

Received May 11, 2020, accepted May 25, 2020, date of publication May 28, 2020, date of current version June 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2998159

Blockchain-Based Solution for the Traceability of Spare Parts in Manufacturing

HAYA R. HASAN^{ID1}, KHALED SALAH^{ID1}, (Senior Member, IEEE), RAJA JAYARAMAN^{ID2}, RAJA WASIM AHMAD^{ID1}, IBRAR YAQOOB^{ID1}, (Senior Member, IEEE), AND MOHAMMED OMAR^{ID2}

¹Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi 127788, United Arab Emirates

²Department of Industrial and Systems Engineering, Khalifa University, Abu Dhabi 127788, United Arab Emirates

Corresponding author: Ibrar Yaqoob (ibraryaqoob@ieee.org)

This work was supported in part by the Khalifa University of Science and Technology under Award RCII-2019-002, and in part by the Center for Digital Supply Chain and Operations Management.

ABSTRACT Spare parts are essential assets to maintain productivity and operations. A spare part is used for the repair or replacement of failed units. Tracing and tracking of spare parts ownership can help to ensure reliable outcomes in terms of manufacturing operations and service delivery. However, today's spare part inventory systems fall short of providing reliable tracing and tracking of spare parts ownership which poses serious threats to their authenticity. Also, current approaches and systems leveraged for spare part inventory management are vulnerable to the single point of failures. In this paper, we propose a blockchain-based smart contract to trace and track the spare parts ownership details from the original equipment manufacturer to the supplier and end-users. We exploit blockchain features to build a secure and trusted spare part inventory system that is tamper-proof, traceable, trackable, accessible immutable, resilient, and reliable. The proposed system integrates decentralized storage of interplanetary file systems (IPFS) to store and share spare parts data. We present algorithms along with their full implementation details. We perform testing and validation of the smart contract. Furthermore, we provide security and cost analysis, and show how the proposed system ensures reliable tracing and tracking of spare parts ownership. We make the smart contract source code publicly available on Github.

INDEX TERMS Spare parts, blockchain, smart contracts, data provenance, manufacturing, traceability.

I. INTRODUCTION

The rapidly growing automotive industries and technological advancements in the design of machinery have derived the demand for spare part components [1]. The spare parts supply chain management is complex and challenging as it involves globally interconnected partners and supply chain services. The transparency and visibility of operations during the supply chain process affect the conformity and safety of spare part components. The absence of reliable protocols to exchange data, dependency on human mediation to transfer valuable assets, and limited participation of stakeholders during the supply chain process can affect components' reliability, availability, maintenance, and total return on investment (ROI) [2], [3]. Blockchain technology can improve supply chain operations to increase system efficiency, minimize

The associate editor coordinating the review of this manuscript and approving it for publication was Fabrizio Messina^{ID}.

maintenance costs, and secure business activities. According to the Accenture technology vision report, 86% of aerospace organizations will integrate blockchain technology into their corporate operations by 2021 [4]. One of the aerospace giants named Honeywell has employed blockchain technology to prove the provenance of Boeing parts and ensure they comply with safety standards [5].

A spare part is an interchangeable component of a system that is appropriate to replace a failed or damaged part [1]. Spare parts management is of critical importance as it is a maintenance service that maintains an adequate amount of spare parts to increase mean times between system failure and reduce the budget in the repair-and-reliability value chain. As a result, it significantly minimizes the spare part associated delays to increase the operational reliability of a system. The maintenance service to a system or component helps to restore its reliability and safety by enhancing the system's performance [6], [7]. For instance, in the aviation

industry, airworthiness investigative authority (AIA) assigns a certificate of airworthiness (CoA) based on the health level of aircraft or components. Spare parts are heterogeneous and vary depending on their life cycle, design, type, and scrap rate. For instance, spare parts of an aircraft can be rotatable (e.g., brakes and wheels), repairable, expandable (e.g., fasteners), recoverable, and life-limited. The high-speed turbine rotor is a life limited spare part and its life cycle is subject to hours only [7], [8].

Traditional spare parts supply chain management is based on manual inventory management systems that are very slow, expensive, and difficult to manage and process. In the traditional systems, during the maintenance operations, tracing the provenance of a component is very difficult and it takes days to trace the history of a spare part. Also, the manual systems follow centralized storage architecture that makes it vulnerable to hackers and a single point of failure [9], [10]. Typically, web-based approaches are used to handle orders for purchasing and selling components. However, most of the organizations have restricted control over the supply chain process in recent years due to limited visibility and transparency that hinder the assessment of customer demand [11]. In comparison, blockchain technology presents a single, synchronized, secure, and immutable view of tamper-proof data [3], [7], [12].

To handle the aforementioned spare parts related supply chain challenges, blockchain technology proposes to permanently record all movements, modifications, restorations, and ownership details of spare parts on a distributed transparent and tamper-proof ledger. Furthermore, transparency of transaction records (e.g., frequent maintenance of components) increases the trust of customers and regularity authorities [8]. Using immutable transaction records, blockchain technology assists to identify counterfeit parts, minimize frauds, reduce components' maintenance costs and delays, and establish the provenance of assets based on the traceability feature [13].

A. RELATED WORK AND CONTRIBUTIONS

Tracing and Tracking are the most critical aspects of the existing supply chain management of spare parts. Moreover, their logistical operations are challenging to maintain because of the complicated architecture of industrial machines. Additionally, each component has different characteristics and features. For instance, many of the components can be installed, removed, maintained, and shipped within the organizations during their lifetime. In traditional systems, the data storage and sharing using web-interfaces, emails, and fax open the door for hackers to perform modification within the documents to falsify the truth [8], [14]. Also, the increasing demand for spare part components motivates the original equipment manufacturer (OEM) to directly provide support by-the-hour basis along with high-quality service assurance. The type, nature, and complexity of spare part components put forward demands for highly optimized algorithms that can ensure efficient planning and decision making during supply chain management [15].

The study conducted in [16] has proposed a tool to analyze and assure the high availability of spare parts for the uninterrupted production process. The proposed method aims to satisfy the customers to proactively handle interruptions in the production process resulting from planned and unplanned maintenance, repair, and overhaul operations. The presented tool managed the spare parts components into two classes based on their frequency of use and critical priority criterion. The critical spare parts are crucial to handle to assure the continuous production process. The criticality of components is forecasted based on a multi-criteria decision-making method called the analytic hierarchy process. The decision engine considered logistics costs, upper bound of delivery waiting time, spare parts warehouse location, and repairing time factors to schedule spare parts associated operations to minimize the downtime. The primary focus of the study was to manage the inventory of the spare parts. However, it lacks tracing and tracking of the spare parts ownership.

Similarly, in [17], the authors did not provide a mechanism to trace the spare part as its ownership is revoked from the OEM to the suppliers and end-users. The authors have proposed the integration of an intelligent maintenance system (IMS) to the supply chain to forecast the failure of components to improve the operational planning of spare parts supply chain. The proposed system incorporates and follows a heuristic approach, mixed linear programming (MILP) model, and the demand for spare parts indicated by the IMS module. It ensures that during erratic spare parts demand, components breakdowns are proactively handled without disrupting the production and logistic operations for the preset deadlines. The linear programming model optimizes the behavior of the system under certain conditions such as system capabilities, forecasts, and spare parts delivery dates (Stochastic variables). The scheduler for distribution and maintenance keeps a balance between cost and service level. Also, it ensures that spare parts are delivered before the deadline for higher customer satisfaction.

Furthermore, the system proposed in [10], called FIEGE, has analyzed a web service-based logistics case study to establish a global information state about spare part components and documents exchanged among participating organizations. The key services of the proposed logistic system include shipment management, consolidation, freight calculator, tracking and tracing information and documents, event management, e-invoicing, and reconciliation. Among all services, consolidation combines multiple transport orders into a single shipment unit to reduce the total cost. During the shipment process, the proposed system offers two modes called manual and automated to update the location of the shipment for tracking purposes. On the other hand, tracking and traceability service helps to keep the track record of a component or document during its shipment. The proposed system has integrated several logistic systems for greater efficiency. The system was implemented on a four-layer web architecture server. The proposed system is viable to support the transfer of standard documents such as EDIFACT, VDA,

XML, and FORTRAS. However, the proposed system is prone to errors, frauds, and vulnerable to a single point of failure.

In general, the spare part components are mostly managed based on cooperative pooling, commercial pooling, Adhoc, and in-house sourcing methods. In [18], the authors have proposed an Internet of things (IoT) based study to assess the four spare parts inventory management techniques as stated above. During the maintenance operation, the IoT-based system assists to optimize predictive and scheduled maintenance based on real-time monitoring data of maintenance operations, health history of component, and its remaining life expectancy. The proposed IoT-based solution does not cover a methodology to keep track of the ownership of the spare parts in case of accidents or other use cases.

On the other hand, the authors in [3], [7], [12], [19] have highlighted the importance of having a decentralized supply chain procedure for the management of spare parts. However, they did not discuss any system architecture design or algorithms in this regard. Although they have presented the various challenges that blockchain can help to resolve, they did not explore or study the issues related to tracing or tracking of the spare parts ownership.

Furthermore, the authors in [20], [21] proposed the use of cloud computing technology to support and facilitate spare parts supply chain management, operations, and services. Cloud computing is a technology that is widely used nowadays to provide computation and storage at scale with high availability and reliability. However, cloud technology is centralized in which the data and computation are done by a single platform which can be subject to data tampering and manipulation, hacking, mistrust, failure, and privacy evasion. By design, blockchain is decentralized, and computation and storage are done in a secure and trusted manner. This makes it ideal for supply chain tracking and tracing. In short, some of the key blockchain features include trusted data provenance [22], [23], transparency and visibility as well as data authentication and signature of origin, immutability and integrity—in a fully secure, trusted and decentralized manner. Furthermore, blockchain smart contracts can offer programmability to ensure execution of business logic with ordered execution of transactions according to predefined set of rules and policies, in a way that is open and trusted by all actors and organizations who are interacting with the blockchain network.

In summary, most of the above-mentioned solutions are primarily focused on inventory management systems that follow centralized management and record keeping. They are also unable to trace and track spare parts ownership history in a tamper-proof and transparent manner. This can be achieved using blockchain technology that is based on distributed and decentralized architecture. The lack of enough research on the tracing and tracking of spare part details and ownership in a decentralized way motivated us to conduct this study. Unlike the existing works, the key contributions of our paper can be summarized as follows:

- We propose a blockchain-based solution that guarantees secure and trusted traceability, trackability, accessibility, immutability, and data provenance of spare parts.
- We introduce and develop smart contracts code and algorithms to implement the functions, modifiers, and trigger events.
- We integrate our blockchain-based system with the InterPlanetary File System (IPFS) to securely store the details of the spare part components in a decentralized way.
- We perform security and cost analysis to evaluate the reliability and practicality of the proposed solution.
- We propose a generic solution that can be implemented on a permissioned or permissionless blockchain depending on the needs of industries.
- We present the full implementation details, smart contract code,¹ and testing details.

The remainder of the paper is organized as follows. Section II presents the proposed blockchain-based approach. Section III describes the key implementation details. Section IV provides testing and validation details. Security and cost analysis are provided in section V. Section VI concludes the paper.

II. PROPOSED BLOCKCHAIN-BASED SOLUTION

The details of the proposed blockchain-based system to trace and track the spare parts in manufacturing are provided in this section. Figure 1 shows the components of the system along with the interactions between the different participating entities and the smart contracts.

- **Participating Entities:** Suppliers and the Original Equipment Manufacturer (OEM) usually interact with the participating entities of a company through the procurement manager. The participating entities include engineer, line manager, inventory manager, purchase manager, and procurement manager. All the participating entities possess Ethereum addresses to interact with each other through the blockchain network. The participating entities would access a decentralized application to communicate with the blockchain client through web3 and JSON-RPC. The engineer initiates the requests such as the purchase request (PR) and the purchase order (PO). The line, inventory, and purchase managers approve the requests and make confirmations based on their roles. To satisfy the privacy needs of an organization, the interactions between the suppliers and OEM are only viewed/heard by the procurement manager. Moreover, all the interactions of the organization's employees are restricted to a certain number of entities (viewed/heard). This can be achieved (if necessary) by using a private Ethereum blockchain or the Hyperledger. However, if the objective is to just trace back a spare part to the OEM through the suppliers and

¹<https://github.com/smartcontract694/SP/blob/master/code>

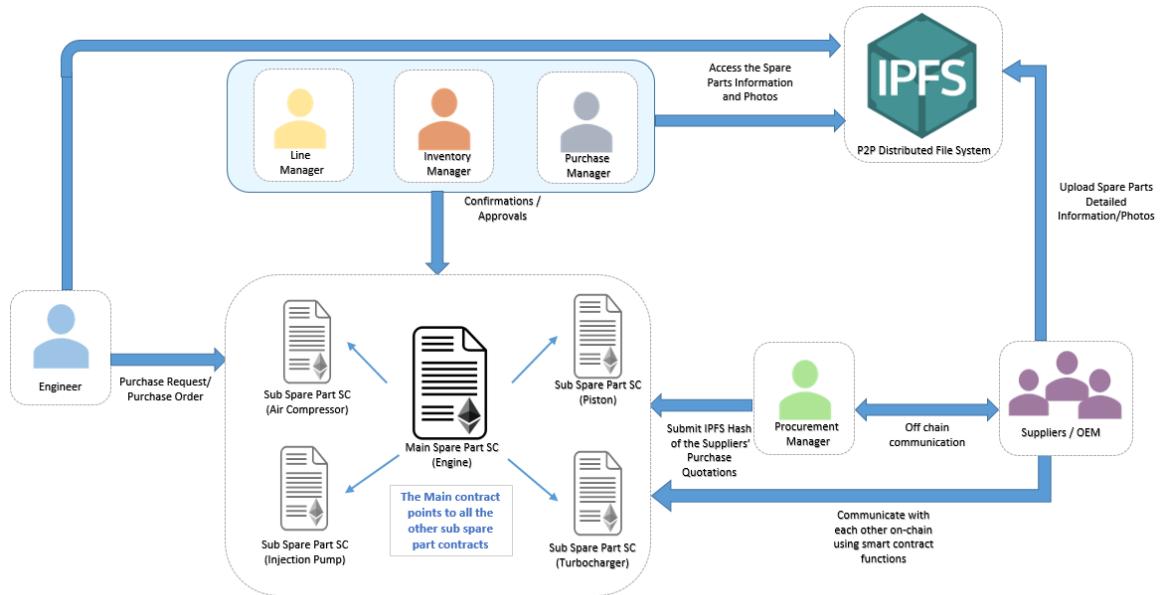


FIGURE 1. A system diagram showing the main components interacting with the smart contract and the resulting on-chain resources.

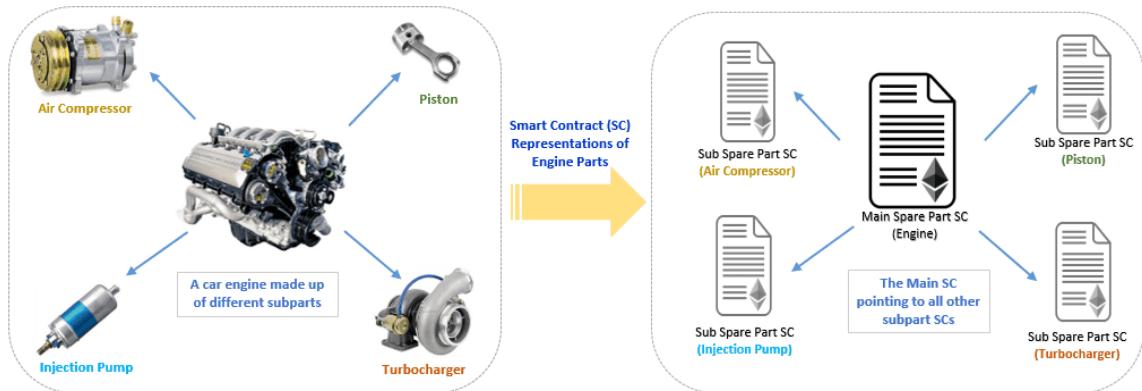


FIGURE 2. The smart contract representation of the different spare parts that form a car engine.

random purchasers, a public blockchain can be used to meet such a requirement.

- **IPFS Server:** IPFS [24] is an essential part of the proposed blockchain-based solution. It stores all the images of the spare part from different angles as well as the purchase quotations (PQ) of the suppliers and OEM. The bidders provide their IPFS hashes when they present their PQs to the procurement manager and other entities of an organization. Storing such data on the chain can be extremely expensive. Alternatively, the IPFS is an emerging storage option that makes the solution feasible and tamper-proof.
- **Smart Contracts:** Each spare part has its smart contract. A spare part can be assembled from other spare parts. Consequently, every assembled spare part points to other smart contracts. Figure 2 shows in details how the smart contracts in Figure 1 represent the real spare parts of a car engine. A car engine is assembled from other spare

parts such as an injection pump, air compressor, piston, and turbocharger. Each of those has a separate smart contract and the original parent contract is the engine's smart contract. Hence, the parent smart contract will have addresses of all sub-smart contracts.

A. COMMUNICATION BETWEEN THE PARTICIPATING ENTITIES

A public blockchain can be used to keep track of the spare part owners, if the participating entities have no privacy concerns. However, if the participating entities are employees of an organization then it is important to maintain the privacy of their logs and communication. At the same time, it is important to listen to the interactions between the suppliers and the OEM. Therefore, to maintain the transparency between the different vendors, a permissioned blockchain can be used. Hyperledger Fabric (HLF) is a private blockchain that allows

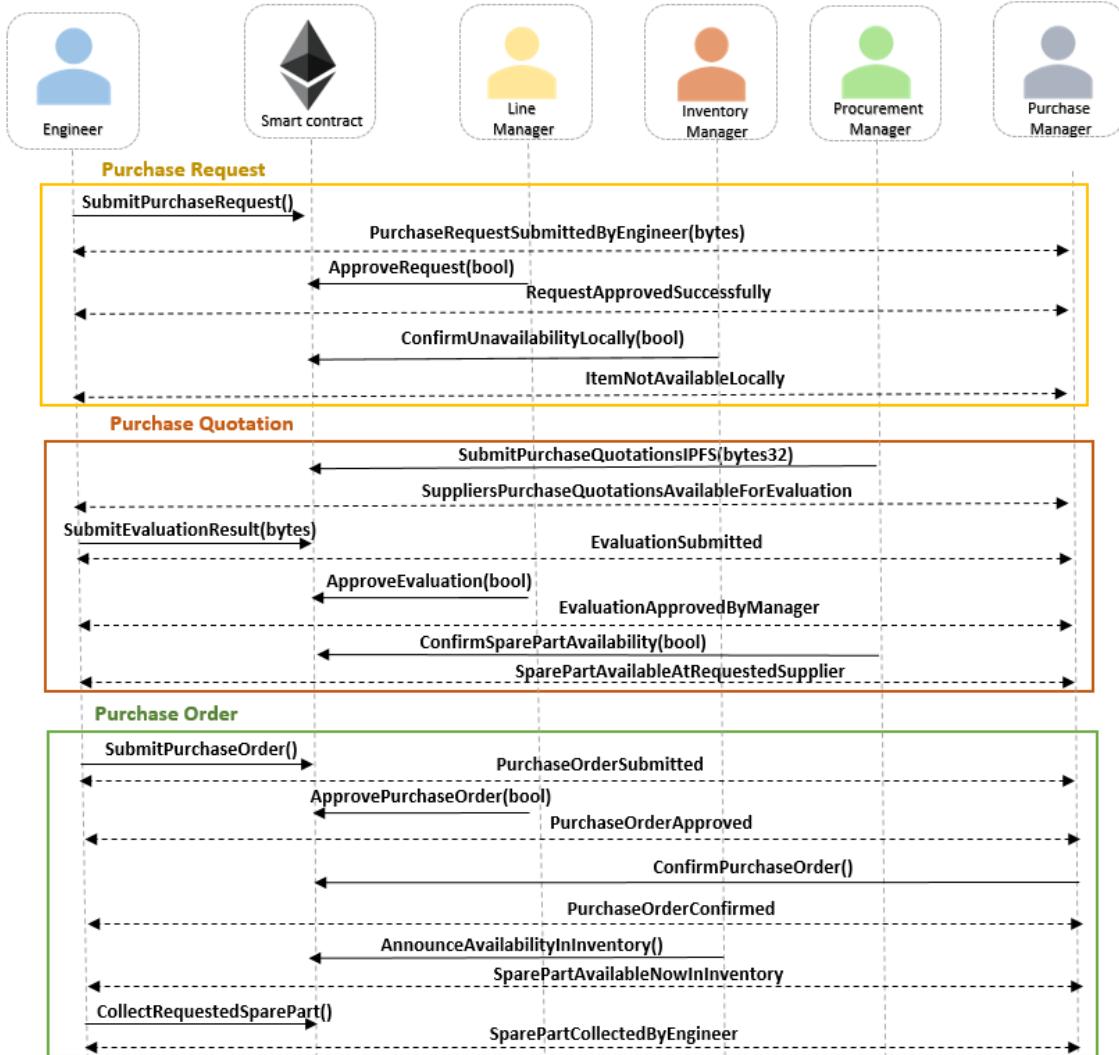


FIGURE 3. Sequence diagram showing the different interactions between the company's participating entities with the smart contract.

certain users to use or access the blockchain. Communication channels are used in Fabric to ensure confidential communication between two or more parties. A channel can be designed by organization members and communicating members who must be authorized and authenticated. Moreover, privacy can be maintained using Hyperledger Besu which is an Ethereum client that has access control and management through its transaction manager 'Orion' [25], [26]. Additionally, it also has 'EthSigner' for managing identities and private keys [25].

III. IMPLEMENTATION DETAILS

We develop the smart contract in Solidity language using the Remix IDE [27]. The blockchain-based system is verified using the Ethereum smart contract that logs all the communication information between the participating entities. In this section, we discuss full implementation details along with algorithms. The implemented source code is made publicly available on GitHub.

Our proposed decentralized approach to trace and track the spare parts ownership is generic, and thus can be implemented on public or private blockchain infrastructure. It all depends on the use case. The participating entities can be only the Original Equipment Manufacturer (OEM), suppliers, and people who are interested in owning the spare parts, or it could be engineers and managers from a company in addition to the OEM and suppliers. The latter case requires additional privacy and access control to authorize only certain entities to listen/view the on-chain communications. In our implementation, we will explain the details of the algorithms that can be applied in a real scenario in a company. For instance, an engine needs a replacement, the whole on-chain procedure is demonstrated in the implementation from the Purchase Request (PR) to the Purchase Quotations (PQ), and finally the Purchase Order (PO).

Figure 3 shows the sequential interactions between the members of a company when a spare part is required for maintenance. The engineer initiates the request by making

a PR which is approved by the line manager. The inventory is checked to ensure whether or not the spare part available locally. This is followed by the PQs that are supplied by the procurement manager. The procurement manager is the only member that is allowed to listen/view the communication between the suppliers and the OEM. Hence, the procurement manager forwards IPFS hashes information provided by the suppliers and bidders to the engineer to examine and analyze them based on the requested specifications and criteria. In the final phase, the PO is placed by the engineer and is approved by the line manager and purchase manager.

It is important to mention that the spare part received was originally manufactured by the OEM and then provided to the company either through the OEM or through a supplier who bought it from the OEM or another supplier. Therefore, to trace and track the ownership details of the spare part, the ownership revocation of the spare part needs to be logged on the chain. In our implementation scenario, ensuring privacy of the communication information is achieved using private blockchain infrastructure as it helps to authorize participating entities and provides separate communication channels between suppliers and bidders. The sequential interaction between OEM and suppliers is explained in Figure 4. The spare parts ownership requested by the supplier is transferred to the new owner after the payment is completed using the smart contract.

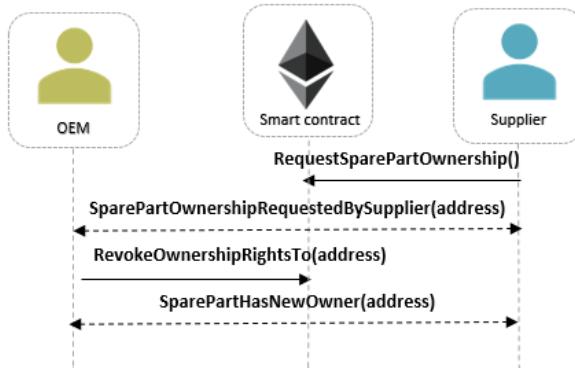


FIGURE 4. Sequence diagram showing the interactions between different suppliers or between the supplier and the OEM.

Tracing and tracking ownership records can be made manageable using blockchain's tamper-proof logging abilities and smart contracts. Details of the function calls and their events are presented as part of the algorithms. Each function holds a state change and is restricted based on the current state of the smart contract as well as modifiers. Modifiers restrict and control who can initiate a function call. Also, the Material and Equipment Standards and Code number (MESC) of the spare part is used in the algorithms to log the spare part identification number. Further details of the algorithms can be found in the next subsections.

A. SUBMITTING A PURCHASE REQUEST

Only an engineer can submit a purchase request for a required spare part when the status is 'Needs PR'. This status

indicates that the spare part is currently unavailable in the local inventory, and thus needs to be purchased. At the end of this request, the state is updated to 'PR Submitted' and an event is triggered to notify the participating entities, as can be seen in algorithm 1.

Algorithm 1 Submitting a Purchase Request

```

Input : caller, SPstate, Engineer, MESC
1 caller holds the Ethereum Address of the function
   caller.
2 if caller == Engineer then
3   if SPstate == NeedsPR then
4     SPstate = PRsubmitted.
5     Emit an event to notify everyone that a purchase
       request with the MESC number is submitted.
6   end
7   else
8     Preview an error after returning the contract to
       its previous state.
9   end
10 end
11 else
12   Preview an error after returning the contract to its
       previous state.
13 end
  
```

B. APPROVING THE PURCHASE REQUEST

Once the PR is submitted by the engineer, the line manager needs to approve the request. Algorithm 2 shows how the state is updated as the line manager approves the request.

C. CONFIRMING THE UNAVAILABILITY OF THE SPARE PART LOCALLY

After the approval of the PR, the line manager creates a notification that notifies the inventory manager to confirm the unavailability of the spare part in the local inventory. The inventory manager follows the algorithm 3 when confirming that the item is not available locally.

D. SUBMITTING THE PURCHASE QUOTATIONS IPFS HASH

The next phase after getting the PR approval is to submit PQ to the engineer for evaluation. The procurement manager who is the only in contact with the suppliers and bidders submits the PQs' IPFS hash on the chain, as can be seen in algorithm 4.

E. SUBMITTING THE EVALUATION RESULTS

The engineer evaluates the quotations of the suppliers based on the company's criteria and requirements. The engineer is responsible to choose the best-fit supplier and submit the supplier's unique identification code as shown in algorithm 5.

Algorithm 2 Approving the Purchase Request

```

Input : result, caller, SPstate, Line Manager
1 caller is the Ethereum Address of the function caller.
2 result is a boolean input by the manager.
3 if caller == LineManager then
4   if SPstate == PRsubmitted then
5     if result == True then
6       SPstate = PRapproved.
7       Emit an event to notify listeners about the
8       successful approval of the PR.
9     end
10    else
11      Emit an event to notify listeners that the PR
12      is not approved.
13    end
14  end
15  else
16    Reverse the contract state to the previous one
17    and show an error.
18  end
19 end

```

Algorithm 3 Confirming the Unavailability of the Spare Part Locally

```

Input : caller, SPstate, Inventory Manager, result
1 caller is the Ethereum Address of the function caller.
2 result is a boolean entered by the Inventory Manager.
3 if caller == InventoryManager then
4   if SPstate == PRapproved then
5     if result == True then
6       SPstate = NotAvailableLocally.
7       Create a notification announcing the
8       unavailability of the spare part locally.
9     end
10    else
11      Create a notification stating that the spare
12      part is available locally.
13    end
14  end
15  else
16    Reverse the contract state to the previous once
17    and show an error.
18  end
19 end

```

F. APPROVING THE ENGINEER's EVALUATION

The line manager is required to approve the evaluation submitted by the engineer to complete the purchase process. Algorithm 6 presents this process.

G. CONFIRMING SPARE PART CURRENT AVAILABILITY AT THE SUPPLIER

An approved evaluation result allows the procurement manager to step in and confirm that the spare part is currently available with the chosen supplier. This is the last step in the purchase quotations phase which leads to complete a purchase order request if all goes well. It is handled by the procurement manager as presented in algorithm 7.

H. SUBMITTING A PURCHASE ORDER

The PO is placed by the engineer through a function call, as can be seen in algorithm 8. When the PO is placed, the state is automatically updated.

I. APPROVING THE PURCHASE ORDER

The line manager approves the engineer's request against the PO as shown in algorithm 9. A boolean is passed as a parameter to the function to approve the request.

J. CONFIRMING THE PURCHASE ORDER BY THE PURCHASE MANAGER

The purchase manager is responsible to confirm the PO. The payment and delivery process can be made on the chain as discussed in detail using a proof of delivery blockchain-based

Algorithm 4 Submitting the Purchase Quotations

```

Input : caller, SPstate, Procurement Manager, ipfs
1 caller is the Ethereum Address of the function caller.
2 ipfs is the IPFS hash of the suppliers PQs.
3 if caller == ProcurementManager then
4   if SPstate == NotAvailableLocally then
5     Create a notification to announce the
6     availability of the suppliers' PQs for evaluation.
7     SPstate = PQsubmitted.
8   end
9   else
10    Error reverse contract state to the previous one.
11  end
12  else
13    Error reverse contract state and to the previous one.
14 end

```

solution [28], [29]. Algorithm 10 explains the details of confirming the PO by the purchase manager.

K. ANNOUNCING THE AVAILABILITY OF SPARE PART IN THE INVENTORY

The successful procurement of the spare part makes it locally available in the inventory. The inventory manager creates a notification and announces the availability of the requested spare part using algorithm 11.

Algorithm 5 Submitting the Evaluation Result

Input : caller, SPstate, Engineer, supplierCode

- 1 *caller* is the Ethereum Address of the function caller.
- 2 *supplierCode* is the input used by the engineer that uniquely identifies a company.
- 3 **if** *caller* == *Engineer* **then**
- 4 **if** *SPstate* == *PQsubmitted* **then**
- 5 *SPstate* = *EvaluationSubmitted*.
- 6 **end**
- 7 **else**
- 8 Show an error and reverse contract state to previous one.
- 9 **end**
- 10 **end**
- 11 **else**
- 12 Show an error and reverse contract state to previous one.
- 13 **end**

Algorithm 6 Approving the Evaluation

Input : caller, SPstate, Line Manager, result

- 1 *caller* is the Ethereum Address of the function caller.
- 2 *result* is a boolean used as an input.
- 3 **if** *caller* == *LineManager* **then**
- 4 **if** *SPstate* == *EvaluationSubmitted* **then**
- 5 **if** *result* == *True* **then**
- 6 Emit an event stating that the evaluation is approved by the manager.
- 7 *SPstate* = *EvaluationApproved*.
- 8 **end**
- 9 **else**
- 10 Emit an event stating that the evaluation is not approved by the manager.
- 11 **end**
- 12 **else**
- 13 Reverse contract state and show an error.
- 14 **end**
- 15 **end**
- 16 **else**
- 17 Reverse contract state and show an error.
- 18 **end**
- 19 **end**

L. COLLECTING THE REQUESTED SPARE PART

Once the spare part is locally available in the inventory, the engineer can collect it for its further distribution. The engineer announces collecting the spare part and updates its state as explained in algorithm 12.

IV. TESTING AND VALIDATION

In this section, we provide testing and validation details of the smart contract. We use an Ethereum smart contract to test the interactions between the employees of an organization.

Algorithm 7 Confirming Spare Part Current Availability at the Chosen Supplier

Input : caller, SPstate, Procurement Manager, result

- 1 *caller* is the Ethereum Address of the function caller.
- 2 *result* is a boolean input.
- 3 **if** *caller* == *ProcurementManager* **then**
- 4 **if** *SPstate* == *EvaluationApproved* **then**
- 5 **if** *result* == *True* **then**
- 6 Emit a notification announcing the availability of the spare part at the requested supplier.
- 7 *SPstate* = *SparePartAvailabilityConfirmed*.
- 8 **end**
- 9 **else**
- 10 Emit a notification announcing the immediate unavailability of the spare part at the requested supplier.
- 11 **end**
- 12 **end**
- 13 **else**
- 14 Reverse contract state and show an error.
- 15 **end**
- 16 **end**
- 17 **else**
- 18 Reverse contract state and show an error.
- 19 **end**

Algorithm 8 Submitting a Purchase Order

Input : caller, SPstate, Engineer

- 1 *caller* holds the Ethereum Address of the function caller.
- 2 **if** *caller* == *Engineer* **then**
- 3 **if** *SPstate* == *SparePartAvailabilityConfirmed* **then**
- 4 *SPstate* = *PurchaseOrderSubmitted*.
- 5 Emit an event to notify everyone that a purchase order is submitted.
- 6 **end**
- 7 **else**
- 8 Preview an error after returning the contract to its previous state.
- 9 **end**
- 10 **end**
- 11 **else**
- 12 Preview an error after returning the contract to its previous state.
- 13 **end**

The purchase of the spare part and the delivery of the item is not tested in this paper, as it has already been implemented and discussed in our previous study [28], [29]. The participants including engineer, line manager, inventory manager, procurement manager, and purchase manager

Algorithm 9 Approving the Purchase Order

```

Input : caller, SPstate, Line Manager, result
1 caller is the Ethereum Address of the function caller.
2 result is a boolean input.
3 if caller == LineManager then
4   if SPstate == PurchaseOrderSubmitted then
5     if result == True then
6       Emit a notification announcing the approval
       of the PO.
7       SPstate = PurchaseOrderApproved.
8     end
9   else
10    Emit a notification announcing the
       declination of the PO request.
11  end
12 end
13 else
14  | Reverse contract state and show an error.
15 end
16 end
17 else
18  | Reverse contract state and show an error.
19 end

```

Algorithm 10 Confirming the Purchase Order

```

Input : caller, SPstate, Purchase Manager
1 caller holds the Ethereum Address of the function
   caller.
2 if caller == PurchaseManager then
3   if SPstate == PurchaseOrderApproved then
4     SPstate = PurchaseOrderConfirmed.
5     Emit an event to notify everyone that a purchase
       order is confirmed.
6   end
7   else
8     | Preview an error after returning the contract to
       its previous state.
9   end
10 end
11 else
12  | Preview an error after returning the contract to its
       previous state.
13 end

```

have the following Ethereum addresses, respectively; “0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c”, “0x14723A09ACff6D2A60DcdF7aA4AFF308FDDC160C”, “0x4B0897b0513fdC7C541B6d9D7E929C4e5364D2dB”, “0x583031D1113aD414F02576BD6afaBfb302140225”, “0xD870fA1b7C4700F2BD7f44238821C26f7392148”.

Modifiers are tested to ensure that each function can only be executed by the authorized entity only. Also, the state

Algorithm 11 Announcing Spare Part Availability

```

Input : caller, SPstate, Inventory Manager
1 caller holds the Ethereum Address of the function
   caller.
2 if caller == InventoryManager then
3   if SPstate == PurchaseOrderConfirmed then
4     SPstate = AvailableInInventory.
5     Emit an event to notify everyone that the
       ordered spare part is now available.
6   end
7   else
8     | Preview an error after returning the contract to
       its previous state.
9   end
10 end
11 else
12  | Preview an error after returning the contract to its
       previous state.
13 end

```

Algorithm 12 Announcing Spare Part Availability

```

Input : caller, SPstate, Engineer
1 caller holds the Ethereum Address of the function
   caller.
2 if caller == Engineer then
3   if SPstate == AvailableInInventory then
4     SPstate = CollectedByEngineer.
5     Emit an event to notify everyone that the spare
       part has been collected.
6     SPstate = NeedsPR.
7   end
8   else
9     | Preview an error after returning the contract to
       its previous state.
10  end
11 end
12 else
13  | Preview an error after returning the contract to its
       previous state.
14 end

```

of the spare part components ensures the correct sequence is followed during execution without any repetition unless allowed. For instance, a new purchase request can only be submitted by an engineer if the old request is already completed and the spare part is not locally available. In the following subsections, we present the logs produced when the functions are successfully executed using the Remix IDE [27]. We further present a complete scenario of an engineer requesting a spare part by showing the logs of the main functions.

A. SUBMITTING A PURCHASE REQUEST

The engineer initiates the process by submitting a purchase request using a function called *SubmitPurchaseRequest* which changes the state and emits an event as shown in the logs in Figure 5.

FIGURE 5. Logs showing a successful purchase request submitted by the Engineer.

B. APPROVING THE PURCHASE REQUEST

The line manager approves the PR of the engineer by using a boolean as an input in *ApproveRequest* function, as can be seen in Figure 6. After the purchase request is approved by the line manager, the inventory manager executes a successful transaction to ensure that the spare part is not available locally in the inventory. This updates the state of the spare part to ‘NotAvailableLocally’ which allows the procurement manager to supply on the chain the IPFS hash of the purchase quotations available from the bidders and suppliers.

```
status          0x1 Transaction mined and execution succeed
transaction hash 0xedc950dd07cb04ddcc1c48259c1198e5480f549a6dbe41c3fe1cec135d05e1
from           0x14723a09acf6d2a06dcdf7faa4ff7a08fddc160c
to             SpareParts.ApplicationRequest(bool)
gas            3000000 gas
transaction cost 30298 gas
execution cost 8834 gas
hash           0xedc950dd07cb04ddcc1c48259c1198e5480f549a6dbe41c3fe1cec135d05e1
input          0x584...000001
decoded input  {
                  "bool result": true
            }
decoded output  {}
logs           [
                  {
                    "from": "0x15e08fa0fe3e3aa3607ac57a29f2b58cb154a2",
                    "topic": "0x093dceee53c696125a3c464bd803f95d8aae00e655be9d27ec75a0d63afce08242",
                    "event": "PurchaseRequestApprovedSuccessfully"
                  }
            ]
```

FIGURE 6. A successful PR approval by the Line Manager.

C. SUBMITTING THE PURCHASE QUOTATIONS IPFS HASH

The procurement manager is the only authorized entity to access and communicate with the suppliers and OEM. The procurement manager also transfers the purchase quotations of the bidders to the organization's employees on the chain using the function *SubmitPurchaseQuotationsIPFS*. The IPFS hash is entered using an input parameter of type bytes32. In this testing scenario, the IPFS hash ““0x64EC88CA00B268E5BA1A35678A1B5316D212F4F3 66B2477232534A8AECA37F3C”” was entered in the following format; “[““0x64”, ““0xEC”, ““0x88”, ““0xCA”,

“0x00”, “0xB2”, “0x68”, “0xE5”, “0xBA”, “0x1A”, “0x35”, “0x67”, “0x8A”, “0x1B”, “0x53”, “0x16”, “0xD2”, “0x12”, “0xF4”, “0xF3”, “0x66”, “0xB2”, “0x47”, “0x72”, “0x32”, “0x53”, “0x4A”, “0x8A”, “0xEC”, “0xA3”, “0x7F”, “0x3C”]” to the function for execution. The result was a successful transaction, as can be seen in Figure 7. Consequently, the engineer evaluates the quotations from each bidder and sends reply back by submitting the company’s unique code (as a bytes32) on the chain. This evaluation requires approval from the line manager to proceed further in the procurement procedure.

FIGURE 7. Logs showing successful passing of the PQ IPFS hash.

D. CONFIRMING SPARE PART AVAILABILITY AT THE REQUESTED SUPPLIER

The last step before the engineer submits a purchase order is performed by the procurement manager where the spare part is confirmed to be available immediately at the requested and approved supplier. This is done using a boolean as an input parameter in the function *ConfirmSparePartAvailability*, as can be seen in Figure 8.

```
status          0x1 Transaction mined and execution succeed
transaction hash 0xaa3b4a866bdfc7aa885b95f400e3d8f2ea15c71f33a468421cf89ce4fe3821b
from           0x583031d11113ad414f02576bdafabfb302140225
to             SparreParts.ConfirmSparePartAvailability(bool) 0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2
gas            3000000 gas
transaction cost 30231 gas
execution cost 8767 gas
hash           0xaa3b4a866bdfc7aa885b95f400e3d8f2ea15c71f33a468421cf89ce4fe3821b
input          0x619...00001
decoded input  {
                  "bool result": true
            } 
decoded output () 
logs           [
                  {
                    "from": "0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2",
                    "topic": "0xd9dc927459c2ed6f2406fb8ab9d8865775f540dec8deba14548339d7814d1a",
                    "event": "SparePartAvailableAtRequestedSupplier"
                  }
                ]
```

FIGURE 8. Logs showing an event emitted announcing the availability of the requested spare part at the supplier.

E. SUBMITTING THE PURCHASE ORDER

The engineer submits the PO request which emits an event announcing the submission of the request and updates the state to 'PurchaseOrderSubmitted' as presented in Figure 9. This PO request is approved successfully by the Line Manager before the Purchase Manager confirms the purchase.

```

status 0x1 Transaction mined and execution succeed
transaction hash 0x821144f431ab5b269dfbba35e3e21110e670b5d7c50461aedbb0e75568a3ae21 □
from 0xca35b7d915458ef540adec6088dfc2f44e8fa73c □
to SpareParts.SubmitPurchaseOrder() 0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2 □
gas 3000000 gas □
transaction cost 29895 gas □
execution cost 8623 gas □
hash 0x821144f431ab5b269dfbba35e3e21110e670b5d7c50461aedbb0e75568a3ae21 □
input 0xda6...350ae □
decoded input () □
decoded output () □
logs [
  {
    "from": "0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2",
    "topic": "0x44d48af3218df6357e7?fafdc0de701e0ae685df4d1ce8b44759ca976b27a7",
    "event": "PurchaseOrderSubmitted",
  }
]

```

FIGURE 9. Logs showing the PO request submitted successfully.

```

status 0x1 Transaction mined and execution succeed
transaction hash 0xf885774b0517889c5602457a20f7de6341f59736989531ef2ed420be637678c2 □
from 0xd8d70fa1b74700f21d744238b2126f7392148 □
to SpareParts.ConfirmPurchaseOrder() 0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2 □
gas 3000000 gas □
transaction cost 29961 gas □
execution cost 8689 gas □
hash 0xf885774b0517889c5602457a20f7de6341f59736989531ef2ed420be637678c2 □
input 0x49...04a7b □
decoded input () □
decoded output () □
logs [
  {
    "from": "0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2",
    "topic": "0x75f7e141937c40da40b0c97d8c18cbe8d274d85a07c76b8934243fae3",
    "event": "PurchaseOrderConfirmed",
  }
]

```

FIGURE 10. Logs showing a successful purchase confirmation by the Purchase Manager.

F. CONFIRMING THE PURCHASE ORDER REQUEST

The engineer submits the PO request which triggers an event announcing the submission of the request and updates the state to ‘PurchaseOrderSubmitted’, as presented in Figure 9. This PO request is approved by the line manager before the purchase manager confirms the purchase.

G. ANNOUNCING SPARE PART AVAILABILITY IN THE INVENTORY

When the spare part is successfully procured and made available in the inventory, the inventory manager creates a transaction to announce its availability and readiness for collection. This is performed using the function *AnnounceAvailabilityInInventory*, as seen in Figure 11.

```

status 0x1 Transaction mined and execution succeed
transaction hash 0xd4538ae116b7807cd818ee0a2ddb1e58b5a5471a397154d7130789e5fe6b1a □
from 0xb897b0511fd7c541bd9d7e929c4e53ed42d2b □
to SpareParts.AnnounceAvailabilityInInventory() 0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2 □
gas 3000000 gas □
transaction cost 291919 gas □
execution cost 8647 gas □
hash 0xd4538ae116b7807cd818ee0a2ddb1e58b5a5471a397154d7130789e5fe6b1a □
input 0x146...fb1bc □
decoded input () □
decoded output () □
logs [
  {
    "from": "0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2",
    "topic": "0xb66c6871bc9d9836132d20a60fd857b7740fabfa08890ca6defe3618751b7a00",
    "event": "SparePartNowAvailableInInventory",
  }
]

```

FIGURE 11. Logs indicating that the spare part is now available locally.

H. COLLECTING THE SPARE PART

The engineer collects the spare part from the local inventory and logs the event using the function *CollectRequestedSpare*

Part to confirm and announce its collection. Also, the state of the spare part is updated to ‘NeedsPR’ which indicates that a new purchase request needs to be filed to place the order of a new spare part. A successful transaction can be seen in Figure 12.

```

status 0x1 Transaction mined and execution succeed
transaction hash 0xf6556e646886645b1cad39b011ad363a101f4da68a4c5ac7ba0e7bf44efc8c93a □
from 0xca35b7d915458ef5340de068df2e744e8fa73c □
to SpareParts.CollectRequestedSparePart() 0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2 □
gas 3000000 gas □
transaction cost 31611 gas □
execution cost 10339 gas □
hash 0xf6556e646886645b1cad39b011ad363a101f4da68a4c5ac7ba0e7bf44efc8c93a □
input 0x900...ed705 □
decoded input () □
decoded output () □
logs [
  {
    "from": "0x15e08fa9fe3e3aa3607ac57a29f92b5d8cb154a2",
    "topic": "0x000034eef26e86f30ed41d4ac3f783bb4d80e617b4f71dd7a045e2d",
    "event": "SparePartCollectedByEngineer",
  }
]

```

FIGURE 12. Logs showing a successful collection of the spare part by the Engineer.

V. DISCUSSION

The proposed blockchain-based system is evaluated in terms of security and cost to verify the reliability and feasibility of the solution.

A. SECURITY ANALYSIS

Blockchain can be vulnerable to attacks that target mining pools, smart contracts, and even digital wallets. For instance, a smart contract code of the Ethereum based Decentralized Autonomous Organization (DAO) has recently been exploited through a well-known hack named as the ‘never ending ATM’. Sybil attacks are another type of attacks that can target the blockchain nodes. They empower a single node to take control over the network and block other users from using it. The attack that happened on the DAO can be prevented by examining the smart contract code for possible vulnerabilities. Moreover, consensus algorithms (e.g., proof of work and proof of stake) in blockchain can make the Sybil attacks impractical for the hacker [30]. In this way, blockchain can help to reduce the chances of attacks drastically.

The implementation code that is publicly made available on GitHub has been validated using a security tool named ‘SmartCheck’ and verified successfully against known vulnerabilities. The tool helps to evaluate and eradicate weaknesses in the code. Exploits and errors that can cause great losses and pose serious threats to data security and the participating entities are detected. Such bugs include time stamp dependence, malicious libraries, reentrancy, and locked money. Therefore, the code is free from the known threats and since it has no fall back functions or loops it is less vulnerable compared to other smart contracts.

On the other hand, by design, Blockchain has built-in security features that allow for building a trusted, secure, and resilient services and networks. For instance, security requirements such as integrity, authorization, non-repudiation, availability, and privacy can be easily achieved using blockchain [31]. Table 1 highlights key spare parts supply

TABLE 1. An overview of spare parts supply chain challenges that are addressed using the blockchain technology.

Challenge	Description (How blockchain deals with it)
Integrity	(a) The participating stakeholders in the spare parts supply chain such as OEM and supplier can digitally sign transactions to ensure that the integrity of spares part data will be preserved. (b) The immutability of recorded transactions and data also assures the high integrity in the spare parts supply chain.
Non-repudiation	(a) All transactions of spare parts supply chain are digitally and cryptographically signed by their actors. This is done using the Ethereum address in permissionless blockchain. Also this is achieved by the role of the identity authenticator in permissioned blockchain. (b) The Membership Service Provider (MSP) in HLF determines the identity and access privileges of each user. Therefore, blockchain participants and actors involved in spare parts supply chain cannot deny their actions.
Authorization	(a) The role of different stakeholders in spare parts supply chain is different. The authorization feature assures that only authorized stakeholder can perform a specific task. For example, only engineer can trigger a smart contract function to purchase a spare part from the OEM.
Privacy	(a) The channels and orion node features of HLF and BESU can be used to ensure that the stakeholder of spare parts supply chain can communicate in a highly private environment.
Availability	(a) The transaction logs of stakeholders involved in the spare parts supply chain are always available that can assist to trace the provenance of a spare part. (b) Duplicated spare part records are stored on the blockchain nodes, thereby making the system robust and resilient against single point of failure.

chain challenges and how blockchain can help to resolve such challenges.

B. COST ANALYSIS

Any transaction on the chain costs a certain amount of Ether or cryptocurrency. The transaction fee of every function is calculated as gas in the Ethereum blockchain, but the cost is paid in ‘Ether’. On the other hand, the ‘gas price’ provides the cost in Ether for a unit of gas, which is given the unit ‘Gwei’. Therefore, a higher amount of Gwei would incentivize the miners and elevate the transaction’s priority. Hence, it is important to study and examine the code for possible ways of decreasing the cost and making it more efficient. In our Ethereum smart contract, we have a dependency on the logs to trigger events and trace the spare parts details. There are no arrays or mappings used. Moreover, we have used bytes32 whenever possible instead of strings to increase the efficiency and the feasibility of the solution.

Table 2 shows the cost details of the algorithms used in the smart contract. The table shows the algorithm number based on the execution order in the smart contract and the order of appearance in section III (the Implementation Details). The table shows the transaction cost, which includes the execution cost as gas. To calculate the cost in USD (\$), we have used the gas price in the Eth Gas Station. Based on the gas prices

offered by the ETH Gas Station on May 7, 2020, the fastest, fast, average, and cheap gas prices were found as 10, 6.6, 4, and 4 Gwei, respectively [32]. In table 2, we have used the average gas price of 4 Gwei to calculate the cost in USD. The highest cost is \$ 0.0264 which is almost negligible. The calculated cost for the algorithms is as expected as they follow a constant time complexity O(1). The algorithms do not have any loops or arrays as they only used to trigger notifications. Hence, the cost is very minimal.

C. TRACING SPARE PARTS REQUIREMENTS

The spare parts are reliably being tracked from the OEM to the suppliers and end-users using the proposed blockchain system. Provenance data is available on the chain using logs and events which capture the details of all on-chain transactions. The proposed solution is decentralized that ensures non-repudiation and accountability as every transaction is signed by the address of the caller. Each function is restricted using modifiers to only authorize a specific participant to execute it. Furthermore, because of data integrity, transparent, and tamper-proof features of the blockchain, trust is easily acquired and is part of the solution. Also, the proposed solution relies on IPFS for the storage of the spare part details and information. The proposed solution relies on the security features of the blockchain technology that make it resilient and reliable. Finally, our solution can be easily implemented on permissioned or permissionless blockchain based on the needs and requirements of enterprises.

TABLE 2. Gas cost in USD of the smart contract algorithms.

Algorithm Number	Transaction Gas	Execution Gas	Cost USD
1	30993	9721	0.02542
2	30298	8834	0.02485
3	30296	88321	0.02485
4	32136	8752	0.02634
5	32201	8817	0.0264
6	30297	8833	0.02485
7	30231	8767	0.02478
8	29895	8623	0.02452
9	30319	8855	0.02487
10	29961	8689	0.02456
11	29919	8647	0.02454
12	31611	10339	0.02591

D. OPEN CHALLENGES

Blockchain is a promising technology that can play a key role in shaping and transforming the way spare parts are managed, traced, and tracked. However, Blockchain is still maturing and is relatively a new technology facing many challenges that need to be overcome before being widely adopted and deployed in the supply chain management domain. Some of these challenges include scalability, governance, energy consumption, privacy, security of smart contract codes, transaction costs, and interoperability [33].

It is worth noting that blockchain is the preferred platform to record and store alerts about serious notifications and violations that may be triggered by blockchain-enabled IoT devices to trace and track shipment, products, or equipment during the supply chain phases [34], [35]. In today's Cyber Physical Systems (CPS), the accuracy of the reported data to the blockchain are as accurate as the devices reporting them. More specifically, the accuracy of the data stored on-chain depends on the source that is feeding the data to the network. By design, blockchain ensures that the stored data and relevant transactions are immutable, secure, and tampered-proof with trusted data provenance and history [36]. Blockchain also ensures authentication of data origins. It is to be noted that blockchain can be a computational expensive platform for storing large-size data. Therefore, blockchain solution and systems usually integrate decentralized storage technologies like IPFS or SWARM, or cloud platforms, for storing large-size data and files. The hashes of these large-size data are then stored in blockchain.

VI. CONCLUSION

In this paper, we have proposed blockchain-based tracing and tracking of spare parts ownership from the OEM to the suppliers and end-users. We developed smart contracts to implement the functions, modifiers, and trigger events. In our proposed solution, we integrated decentralized storage services (IPFS) to store the details of the spare parts and quotations of the suppliers. We provided detailed discussions on implementation, testing, and validation aspects. Our proposed solution is generic enough and can be adopted for permissioned or permissionless blockchain platforms depending on the use case. Hyperledger Fabric and Hyperledger Besu are two possible permissioned blockchains where authorization, access control, and confidential communication can be achieved. For future work, we plan to implement and evaluate the cost and performance of our blockchain solution when deployed on these two permissioned networks, and show how such deployment and implementation can satisfy defined system requirements. The implemented code of the smart contract was made publicly available on GitHub and it has been successfully verified for the known vulnerabilities using the SmartCheck tool. We presented cost analysis to demonstrate the practicality to use and adopt our proposed solution by users and organizations involved in the trading, management, and traceability of spare parts. Overall, our technical details presented in this paper (with regards to blockchain system design, architecture, smart contracts and algorithms) are generic enough and can be used and integrated in most popular blockchain platforms.

REFERENCES

- [1] *Global Spare Parts Logistics Market is Driven by the Rapidly Growing Automotive Industry Across the Worldwide*. Accessed: Jan. 2, 2020. [Online]. Available: <https://apnews.com/Wired%20Release/65af4943529532114867d455bac779c4>
- [2] Y. Madhwal and P. B. Panfilov, "Industrial case: Blockchain on aircraft's parts supply chain management," in *Proc. Amer. Conf. Inf. Syst. Workshop Smart Manuf.*, vol. 6, 2017, p. 1–6
- [3] Y. Chang, E. Iakovou, and W. Shi4, "Blockchain in global supply chains and cross border trade: A critical synthesis of the State-of-the-art, challenges and opportunities," 2019, *arXiv:1901.02715*. [Online]. Available: <http://arxiv.org/abs/1901.02715>
- [4] *Launchpad to Relevance: Aerospace & Defense Technology Vision 2018*. Accessed: Jan. 2, 2020. [Online]. Available: <https://www.accenture.com/us-en/insights/high-tech/launchpad-relevance>
- [5] *Honeywell is now Tracking \$1 Billion in Boeing Parts on a Blockchain*. Accessed: Mar. 17, 2020. [Online]. Available: [https://www.forbes.com/sites/michaeldelcastillo/2020/03/07/honeywell-is-now-tracking-1-billion-in-boeing-parts-on-a-blockchain/65643d0f78bf](https://www.forbes.com/sites/michaeldelcastillo/2020/03/07/honeywell-is-now-tracking-1-billion-in-boeing-parts-on-a-blockchain/)
- [6] D. Mocenco, "Supply chain features of the aerospace industry. Particular case airbus and Boeing," *Sci. Bull.-Econ. Sci.*, vol. 14, no. 2, pp. 17–25, 2015.
- [7] Y. Madhwal and P. B. Panfilov, "Blockchain and supply chain management: Aircrafts' parts' business case," in *Proc. Ann. DAAAM*, vol. 28, 2017, pp. 1–7.
- [8] D. Rajkov, "Blockchain for aircraft spare part management: Evaluating the robustness of the maintenance, repair and overhaul business model," M.S. thesis, Delft Univ. Technol., Delft, The Netherlads, 2018.
- [9] J. Müller, "Supporting change in business process models using pattern-based constraints," in *Enterprise, Business-Process and Information System Modeling*. Berlin, Germany: Springer, 2009, pp. 27–32.
- [10] R. Klueber, H. McCann, R. Wehrle, and T. Knopf, "Service innovation in spare parts logistics in the business aviation industry," in *Proc. IEEE Conf. Commerce Enterprise Comput.*, Vienna, Austria, Jul. 2009, pp. 317–323.
- [11] *How Manufacturers can use Blockchain for Inventory Management*. Accessed: Mar. 22, 2020. [Online]. Available: <https://www.bbntimes.com/technology/how-manufacturers-can-use-blockchain-for-inventory-management>
- [12] C. Wickboldt and N. Kliewer, "Blockchain for workshop event certificates—A proof of concept in the aviation industry," in *Proc. 27th Eur. Conf. Inf. Syst.*, Stockholm, Sweden, 2019, pp. 1–15.
- [13] T. K. Dasaklis, F. Casino, C. Patsakis, and C. Douligeris, "A framework for supply chain traceability based on blockchain tokens," in *Proc. Int. Conf. Bus. Process Manage.* Vienna, Austria: Springer, 2019, pp. 704–716.
- [14] S. I. Satoglu and A. Ustundag, "Value of RFID enhanced maintenance in aerospace industry," in *The Value of RFID*. London, U.K.: Springer, 2013, pp. 141–153.
- [15] N. Julka, A. Thirunavukkarasu, P. Lendermann, B. P. Gan, A. Schirrmann, H. Fromm, and E. Wong, "Making use of prognostics health management information for aerospace spare components logistics network optimisation," *Comput. Ind.*, vol. 62, no. 6, pp. 613–622, Aug. 2011.
- [16] K. Kolinska, B. Sliwczynski, L. Hadas, and A. Kolinski, "Analysis of spare parts in terms of their availability management for the production processes needs," in *Proc. Bus. Logistics Modern Manage.*, 2018, pp. 191–204.
- [17] E. F. Israel, A. Albrecht, E. M. Frazzon, and B. Hellingrath, "Operational supply chain planning method for integrating spare parts supply chains and intelligent maintenance systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12428–12433, 2017.
- [18] S. Keivanpour and D. A. Kadi, "The effect of 'Internet of Things' on aircraft spare parts inventory management," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 2343–2347, 2019.
- [19] G. Meroni and P. Plebani, "Combining artifact-driven monitoring with blockchain: Analysis and solutions," in *Proc. Int. Conf. Adv. Inf. Syst. Eng.* Cham, Switzerland: Springer, 2018, pp. 103–114.
- [20] Q. Cao, D. G. Schniederjans, and M. Schniederjans, "Establishing the use of cloud computing in supply chain management," *Oper. Manage. Res.*, vol. 10, nos. 1–2, pp. 47–63, Jun. 2017.
- [21] Y. Wu, C. G. Cegielski, B. T. Hazen, and D. J. Hall, "Cloud computing in support of supply chain information system infrastructure: Understanding when to go to the cloud," *J. Supply Chain Manage.*, vol. 49, no. 3, pp. 25–41, Jul. 2013.
- [22] H. M. Kim and M. Laskowski, "Toward an ontology-driven blockchain design for supply-chain provenance," *Intell. Syst. Accounting, Finance Manage.*, vol. 25, no. 1, pp. 18–27, Jan. 2018.
- [23] S. K. Roy, R. Ganguli, and S. Goswami, "Transformation of supply chain provenance using blockchain—A short review," in *Emerging Technology in Modelling and Graphics*. Singapore: Springer, 2020, pp. 583–593.
- [24] *IPFS is the Distributed Web*. Accessed: Jun. 13, 2018. [Online]. Available: <https://ipfs.io/>

- [25] *Announcing Hyperledger Besu*. Accessed: Nov. 19, 2019. [Online]. Available: <https://www.hyperledger.org/blog/2019/08/29/announcing-hyperledger-besu>
- [26] *Ethereum Blockchain Solutions*. Accessed: Nov. 19, 2019. [Online]. Available: <https://pegasys.tech/>
- [27] *Remix*. Accessed: Nov. 19, 2019. [Online]. Available: <https://remix.ethereum.org/>
- [28] H. Hasan and K. Salah, "Blockchain-based solution for proof of delivery of physical assets," *Proc. Int. Conf. Blockchain (ICBC)*, in Lecture Notes in Computer Science book series, vol. 10974, Seattle, WA, USA: Springer, Jun. 2018, pp. 139–152.
- [29] H. R. Hasan and K. Salah, "Blockchain-based proof of delivery of physical assets with single and multiple transporters," *IEEE Access*, vol. 6, pp. 46781–46793, 2018.
- [30] J. H. Mosakheil, "Security threats classification in blockchains." M.S. thesis, Dept. Graduate, St. Cloud State Univ., St. Cloud, MN, USA, 2018.
- [31] H. Min, "Blockchain technology for enhancing supply chain resilience," *Bus. Horizons*, vol. 62, no. 1, pp. 35–45, Jan. 2019.
- [32] *Eth Gas Station*. Accessed: Nov. 11, 2018. [Online]. Available: <https://ethgasstation.info/>
- [33] *The Hurdles & Current Challenges of Using Blockchain in Supply Chain & Logistics*. Accessed: Feb. 24, 2020. [Online]. Available: <https://cerasis.com/using-blockchain-in-supply-chain-logistics/>
- [34] K. Salah, A. Alfalesi, M. Alfalesi, M. Alharmoudi, M. Alzaabi, A. Alzyeodi, and R. W. Ahmad, "IoT-enabled shipping container with environmental monitoring and location tracking," in *Proc. IEEE 17th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Lasvegas, NV, USA, Jan. 2020, pp. 1–7.
- [35] P. J. Zelbst, K. W. Green, V. E. Sower, and P. L. Bond, "The impact of RFID, IIoT, and blockchain technologies on supply chain transparency," *J. Manuf. Technol. Manage.*, vol. 31, no. 3, pp. 441–457, Oct. 2019.
- [36] G. Perboli, S. Musso, and M. Rosano, "Blockchain in logistics and supply chain: A lean approach for designing real-world use cases," *IEEE Access*, vol. 6, pp. 62018–62028, 2018.



HAYA R. HASAN received the B.S. degree in computer engineering from the American University of Sharjah, United Arab Emirates, in 2014, and the master's degree in electrical and computer engineering from Khalifa University, United Arab Emirates, in 2018. She is currently a Research Associate with the Department of Industrial and Systems Engineering, Khalifa University. She is passionate about research especially in the field of blockchain and smart contracts. She has publications in her area of interest, blockchain as well as in security.



KHALED SALAH (Senior Member, IEEE) received the B.S. degree in computer engineering with a minor in computer science from Iowa State University, USA, in 1990, the M.S. degree in computer systems engineering and the Ph.D. degree in computer science from the Illinois Institute of Technology, USA, in 1994 and 2000, respectively. He is currently a Full Professor with the Department of Electrical and Computer Engineering, Khalifa University, United Arab Emirates. He has over 220 publications and three U.S. patents, has been giving a number of international keynote speeches, invited talks, tutorials, and research seminars on the subjects of blockchain, the IoT, fog, and cloud computing, and cybersecurity. He has served as the Chair of the Track Chair of the IEEE Globecom 2018 on Cloud Computing. He is an Associate Editor of the IEEE Blockchain Tech Briefs, and a member of IEEE Blockchain Education Committee. He is also leading a number of projects on how to leverage blockchain for healthcare, 5G networks, combating deepfake videos, supply chain management, and AI.



RAJA JAYARAMAN received the bachelor's and master's degree in mathematics from India, the Master of Science degree in industrial engineering from New Mexico State University, and the Ph.D. degree in industrial engineering from Texas Tech University. He is currently an Associate Professor with the Department of Industrial and Systems Engineering, Khalifa University, Abu Dhabi, United Arab Emirates. His expertise is in multi-criteria optimization techniques applied to diverse applications including supply chain and logistics, healthcare, energy, environment, and sustainability. His research interests include primarily focused on using blockchain technology, systems engineering and process optimization techniques to characterize, model and analyze complex systems with applications to supply chains, maintenance operations planning and healthcare delivery. His Postdoctoral Research was centered on technology adoption and implementation of innovative practices in the healthcare supply chains and service delivery. He has led several successful research projects and pilot implementations in the area of supply chain data standards adoption in the US healthcare system. His research has appeared in top-rated journals including: *Annals of Operations Research*, IIE Transactions, *Energy Policy*, *Applied Energy*, *Knowledge Based Systems*, *IEEE Access*, the *Journal of Theoretical Biology*, *Engineering Management Journal*, and others.



RAJA WASIM AHMAD received the master's degree in computer science from COMSATS University Islamabad (CUI), Abbottabad Campus, in 2011, and the Ph.D. degree in computer science from the Center for Mobile Cloud Computing Research (C4MCCR), Faculty of Computer Science and Information Technology (FSKTM), University of Malaya, Malaysia. He is currently working as a Postdoctoral Research Fellow with Khalifa University, Abu Dhabi, United Arab Emirates. He has published several research articles in leading journals and conferences. Most of his research contributions are published in top-cited journals, such as the *IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING*, the *IEEE SYSTEMS JOURNAL*, *IEEE Access*, the *Journal of Networks and Computer Applications*, the *Journal of Grid Computing*, *Cluster Computing*, the *International Journal of Communication Systems*, the *International Journal of Information Management*, the *Journal of Systems and Software*, the *Journal of Supercomputing*, *IET Intelligent Transport Systems*, and *Renewable and Sustainable Energy Reviews*. He is also serving as a reviewer for several journals and conferences. His research work is well acknowledged in national and international conferences. One of his research articles has received the Best Paper Award in SPECTS symposium held, in France.



IBRAR YAQOOB (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Malaya, Malaysia, in 2017. He is currently working with the Department of Electrical Engineering and Computer Science, Khalifa University, United Arab Emirates. Previously, he worked as a Research Professor at the Department of Computer Science and Engineering, Kyung Hee University, South Korea, where he completed his postdoctoral fellowship under the prestigious grant of Brain Korea 21st Century Plus. He worked as a Researcher and Developer at the Centre for Mobile Cloud Computing Research (C4MCCR), University of Malaya. His numerous research articles are very famous and among the most downloaded in top journals. He has been listed among top researchers by Thomson Reuters (Web of Science) based on the number of citations earned in the last three years in six categories of Computer Science. He is currently serving/served as a Guest/Associate Editor in various journals. He has been involved in a number of conferences and workshops in various capacities. His research interests include big data, blockchain, edge computing, mobile cloud computing, the Internet of Things, healthcare, and computer networks.



MOHAMMED OMAR is currently a Full Professor and the Founding Chair of the Department of Engineering Systems and Management (currently renamed industrial and systems engineering). Prior to joining the Masdar Institute/KUST, he was an Associate Professor and a Graduate Coordinator with Clemson University, Clemson, SC, USA. He was a part of the Founding Faculty Cohort of Clemson University Research Park, Greenville, SC. He has over 100 publications in the areas of product lifecycle management, knowledge-based manufacturing, and automated testing systems, in addition to authoring several books and book chapters. He holds four U.S. and international patents. He was named a Tennessee Valley Authority Fellow of two consecutive years during the Ph.D. degree, in addition to being a Toyota Manufacturing Fellow. His professional career includes a Postdoctoral service at the Center for Robotics and Manufacturing Systems CRMS, and a Visiting Scholar at the Toyota Instrumentation and Engineering Division, Toyota Motor Company, Japan. His group graduated seven Ph.D. dissertations and over

35 M.Sc. theses. Four Ph.D. students are currently on academic ranks in U.S. universities. His current research interests include capabilities in composite fabrication and manufacturing analytics at a laboratory Masdar City Campus. His work has been recognized by the U.S. Society of manufacturing engineers SME through the Richard L. Kegg Award. He received the SAE Foundation Award for Manufacturing Leadership and the Murray Stokely Award from the College of Engineering, Clemson University. He has also led an NSF I/UCRC Center and a part of the DoE GATE Center of Excellence in Sustainable Mobility Systems. His current research group supported two Postdoctoral Scholar's Career Planning to become an Assistant Professor at the Texas A&M (TAMUQ), in 2013, and the University of Sharjah, in 2015. He currently serves as an Editor-in-Chief for the *Journal of Material Science Research* (Part of the Canadian Research Center), and as an Associate Editor for the *Journal of Soft Computing* (Springer), handling the areas of decision science, knowledge-based systems, in addition to his membership on several editorial boards and conference organizations. Furthermore, he serves on the Advisory Board of the Strata PJSC (part of Mubadala Aerospace).

• • •