International Research Journal of Education and Technology
ISSN 2581-7795

Smart Contract-Based Sustainable supply chain for Agriculture Using Blockchain

Suriya K[1]| Shameel R[2] | Vignesh N M[3]

[1,2,3] - SRM Valliammai Engineering College - Kattankulathur, Kanchipuram, Tamil Nadu.

Abstract: ***The intricacy of a supply chain makes tracking product safety or quality issues extremely challenging, especially for the basic agricultural food supply networks that people eat every day. Existing agricultural food supply chains have several significant flaws, including a large number of players, cumbersome communication due to protracted supply chain cycles, and data distrust between participants and the centralised system. The introduction of blockchain technology efficiently solves the pain-point problem that exists in the agricultural food supply chain traceability system. The blockchain is a shared record of accounts and transactions that all members write and store. It promises to be a trustworthy source of information about the state of farms, inventory, and contracts in agriculture, where gathering such data is typically prohibitively expensive. Food provenance can be tracked using blockchain technology, which aids in the creation of trustworthy food supply chains and the development of trust between producers and consumers. It promotes the use of data-driven technologies to make farming smarter by providing a secure means to store data. Furthermore, when used in conjunction with smart contracts, it enables timely payments between stakeholders that are triggered by data changes in the blockchain. For both theoretical and practical purposes, this study investigates the use of blockchain technology in food supply chains and agricultural product transactions. We also go over the difficulties of tracking smallholder farmer transactions and developing an ecosystem for using blockchain technology in agriculture.***

Keywords: Blockchain, Supply chain, Smart Contract.

1. Introduction:

Farmers, who account for the vast majority of farmers in developing countries, are predominantly marginal and submarginal farm households with fewer than ten hectares of land under cultivation. In these supply chains, transporters and distributors, agricultural extension officers, financial institutions, wholesalers, retailers and customers, and local manufacturers all play key roles. These players have a huge say in how smallholder farmers are treated and how they react. These smallholder farmers must deal with a wide range of activities and problems. They prepare the soil and plant crops such as sugar, tea, rice, and maize. Before reaching the end customer, rice, for example, travels via several intermediaries in the agriculture supply chain. Along this convoluted supply chain, smallholder farmers face exploitation, child and slave labour, fraud, and corruption. Increased food consumption raises new issues, such as counterfeit goods, which pose a threat to food supply systems on multiple levels. Due to inefficiency and lack of transparency, farmers and consumers are at a disadvantage. Finally, blockchain farming and distributed ledger technology (DLT) have the potential to improve efficiency, transparency, and confidence in the agricultural supply chain. Blockchain for the agriculture supply chain can empower all market participants by establishing reliable connections. The agricultural business could be changed if blockchain is used to simplify all phases of the agricultural supply chain. reducing financial risks and promoting open commerce In agriculture, data science is producing smarter market data for better decision-making.

2. Methodology

Blockchain:

A blockchain is a growing list of documents, known as blocks, that are cryptographically linked together. A cryptographic hash of the preceding block, a timestamp, and transaction data are all included in each block. The timestamp verifies that the transaction data was there at the time the block was released, allowing it to be hashed. Because each block contains information about the one before it, they form a chain, with each new block reinforcing the preceding ones. As a result, blockchains are resistant to data tampering since the data in any given block, once recorded, cannot be changed retrospectively without affecting all subsequent blocks. Blockchains are often administered via a peer-to-peer network for use as a publicly distributed ledger, with nodes communicating and validating new blocks using a protocol. Although forks are possible, blockchain records can be deemed secure by design, and they demonstrate a distributed computing system with excellent Byzantine fault tolerance.

Proof of Work:

Proof of work (PoW) is a type of cryptographic proof in which one party (the prover) demonstrates to others (the verifiers) that a certain amount of computational effort has been invested. Following that, with no effort on their part, verifiers can authenticate this expenditure. Cynthia Dwork and Moni Naor introduced the notion in 1993 as a technique to prevent denial-of-service attacks and other service abuses like spam on a network by requiring some work from the service requester, usually in the form of a computer processing time. Markus Jakobsson and Ari Juels invented and standardised the term "proof of work" in a 1999 publication.

Bitcoin popularised proof of work as a foundation for consensus in a permissionless decentralised network, in which miners compete to append blocks and mint new currency, with each miner having a success chance proportional to the computational effort expended. The asymmetry of proof-of-work systems is a critical feature: the work – the calculation – on the prover or requester side must be fairly difficult (though possible), but easy to check for the verifier or service provider. A CPU cost function, client puzzle, computational puzzle, or CPU price function are all terms for the same concept. Built-in incentive systems that encourage allocating processing capability to the network with monetary value are another prevalent characteristic.

Smart Contract:

A smart contract is a contract in which the conditions of the buyer-seller agreement are encoded directly into code lines. A distributed, decentralised blockchain network hosts the code and the agreements it contains. Transactions are trackable and irreversible, and the code controls the execution. Without the requirement for a central authority, legal system, or external enforcement mechanism, smart contracts enable trustworthy transactions and agreements to be carried out among disparate, anonymous parties. Smart contracts are self-executing lines of code that automatically verify and execute the conditions of a buyer-seller agreement via the internet.

Smart contracts are computerised transaction protocols that execute contract conditions, according to Nick Szabo, an American computer scientist who devised a virtual currency called "Bit Gold" in 1998.

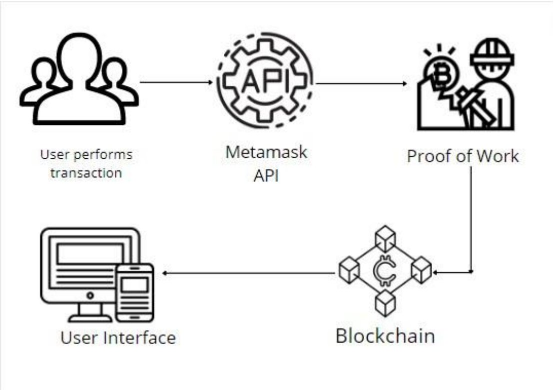
Transactions are traceable, transparent, and irrevocable with smart contracts deployed on blockchains.

International Research Journal of Education and Technology
ISSN 2581-7795

3. PROPOSED SYSTEM:

The proposed system's architecture is as follows. In our proposed system, the user purchases crops from a farmer or dealer at the address provided to them. When the user presses the submit button, the website prompts them to finish the ETH transaction. The user is directed to Metamask as a result of this. Metamask is the API that connects the User Interface to the Blockchain. Metamask functions as a cryptocurrency wallet, storing the user's cryptocurrency and completing transactions if there are sufficient coins in the wallet. Our request is now sent to the blockchain, where miners will validate our transaction, which is known as Proof of Work. The transaction is posted to the blockchain as a block once the Proof of Work is completed successfully. The blockchain does the hashing for each transaction.

ARCHITECTURE:



IMPLEMENTATION

PSEUDOCODE

BUYING CROPS

Input:

- Address of the farmer or retailer,
- Quantity,
- Type of crop.

1. If Sale is agreed and currency = paid then
    a. Contract state changes to success
    b. Change the state of the seller to sold
    c. Change buyer state to crop bought successfully
    d. Send a success notification to the buyer
2. end
3. else
    a. Contract state changes to failed
    b. Change the state of the seller to not sold
    c. Change buyer state to crops not brought successfully
    d. Send a failure notification to the buyer
4. end
5. else
    a. Reset contract
    b. Show an error message
6. end

ADDING CROPS

Input:

- Username and Password of the farmer,
- Quantity,
- Type of crop.

1. If login is agreed then
    a. Choose the type of the crop
    b. Add the quantity
    c. Click submit

i. If the sale is agreed and gas amount = paid then
    1. Contract state changes to success
    2. Send crops added successfully to the farmer

end

end

2. else
    a. Reset contract.
    b. Show an error message.
3. end

REGISTER FARMER AND RETAILER USERS

Input: The user accounts for Farmer and Retailer can be created only by the Super User. The Super User requires:

- User Account Number,
- User Password,
- User Type.

1. If login is agreed then click add user
    a. Enter the user account number
    b. Enter the Password
    c. Enter the User type
    d. Click submit
        i. If the sale is agreed and gas amount = paid then
            1. Contract state changes to success
            2. Send user-added successfully to the Super User

end

i. else
    1. Contract state changes to failure
    2. Send user not successfully added to the Super User.

end

ii. else
    1. Reset Contract
    2. Send error message

2. end

REGISTER CUSTOMER USERS

Input:

- User Account Number,
- User Password,
- User Type.

1. Click Register user
    a. Enter the user account number.
    b. Enter the Password.
    c. Enter the User type.
    d. Click submit

i. If the sale is agreed and gas amount = paid then
    1. Contract state changes to success
    2. Send user-added successfully to the Super User

  end

ii. else
    1. Contract state changes to failure
    2. Send user not successfully added to the Super User.

  end

iii. else
    1. Reset Contract
    2. Send error message

2. end

5. SMART CONTRACT CODE:

```
pragma solidity >=0.7.0 <0.9.0;
import
"@openzeppelin/contracts/utils/Strings.sol";

library StringUtils {
   /// @dev Does a byte-by-byte
lexicographical comparison of two strings.
   /// @return a negative number if `_a` is
smaller, zero if they are equal
   /// and a positive numbe if `_b` is smaller.
   function compare(string memory _a, string
memory _b) private returns (int) {
       bytes memory a = bytes(_a);
       bytes memory b = bytes(_b);
       uint minLength = a.length;
       if (b.length < minLength) minLength =
b.length;
       for (uint i = 0; i < minLength; i ++)
          if (a[i] < b[i])
             return -1;
          else if (a[i] > b[i])
             return 1;
       if (a.length < b.length)
          return -1;
       else if (a.length > b.length)
          return 1;
       else
          return 0;
   }
   /// @dev Compares two strings and returns
true iff they are equal.
   function equal(string memory _a, string
memory _b) public returns (bool) {
       return compare(_a, _b) == 0;
   }
   /// @dev Finds the index of the first
occurrence of _needle in _haystack
   function indexOf(string memory _haystack,
string memory _needle) private returns (int)
   {
          bytes memory h = bytes(_haystack);
          bytes memory n = bytes(_needle);
          if(h.length < 1 || n.length < 1 ||
(n.length > h.length))
                  return -1;
          else if(h.length > (2**128 -1)) // since
we have to be able to return -1 (if the char isn't
found or input error), this function must return
an "int" type with a max length of (2^128 - 1)
                  return -1;


          else
          {
                  uint subindex = 0;
                  for (uint i = 0; i < h.length; i
++)
```

```
{
                              if (h[i] == n[0]) // found
the first char of b
                              {
                                     subindex = 1;

 while(subindex < n.length && (i + subindex) <
 h.length && h[i + subindex] == n[subindex]) //
 search until the chars don't match or until we
 reach the end of a or b
                                     {

 subindex++;
                                     }
                                     if(subindex ==
 n.length)
                                            return
 int(i);
                              }
                       }
                       return -1;
                }
         }
 }

 contract CropSupplyChain {

    struct Crop {
       string      cropName;
       uint256 supplyCount;
    }

    struct   User   {
       string userId;
       string   userType;
       string userName;
    }

    struct PassbookEntry {
       string user;
       string        data;
       string        crop;
       uint256 amount;
    }
    PassbookEntry[] passbook;
    Crop[] crops;
    User[] users;

    // customers need to login into the portal
    // userID = eth wallet address

                            }
```

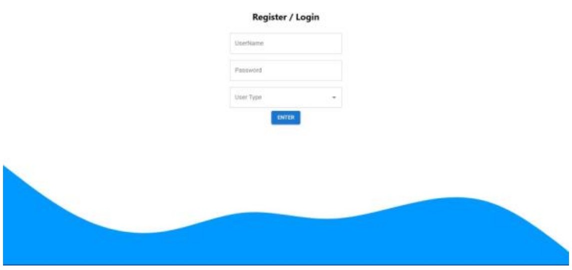
```
   function login (string memory userId,string
memory userType,string memory userName)
public returns (string memory) {
       string memory out = "";
       if (users.length == 0) {
          User memory newUser =
User(userId,userType,userName);
          users.push(newUser);
          out = "user-created";
       }else {
          for (uint i=0; i<users.length; ++i) {
             // make password check
             if
(StringUtils.equal(users[i].userId,userId)     &&
StringUtils.equal(users[i].userType,userType)
) {
                out = "success";
             }else{
                User memory newUser =
User(userId,userType,userName);
                users.push(newUser);
                out = "user-created";
             }
          }
       }
       return out;
   }

   // customer buys the crops
   function  buyCrops  (uint256  quantity,string
memory  cropType,string  memory  userName)
public returns (string memory) {
       string memory out = "";
       for (uint i=0; i<crops.length; ++i) {
        Crop storage curr = crops[i];
          if
(StringUtils.equal(curr.cropName,cropType)) {
             // transaction occurs here
             if (curr.supplyCount < quantity) {
                out = "insufficent-crops";
             }
             curr.supplyCount -= quantity;
             string memory passbookStatement =
"bought crops";
             PassbookEntry   memory   entry   =
PassbookEntry(userName,passbookStatemen
t,cropType,quantity);
             passbook.push(entry);
             out = "crop bought";
          }else {
             out = "crop-not-found";
```

International Research Journal of Education and Technology
ISSN 2581-7795

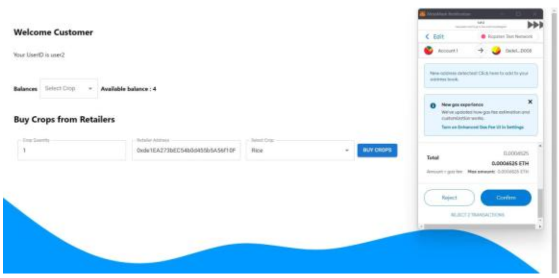
```
    }
    return out;
  }

  // Farmer adds the crops
  function addCrops (uint256 quantity,string memory cropType,string memory userName) public returns (string memory) {
    string memory out = "";
    string memory passbookStatement = "added crops";
    if (crops.length == 0) {
      // add new crop
      Crop memory newCrop = Crop(cropType,quantity);
      crops.push(newCrop);
      PassbookEntry memory entry = PassbookEntry(userName,passbookStatement,cropType,quantity);
      passbook.push(entry);
      out = "added new crops";
    } else {
      for (uint i=0; i<crops.length; ++i) {
        Crop storage curr = crops[i];
        if (StringUtils.equal(curr.cropName,cropType)) {
          // make transaction
          curr.supplyCount += quantity;
          PassbookEntry memory entry = PassbookEntry(userName,passbookStatement,cropType,quantity);
          passbook.push(entry);
          out = "added crops";
        }else {
          Crop memory newCrop = Crop(cropType,quantity);
          crops.push(newCrop);
          PassbookEntry memory entry = PassbookEntry(userName,passbookStatement,cropType,quantity);
          passbook.push(entry);
          out = "added new crops";
        }
      }
    }

    return out;
  }

  function showCrops() public view returns (Crop[] memory) {
    return crops;
  }

  function showUsers() public view returns (User[] memory) {
    return users;
  }

  function showPassbook() public view returns (PassbookEntry[] memory) {
    return passbook;
  }
}
```

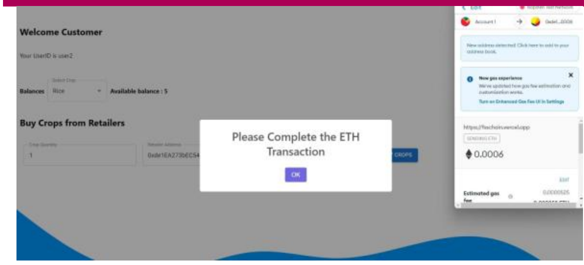
6. OUTPUTS:

1. The Login Portal enables the user to login into a respective account using their credentials.

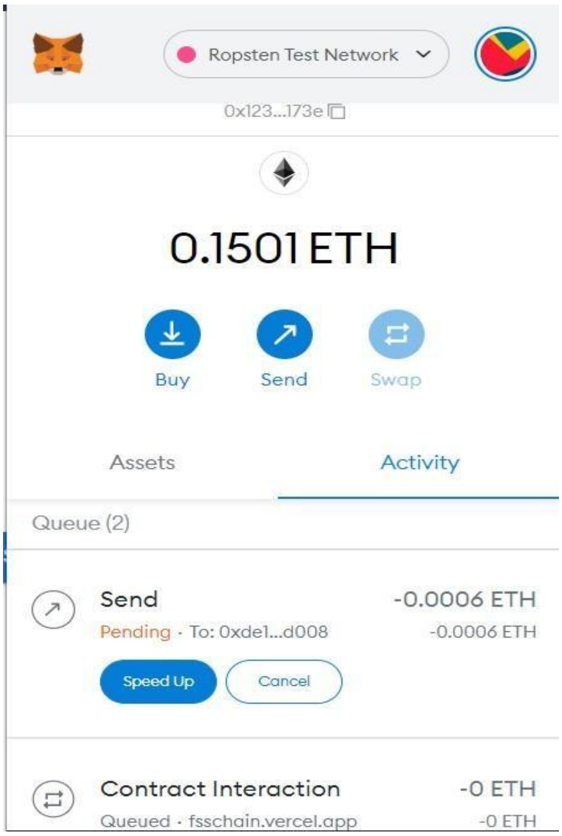


2. The customer portal helps the customer to buy the crop from the retailer using the retailer transaction address. The customer needs to mention the number of crops and the type of crop in the portal. After clicking the buy crop button it will ask the customer to complete the ETH transaction which will direct them to the Metamask wallet.

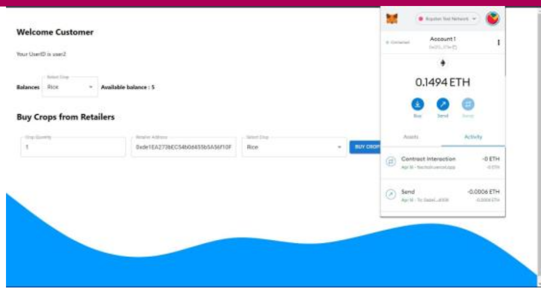




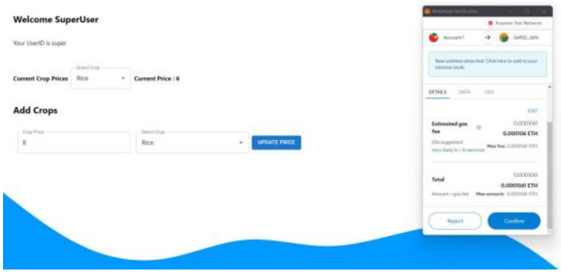
3. The transaction request is submitted to Metamask, if there are sufficient funds in the crypto wallet, the transaction is made successful. Now Metamask will send the request to the blockchain to perform the proof of work. When the proof of work is verified by the miners in the blockchain, the transaction is made successful and added to the chain.

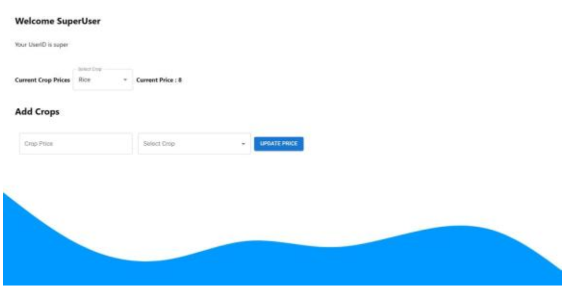


4. This is the updated user interface after performing the transaction.



5. The price is set by the government for each crop and the customer and the dealer can buy the crops from the farmer based on the fixed price.





7. ADVANTAGES OF THE PROPOSED SYSTEM:

1. Our Proposed System will break all the monopolies in the farmer Supply chain, unlike the existing system it is inclusive and incorporates every stakeholder.

2. Transparency will be achieved in the proposed system because the transaction details are made public.
3. In our proposed system farmers can reach out to retailers, and customers directly without the help of intermediaries.
4. Hence the price controlling authority is the government, we can prevent the price fluctuation in the market.
5. Our System will ensure the farmer will get their fair cut for their produce.

7. CONCLUSION:

We suggested a framework for performing transactions using smart contracts, which alters the centralised model of the agricultural food supply chain, eliminates intermediaries and intermediary nodes, and realises the decentralised model, thus addressing the requirement for agricultural food traceability. However, in response to the current issues of blockchain scalability, privacy, and regulation, we have proposed a solution that ignores the reliability and suitability of data transactions and payments, and with the growth of the agricultural food supply chain, a decentralised automatic payment mechanism is required to ensure that all system entities adhere to the promise of defects in the deal. We intend to examine associated challenges and be able to mitigate and solve them as a goal of our future work.

8. REFERENCES:

[1] Awwad M, Kalluru SR, Airpulli VK, Zambre M, Marathe A, Jain P. 2018. Blockchain technology for efficient management of the supply chain. In: The International Conference on Industrial Engineering and Operations Management. Washington DC, USA;. p. 440-50.

[2] B. Amuzu-sefordzi, K. Martinus, P. Tschakert, and R. Wills, 2018, “Disruptive innovations and decentralised renewable energy systems in Africa: A socio-technical review,” Energy Res. Social Sci., vol. 46, pp. 140–154,, doi: 10.1016/j.erss.2018.06.014.

[3] C. Andrew, “Modern slavery as a management practice: Exploring the conditions and capabilities for human exploitation,” Acad. Manage. Rev., vol. 38 , no. 1, pp. 45–69, 2013, doi: 10.2307/23416302.

[4] C. Bai and J. Sarkis, “A supply chain transparency and sustainability technology appraisal model for blockchain technology,” Int. J. Prod. Res., vol. 58, no. 7, pp. 2142–2162, 2020, doi: 10.1080/00207543.2019.1708989.

[5] C. Bai, B. Shi, F. Liu, and J. Sarkis,2019, “Banking credit worthiness: Evaluating the complex relationships,” Omega, vol. 83, pp. 26–38.

[6] FAO Statistics 2019, World food and agriculture, statistical pocketbook, Rome, www.fao.org/publications.

[7] Feng H, Wang X, Duan Y, Zhang J, Zhang X. Applying blockchain technology to improve agri-food traceability: A review of development methods, benefits and challenges. J. Clean. Prod. 2020; 260: 121031. In the press. https://doi.org/10.1016/j.jclepro.2020.121031 [Accessed 1 July 2020].

[8] Frizzo-Barker J, Chow-White PA, Adams PR, Mentanko J, Ha D, Green S. Blockchain as a disruptive technology for business: A systematic review. Int J Inf Manag. 2019; 51: 102029. In the press. https://doi.org/10.1016/j.ijinfomgt.2019.10.014 [Accessed 1 April 2020].

[9] Ghosh A, Gupta S, Dua A, Kumar N. Security of Cryptocurrencies in blockchain technology: State-of-art, challenges and future prospects. J Netw Comput Appl. 2020; 163: 102635. In press, ttps://doi.org/10.1016/j.jnca.2020.102635 [Accessed 1 August 2020].

[10] G. LeBaron and E. Gore, “Gender and forced labour: Understanding the links in global cocoa supply chains,” J. Develop. Stud., vol. 56, no. 6, pp. 1095–1117, 2019, doi: 10.1080/00220388.2019.1657570.

[11] I. Protopop and A. Shanoyan, "Big data and smallholder farmers: Big data applications in the agri-food supply chain in developing countries," Int. Food Agribus. Manage. Rev., vol. 19, pp. 173–190. 2016, doi: 10.22004/ag.econ.240705.

[12] J. Wiegratz, "Fake capitalism? The dynamics of neoliberal moral restructuring and pseudo-development: The case of Uganda," Rev. Afr. Political Econ.,37, no. 124, pp. 123–137, 2010, doi: 10.1080/03056244.2010.484525.

[13] J. H. Grimm, J. S. Hofstetter, and J. Sarkis, "Exploring sub-suppliers' compliance with corporate sustainability standards," J. Cleaner Prod., vol. 112, pp. 1971–1984, 2016, doi: 10.1016/j.jclepro.2014.11.036.

[14] Kamble SS, Gunasekaran A, Sharma R. Modelling the blockchain enabled traceability in the agriculture supply chain. Int J Inf Manag. 2019; 52: 101967. In the press. https://doi.org/ 10.1016/j.ijinfomgt.2019.05.023 [Accessed 1 June 2020].

[15] Liu Z, Li Z. A blockchain-based framework of cross-border e-commerce supply chain. Int. J Inf Manag 2019; 52: 102059. In the press. https://doi.org/10.1016/j.ijinfomgt.2019.102059 [Accessed 1 June 2020].

[16] M. Janssen, V. Weerakkody, Ismagilova, U. Sivarajah, and Z. Irani, "A framework for analysing blockchain technology adoption: Integrating institutional, market and technical factors," Int. J. Inf. Manage., vol. 50, pp. 302–309, 2020, doi: 10.1016/j.ijinfomgt.2019.08.012.

[17] P. Kittipanya-ngam and K. H. Tan, "A framework for food supply chain digitization: Lessons from Thailand," Prod. Planning control, vol. 31, no. 2/3, pp. 158–172, 2020, doi: 10.1080/09537287.2019.1631462.

[18] P. Meyfroidt, D. Abeygunawardane, N. Ramankutty, A. Thomson, and G. Zeleke, "Interactions between land systems and food systems," Current Opinion Environ. Sustain., vol. 38, pp. 60–67, 2019, doi: 10.1016/j.cosust.2019.04.010.

[19] Pan X, Pan X, Song M, Ai B, Ming Y. Blockchain technology and enterprise operational capabilities: An empirical test. Int J Inf Manag. 2019; 52: 101946. In the press. https://doi.org/10.1016/j.ijinfomgt.2019.05.002 [Accessed 1 June 2020].

[20] P. R. Ntakyo andM. van den Berg, "Smallholder food marketing behaviour: Exploring the role of informal credit and traders in stabilisation of food crop prices," Appl. Stud. Agribus. Commerce, vol. 12, pp. 67–82, 2018.

[21] Ronaghi MH, Forouharfar A, A contextualised study of the usage of the Internet of things (IoTs) in smart farming in a typical Middle Eastern country within the context of Unified Theory of Acceptance and Use of Technology model (UTAUT). Technol Soc. 2020; 63:101415. In press,https://doi.org/ 10.1016/j.techsoc.2020.101415 (Accessed 1 November 2020).

[22] S. Baurzhan and G. P. Jenkins, "Off-grid solar PV: Is it an affordable or appropriate solution for rural electrification in Sub-Saharan African countries?"Renewable Sustain. Energy Rev., vol. 60, pp. 1405–1418, 2016, doi: 10.1016/j.rser.2016.03.016.

[23] Sternad M, Lerher T, Gajsek B, Maturity Levels for Logistics 4.0 Based on Nrw'S industry 4.0 maturity model. In: 18th international scientific conference Business Logistics in Modern Management, October 11-12, 2018. Osijek, Croatia; 2018. p. 695–708.

[24] Stranieri S, Riccardi F, Meuwissen PM, Soregaroli C. Exploring the impact of blockchain on the performance of agri-food supply chains, Food Control. 2021; 119: 107495. In the press. https://doi.org/10.1016/j.foodcont.2020.107495 [(Accessed 20 January 2020].

[25] Thakur V, Doja MN, Dwivedi YK, Ahmad T, Khadanga G. Land records on blockchain

for implementation of land titling in India. Int J Inf Manag. 2019; 52: 101940. In the press. https://doi. org/10.1016/j.ijinfomgt.2019.04.013 [Accessed 1 June 2020].
.

[26] T. Salerno, “Cargill’s corporate growth in times of crises: How agro-commodity traders are increasing profits in the midst of volatility,” Agriculture Human Values, vol. 34, no. 1, pp. 211–222, 2017, doi: 10.1007/s10460-016-9681-8.

[27] Venkatesh VG, Kang K, Wang B, Zhong R, Zhang A. System architecture for blockchain based transparency of supply chain social sustainability. Robot Comput Integr Manuf. 2020; 63: 101896. In press https://doi.org/10.1016/j.rcim.2019.101896 [Accessed 1 June 2020].

[28] Verdouw C, Sundmaeker H, Tekinerdogan B, Conzon D, Montanarod T. Architecture framework of IoT-based food and farm systems: A multiple case study. Comput Electron Agric. 2019; 165: 104939. In the press. https://doi.org/10.1016/ j.compag.2019.104939 [Accessed 1 October 2019].

[29] Wong LW, Leong LY, Hew JJ, Tan GWH, Ooi KB. Time to seize the digital evolution: Adoption of blockchain in operations and supply chain management among Malaysian SMEs. Int J Inf Manag. 2019; 52: 101997. In the press. https://doi.org/10.1016/j.ijinfomgt.2019.08.005 [Accessed 1 June 2020].