engineering proceedings

Proceeding Paper

Design and Implementation of Smart Contract in Supply Chain Management Using Blockchain and Internet of Things †

Fatima Haider Naqvi [1], Sundus Ali [1,*], Binish Haseeb [1], Namra Khan [1], Soomal Qureshi [1], Taha Sajid [2] and Muhammad Imran Aslam [1]

1 Department of Telecommunications Engineering, NED University of Engineering and Technology, Karachi 75270, Pakistan; naqvi4105572@cloud.neduet.edu.pk (F.H.N.); haseeb4102235@cloud.neduet.edu.pk (B.H.); khan4200613@cloud.neduet.edu.pk (N.K.); soomal4102751@cloud.neduet.edu.pk (S.Q.); iaslam@neduet.edu.pk (M.I.A.)
2 Comcast Corporation, 1701 JFK Boulevard, Philadelphia, PA 19103, USA; tahasajid04@hotmail.com
* Correspondence: sundus@neduet.edu.pk

† Presented at the 2nd International Conference on Emerging Trends in Electronic and Telecommunication Engineering, Karachi, Pakistan, 15–16 March 2023.

Abstract: In this paper, we have presented the design and implementation of a blockchain-based approach for ensuring reliable supply chain management for commodities transported through smart containers. To administer interactions between the sender and receiver, our developed system makes use of the Ethereum blockchain's smart contract features. Smart containers equipped with Internet of Things (IoT)-enabled sensors are used to monitor shipping conditions to check predefined shipping requirements. Smart contracts on Ethereum are used to automate payments, validate receivers, and give refunds in the case of violation of predefined requirements. We have also implemented our designed front-end decentralized WebApp and wallet that allows the sender and receiver to communicate with Ethereum smart contracts.

Keywords: smart contract; Internet of Things; supply chain management; blockchain; Ethereum

Citation: Naqvi, F.H.; Ali, S.; Haseeb, B.; Khan, N.; Qureshi, S.; Sajid, T.; Aslam, M.I. Design and Implementation of Smart Contract in Supply Chain Management Using Blockchain and Internet of Things. *Eng. Proc.* **2023**, *32*, 15. https://doi.org/10.3390/engproc2023032015

Academic Editors: Muhammad Faizan Shirazi and Saba Javed

Published: 25 April 2023

1. Introduction

Supply chain management (SCM) has a tremendous impact on the global economy in today's market. The transfer of goods from producer to consumer is a common definition of SCM. It involves the manufacturer, distributor, and retailer and is divided into several phases, beginning with the delivery of raw materials and ending with the client. It is a worldwide process that involves sourcing components from a single location, packaging them, and shipping them all over the world. With each passing day, technological advancements and their benefits invade more of our lives; more government, economic, and social functions, among other things, are accessible via the internet [1]. We bank online, shop online, and log into applications and services to create our digital selves and send data back and forth. Because of its decentralized, peer-to-peer transaction, distributed consensus, and anonymity qualities, blockchain technology has become a popular research domain in the last few years [2]. The blockchain is powered by executing smart contracts. A smart contract is a collection of self-verifying, self-executing, and tamper-resistant programs. Smart contracts that incorporate blockchain technology are capable of doing tasks in real-time at a lower cost and with a higher level of security [2,3]. In blockchain technology, smart contracts are implemented by digital nodes without any human intervention [4]. As a result, smart contracts may run independently and precisely as planned, with no risk of downtime, censorship, fraud, or third-party interference [5]. Smart contracts eliminate the need for a middleman to oversee transactions, allowing equitable transactions to take place directly between participants. Blockchain has the ability to solve the single point of failure problem in cloud-based IoT infrastructure, resulting in a more resilient ecosystem [6].

IoT solutions can enable communications between devices by leveraging smart contracts, which model agreement between the two devices, by leveraging blockchain [7].

In this paper, we have presented the design, development, and working of our developed prototype, which is a blockchain-based, secure, cold supply chain management system that uses an IoT network [8] to monitor the shipping condition of vaccines to ensure compliance to the requirements under secure and scalable methods [9]. Recently, authors in [10] have emphasized the significance of integrating blockchain technology in the IoT environment to ensure trust among IoT devices. They have outlined some important challenges and issues of trusted IoT with how these can be solved by blockchain. They have also presented a comparative analysis between traditional and blockchain-based trust management techniques. The rest of the paper is structured as follows: in Section 2, we present our system model, elaborating on both the hardware and software components of the system. The system design and interconnection are discussed in Section 3. Implementation results are discussed in Section 4. Section 5 concludes the paper.

2. System Model

We have presented the system model in this section. The system model includes several hardware and software components. The hardware components are discussed in the following sub-section.

2.1. System Hardware

There are several hardware components that were used to develop the prototype of the system. The hardware and its functionality are discussed below:

- The NodeMCU ESP8266 Microcontroller is a compact integrated circuit used to perform specific operations in an embedded system. We have used NodeMCU ESP8266 to make the IoT-based sensor node.
- The DHT11 sensor is an inexpensive digital sensor for sensing the temperature and humidity of the deployed environment.
- The IoT-enabled Container is designed by developing hardware and software interfaces and equipment that will enable remote monitoring of temperature and humidity requirements.

2.2. Software Components

All the software used to build a working mechanism of the system is described below:

- Meta Mask is a popular cryptocurrency wallet. It is a browser plugin that serves as an Ethereum wallet and installs like any other plugin. With the Meta Mask connection to Ethereum-based applications, users can spend their amounts or coins in games, stake tokens, etc.
- Ether scan is a block explorer and block analytics medium that allows users to access details of any pending or confirmed Ethereum blockchain transactions. Using Etherscan, you can see the current state of balance and transaction history by searching with the Ethereum address of the wallet.
- Ropsten Test Network enables builders to experiment with protocol upgrades and decentralized websites and applications before the actual deployment, i.e., on the Ethereum main net. Due to its proof-of-work consensus model, the test net is similar to the mainnet. The Ropsten test net is used by developers due to it presents the production environment of the mainnet of Ethereum, which is based on a proof-of-work mechanism, and to prepare for the upcoming Ethereum mainnet merge.
- FireBase is a platform for development known originally for its real-time database. The applications based on FireBase allow secure and direct access to the database by using client-side code. The changes made by the user on their device are retained and synchronized automatically while being offline with the remote database.

- Arduino IDE Arduino Software Integrated Development Environment (IDE) contains a text editor for code writing, a message area, a text console, a toolbar with buttons for common functions, and a series of menus.
- Remix IDE allows the development and deployment of smart contracts for the Ethereum blockchain. Solidity language is used to write and implement smart contracts. The programming languages and libraries used to build the system include Web3.js, Solidity programming language, React, and React.js, which offer a range of extensions for application architectural support, for instance, Flux and React Native.

3. System Design

For our IoT module, we connected the DHT11 sensor to the ESP8266 node MCU and then placed it in the cold storage container. This IoT module will detect the temperature and humidity of the container and send it to Arduino IDE, where we will view it in the Arduino IDE terminal as well as in FireBase real-time database for storage. Figure 1 depicts the end-to-end system design.

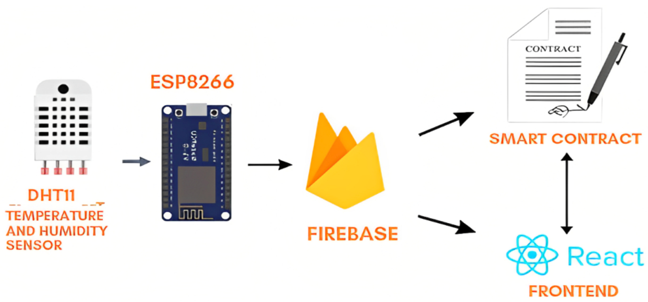
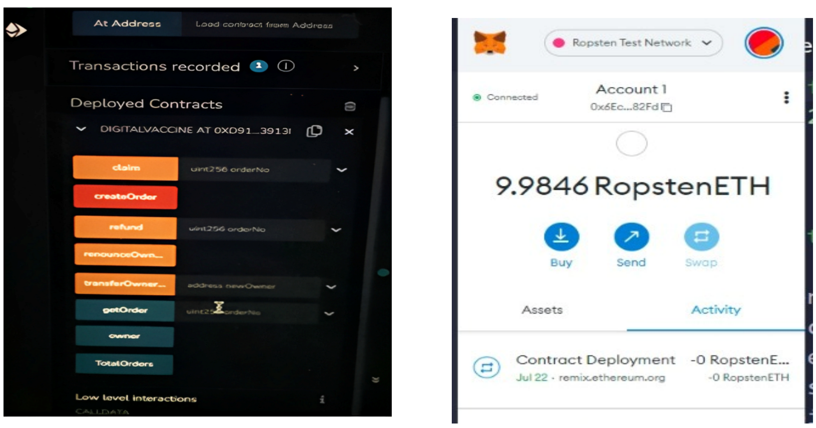


Figure 1. End-to-end system diagram.

A smart contract is written in the Solidity programming language, defining all the triggering events of the shipment based on temperature and humidity data. The contract is tested using Remix IDE and deployed using Truffle suite with Infura. Etherscan block explorer is employed to track all transactions, balance inquiries, and block information while the transactions are handled by Metamask. The Metamask wallet and main functions of the smart contracts are shown in Figure 2a,b. For the purpose of integration, we have used React JS. We call the data from Firebase using a secret key and host link to React JS. Once data is called, the next step is to call the already deployed smart contract to React Js. For this, we used Web3. After calling both the data and the smart contract, we will integrate the data from the IoT module and smart contract; the conditions for temperature are tested, and based on whether the conditions are met or not, the functions in the smart contract are called. A complete system design flow is illustrated in Figure 3.



(**a**) (**b**)

Figure 2. Meta mask wallet (**a**) deployed using Remix IDE and (**b**) showing transactions.

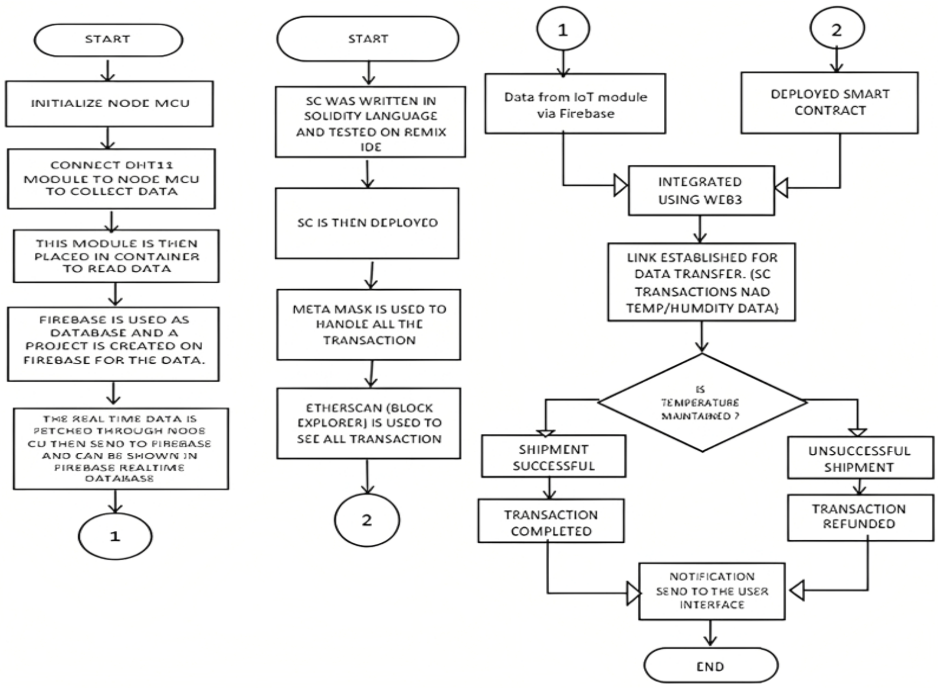
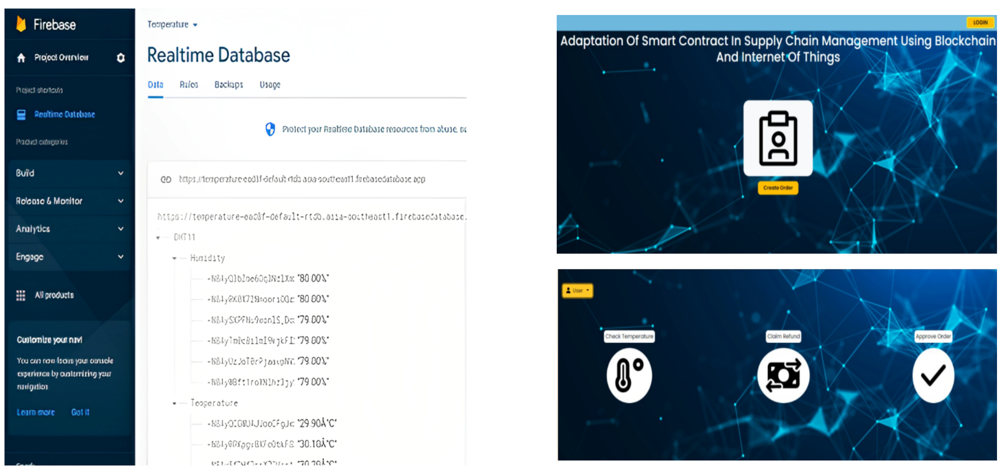


Figure 3. System design flowchart.

4. Results and Discussion

The fetched data is visualized using an Arduino monitor, and the data fetched is collected over the FireBase database (shown in Figure 4a). Vaccines will be placed in smart containers equipped with sensors that are connected to Arduino, which will be used to track and monitor shipment conditions (temperature) against predefined thresholds. We have used node MCU with sensor DHT11 and connected it to Arduino IDE, where after compiling and uploading the code, we can see the monitored temperature and humidity displayed on the serial monitor. The FireBase real-time database is hosted in the cloud to aggregate, store, and issue all the data generated from the sensors placed within the shipment. The smart contract will manage the transactions, whereas a user interface is created using React.js to interact with the smart contract, payment method, and monitoring of shipment using temperature data shown in Figure 4b.



(**a**) (**b**)

Figure 4. (**a**) Collected data stored in FireBase database. (**b**) User-Interface.

5. Conclusions

In this paper, we have presented the results of adapting smart contracts in supply chain management using IoT. We have developed the hardware composed of NodeMCU and DHT11 sensors to detect and record the temperature and humidity during shipment. For solidity-programmed smart contracts, Remix IDE and Truffle suite with Infura is used for testing and deployment, and Metamask is used for transaction handling, which are all based on data received by the IoT system. Self-reporting, self-monitoring, immutability, temper-proofing, accuracy, and transparency are among the features offered by the proposed prototype. The suggested solution's results show that it is feasible in terms of GAS computation and transaction throughput.

Author Contributions: Hardware development: F.H.N. and B.H.; software and app development: N.K. and S.Q.; project coordination and technical guidance: S.A. and M.I.A.; conceptualization and project management: S.A., M.I.A. and T.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by IGNITE (Ministry of IT and Telecommunication Pakistan), National Grassroot ICT Research Initiative (NGIRI), project no. NGIRI-2022-15682.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to extend our gratitude to IGNITE National Grass Root ICT Research Initiative for NGIRI for funding the project.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Saputhanthri, A.; Alwis, C.D.; Liyanage, M. Emergence of Blockchain based IoT Marketplaces. In Proceedings of the Joint European Conference on Networks and Communications (EuCNC) & 6G Summit, Porto, Portugal, 8–11 June 2021.
2. Wang, Z.; Jin, H.; Dai, W.; Choo, K.K.; Zou, D. Ethereum smart contract security research: Survey and future research opportunities. *Front. Comput. Sci.* **2021**, *15*, 152802. [CrossRef]
3. Reyna, A.; Martín, C.; Chen, J.; Soler, E.; Díaz, M. On blockchain and its integration with IoT. Challenges and opportunities. *Future Gener. Comput. Syst.* **2018**, *88*, 173–190. [CrossRef]
4. Ashok, A.; Brison, M.; LeTallec, Y. Improving cold chain systems: Challenges and solutions. *Vaccine* **2018**, *35*, 2217–2223. [CrossRef] [PubMed]
5. Lou, M.; Dong, X.; Cao, Z.; Shen, J. SESCF: A Secure and Efficient Supply Chain Framework via Blockchain-Based Smart Contracts. *Secur. Commun. Netw.* **2021**, *2021*, 8884478. [CrossRef]
6. Hasan, H.; AlHadhrami, E.; AlDhaheri, A.; Salah, K.; Jayaraman, R. Smart contract-based approach for efficient shipment management. *Comput. Ind. Eng.* **2019**, *136*, 149–159. [CrossRef]
7. Anbunathan, R. Smart Home Solution Using Smart Contract. 2022. Available online: https://www.electronicsforu.com/electronics-projects/hardware-diy/smart-home-solution-using-smart-contract (accessed on 31 December 2022).
8. Goda, M.P. Blockchain Enabled Vaccine Supply Chain Provenance. Master's Thesis, The Pennsylvania State University, State College, PA, USA, 2021.
9. Antal, C.; Cioara, T.; Antal, M.; Anghel, I. Blockchain Platform For COVID-19 Vaccine Supply Management. *IEEE Open J. Comput. Soc.* **2021**, *2*, 164–178. [CrossRef]
10. Kumar, R.; Sharma, R. Leveraging blockchain for ensuring trust in IoT: A survey. *J. King Saud Univ.—Comput. Inf. Sci.* **2022**, *34*, 8599–8622. [CrossRef]