Smart Contract-based Rice Supply Chain Traceability: Perspective of Bangladesh

Maliha Rajwana Haque, Sadia Tabassum
Department of Industrial & Production Engineering
Rajshahi University of Engineering & Technology
Rajshahi, Bangladesh
m.rajwana@gmail.com, sadia.tabassum16022@gmail.com
&
Khairun Nahar
Asst. Professor, Department of Industrial & Production Engineering
Rajshahi University of Engineering & Technology
Rajshahi, Bangladesh
khairun.nahar@ipe.ruet.ac.bd

Abstract

Bangladesh's economy draws its main strength from the agriculture sector. Rice is the staple food of the country, and is the most important crop in Bangladesh's agricultural sector. However, this sector is facing numerous challenges. The main challenge is proper tracking of agricultural products and ensuring payment after delivery. As the whole transaction process of products and money currently depends on third parties, both buyers and sellers use agents for delivery and payment of food. Ultimately these agents add additional cost to the system. Blockchain is a technology that can provide a remarkable solution for product traceability in agriculture and food supply chains. In this paper, we have conceptualized a Smart contract-based agricultural system through which buyers and sellers can directly interact with each other. With leveraging the smart contract facility of blockchain, transactions among stakeholders can be recorded, and the need for intermediaries will be eliminated prevailing a central authority system. All transactions will be recorded in Blockchain's immutable ledger which provides a high level of transparency and traceability to the supply chain. Furthermore, in this paper, we have provided data privacy protection solutions through cryptographic primitives and Markle trees that can avoid enterprise privacy and sensitive data leakage. We have also shown cost analysis and performance analysis. We made our smart contract codes publicly available on GitHub.

Keywords

Blockchain, Ethereum, Merkle tree, Smart contract.

1. Introduction

Being an agricultural country, Bangladesh faces numerous problems related to the most complex risk issues in the paddy-rice market. The rice market of Bangladesh plays a significant role in the overall economic performance of the country by means of its contribution to GDP, employment generation, and food security. Any damage that happens in this sector has a huge impact on the well-being of most of the people of this country (M. C. Rahman et al. 2018). Agricultural supply chains of the rice industry are complicated system network that involves various stakeholders such as farmers, millers, retailers, traders, wholesalers, and customers. In the Agricultural food supply chain domain, to maintain the transparency and trust between the total supply chain network and existing participants, it is a must that the collected records have to be tamperproof, which will resolve the case of ensuring transparency in the transaction of payment in between the participants without relying on any kind of integrated third-party intermediary (Q. Y. A. Zhang 2022). In Bangladesh, there is seen price domination by intermediaries in the supply chain of rice and there is a lack of information flow between entities a major setback as well (D. Roy et al. 2019). Lack of transparency in transactions, false storage claim, and losses of input like the seed, and fertilizers are also seen. These irregularities happen as there is a lack of transparency in this rice supply chain (Ahmed, Akhter et al. 2021) And therefore, the potential explication for such a situation and concern is the use of Ethereum Blockchain, which does not depend on any centralized authorities and servers. As each and every record is stored in Ethereum Blockchain which is based on the nature of this technology network, this distributed chain of database ledger is immutable by means of its own configuration of design and provides a transparent source of information flow (Khaled Salah et al.,2019). Our proposed model can eliminate the requirement for a trusted and centralized authority. This model addresses the application of the smart contract-based Blockchain network and secures transparency in the interactions and transactions among all the existing participants who are involved in the supply chain of rice paddy. After each and every transaction, the history of payments is stored and recorded in the Ethereum Blockchain data storage system, Every time when the product changes hands in the network, the information regarding the product will be uploaded to the Ethereum network. There will be an immutable history of the information based on activities related to the transfer of the product from the manufacturer to consumers.

2. Research Framework

Figure 1. The Participants of Rice Supply chain.

In the rice supply chain of Bangladesh between farmers and consumers, there is a group of intermediaries. It starts with the seed company and then to the grower. The product is then sent to traders who transfer it to a rice miller for husking processes. The rice miller sometimes sends it back to traders, while traders sell it to the wholesaler. In some cases, the rice miller directly transmits to the wholesaler. The retailer collects from the wholesaler. In this chain, the consumer is the ending point (M. C. Rahman el al.,2018). All the participants of the rice supply chain will be registered in the Blockchain network. Transference of data will occur through smart contracts. When functions of the smart contracts are called, the caller will receive traceability information about the product. The corresponding actions of the stakeholders of the rice supply chain in the Blockchain network are shown in Figure 1.

2.1 Entity Relationship Diagram

Figure 2 shows the relationship between the entities which are seed company, Farmer, Trader, Miller, Wholesaler, Retailer, Customer in the rice supply chain network of Bangladesh. This illustrates the smart contract attributes and functions and the relationship between the participating entities and the smart contract. Such data and relations are key to implement smart contracts. This diagram shows the entity types and establishes relationship that can exist between them. Here we have considered eight entities and showed how they are interconnected through the help of the smart contracts.

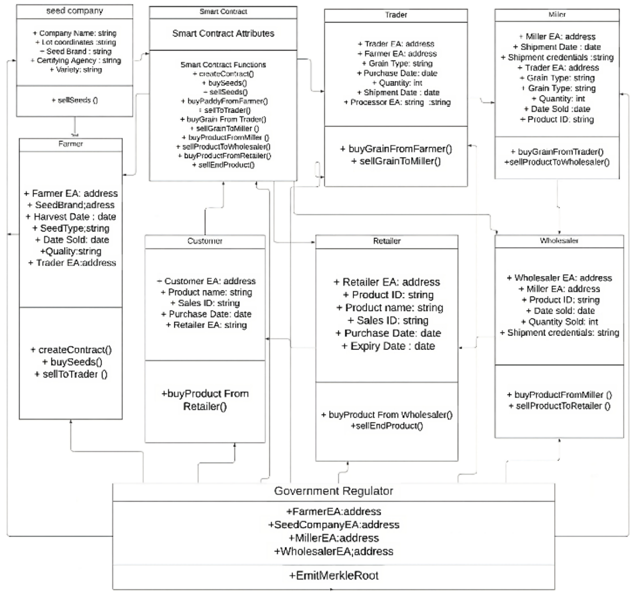

Figure 2. Entity Relationship Diagram

2.2 Work Sequence

These diagrams are created in the platform called Lucid Chart. We created four sequence diagrams to show how these participants are operating the sequence for each and every event.

Figure 3. shows the communication between the seed company, farmer, smart contract, and retailer. The seed company sells the seeds to a farmer and this message is sent to the smart contract. If the seeds are sold successfully then the smart contract will receive and store a message of "sold to farmer". When the smart contract is created, it gives a reaction message named "contract created".

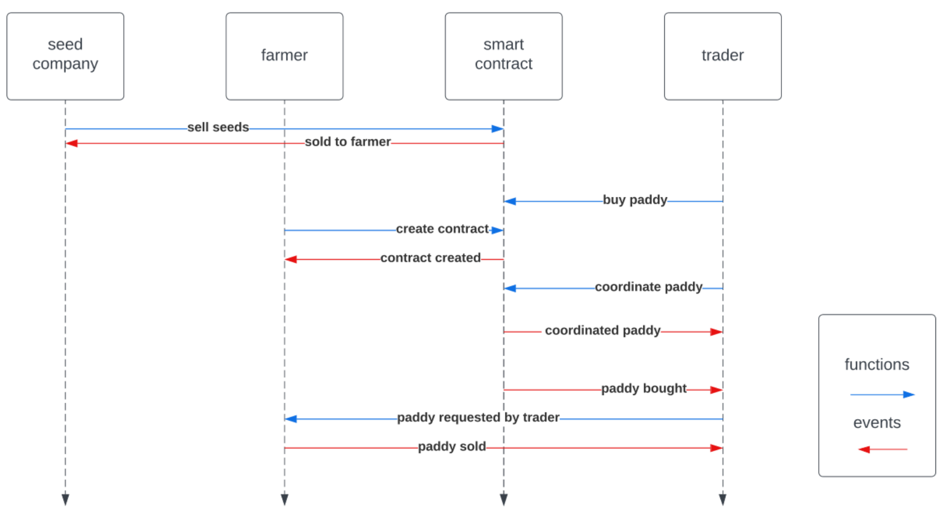

Figure 3. Interaction Between Seed company, Farmer, Smart contract and Trader.

Figure 4 represents the communication between the trader, smart company, miller, and wholesaler. The trader buys paddy and it is then sent to the smart contract as a message. When the wholesaler wants to store rice, he sends the message to the smart contact. Upon completion of the action, "rice stored" is the reaction message given by the smart contract to the wholesaler. And this information will be accessible to all the participants who have access to this Ethereum network. Again as the wholesaler gives the message to buy rice from the miller, the miller will send the message to smart contact as “Collect rice”. And if the rice is collected then the reaction message sent from the smart contract is called “rice collected”. If the coordination of paddy is done, then “paddy coordinated” is the reaction message sent by smart contract to the trader.

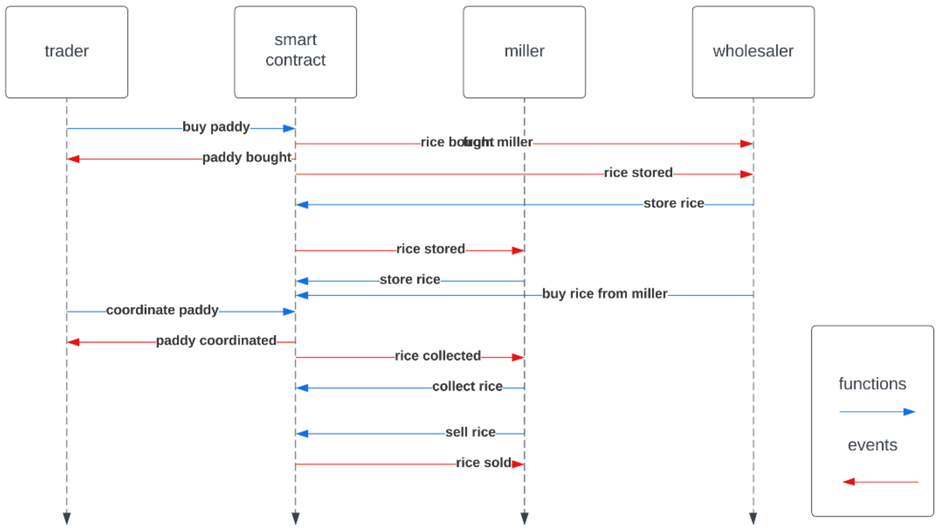

Figure 4. Sequence Diagram for Representing the Interaction Between Trader, Smart contract, Miller and Wholesaler.

Figure 5. demonstrates the sequential communication between wholesalers, smart contracts, retailers, and customers. If the rice is stored, then the message is sent to the wholesaler from the smart contract as “rice stored”. If the wholesaler wants to sell the rice, he sends the message to the smart contract. As a reaction message the smart contract returns “rice sold” to the wholesaler. When the Customer wants to receive the end product the message is also given to the smart contract and a reaction message named “end product received”. When retail wants to supply rice he gives the command to the smart contract and the smart contract then gives a reaction message called “rice supplied”.

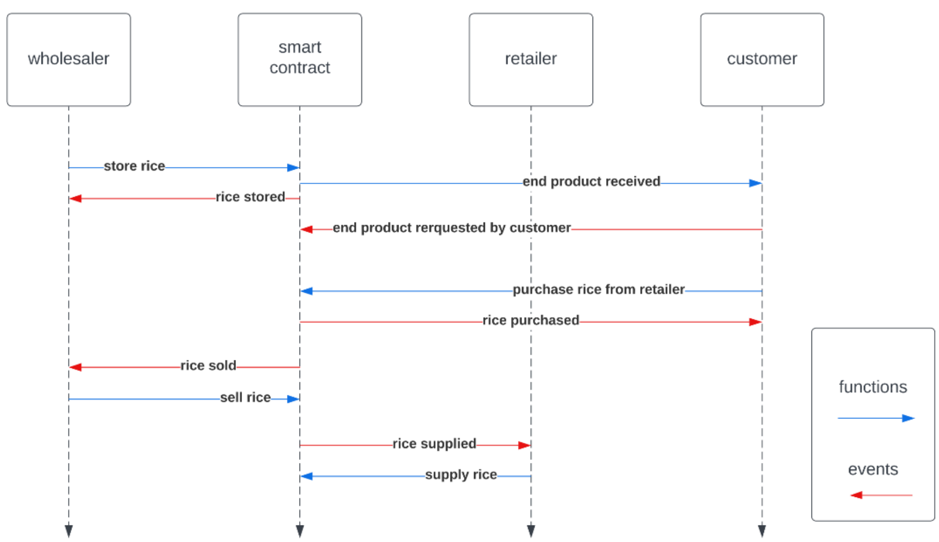

Figure 5. Sequence Diagram for Representing the Interaction Between Wholesaler, Smart contract, Retailer and Customer.

Figure6. shows the sequential communication between the farmer, seed company, miller, wholesaler, retailer, government regulator, and customer. It indicates only the whitelisted participants have access to this ledger. And only if they have access to this network, they can request business qualification. If the participants are not registered or if

the sender's embedded address does not match the registered embedded address, then access will be denied. As a result, the non-Registered participants won't have access to this network.

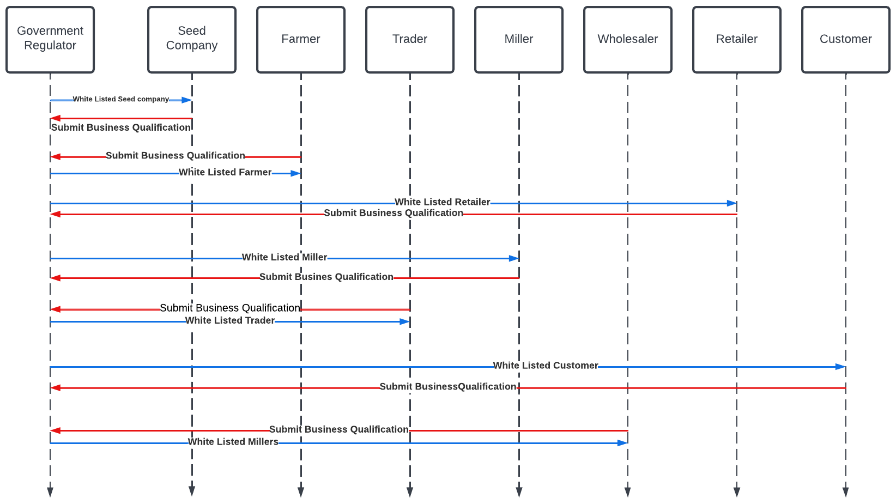

Figure 6. Sequence Diagram Showing Interaction Between All the Participants and Government Regulator.

3. System Testing and Validation

Remix IDE in-browser development and testing environment has been used to test and validate the key functions of the smart contracts. The data was accumulated from DAM (Department of agricultural marketing), DAE (Department of agricultural extension) of Khamarbari, Dhaka for the year 2020-2021 fiscal year.

Algorithm 1: Seed Company Sells Seeds to Farmer

Input: F is the list of registered farmers
Ethereumaddress(EA) of farmer.
Ethereumaddress(EA) of SeedCompany,
Quantity,
Quality,
SeedType,
SeedBrand,
SeedPrice.
Output: Emit an event "seeds sold to farmer"
Contractstate is **Created**
State of the farmer is **Seeds Requested**
Seed Company state is **Ready**
Restrict access to only f € F i.e., registered Farmers
if farmer = registered and SeedPrice = paid **then**
    Contract state changes to *SeedRequestSubmitted*.
    Change State of the farmer to *WaitForSeeds*.
    Seed Company state is *AgreeToSell*
    Create a notification message stating sale of seeds
end
else
    Revert contract state and show an error.
end

Algorithm 2: Miller Buys Grain from Trader

Input: M is the list of registered millers
Ethereumaddress(EA) of miller,

Ethereumaddress(EA) of trader,
Date Purchased,
Purchased quantity,
paddyPrice.

Output: Emit an event "requested paddy from miller sold"
Contractstate is **Buy From Trader**
State of the miller is **paddy Requested**
Trader state is PaddyBought From Farmer
Restrict access to only M € Miller
if PaddySale is agreed and Paddy Price = paid **then**
| Contract state changes to *PaddyRequestAgreed*.
| Change State of the miller to *WaitForMillerFromElevator*.
| Trader state is *SellPaddyToMiller*
| Create a notification message stating sale of paddy to requesting miller
end
else
| Contract state changes to *PaddyRequestFailed*.
| State of miller is Request Failure.
| Grain Elevator state is *CancelRequestOfTrader*
| Create a notification message stating request failure
end
else
| Revert contract state and show an error.
end

1)Transaction between Farmer and Seed Company (Event 1): For the contract between the seed company and the farmer Algorithm 1 describes the process that observes the sale of the seeds by the seed company to the farmer. After the initial state of the contract is declared, the smart contract checks if the farmer requesting the seed is already registered and if the price of the seed is paid. If both the conditions are true, then the state of the contract changes to 'SeedRequestSubmitted', the farmer state changes to 'WaitForSeeds' and the state of the seed company changes to 'AgreeToSell'. It notifies the system about the change of states. Otherwise, the contract state of all participants reverts to the initial state and the transaction is dismissed. Only the registered participants of the Ethereum network can call the functions of the smart contract and an event will trigger notifying about all the traceability information such as the Ethereum address of the farmer, the Ethereum address of the caller, Ethereum address of seed company, purchasing price and quantity and the contract state.

Algorithm 3. Wholesaler Sells Rice to Retailer

Input: 'R' is the list of registered Retailers
Etherenumaddress(EA) of Wholesalers,
Ethereumaddress(EA) of Retailer,
Date of manufacturing,
Price of the quantity sold,
Quantity Sold,
Date Purchased.
Output: Emit an event "product sold to retailer"
Contractstate is *RiceSoldToWholesaler*
Wholesaler state is *RiceReceivedFromMiller*
Retailer state is *ReadyToPurchase*
Restrict access to only R € Retailer
if Sale = agreed and Rice Payment = successful **then**
| Contract state changes to
| *SaleRequestAgreedSuccess.*
| Wholesaler state changes to *RiceSoldToRetailer*.
| Retailer state is Rice *DeliveredSuccessfully*
| Create a 'success' notification message.

```
end
else
  | Contract state changes to SaleRequestDenied.
  | Wholesaler state changes to Request Failed.
  | Retailer state is Rice DeliveryFailure
  | Create a request failure notification message.
end
else
  | Revert contract state and show an error.
end
```

2)Transaction between Trader and Miller (Event 2): For the contract between the trader and miller Algorithm 2 shows the method that observes the sale of paddy by the trader to the miller. If the requesting miller is registered and if the sale of grain is agreed upon and the purchase price is paid; the contract state changes to 'PaddyRequestAgreed', the Miller state changes to 'WaitForMillerFromTrader', Trader state changes to 'SellPaddyToMiller', It notifies the system about the change of states. Otherwise, the contract state of all participants reverts to the initial state and the transaction is dismissed.

3)Transaction between Wholesaler and Retailer (Event 3): For the contract between the wholesaler and retailer Algorithm 3 describes the sale of seeds by the wholesaler to the retailer. If the requesting wholesaler is registered, the sale request is agreed upon and the purchase price is paid; the contract state changes to 'SaleRequestAgreed', the wholesaler state changes to 'RiceReceivedFromMiller', the retailer state changes to 'RiceDeliveredSuccessfully', and all the active entities are notified with a message on the sale of grain. It notifies the system about the change of states. Otherwise, the contract state of all participants reverts to the initial state and the transaction is dismissed.

4)Transaction between Retailer and Customer (Event 4): For the contract between the wholesaler and retailer: Algorithm 4 describes the process of the sale of seeds by the retailer to the customer. If the requesting customer is registered and if the payment is successful; the contract state changes to 'RiceSoldToCustomer', the retailer state changes to 'RiceSaleSuccessful', and the customer state changes to 'SuccessfulPurchase'. The contract notifies all the entities in the chain about the change of states. Otherwise every participant revert to the initial state, and the transaction is terminated.

Algorithm 4. Customer Buys from Retailer

```
Input: Ethereumaddress(EA) of Retailer
        Ethereumaddress(EA) of Customer,
        Date Purchased,
        Total quantity sold,
        Product ID,
        SalesID.
Output: Emit an event "customer rice bought"
Contractstate is SaleRequestAgreedSuccess
Retailer state is RiceDeliveredSuccessfully
Customer state is ReadyToBuy
Restrict access to only Customers
if Rice Payment = successful then
  | Contract state changes to RiceSoldToCustomer.
  | Retailer state is RiceSaleSuccessful
  | Customer state is SuccessfulPurchase
  | Create a 'purchase success' notification message.
end
else
```

```
    Contract state changes to SaleOfRiceDenied.
    Retailer state is RiceSaleFailure
    Customer state is FailedPurchase
    Notify with a 'purchase failure' message.
end
else
    Revert contract state and show an error.
end
```

5)Transaction between Government Regulator to all Stakeholders (Event 5): For the contract between the government regulator to all stakeholders Algorithm 5 describes the process that describes the uploading of the Markle root by the government regulators to the blockchain. This contract will hold the Markle tree root created from the transaction hashes of the other four contracts. The Markle root will be uploaded by the government regulators through this contract. All the participants can get the desired leaf node of the tree containing desired transaction information. Only the registered participants can call this contract and the triggered event with the detailed information (see Figure 7)

Algorithm 5. Government regulator uploads traceability information.

Input: Ethereumaddress(EA) of farmer,
Ethereumaddress(EA) of SeedCompany,
Ethereumaddress(EA) of miller,
Ethereumaddress(EA) of trader,
Etherenumaddress(EA) of Wholesalers,
Ethereumaddress(EA) of Retailer,
Ethereumaddress(EA) of customer.

Output: Emit an event "The markle root of the markle tree"

```
if caller is a registered participant then
    emit an event showing the markle root
end
else
    Revert contract state and show an error.
end
```

3.1 Smart Contract Execution Records

Deployment and Execution of Contract: The solidity program of smart contracts is created using Remix IDE and it is tested and validated through the Goerli test network. The transactions were made using the Metamask wallet. To validate the contracts, implementation and execution costs were collected from the Etherscan website. The smart contract codes are available on GitHub.

1. Creation of Contract between Farmer and Seed Company: After the contract is created between the seed company and farmer, the log of the transaction history is shown in Table 1. The log contains the transaction hash of the smart contract, the sender and receiver of the contract, gas usage, and transaction fee.

Table 1. Seed Sold to Farmer

Transaction Hash	0xa11a91a0373bf3b21b18755ec981fb3860de9e2b66 ae87d61729b9cdc0274dc7
Status	True Transaction mined and execution succeed
From	0x1aE0EA34a72D944a8C7603FfB3eC30a6669E454C
To	0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2
Gas Used	528237
Price (Gwei)	28
Transaction fee (Ether)	0.01479 ether

2. Creation of Contract between Trader and Miller: After the contract creation between the trader and miller, the log of the transaction history is shown in Table 2. The log contains transaction hash of the smart contract, the sender and receiver of the contract, gas usage and transaction fee.

Table 2. Paddy Sold to Miller

Transaction Hash	0xa0ad9718875e0e2434e0ec4504f7e3bc95ad6745a6f d83af6eb5dc9f35532176
Status	True Transaction mined and execution succeed
From	0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02dbE454C
To	0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB
Gas Used	571493
Price (Gwei)	28
Transaction fee (Ether)	0.01600 ether

3. Creation of Contract between Wholesaler and Retailer: After the contract creation between the wholesaler and retailer, the log of the transaction history is shown in Table 3. The log contains transaction hash of the smart contract, the sender and receiver of the contract, gas usage and transaction fee.

Table 3. Rice Sold to Retailer

Transaction Hash	x35a3e83232153ea86d75b00fff95ebad5e9f82afe0dc bea5bbfb9e1ce3e71ae2
Status	True Transaction mined and execution succeed
From	0x617F2E2fD72FD9D5503197092aC168c91465E7f2
To	0x17F6AD8Ef982297579C203069C1DbfFE4348c372
Gas Used	571385
Price (Gwei)	28
Transaction fee (Ether)	0.01599 ether

4. Creation of Contract between Retailer and Customer: After the contract creation between the retailer and customer, the log of the transaction history is shown in Table 4. The log contains transaction hash of the smart contract, the sender and receiver of the contract, gas usage and transaction fee.

Table 4. Rice Sold to Customer

Transaction Hash	0xb5168450de065cb1df6377b0ee6627b1eabcea0c312cba386a99b9dac337205b
Status	True Transaction mined and execution succeed
From	0x17F6AD8Ef982297579C203069C1DbfFE4348c372
To	0x5c6B0f7Bf3E7ce046039Bd8FABdfD3f
Gas Used	519339
Price (Gwei)	28
Transaction fee (Ether)	0.01454 ether

5. Creation of Contract for uploading Traceability Information: After the contract creation by the government regulators, the log of the transaction history is shown in Table 5. The log contains transaction hash of the smart contract, gas usage and transaction fee.

Table 5. Upload of Traceability Information

Transaction Hash	0xab00ad5c5778dcb286e279f3d0c2f0f057f9d25ac526a6be81f41827b02dbd84
Status	True Transaction mined and execution succeed

Gas Used	423437
Price (Gwei)	40
Transaction fee (Ether)	0.01693 ether

3.2 Security Analysis

As Ethereum network is a decentralized platform and does not need a centralized server, the suggested system employs the on-chain storage model of Blockchain, where the key information is stored on chain. The functions of smart contracts identify the whitelisted users, verify their status of contract and restrict access to the system. All callers of the functions can be traced and they will be accountable for their actions. The traceability information is also protected against forgeries in the agricultural product supply chain. The data privacy protection is demonstrated through the Markle tree analysis. Merkle trees are also known as Binary hash trees, it is a mathematical data structure that is made up of hashes constructing various data blocks that summarizes all transactions in a block. This also enables quick, data consistency and content are verified across large databases in a secure manner. A Merkle tree allows the user to identify whether a transaction is included in the block by displaying the total of all transactions in the block and creating a digital fingerprint of the complete set of operations. Merkle trees are created by continuously hashing node pairs until only one hash is left (Q. Y. A. Zhang et al.2022). This final hashed data is the markle root. The hash function is irreversible and one way, which assures that the data content cannot be determined by the hash value of the data, which is due to its one-way property. Any data manipulation will result in a complete modification to the markle root hash, announcing the transmission of data. Furthermore, for the feature of collision resistance of the hash function, once the data provided by the enterprise to the consumer are altered or incomplete, the computed hash value must alter which leads to an inconsistent hash value.

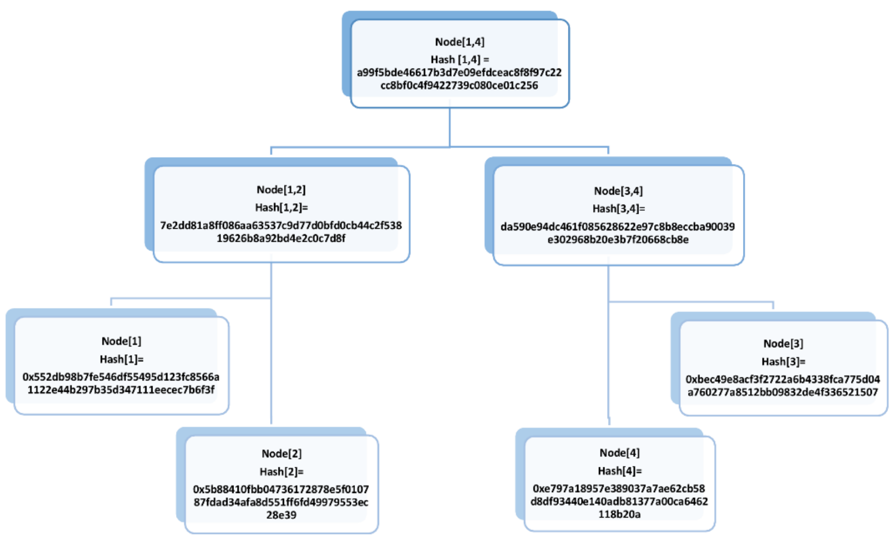

Figure 7. Merkle Tree for Traceability Information.

3.3 Gas Cost of Ether in the Function of Smart Contracts

After deploying the smart contracts, we get the total gas required for each contract and ether amount that has to be paid to execute them. In Table 6. the total amount of ether and gas cost is given. The transaction fee for each smart contract was calculated by multiplying the Gas usage by the unit gas price. At the time of calculation the gas price was 28 Gwei. The dollar value per unit ether was 1260 USD. After calculating the transaction fee in dollars, it was converted to BDT amount. The total transaction fee to deploy all the smart contracts will cost 9991 BDT.

Table 6. Deployment Cost of Smart Contracts

Charge Type	Amount (Ether)	execution (Gas)	Amount (BDT)
Farmer buys from seed company	0.01479	528237	1888.40
Miller buys paddy from trader	0.01600	571493	2042.89

Rice sold to retailer	0.01599	571385	2041.61
Rice sold to customers	0.01454	519339	1856.48
Information stored on chain	0.01693	423449	2161.63
Total(BDT)	0.07825	2613903	9991

3.4 Performance Analysis

The data privacy solution has been demonstrated through the hash function and use of Merkle Tree. We have analyzed the performance of this data privacy solution. We computed the hash of each piece of data in the traceability information. Then the computed hash value constitutes the leaf nodes of the Merkle Tree. We have ran simulation on different data size on Tenderly Platform. Table 7. shows the time taken to hash data of different size using SHA256 hash function. We can see that for a data size of 1000000 KB, it will take only about 2.124 seconds.

Table 7. Hash Performance

Data Size (kb)	Time (s)
1	0.017
10	0.025
100	0.035
1000	0.041
10000	0.087
100000	0.312
1000000	2.124

A significant amount of time is taken for the execution of the smart contracts. Although the amount is negligible, with increment of the data nodes, this time may increase. The average execution time of smart contract from the miner registration phase to the block check phase recorded by simulation results are shown in Table 8. It can be seen that the highest execution time of the smart contracts are 31 seconds. The total time taken for the execution of the smart contracts is 98 seconds.

Table 8. The Average Execution time of smart contracts

Contracts	Time (s)
Smart Contract 1 (Farmer to Seed Company)	13
Smart Contract 2 (Trader to Miller)	11
Smart Contract 3 (Wholesaler to Retailer)	31
Smart Contract 4 (Retailer to Customer)	17
Smart Contract 5 (Upload of Information)	26
Total	98

3.5 Cost Analysis of Smart Contract-Based Traceability System

As each transaction will occur in the supply chain through participants, data regarding the transaction is recorded and stored in the immutable ledger of the Blockchain network. It gives a higher degree of transparency and traceability to the data management system of the rice supply chain. Every time the product changes hands in the network, the information gets stored. The cost of smart contract deployment in our model is related to the Gas usage associated. Gas refers to the unit that measures the amount of computational effort required to execute specific operations on the Ethereum network. The more computational power we need to execute a smart contract, the more Gas will be required. The Gas required to execute a transaction is set, however, the actual price of the Gas varies based on supply and demand factors. The total deployment cost of the smart contracts in the Blockchain network is obtained to be 9991 BDT(Table 6.). Reducing Solidity Gas costs can be accomplished through Gas fee optimization. Gas is both a measurement and fuel connected to EVM usage (K. A. S. Murshid et al., 2011). Usually, variables affecting Gas price are 1) computational power and 2) supply and demand. One way of optimizing Gas fees is to execute functions and contracts at specific times during the day when a network is less congested (T. S. A. Himura et al.,2018). Another way of reducing the overall cost of smart contracts is by creating memory variables allowing the contract to avoid unnecessary Blockchain interactions within loops. When used in public platforms the cost can be further minimized by the moderation of codes.

4. Conclusion

In previous years, the agricultural supply chain traceability network for rice was overlooked. Nowadays it has raised a public concern. To keep a record of specific information and database about the purchased product (rice) and its raw material status throughout the entire rice supply chain of Bangladesh. To conclude, we have tried to create a trusted Smart contract-based Blockchain traceability system to ensure transparent information flow between the stakeholders of the rice supply chain. Smart contracts were created to ensure proper product tracking and reduce the uncertainty regarding payment issues. This paper addressed issues of data storage, data integrity, low traceability of products, and lack of transparency in the transaction of funds and products of the rice supply chain of Bangladesh. We have focused on the issue of the lack of information technology use in the rice supply chain of Bangladesh and tried to overcome this problem with a Smart contract-based solution. cost, security and performance analysis was done based on testing and validating the Smart contracts. Any attempt to make the functions of the smart contract more optimized is likely to be made. This will reduce the Gas consumption of smart contracts. When Ethereum 2.0 is published, this can be made to reinforce the contracts. How product conditions can be verified with the help of decentralized consensus needs further analysis. Further analysis can be done on how to reverse a transaction through a smart contract when arrived product is damaged.

References

Q. Y. A. Zhang, “Improving Agricultural Product Traceability Using Blockchain,” *Sensors,* 2022, 22, 3388, 2022.

Khaled Salah, Nishpra Nizamuddin, Raja Jayaraman, *Block chain-Based Soybean Traceability in Agricultural Supply Chain.* 2019.

Raja Wasim Ahmad, Khaled Salah, Raja Jayaraman, Ibrar Yaqoob, “Blockchain-Based Forward Supply Chain and Waste Management for COVID-19 Medical Equipment and Supplies.” 2020.

Sm Al-Amin , Shipra Rani , M.Shamim Kaiser, “Towards a Blockchain Based Supply Chain Management for E-Agro Business System,” 2020.

Ferdous, M. S., Sultana, J., Reza, M. S.,and Ahmed, S, “National Blockchain Strategy: Bangladesh.,” 2020.

Dianhui Mao, Fan Wang *, Zhihao Hao and Haisheng Li, “Credit Evaluation System Based on Blockchain for Multiple Stakeholders in the Food Supply Chain, 2019.

M.F.W.H.A. Janssen (Marijn), “Boundary conditions for traceability in food supply chains using blockchain Technology,” Kay Behnkea, 2020.

Yousuf, M. A. I., Ratna, S. A., and Rahman, F., “A Research Framework of Smart Food Supply Chain Harnessing Blockchain Technology for Developing Countries,” 2019.

Tahmid Hasan Pranto, Abdulla All Noman, Atik Mahmud, AKM Bahalul Haque, “Blockchain and smart contract for IoT enabled smart agriculture.,” 2020.

Vinay Surendra Yadava, A.R. Singha, Rakesh D. Rautb, Usharani Hareesh Govindarajanc, “Blockchain technology adoption barriers in the Indian agricultural supplychain: an integrated approach.” 2020.

M. Shamim Kaiser, Anirban Bandyopadhyay, Mufti Mahmud, Kanad Ray *et al.*, Towards a Blockchain-Based SupplyChain Management for E-Agro Business System,” vol. 1309, 2020.

Mahadi Hasan Miraz, Mohammad Tariq Hasan, Farhana Rahman Sumi,Shumi Sarkar and Mohammad Ismail, “The Innovative of Blockchain Transparency & Traceability in Logistic Food Chain.,” 2020.

Malaya Dutta Borah, Vadithya Bharath Naik, Ripon Patgiri, Aditya Bhargav, Barneel Phukan and Shiva G. M. Basani, “Supply Chain Management in Agriculture Using Blockchain and IoT,” Springer Nature Singapore Pte Ltd., pp. 227–242, 2020.

J. W. X. Z. Xu, Ed. *et al.*, Blockchain-Based Information Supervision Model for Rice Supply Chains,” vol. 2022, p. 17, Mar. 2020.

Bello Musa Yakubu , Rabia Latif , Aisha Yakubu , Majid Iqbal Khan and Auwal Ibrahim Magashi, “RiceChain: secure and traceable rice supply chain framework using blockchain technology.,” 2022.

K. A. S. Murshid, Ed. *et al.*, Exploring Transition and Change in the Bangladesh Rice Market,” *CPD-CMI Working Paper 1.*, 2011.

Ahmed, Akhter, ed., Islam, Nurul, ed., Mujeri, Mustafa K, “Securing food for all in Bangladesh,” 2021.

D. Roy, D. S. D. MSc, and M. S. c. &. Sheheli “Food Security in Bangladesh: Insight from Available Literature,”2019.

M. A. I. Gazi, “Supply Chain Management for Agro Products in Bangladesh; Logistics Support for Capturing Market by Ensuring Balanced Distribution,” 6, June 2020.

Md. Mizanur Rahman1 and Shahnaz Begum Neena, "The Marketing System of Agricultural Products in Bangladesh: A Case Study from Sylhet District.," *Bangladesh Journal of Public Administration (BJPA)*,2018.
M. C. Rahman, "Welfare Impact of Asymmetric Price Transmission on Bangladesh Rice Consumers," 2018.
Mudabbir Kaleem, Keshav Kasichainula, Rabimba Karanjai, Lei Xu, *et al.*, An Event Driven Framework for Smart Contract Execution.," *Virtual Event, Italy: DEBS 2021 Research Track.*
T. S. A. Himura, "Smart-Contract based System Operations for Permissioned Blockchain," 2018.
J.-H. L. ShIjie Zhang, "Smart Contract-Based Secure Model for Miner Registration and Block Validation," 2019.

Biographies
Maliha Rajwana Haque received her B.Sc. degree in Industrial & Production Engineering from Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh in 2022. She is currently working in the supply chain department of H&M Group Bangladesh. Her research interests include supply chain management, operation research, cyber security, manufacturing, automation, and machine learning.

Sadia Tabassum received her B.Sc. degree in Industrial & Production Engineering from Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh in 2022. She is currently working in the department of supply chain of OMERA Petroleum Bangladesh. Her research interest includes optimization techniques, supply Chain Management, quality control management.

Khairun Nahar received her B.Sc. degree in Industrial & Production Engineering from Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh. She is currently working as an assistant professor in the department of Industrial & Production Engineering in Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh. Her research interest includes human factors engineering and ergonomics, forecasting Accuracy, operations and supply chain management, manufacturing systems modeling and optimization, additive manufacturing and machine learning.