Garm: A Blockchain based platform for supply chain management

Garm: Uma Plataforma baseada em Blockchain para gerenciamento de cadeia de suprimentos

DOI:10.34117/bjdv6n12-587

Recebimento dos originais: 22/11/2020
Aceitação para publicação: 22/12/2020

Edivaldo Mascarenhas Ferreira de Jesus Júnior
Specialist on Ubiquitous and Distributed Computing
Instituto Federal da Bahia (IFBA), Salvador, Bahia, Brazil; Master Student
Graduate Program in Systems and Products Engineering (PPGESP-IFBA)
Endereço: Rua Emídio dos Santos, IFBA, GSORT
E-mail: edivaldojunior@ifba.edu.br

Manoel Carvalho Marques Neto
Doctoral degree in Computer Science
Instituto Federal da Bahia (IFBA), Salvador, Bahia, Brazil; Professor
Graduate Program in Systems and Products Engineering (PPGESP-IFBA)
Endereço: Rua Emídio dos Santos, IFBA, GSORT (sala A-403), 40.301-015
E-mail: manoelnetom@ifba.edu.br

Allan Edgard Silva Freitas
Doctoral degree in Computer Science
Instituto Federal da Bahia (IFBA), Salvador, Bahia, Brazil; Full Professor
Graduate Program in Systems and Products Engineering (PPGESP-IFBA)
Endereço: Rua Emídio dos Santos, IFBA, GSORT (sala A-403), 40.301-015
E-mail: allan@ifba.edu.br

ABSTRACT
A complex web of relationships provides goods for manufacturing, assembling and delivering of final products known as supply chain. Emerging technologies have been used in supply chain systems in order to provide traceability. However, these systems tend to be centralized, monopolistic, asymmetric and opaque. As a con- sequence, these systems may result in trusting problems, such as fraud, corruption and tampering. Blockchain technology provides a new approach for information systems based on decentralization, that can apply for these supply chain systems. This work presents a Blockchain-based platform for developing applications that provide such traceability for Supply Chain Management.

Keywords: Blockchain, supply chain management, traceability, smart contract.

RESUMO
Uma complexa rede de relacionamentos fornece mercadorias para fabricação, montagem e entrega de produtos finais, conhecida como cadeia de suprimentos. Tecnologias emergentes têm sido usadas em sistemas de cadeia de suprimentos para fornecer rastreabilidade. No entanto, esses sistemas tendem a ser centralizados, monopolísticos, assimétricos e opacos. Como consequência, esses sistemas podem resultar em problemas de confiança, como fraude, corrupção e adulteração. A tecnologia Blockchain

fornece uma nova abordagem para sistemas de informação com base na descentralização, que pode ser aplicada a esses sistemas de cadeia de suprimentos. Este trabalho apresenta uma plataforma baseada em Blockchain para o desenvolvimento de aplicações que fornecem tal rastreabilidade para o Gerenciamento da Cadeia de Suprimentos.

Palavras-chave: Blockchain, gerenciamento de cadeia de suprimentos, rastreabilidade, contratos inteligentes.

1 INTRODUCTION

Hundreds of years ago, supply chains were reasonably straightforward. Mines and farms provided natural resources to skilled craftsman like blacksmiths and tailors who then created and sold goods. Nowadays, supply chains are much more complicated, fragmented and difficult to understand. Most of the time, the various companies don't know about each other and a final consumer likely don't know anything about how, where, when or under what condition the products passed through. This isn't just a problem for consumers. Today's supply chains are so complex that even big industry players have difficulty tracking how their goods get made (Swan, 2015).

In order to solve some problems that come with this complexity, such as supply chain visibility and traceability, many systems have been developed. However, these systems typically store information in standard databases controlled by service providers. This centralized data storage becomes a single point of failure and risks tampering. As a centralized organization, it can become a vulnerable target for bribery, and then the whole system can not be trusted anymore (Tian, 2017).

Blockchain and smart contracts could make supply chain management more straightforward and more transparent. The idea is to create a single source of information about products and supply chain via global ledger. Each component would have its own entry on the blockchain that gets tracked over time. Both untrust companies could then update the status of a good in real-time. The end result is once the clients receive their products they could track every piece back to its manufacturer (Greve et al., 2018).

Companies can also use the blockchain supply chain as a single source of truth for their products. They can manage and monitor risks within the supply chain, ensure quality of delivery and track the status of all components. Additionally companies can use smart contracts to manage and pay for supply chain autonomously (Tian, 2017).

This would reduce the need for large contract invoices on the back-and-forth of refund requests for faulty components. Those same smart contracts could assist with shipping and logistics tracking valuable products as they travel around the world. Companies using blockchain can finally have a complete picture of their products at every stage in the supply chain, bringing transparency to the production process while reducing the cost of manufactured goods (Swan, 2015).

This work presents Garm, a pure web based platform intended to be used in any kind of supply chain correlated to assets and products. The platform is in line with the topic of interest "Distribution, sharing and content security".

This paper is structured as follow: Section 2 presents several technologies involved in the preparation of this article, aimed to introduce important concepts of Computer area in which the context of this project is inserted. Section 3 presents business concepts related to supply chain management. Section 4 shows correlated works. Section 5 presents the solution and its architecture. Section 6 exposes details of the executed implementation. Section 7 exposes the validation and its results. The last section presents the conclusions.

2 BLOCKCHAIN

While the system of financial institutions that serve as third parties reliable processors for processing payments work well for most still suffer from the shortcomings inherent in the model based on confidence. Also, the cost of mediation increases transaction costs, which limits the minimum practical size of the transaction and eliminates the possibility of occasional small transactions. To solve these problems, (Nakamoto, 2008) defined an electronic payment system called Bitcoin, based on cryptographic proof rather than reliable, allowing either party willing to transact directly with each other without the need to a responsible third party.

Blockchain can be considered as a public ledger, in which all committed transactions are stored in a blockchain (Zheng et al., 2016). For (Swan, 2015), besides the currency ("Blockchain 1.0"), smart contracts ("2.0") demonstrate how the Blockchain is in a position to become the fifth disruptive computing paradigm after mainframes, PCs, Internet and mobile/ social networks. Smart contracts mediate registering in a public ledger through an application executed in Blockchain itself.

Blockchain technology has critical features, such as decentralization, persistence, anonymity and auditability. Blockchain can function in a decentralized environment that is activated by the integration of several key technologies such as cryptographic hash, digital signature and distributed consensus engine, significantly save the cost and improve efficiency (Zheng et al., 2016).

2.1 PUBLIC BLOCKCHAIN VERSUS PRIVATE BLOCKCHAIN

On a public blockchain, any person can participate without a specific identity. Public blockchains typically involve a native cryptocurrency and often use proof-of-work (PoW) consensus and economic incentives (Androulaki et al., 2018). Public Blockchain can be audited by anyone, and each node has as much transmission power as any other. For a transaction to be considered valid, it must be authorized by all nodes constituents via the consensus process. As long as each node meets protocol-specific stipulations, their transactions can be validated and thus added to the chain (Greve et al., 2018).

Private blockchains, on the other hand, perform a blockchain between a set of known and identified participants. A private blockchain provides a way to protect the interactions between a group of entities that have a common goal but that don't totally trust each other, like companies that trade funds, assets or information. Relying on peer identities, one private Blockchain may use the traditional consensus of Byzantine fault tolerance (BFT) (Androulaki et al., 2018).

2.2 SMART CONTRACTS

A smart contract is a computerized transaction protocol that executes the terms of a contract (Szabo,1997). The term smart contract (SC) means: "an internal transaction protocol format that executes the terms of a contract. Their overall goals are to ensure common contractual conditions, minimize malicious and accidental exceptions and the need for reliable intermediaries. Related economic objectives include reducing fraud losses, arbitration and execution costs, and other transaction costs." (Szabo,1997).

Smart contracts are created as scripts, stored in with private addressing on the Blockchain itself. They are triggered when addressing a transaction to it. Then the script is executed independently and automatically, as prescribed in all nodes in the network according to the data included in the transaction (Greve et al., 2018). Smart contracts interpret the code objectively - "The Code is the law".

3 SUPPLY CHAIN MANAGEMENT

Billions of products have been manufactured every day through complex supply chains that can extend to all parts of the world. However, tracing good flows from harvesting and manufacturing to the final consumer is hard. To achieve this flow, we need to improve Supply Chain visibility (Galvez et al., 2018). Traceability is one of the key challenges encountered in the business world, with most companies having little or no information about their own second and third-tier suppliers. Transparency and end-to-end visibility of the supply chain can help shape product, raw material, test control, and end product flow, enabling better operations and risk analysis to ensure better chain productivity (Abeyratne and Monfared, 2016).

Traceability systems typically store information in standard databases controlled by service providers. This centralized data storage becomes a single point of failure and tampering risks. As a consequence, these systems result in trusting problems, such as fraud, corruption and tampering. Likewise, as a single point of failure, a centralized system is vulnerable to collapse (Tian, 2017).

Nowadays, Blockchain presents a whole new approach based on decentralization, enabling end-to-end traceability, allowing consumers to access the asset's history of these products through a software application (Galvez et al., 2018).

SCM requires to control who can write and read data to/from the Blockchain. In order to do that, the first step is identity. In the SCM context, the peers are known, and the system needs to know who a user is, to define rules about what data they can commit, and what data they can consume from the ledger. So, in a corporate case scenario, Blockchain for the business, Blockchain for supply value chains, a private Blockchain provides this needed characteristic.

4 RELATED WORK

The rapid growth of internet technologies allowed the onset of lots of technologies applied in traceability systems. However, these systems tend to be centralized, monopolistic, and asymmetric. As a consequence, these systems result in trust problems, such as fraud, corruption, tampering, and falsifying information. Likewise, by being a single point of failure, a centralized system is vulnerable to collapse.

Blockchain presents a whole new approach based on decentralization. Nonetheless, by being in its early stages, it has some challenges to deal with, in which scalability and performance become mainly defiance to face the massive amount of data in the real world. Through this technology, some solutions have been raised, as follows.

In order to solve some problems with Supply Chain traceability, many Internet of Things (IoT) technologies, such as RFID and wireless sensor network-based architectures have been applied. However, these technologies do not guarantee that the information shared by supply chain members in the traceability systems can be trusted (Tian, 2017).

In (Tian, 2017), it is proposed a system that combines HACCP (a food safety protocol), Blockchain and IoT in order to provide food safety traceability, involving suppliers, producers, manufacturers, distributors, retailers, consumers and certifiers. Each of these members can add, update and check the information about the product on the Blockchain as long as they register as a user in the system. Each product also has a unique digital cryptographic identifier that connects the physical items to their virtual identity in the system. This virtual identity can be seen as a product information profile.

There are advantages of applying the Blockchain concept to a supply chain. One of the most important is: all stakeholders involved in the supply chain are motivated by the need to demonstrate to customers the superior quality of their methods and products (Lu and Xu, 2017).

In addition to serving the functions of a traceability system, a Blockchain can be used as a marketing tool. As Blockchains are fully transparent and participants can control the assets in them, they can be used to enhance image and reputation of a company (Van Riel and Fombrun, 2007), drive loyalty among existing customers (Pizzuti and Mirabelli, 2015) and attract new ones (Svensson, 2009). In fact, companies can easily distinguish themselves from competitors by emphasizing transparency and monitoring product flow along the chain.

The Everledger Diamonds project provides a Blockchain-based solution to facilitate tracking from mine to consumer, enabling easier compliance against increasingly strict measures from diamonds produced (Crosby et al., 2016).

IBM Food Trust is a pilot project motivated by food contamination scandals worldwide. The main goal is to tackling food safety in the supply chain using Blockchain technology. This platform tracked pork in China and mangoes in the Americas (Kamath, 2018).

These projects are focused on specific products only and are closed projects. Still, there is a general lack of standards for implementation of a Blockchain approach for traceability. A Blockchain must be universal and adaptable to specific situations (Valenta and Sandner, 2017). In addition, the need to agree on a particular type of Blockchain to be used puts the parties under pressure.

Our work is intended to provide a Blockchain-based platform in order to facilitate the development of applications for traceability in supply chain management.

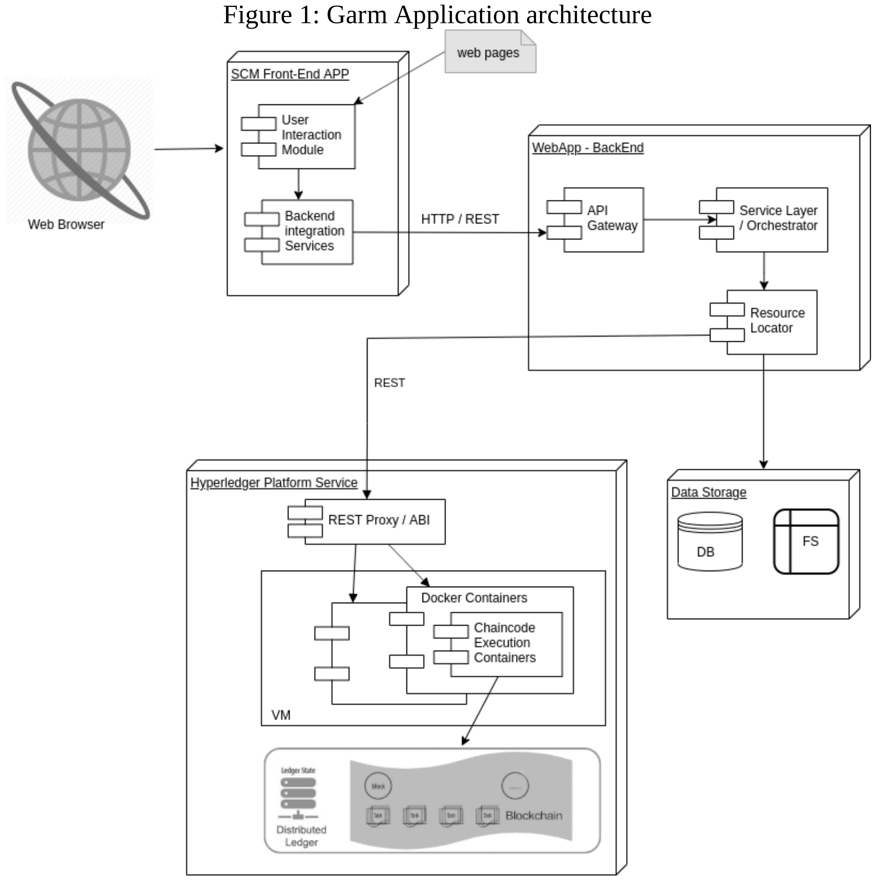
5 PROPOSED SOLUTION

The main objective of this work is to create a generic platform for Supply Chain Management (SCM). Garm project is divided into three main modules described below: Frond-End WebApp, Back-End WebApp and Data Storage. Figure 1 shows the application architecture and its components.

An SCM platform relies on three main items: assets (the goods itself or a token related to it), steps (phases which products go through) and actors (people who transact assets during the steps). Our approach is based on this triad, that must be defined on the creation of a new supply chain.

Initially, a configuration file in JSON format is generated and read in the Blockchain platform, adding the main information for the correct functioning of the chain. The mechanism for creating this configuration file is detailed in the sub-sections 5.1.1 e 5.2.2.

Figure 1: Garm Application architecture



5.1 FRONT-END WEBAPP

Front-End WebApp is a client-server application which the client (including the user interface and client-side logic) runs in a web browser. This is a single-page application (SPA) that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from a server. The application is built with React (also known as React.js or ReactJS). The Front-End Webapp is divided into two main blocks, and these are classified according to the interactions: User Interaction Modules and Backend Interaction Services.

5.1.1 User Interaction Modules

User Interaction modules are responsible for providing web pages that will be rendered on the client's web browser. These interactions are provided by modules described below.

The Login Module is responsible for display the login and authentication alternatives pages (e.g. 'forgot my password', 'reset my password'). The Application Configuration module provides the features of the creation/configuration of supply chain items and supply chain flows (steps). This module is responsible for getting the information from the user to generate the configuration JSON file in the backend. User handling module provides the features for the creation/configuration of Actors and Steps, complementing configuration file. The Data Entry module provides form pages that allow the actors to enter data in the application, search and move assets from a step to another. The Data Visualization module is responsible for displaying the information about assets in the supply chain flow through steps. In the Reporting module users can generate reports/files containing information organized in a narrative, graphic, or tabular form, prepared on ad hoc, periodic, recurring, regular, or as required basis. Reports may refer to specific periods, events, occurrences, or subjects, presented in written form or any other format.

5.1.2 Backend Interaction

Backend interactions happen via a set of services described below.

Authentication service is responsible to request information from an authenticating party, and validate it against the configured identity repository using the specified authentication module. After successful authentication, the user session is activated and validated across all the web application. Application Setup service provides methods to configure and edit supply chain items, and supply chain flows, defining which steps and sub-tasks will be present in this flow and which information will be present in these steps. The User Creation Service is responsible for the creation of users and roles, to allow them to log in and use the application's features. Data Entry service receives data from UI forms and sends them to the backend to be processed and stored. Data visualization services provide information about the supply chain: assets, actors, steps, and entire transactions, to be used by the data visualization module. Report services generate files in different formats (e.g. Doc, PDF, XSL) from a specific period with information about the supply chain.

5.2 BACK-END WEBAPP

WebApp - BackEnd is a Middleware that runs on the server facilitating the client-server connectivity, forming a middle layer between the app and the network: the server, the database, the operating system, and more. It receives requests from the WebApp - FrontEnd and contains the logic to send the appropriate data back to the applicant, over HTTP and REST. Built with Node.js, This module is composed by the API Gateway, Service Layer and Resource Locator more detailed below.

5.2.1 API Gateway

API Gateway is a managed service that enables easily create, publish, maintain, monitor and secure REST APIs to act as a "gateway" for applications to access data, business logic, or functionality in the backend services. The API Gateway provides a simple uniform view of external resources to the internals of an application. It manages all tasks involved in receiving and processing API calls, including traffic management, authorization and access control.

Gateway access can be done from many different devices. Therefore, it must have the power to unify outgoing calls and be able to deliver to the user content that can be accessed from any browser and system. Gateways as a Security Feature: In the APIs world, one of the most subjects talked about issues is always security, and having an API Gateway is one of the best solutions on the market to get full control of APIs because this pattern addresses the so-called CIA (Confidentiality, Integrity, Availability) almost flawlessly.

5.2.2 Service Layer

A Service Layer defines an application’s boundary and its set of available operations from the perspective of interfacing client layers. It encapsulates the application’s business logic, controlling transactions and coordinating responses in the implementation of its operations. This module implements the service layer pattern and provides some benefits:

1. Centralizes external access to data and functions.
2. Hides (abstracts) internal implementation and changes.
3. Allows for versioning of the services.

The service layer acts as an orchestrator, controlling the flow of incoming and outcoming information requests and responses. Orchestration allows to directly link process logic to service interaction within workflow logic. This combines business process modelling with service-oriented modelling and design, realizing workflow management through a process service model. Orchestration brings the business process into the service layer, positioning it as a master composition controller.

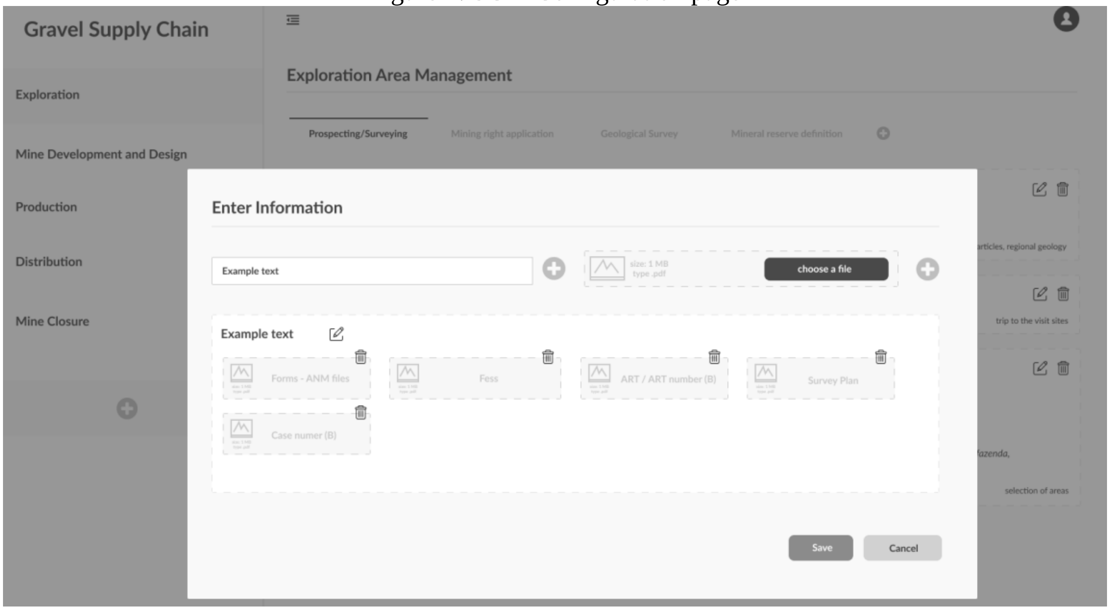
5.2.3 Resource Locator

Resource locators are components that abstract the persistence layer. Their job is to provide an object that can help services to discover and persist information from/to the Data Storage Module. Information can be stored in the Blockchain, Filesystem or Database and resource locators should know exactly where get/put data within them.

5.3 DATA STORAGE

Data storage is a general term for archiving data in electromagnetic or other forms for use by a computer or device. Different types of data storage play different roles in a computing environment. In addition to forms of hard data storage, there are now new options for remote data storage, such as cloud computing, and Blockchain that can revolutionize the ways that users save and access data.

Figure 2: SCM Configuration page



Garm uses three applications as data storages: Blockchain, Cloud filesystem and relational database better detailed on next subsections. Blockchains grow continuously because of the amount of data and code in them, which is unchanging. Therefore, an important design decision is to choose which data and calculations to keep in and out of the chain.

5.3.1 Filesystem

A cloud file system is a tiered storage system that provides shared access to file data. Users can create, delete, modify, read and write files. All these actions generate a digital fingerprint (hash) information that is stored in the Blockchain, separately from the original files or content, to keep consistency, traceability and auditability to the files.

A Cloud file sharing service gives multiple users simultaneous access to a cloud file data set. Cloud file sharing security is managed with user and group permissions, allowing administrators to tightly control access to shared file data.

5.3.2 Database

A relational database is a set of formally described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables. The standard user and application programming interface (API) of a relational database is the Structured Query Language (SQL). SQL statements are used both for interactive queries for information from a relational database and for gathering data for reports.

5.3.3 Blockchain

The platform uses Blockchain to supply chain management tracking parts and service provenance, ensuring the authenticity of goods, block counterfeits and reducing conflicts. This usually involves a limited and known number of actors, suggesting the use of a permissioned Blockchain, that is, a Blockchain where all nodes must be allowed to be part of the system. To implement that, Hyperledger Fabric is used (Cachin et al., 2016). Hyperledger is an open-source collaborative effort created to advance cross-industry Blockchain technologies.

Hyperledger Fabric is an enterprise-grade permissioned distributed ledger framework for developing solutions and applications. Its modular and versatile design satisfies a broad range of industry use cases. It offers a unique approach to the consensus that enables performance at scale while preserving privacy.

In the context of Garm, the Blockchain module consists of smart contracts and the ledger. From the application developer’s perspective, a smart contract, together with the ledger, form the heart of a Hyperledger Fabric Blockchain system. Whereas a ledger holds facts about the current and historical state of a set of business objects, a smart contract defines the executable logic that generates new facts that are added to the ledger.

5.3.4 Chaincode

Hyperledger Fabric implements smart contracts through chaincode. A chaincode is typically used by administrators to group related smart contracts for deployment, but can also be used for low-level system programming of Fabric. These terms, smart contract and chaincode are used interchangeably in a Hyperledger context. In general, a smart contract defines the transaction logic that controls the lifecycle of a business object contained in the world state. It is then packaged into a chaincode which is then deployed to a Blockchain network. So, smart contracts rule transactions, whereas chaincode rules how smart contracts are packaged for deployment.

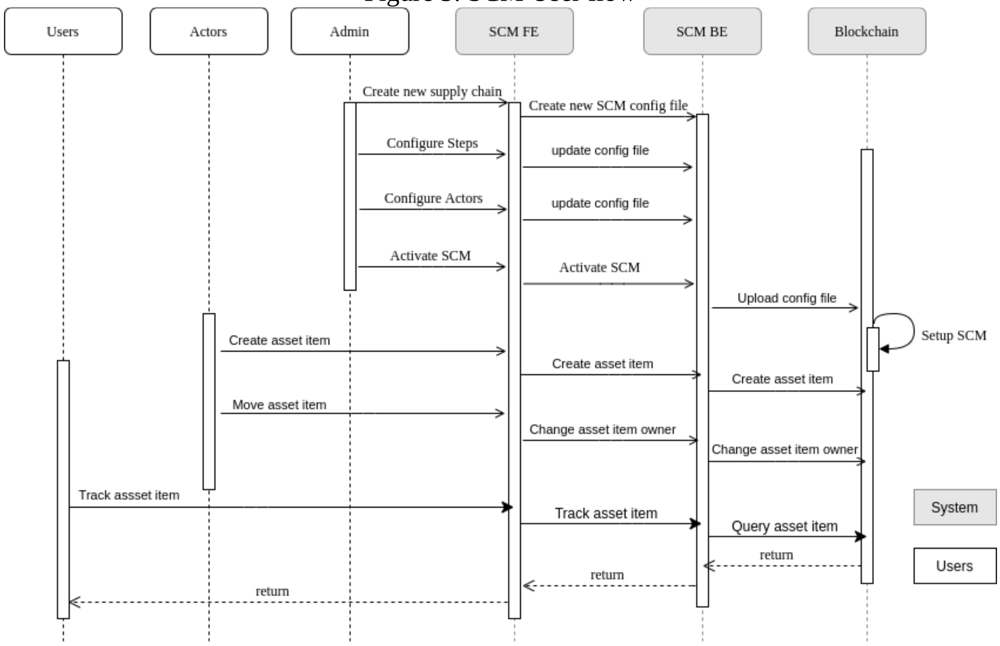
Before businesses transact with each other, they must define a common set of contracts covering common terms, data, rules, concept definitions, and processes. These contracts lay out the business model that govern all of the interactions between transacting parties. A smart contract defines the rules between different organizations in executable code. Applications invoke a smart contract to generate transactions that are recorded on the ledger.

Smart contracts have many APIs available to them. Critically, in all cases, whether transactions create, read, update or delete business objects in the world state, the Blockchain contains an immutable record of these changes.

6 IMPLEMENTATION DETAILS

Our chaincode is written in Golang and provides all contracts needed to proceed traceability in our application. All contracts for use in chaincode must implement *contractapi.ContractInterface*.

Figure 3: SCM User flow



The first step is to create a JSON config file providing all information about these three items. A configuration file includes *assetId*, a list of actors and a list of ordered steps. Our chaincode processes this file through *initLedger* and *createNewAsset* functions. Here follows a template for config file

Code 1: Template for Config File

```
{
  "AssetId":"assetName","Actors":[
    {
      "actorType":"type","aditionalInfo":[
        {
          "key":"value"
        }
      ]
    }
  ],
  "Steps":[
    {
```

```
        "step":"stepName","stepOrder":1,
        "actorType":"actorType"
      }
   ]
}
```

Front-end WebApp enables a user to define settings through a Configuration Page, adding these to the configuration file, as shown in Figure 2.

Assets, asset items, steps and actors are described as *structs*. Respective create methods are associated with each one, for creating instances of these *structs* and saving the state into the Blockchain. Query methods are responsible for interact with the information of any item in the Blockchain. The function *main* invokes the *initLedger*, reads the configuration files and raises the platform enabling users to interact with the Blockchain via exposing its API.

When creating an asset item, an *AssetItemId* is generated. Each entity in the chain will have its unique entity ID and timestamp when it starts processing the transaction. By querying *AssetItemId*, the user can easily track the current transaction information and status. Finally, completed all steps, the Blockchain will update *deliverDate* and mark the status as completed once the final actor has received the order.

ChangeAssetItemOwner is the method called to update an asset item when it is moved from a step to another. It updates the *CurrentOwnerId*, the *ProcessDate*, information about prices and many other details of the transactions by the key/value map *aditionalInfo*.

The *main* function of chaincode invokes the *initLedger* function, reads the configuration files and raises the platform enabling users to interact with the Blockchain via exposing its API.

Figure 3 shows the interaction flow from users with Garm platform. Initially, an admin persona creates and configure the SCM, adding information about the steps and the users. After that, the admin can activate this SCM, and, from that point, the actors can interact with the SCM to provide information about an asset item and also move this asset item through the supply chain. From that point too, any user can track an asset item to get information about the required good.

7 CONCEPT PROOF

The primary purpose of the Garm platform is to decrease manual effort during the development of Web-based supply chain management applications. Therefore, to evaluate our proposal, we used the technology acceptance model (TAM) to verify ease of use and usefulness predict applications usage. The current research investigated TAM for work-related tasks with the World Wide Web as the application. TAM posits that perceived ease of use and perceived usefulness can predict attitudes toward technology that can predict the usage of that technology.

To accomplish the results, an e-mail survey instrument that contained instructions asking the respondent to identify certain desired characteristics was used—focusing a subject on a specific site follows Churchill's recommendation to define a unit of analysis for a more precise response and greater (Churchill, 1979).

Survey respondents were generally well educated with over 35% holding an advanced degree and another 36% having a 4 year degree. The sample of subjects was first split randomly into two groups. Two-factor analyses were performed on 95 subjects. One examined the Web-specific ease of use antecedent items and the other analyzed the Web-specific usefulness antecedent items. The purpose of these analyses was to reduce the number of those items and identify the dimensions of the antecedents to ease of use and usefulness. This group had 95 subjects to preserve a ratio of five subjects to each item for the usefulness items the larger of the two sets of items:

- Seventeen items were asking the extent to which the platform meets ease of use characteristics. Ratings on a 1–5 scale allowed the respondent to indicate the extent. Table 1A lists the three general items which were used. Sixteen were Web-specific antecedents. Table 1B lists 18 Web-specific measures.
- Twenty-two items were asking the extent to which the Web site meets usefulness characteristics. Three were general measures. Sixteen were Web-specific antecedents.
- Two items were measuring Web site usage. One asked the extent to which the respondent used the platform. The second asked the respondent how many times he/she used the platform in the past 30 days.

The survey is presented at Tableau 1

Tableau 1: Survey Sections

Form Survey

(A) TAM ease of use items

- Getting the information I want from the site is easy
- Learning to use the site was easy
- Becoming skilful at using the site was easy

(B) Antecedent ease of use items

Evaluation of Web prototypes

- The site uses terms familiar to me
- The site makes it easy to recognize key information
- The site displays visually pleasing design
- Each display page focuses on a single topic
- Display pages provide links to more detailed information
- The site provides more than one method of navigation
- I can determine my position within site
- The site allows easy return to previous display pages
- The site uses consistent terms
- The site uses consistent graphic

Web user survey

- The site loads quickly
- The information I need is easy to find within the site
- The site is easy to navigate

Usability testing criteria

- The site uses understandable graphics
- The display pages within site are easy to read
- The site uses understandable terms
- The information I need is easy to find within site
- The site is easy to navigate

(C) TAM usefulness items

- Using this site enhances my effectiveness at my job
- Using this site in my job increases my productivity
- Using this site improves my job performance

Each factor analysis used the interaction of the main component with the Varimax rotation and required proper values of at least 1. Any item that failed to load in a single factor at 0.5 or higher was discarded, and the factor analysis was redone. This process of discarding an item and rerunning, continued until all items were loaded by 0.5 or more in one and only one factor.

The research provided some understanding of the ease of use. The antecedents predicted ease of use ($p<0.01$ and $R^2=0.50$) with ease of understanding ($p<0.01$) having a stronger effect than ease of finding ($p<0.05$). The research also provided some understanding of usefulness. The antecedents predicted usefulness ($p<0.01$ and $R^2=0.58$), but only information quality had a significant effect ($p<0.01$).

8 FINAL REMARKS

Although some companies have launched pilot projects using Blockchain technology to manage their supply chains, no detailed information on the technical implementation of such projects has been reported. Either way, the retail industry has the potential to use this technology to improve traceability. Even if some properties of Blockchain implementation may be useful for supply chain management, there are still a few uses to support this claim.

In this paper, we proposed a platform for new decentralized traceability systems based on Blockchain technology. This system will deliver online information to all supply chain members on the safety status of goods, providing a more secure, distributed, transparent, and collaborative approach to supply chain management. The platform can significantly improve the development time of Supply Chain Management applications, and provide efficiency and transparency for product management on a supply chain. In future work, we will present a proof of concept that exercises the use of the proposed platform.

REFERENCES

Saveen A Abeyratne and Radmehr P Monfared. 2016. Blockchain ready manufacturing supply chain using distributed ledger. International Journal of Research in Engineering and Technology 5, 9 (2016), 1–10.

Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, et al. 2018. Hyperledger fabric: a distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference. ACM, "Porto, Portugal", 30.

Christian Cachin et al. 2016. Architecture of the hyperledger blockchain fabric. In Workshop on distributed cryptocurrencies and consensus ledgers, Vol. 310. "Association for Computing Machinery", "New York, NY", 4.

Gilbert A Churchill Jr. 1979. A paradigm for developing better measures of marketing constructs. Journal of marketing research 16, 1 (1979), 64–73.

Michael Crosby, Pradan Pattanayak, Sanjeev Verma, Vignesh Kalyanaraman, et al. 2016. Blockchain technology: Beyond bitcoin. Applied Innovation 2, 6-10 (2016), 71.

Juan F Galvez, JC Mejuto, and J Simal-Gandara. 2018. Future challenges on the use of blockchain for food traceability analysis. TrAC Trends in Analytical Chemistry 107 (2018), 222–232.

Fabíola Greve, Leobino Sampaio, JW ABIJAUDE, AA COUTINHO, I VALCY, and Sílvio Queiroz. 2018. Blockchain e a Revolução do Consenso sob Demanda. Livro de Minicursos do SBRC 1 (2018), 1–52.

Reshma Kamath. 2018. Food traceability on blockchain: Walmart's pork and mango pilots with IBM. The Journal of the British Blockchain Association 1, 1 (2018), 3712.

Qinghua Lu and Xiwei Xu. 2017. Adaptable blockchain-based systems: A case study for product traceability. IEEE Software 34, 6 (2017), 21–27.

Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. Technical Report. Manubot.

Teresa Pizzuti and Giovanni Mirabelli. 2015. The Global Track&Trace System for food: General framework and functioning principles. Journal of Food Engineering 159 (2015), 16–35.

Göran Svensson. 2009. The transparency of SCM ethics: conceptual framework and empirical illustrations. Supply Chain Management: An International Journal 14, 4 (2009), 259–269.

Melanie Swan. 2015. Blockchain: Blueprint for a new economy. " O'Reilly Media, Inc.", "CA,USA".

Nick Szabo. 1997. The idea of smart contracts. http://szabo.best.vwh.net/smart_contracts_idea.html.

Feng Tian. 2017. A supply chain traceability system for food safety based on HACCP, blockchain & Internet of things. In 2017 International Conference on Service Systems and Service Management. IEEE, Dalian, China, 1–6.

Martin Valenta and Philipp Sandner. 2017. Comparison of ethereum, hyperledger fabric and corda. no. June 5, 9 (2017), 1–8.

Cees BM Van Riel and Charles J Fombrun. 2007. Essentials of corporate communication: Implementing practices for effective reputation management. Routledge, New York, NY.

Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Xiangping Chen, and Huaimin Wang. 2018. Blockchain challenges and opportunities: A survey. International Journal of Web and Grid Services 14, 4 (2018), 352–375.