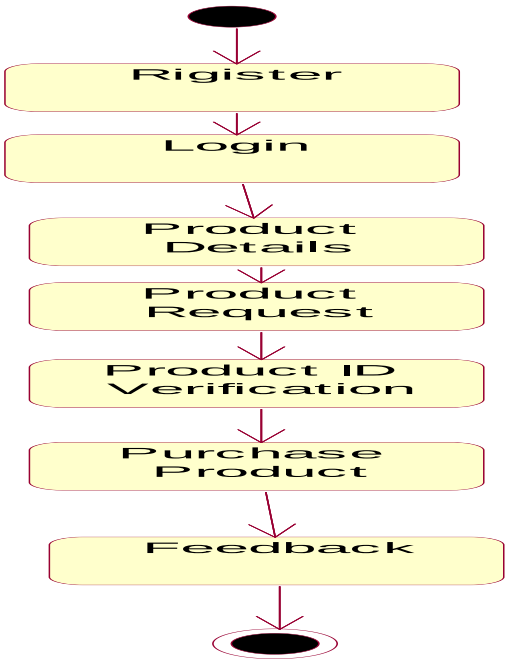
Here, the blueprint of a system, a number of classes are discovered and associate in a class diagram, that helps to determine the static relations between them. The detailed modelling conceptual design is often split into a number of sub classes.



Activity Diagram

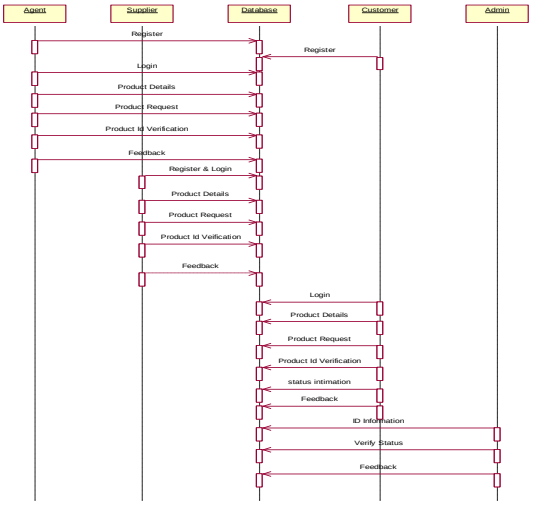
Activity diagram visually presents a sequence of actions or flow of control in a system similar to a data flow diagram. The activity diagrams are often accustomed to business process modelling. Here, the activities modeled can be sequential and simultaneous.

Basic purposes of activity diagrams are related to other four diagrams. It traps the dynamic behavior of the system. Remaining four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another. The activity diagrams are built from a limited number of shapes,connected with arrows.



Sequence Diagram

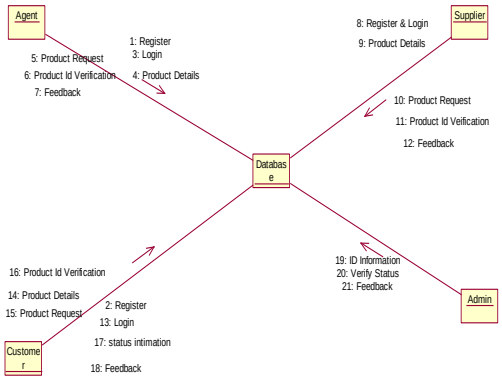
The sequence diagram may be a good diagram to used to document a system's requirements and to flush out a system's design. The rationale the sequence diagram is so useful is because it shows the interaction logic between the objects within the system within the time order that the interactions happen. Sequential diagram display, as parallel vertical lines, various processes or objects that lice concurrently, and as horizontal arrows, the messages transformed between them, within the order during which they occur. This enables the specification of straightforward run time scenarios during a graphical manner.



Collaboration Diagram

The collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the affair and intercommunication among software objects within the Unified Modelling Language. Unified Modeling Language Collaboration diagrams illustrate the connection and interaction between software objects.

They require use cases, system operation contracts, and domain model to exist already. The collaboration diagram explain messages being sent between classes and instances. The communication diagrams model the interplay between objects in sequence. They describe both the static structure, and therefore, the dynamic behavior of a system. In some ways, a communication diagram may be a simplified version of a collaboration diagram introduced in UML 2.0.



Implementation framework

In this section, we describe the algorithms that define the working principles of our proposed blockchain-based approach. As discussed earlier, the customer creates the smart contract. The customer then agrees to the purchase terms (offline) with one of the registered medicine companies.

Algorithm 1 describes the process that govern the sale of medicine by the medicine company to the customer. After the initial state of the contract is established, the smart contract checks to confirm that the requesting customer is already registered and the price of medicine is paid. If the scenario is successful, then the state of the contract changes to Medicine*RequestSubmitted*,

the customer state changes to *WaitForMedicine* and state of medicine company changes to *AgreeToSell*. The contract notifies all the active entities in the chain about the state changes otherwise the state of contract and other active participants reverts to initial state and transaction terminates.

Algorithm 1:Medicine company Sells medicine to customer

Input: *F* is the *list of* registered customers

Ethereum address(EA) of customer.

Ethereum address(EA) of medicine Company

Quantity, medicine Type, medicine WE and,Medicine Price

1 Contract state is ***Created***

2 State of the customer is **medicine *Requested***

3 Medicine Company state is ***Ready***

4 Restrict access to only $f \in F$ i.e., registered Customer

5 if customer= *registered and Medicine Price = paid* **then**

6 Contract state changes to Medicine *Request Submitted.*

7 Change State of the customer to *Wait For Medicine.*

8 Medicine Company state is *AgreeToSell*

9 Create a notification message stating condition of medicines

10 end

11 else

12 Revert contract state and show an error.

13 end

Algorithm 2: Medicine Processor Buys medicine From company

Input: *'gp' is the list of registered Processors*

Ethereumaddress(EA) of MedicineProcessor,

Ethereumaddress(EA) of company Quantity,

DatePurchased, Medicine*Price*

1 Contractstate is ***BuyFromCompany***

2 State of the medicine processor is **Medicine*Requested***

3 Medicine company state is **Medicine*BoughtFromCompany***

4 Restrict access to only *gp* ∈ Medicine*Processor*

5 if Medicine*Sale is agreed and MedicinePrice = paid* **then**

6 Contract state changes to Medicine*RequestAgreed.*

7 Change State of the medicine processor to *WaitForMedicineFromCompany.*

8 Medicine Company state is *SellMedicineToProcessor*

9 Create a notification message stating sale of medicine to requesting processor

10 end

11 else

12 Contract state changes to Medicine*RequestFailed.*

13 State of medicine processor is *RequestFailure.*

14 Medicine Company state is *CancelRequestOfProcessor*

15 Create a notification message stating request failure

16 end

17 else

18 Revert contract state and show an error.

19 end

to check two conditions as shown in Algorithm 2: (i) if the requesting medicine processor is a registered entity and (ii) if the sale of medicine is agreed and purchase price is paid. If these two conditions are true or satisfied, the contract state changes to Medicine*RequestAgreed*, processor state changes to *WaitFor MedicineFromCompany*, company state changes to *SellMedicineTo Processor*, and all the active entities are notified with a message on the sale of medicine to the processor. In the other case, if the above mentioned two conditions are not satisfied, contract state changes toMedicine*RequestFailed*, processor state changes to *RequestFailure*, company state changes to *CancelRequestOfProcessor*.

The medicine processor then sells the finished product to the distributors. Next, we elaborate the state of the system and

the entities where the retailer buys the product from distributor.

Date of product manufacture, Quantity Sold, and Date of Purchase are some of the important parameters to keep a

check. The distributors and retailers will be identified with their Ethereum addresses and state of the contract as shown

in Algorithm 3. At this stage, the contract state is *Product*

SoldToDistributor, and distributor state is *ProductReceived*

FromProcessor. The state of the retailer is *ReadyToPurchase*.

The contract restricts the access to only registered retailers

and checks if sale agreement is accepted and product payment

is completed. If these conditions are met, the contract executes the transaction where the distributor ships the product to the retailer. Here, the state of the contract changes to

---

Algorithm 3: Distributor Ships Product to Retailer

Input: *'r' is the list of registered Retailers*

Etherenumaddress(EA) of Distributor,

Ethereumaddress(EA) of Retailer,

DateManufactured, Quantity Sold,

DatePurchased

1 Contractstate is ***ProductSoldToDistributor***

2 Distributor state is ***ProductReceivedFromProcessor***

3 *i* Retailer state is ***ReadyToPurchase***

4 Restrict access to only *r*e-Jietatler

5 if *Sale = agreed and ProductPayment = successful* **then**

6 Contract state changes to

SaleRequestAgreedSuccess.

7 Distributor state changes to *ProductSoldToRetailer.*

8 Retailer state is *ProductDeliveredSuccessful*

9 Create a 'success' notifification message.

10 end

11 else

12 Contract state changes to *SaleRequestDenied.*

13 Distributor state changes to *RequestFailed.*

14 Retailer state is *ProductDeliveryFailure*

15 is Create a request failure notification message.

16 end

17 else

18 Revert contract state and show an error.

19 end

SaleRequestAgreedSuccess, and the distributor state changes

to *ProductSoldToRetailer*, and Retailer state changes to *ProductDeliveredSuccessful*. For a successful product delivery

done, the contract sends out a notification message stating the

successful delivery to the retailer. Else, for a failure scenario, the contract state changes to *SaleRequestDenied*, state of distributor becomes *RequestFailed* and retailer state changes to

ProductDeliveryFailure and the failure notification message is sent out to all participants. Finally, we describe the algorithm for purchases made by the customer from the retailer in Algorithm 4. The customer is the final entity in the medicine processing and tracking model. The customer state is *ReadyToBuy* initially. The state of the contract and retailer are *SaleRequestAgreed Success* and *ProductDeliveredSuccessful* respectively.

Here, the smart contract restricts access to only Customers to

make a purchase from the registered retailers. The important parameters considered to track the product are Customer Ethereum address, Retailer Ethereum address, Date Purchased, Sales ID, Product ID. Upon successful payment of the product price, the state of contract changes to *ProductSol*

dToCustomer, retailer state changes to *ProductSaleSuccessful* and customer state changes to *SuccessfulPurchase*. If the payment made is not correct, the state of contract changes to *SaleOfProductDenied*, retailer state changes to

ProductSaleFailure and customer state changes to FailedPurchase.

The contract notifies with an event about the sales failure to everyone in the network.

Algorithm 4:

Customer Buys From Retailer

Input:Ethereum address(EA) of Retailer,

Ethereum address(EA) of Customer, Date Purchased,Product ID. Sales ID **1.**Contract state is **SaleRequest AgreedSuccess**

2.Retailer state is **ProductDelivered Successful**

3.Customer state is **ReadyToBuy**

4.Restrict access to only Customers

5.**if** ProductPayment=successful **then**

6.Contract state changes to **ProductSoldToCustomer.**

7.Retailer state is **ProductSaleSuccessful**

8.Customer state is Successful Purchase

9.Create a 'purchase success' notification message.

10.end

11.else

12. Contract state changes to SaleOfProductDenied.

13. Retailer state is ProductSaleFailure

14. Customer state is FailedPurchase **15.**Notify with a 'purchase failure' message.

16.end

17.else

18. is Revert contract state and show an error.

19.end

System Specification

Hardware Requirements:

The hardware must-haves may serve as the support for a contract for the application of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design

- Processor : Intel processor 3.0 GHz
- RAM : 2GB
- Hard disk : 500 GB
- Compact Disk : 650 Mb
- Keyboard : Standard keyboard
- Mouse : Logitech mouse
- Monitor : 15 inch color monitor

Software Requirements:

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is useful in estimating cost, planning team activities and performing tasks throughout the development activity.

Software Requirements:

- Operating system : Windows OS (XP, 2007, 2008)
- Front End : JSP
- IDE for JAVA : Net beans 7.1
- Back End : My SQL 5.0.5 1b
- IDE for MYSQL : Wamp server 2.2.

Software Specification:

JSP Introduction

Java Server Pages or JSP for brief is Sun's solution for developing dynamic internet sites. The java server page to provide excellent server side scripting carry for creating database driven web applications. Java Server Page(JSP) enable the designer to directly insert java code into file java server page, this makes the event process very simple and its maintenance also becomes very easy. JSP pages are efficient, it loads into the online servers' memory on receiving the request very first time, and therefore, the subsequent calls are served within a really short period of your time.
In today's environment most internet sites servers dynamic pages supported user request. Database is extremely convenient thanks to store the info of users and other things. Java Data Base Connectivity(JDBC) to provide an excellent

database connectivity in heterogeneous database environment. Using Java Server Page(JSP) and Java Data Base Connectivity (JDBC) its very easy to develop database driven web application. Java is understood for its characteristic of "write once, run anywhere." JSP pages are plat Java Server Pages.

Evolution of Web Applications

Over the previous couple of years, web server applications have evolved from static to dynamic applications. This evolution became necessary thanks to some deficiencies in earlier internet site design. For instance, to place more of business processes on the online, whether in business-to-consumer (B2C) or business-to-business (B2B) markets, conventional internet site design technologies aren't enough. The most issues, every developer faces when developing web applications, are:

1.*Scalability* — a successful site will have more users and because the number of users is increasing fast, the online applications need to scale correspondingly.

2.*Integration of knowledge and business logic* — the online is simply differently to conduct business, then it should be ready to use an equivalent middle-tier and data-access code.

3.*manageability* — internet sites just keep getting bigger, and that we need some viable mechanism to manage the ever-increasing content and its interaction with business systems.

4.*Personalization* — adding a private touch to the online page becomes an important factor to stay our customer returning again. Knowing their preferences, allowing them to configure the knowledge they view, remembering their past transactions or frequent search keywords are all important in providing feedback and interaction from what's otherwise a reasonably one-sided conversation.

Servlets

Earlier in client- server computing, each application had its own client program and it worked as a interface and wish to be installed on each user's pc . Most web applications use HTML/XHTML that's mostly supported by all the browsers and sites are showed the client as static documents. an internet page can merely displays static content and it also lets the user navigate through the content, but an internet application provides a more interactive experience.

Any computer running Servlets or JSP must have a container. A container is nothing but a bit of software liable for loading, executing and unloading the Servlets and JSP. The servlets are often wont to expand the functionality of any Java-enabled server.They are mostly wont to extend web servers, and are efficient replacement for CGI scripts. CGI was one among the earliest and most prominent server side dynamic content solutions, so before going forward it's vital to understand the difference between CGI and therefore the Servlets.

MySQL

MySQL is the world's most used open source electronic database management system (RDBMS) as of 2008 that run as a server providing multi-user access to variety of databases. The MySQL development project has made its ASCII text file available under the terms of the GNU General Public License, also as under a spread of proprietary agreements. MySQL was owned and sponsored by one for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is the best choice of database to be used in web applications, and may be a central component of the widely used LAMP open source web application software stack—LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that need a full-featured management system often use MySQL.

The economical use, some paid editions are obtainable, and offer additional features. Here, the applications use MySQL databases include: TYPO3, Joomla, Word Press, phpBB, MyBB, Drupal and other software built on the LAMP software stack. MySQL is additionally utilized in many high-profile, large-scale World Wide Web products, including Wikipedia, Google(though not for searches), ImagebookTwitter, Flickr, Nokia.com, and YouTube.

- ***Inter images***

The MySQL is mainly an RDBMS and ships with no Graphical User Interface tools to administer MySQL databases or manage data contained within the databases. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available to be used.

- ***Graphical***

The official MySQL Workbench may be a free integrated environment developed by MySQL AB, that permits users to graphically administer MySQL databases and visually design database structures. The MySQL Workbench replaces the preceding package of software and MySQL Graphical User Interface tools. All the third-party packages, treated as accurate MySQL front end, MySQL Workbench lets users manage database design & modeling, SQL development (replacing MySQL Query Browser) and Database administration (replacing MySQL Administrator).

MySQL Workbench is out there in two editions, the regular free and open source Community Edition which can be downloaded from the MySQL website, and therefore the proprietary Standard Edition(SE) extends and improves the feature set of the Community Edition(CE).

CONCLUSION

An Blockchain based MPSC monitoring architecture has been proposed in this work. Sensing modality was integrated with identification with a small footprint for tracking and quality monitoring of the Medical product packages. When the Medical Product packages are scanned at different retailers, logistics or storage stage within the supply chain, the real time sensor data is updated in a blockchain providing a tamper-proof digital history. Any consumer or retailer can check the public ledger to obtain information regarding the specific medical product packages. The information helps in updating the shelf life, identifying key bottlenecks in the MPSC, implementing targeted recalls and moreover increasing visibility. A single secret ID integration was demonstrated in this work. The proposed architecture takes consensus from participating terminals in the network before updating the blockchain data. The broader participation of all the nodes helps to keep the network decentralized. The security analysis showed that the validation of a fake block drops with a higher number of node participation in the network and multiple consensus stages.

REFERENCES

[1] M. M. Aung and Y. S. Chang, ``Traceability in a food supply chain: Safety and quality perspectives,'' *Food Control*, vol. 39, pp. 172_184, May 2014.
[2] T. Bosona and G. Gebresenbet, ``Food traceability as an integral part of logistics management in food and agricultural supply chain,'' *Food Control*, vol. 33, no. 2, pp. 32_48, 2013.
[3] J. Hobbs, ``Liability and traceability in agri-food supply chains,'' in *Quan- tifying the Agri-Food Supply Chain*. Springer, 2006, pp. 87_102.
[4] D. Mao, Z. Hao, F. Wang, and H. Li, ``Novel automatic food trading system using consortium blockchain,'' *Arabian J. Sci. Eng.*, vol. 44, no. 4, pp. 3439_3455, Apr. 2018.
[5] L. U. Opara and F. Mazaud, ``Food traceability from _eld to plate,'' *Outlook Agricult.*, vol. 30, no. 2, pp. 239_247, 2001.
[6] F. Dabbene and P. Gay, ``Food traceability systems: Performance evaluation and optimization,'' *Comput. Electron. Agricult.*, vol. 75, no. 2, pp. 139_146, 2011.
[7] J. Storoy, M. Thakur, and P. Olsen, ``The TraceFood framework_ Principles and guidelines for implementing traceability in food value chain,'' *J. Food Eng.*, vol. 115, no. 2, pp. 41_48, 2013.
[8] M. A. Khan and K. Salah, ``IoT security: Review, blockchain solutions, and open challenges,'' *Future Gener. Comput. Syst.*, vol. 82, pp. 395_411, May 2018.
[9] L. Lucas. Financial Times. (2018). *From Farm to Plate, Blockchain Dishes Up Simple Food Tracking*. Accessed: Jun. 12, 2018. [Online]. Available: https://www.ft.com/content/225d32bc-4dfa-11e8-97e4-13afc22d86d4
[10] A. Bogner, M. Chanson, and A. Meeuw, ``A decentralised sharing app running a smart contract on the Ethereum blockchain,'' in *Proc. 6th Int. Conf. Internet Things*, 2016, pp. 177_178.
[11] K. Salah, M. Rehman, N. Nizamuddin, and A. Al-Fuqaha, ``Blockchain for AI: Review and open research challenges,'' *IEEE Access*, vol. 7, pp. 10127_10149, 2019.
[12] H. Hasan and K. Salah, ``Combating deepfake videos using blockchain and smart contracts,'' *IEEE Access*, vol. 7, no. 1, pp. 41596_41606, Dec. 2019.
[13] R. Beck, J. S. Czepluch, N. Lollike, and S. Malone, ``Blockchain-the gateway to trust-free cryptographic transactions,'' in *Proc. ECIS*, May 2016, p. 153.
[14] M. E. Peck, ``Blockchains: How they work and why they'll change the world,'' *IEEE Spectr.*, vol. 54, no. 2, pp. 26_35, Sep. 2017.
[15] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, ``A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain,'' *IEEE Access*, vol. 5, pp. 17465_17477, 2017.