AGRICULTURAL DOCS CHAIN

NAAN MUDHALVAN PROJECT

PROJECT

SUBMITTED BY:

M.MUTHU KEERTHANA

IQJAZ JAMEELATH RAAHINA.S.A

C.MURUGESHWARI

ELSI JEBAMALAR.K

TEAM ID : NM2023TMID03106

OCT 2023

NAAN MUDHALVAN REPORT:

1.INTRODUCTION:

1.1 Project Overview:

Develop a smart contract on the Ethereum blockchain to efficiently store and manage agricultural data. The contract will support adding, querying, and updating agricultural information. Detailed planning is crucial to ensure the contract meets the project's requirements effectively. The Agriculture Docs Chain project aims to revolutionize the agricultural industry by leveraging blockchain technology.

It involves creating a decentralized platform where stakeholders in the agriculture sector, such as farmers, suppliers, distributors, and consumers, can securely and transparently exchange information and transactions. Smart Contracts: These are self-executing contracts with predefined rules and conditions.

In this context, they automate processes like payment, delivery, and quality assurance, ensuring trust between parties. Immutable Ledger: All transactions and data entries are recorded on a blockchain , which means they cannot be altered or tampered with, providing a high level of transparency.

1.2 Purpose:

Traceability:

It allows tracking the journey of agricultural products from farm to table, ensuring authenticity and origin verification.

Supply Chain Management:

It helps streamline the movement of goods, reducing fraud, and ensuring quality control.

Smart Contracts:

Enables automated, self-executing contracts, facilitating transactions and agreements within the agriculture industry.

Regulatory Compliance:

Helps in ensuring compliance with industry standards, certifications, and government regulations.

Quality Assurance:

Provides a reliable record of product quality, including certifications, inspections, and testing.

2. LITERATURE SURVAY :

2.1 Existing Problem:

The existing issues and challenges in agriculture documentation chains

Problems:

1. System failures and technical glitches

2. Requiring continuous updates and adjustments to tracking system.

3. Lack of access to technology ,leads to inaccurate data.

4. Lack of measure leads to inaccuracies of the nutritional data intake of people

2.2 References:

This should help us to create a well-organized and informative literature review on the topic of agriculture documentation chains.

From Papers:

1. Food traceability system using blockchain

2. Blockchain for food tracking

3. Blockchain use cases for food traceability and control

4. Food supply chain traceability using block chain

5.Apply blockchain technology to improve agrifood traceability

2.3 Problem Statement Definition:

"The problem of inefficient documentation chains in agriculture." In the agricultural industry, there is a critical need for a secure and efficient system to manage , verify and share various documents and records such as land ownership titles, crop certifications, supply chain information, and contracts.

Current paper based and manual processes are prone to errors, fraud and delays, leading to inefficiencies in the agricultural system.

Additionally ,these documents are often dispersed across multiple stakeholders, making collaboration and transparency challenging. The ‘Agricultural Docs Chain’ project aims to address these challenges by leveraging blockchain technology to establish a decentralized and immutable system for document managementand verification.

This will improve the integrity, accessibility and traceability of crucial agricultural documents , ultimately enhancing trust and efficiency within the industry.

Designing a smart contracts on the ethereum blockchain to store and manage agriculture data , including the ability to add, query and update details, requires careful planning. Below is a conceptual outline of how this can be done along with the simplified smart contracts.

This simplified smart contract allows for adding and updating agricultural data . In practice, additional features ,security measures and control mechanisms would be needed for a fully

functional and secure agriculture data management system on the blockchain.

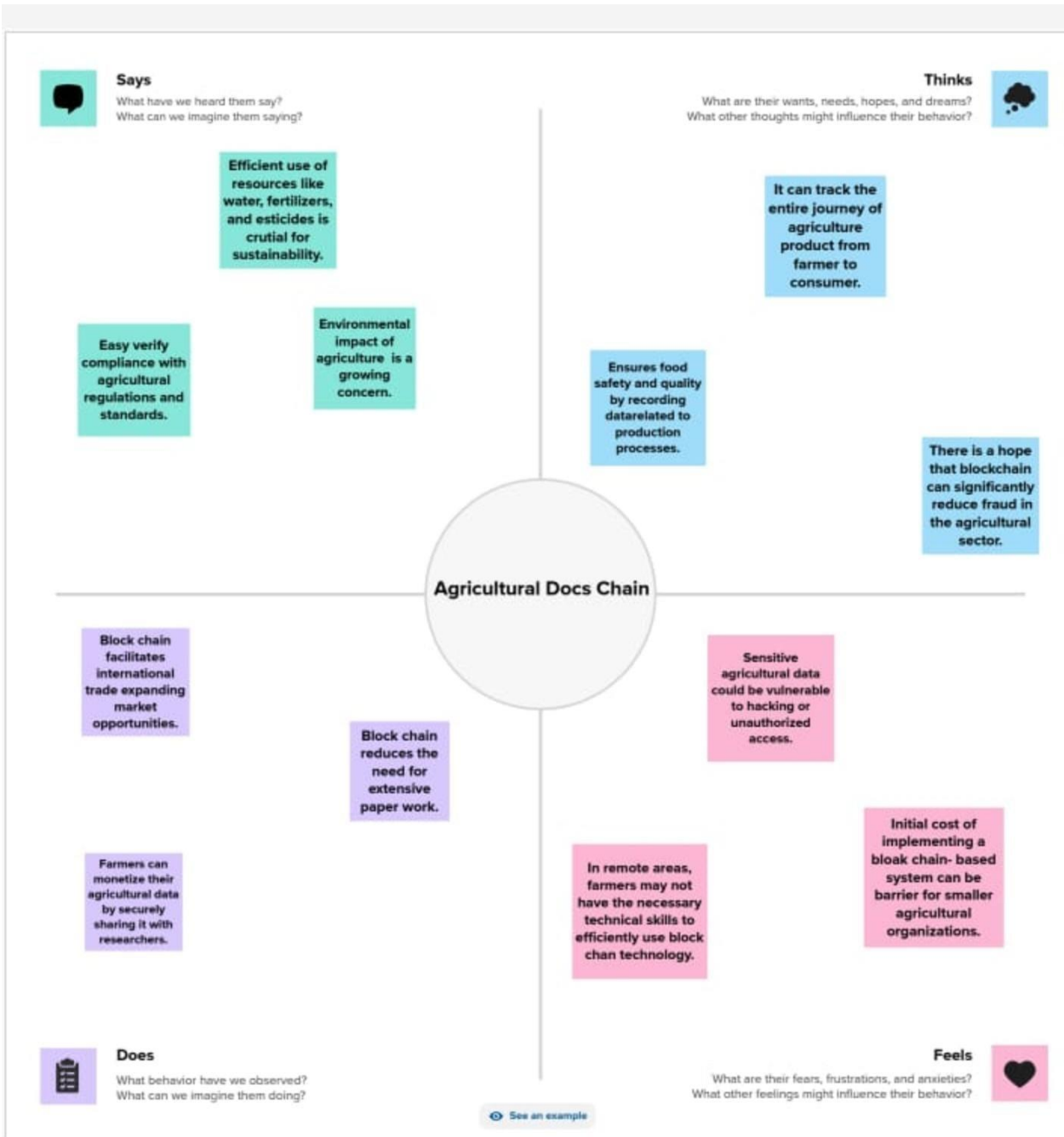
S.NO	Title of the paper	Authors	Algorithm	Advantages	Disadvantages
1.	FOOD TRACEABILITY SYSTEM USING BLOCKCHAIN	IUON-CHANG LIN, HSUAN SHIH, JUI-CHUN LIU, YI-XIANG JIE	Sensors are used in an IOT mode Traceability system	Allows for real-time monitoring of food items,improving efficiency and reducing uncertainties.	System failures, or technical glitches.complex supply chain
2.	**Blockchain for Food Tracking**	Arif Furkan Mendi	Gps , sensor	Helps comply with food safety regulations and standards. Reducing waste and optimizing processes.	Requiring continuous updates and adjustments to tracking systems Leading to complexities in its implementation.
3.	Blockchain use cases for food traceability and control	Axfoundation, SKL Kommentus, Swedish county councils and regions, Martin & Servera, and Kairos Future.	a unique identifier such as a barcode, QR code, or a RfID transmitter	Reduce foodborne illnesses,distribution is secure and tamper-resistant.	Lack of access to technology Leading to incomplete or inaccurate data.
4.	Food Supply Chain Traceability using Block Chain	S.Kayalvizhi, D. Amirtha Sughi, G.Shivasree, J.Shruthi	A radio frequency Identification (RFID)-based sensor,MQTT,IOT.	Aiding in weight management, and helping meet nutritional goals.	The data is not adequately protected. Potentially leading to user fatigue.
5.	Applying blockchain technology to improve agrifood traceability	Huanhuan Fenga,b, Xiang Wanga,b, Yanqing Duan c , Jian Zhangd*, Xiaoshuan Zhanga,b*	Internet of Things Wireless Sensor Network (WSN), QR code, NFC,And RFID	Reduce foodborne illnesses, and enable consumers to make more informed choices about the products they consume.	People may not accurately measure or record their food intake, leading to inaccuracies in the nutritional data.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

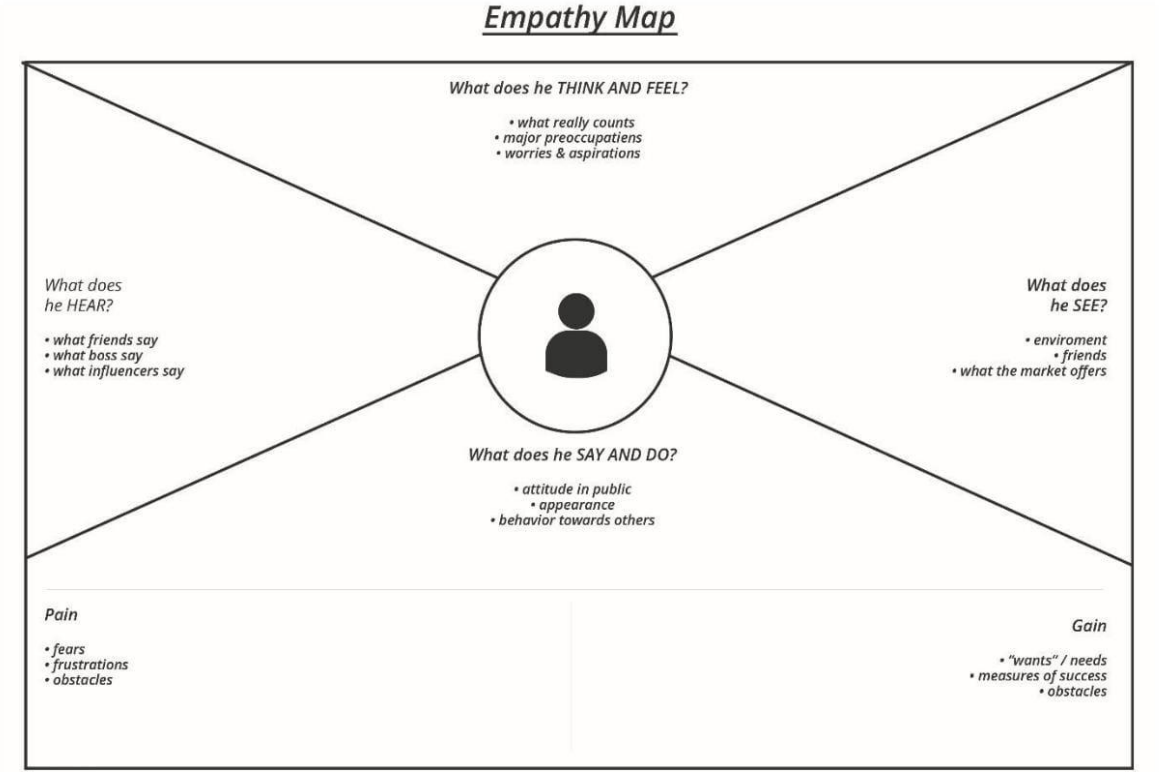
Empathy Map:

Empathy maps are an efficient tool used by designers to not only understand user behavior, but also visually communicate those findings to colleagues, uniting the team under one shared understanding of the user


Says
What have we heard them say?
What can we imagine them saying?
Efficient use of resources like water, fertilizers, and esticides is crutial for sustainability.
Easy verify compliance with agricultural regulations and standards.
Environmental impact of agriculture is a growing concern.
Thinks
What are their wants, needs, hopes, and dreams?
What other thoughts might influence their behavior?
It can track the entire journey of agriculture product from farmer to consumer.
Ensures food safety and quality by recording datarelated to production processes.
There is a hope that blockchain can significantly reduce fraud in the agricultural sector.
Agricultural Docs Chain
Block chain facilitates international trade expanding market opportunities.
Block chain reduces the need for extensive paper work.
Farmers can monetize their agricultural data by securely sharing it with researchers.
Sensitive agricultural data could be vulnerable to hacking or unauthorized access.
Initial cost of implementing a bloak chain- based system can be barrier for smaller agricultural organizations.
In remote areas, farmers may not have the necessary technical skills to efficiently use block chan technology.
Does
What behavior have we observed?
What can we imagine them doing?
Feels
What are their fears, frustrations, and anxieties?
What other feelings might influence their behavior?
See an example

Canvas Ideation:

Brainstorm different aspects of the agriculture document chain. Consider elements like document types, data security, accessibility, and efficiency.



3.2 Ideation & Brainstorming:

Brainstorming:

Gather a diverse group of stakeholders or team members. Encourage open discussion and idea sharing. Focus on creative solutions to the pain points identified in the empathy map.

Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Share template feedback

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

Type your paragraph...

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

Need some inspiration?

See a finished version of this template to kickstart your work.

Open example →

Define your problem statement:

In the agriculture industry, there is a critical need for a secure and efficient system to manage, verify, and share various documents and records such as land ownership titles, crop certifications, supply chain information, and contracts. Current paper-based and manual processes are prone to errors, fraud, and delays, leading to inefficiencies in the agricultural ecosystem. Additionally, these documents are often dispersed across multiple stakeholders, making collaboration and transparency challenging. The 'Agriculture Docs Chain' project aims to address these challenges by leveraging blockchain technology to establish a decentralized and immutable system for document management and verification.

This will improve the integrity, accessibility, and traceability of crucial agricultural documents, ultimately enhancing trust and efficiency within the industry.
Designing a smart contract on the ethereum block chain to store and manage agriclture data, including the ability to add , query, and update details , requires careful planning.
Below is a conceptual outline of how this can be done along with the simplified smart contract. This simplified smart contract allows for adding and updating agriculture data. In practice, additional features , security measures, and access control mechanisms would be needed for a fully functional and secure agriculture data management system on the block chain.

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Team leader

- increase trust among stakeholders
- improve traceability and food safety
- encryption RAFT consesnus algorithm
- errors in smart contracts leads to financial losses

Team member 1

- ensure the integrity of data
- smart contracts
- it is resistive to adapt new technology
- the system relies on technology

Team member 2

- machine learning algorithm
- facilitate contact tracing
- farmers need education and training to effectvely use blockchain
- ability to track agricultural product from source to market

Team member 3

- decentralized consesnus algorithm
- ability to monitor and record data at various stages of production
- integrating with existing system can be complex
- reduce reliance on manual security check

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

ability to monitor and record data at various stages of production

ensure the integrity of data

improve traceability and food safety

encryption RAFT consesnus algorithm

machine learning algorithm

decentralized consesnus algorithm

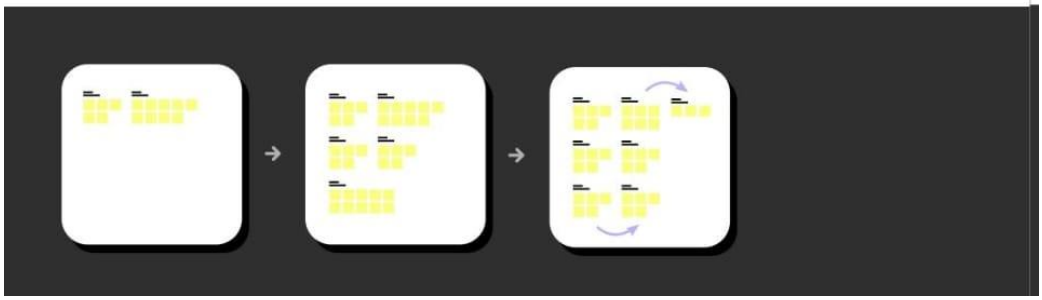
real time monitoring

cloud based storage

the system relies on technology

low light conditions affect the system

aims to reduce reliance on manual security



4

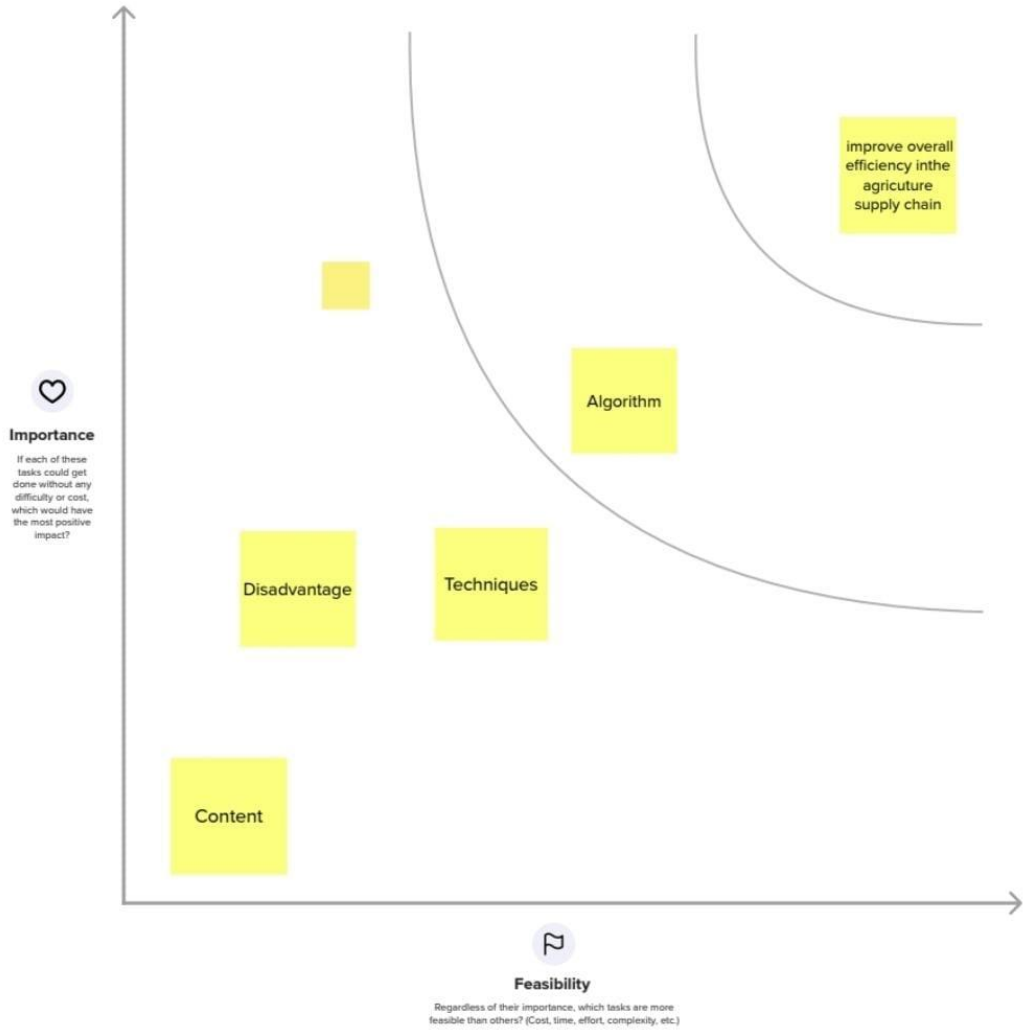
Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

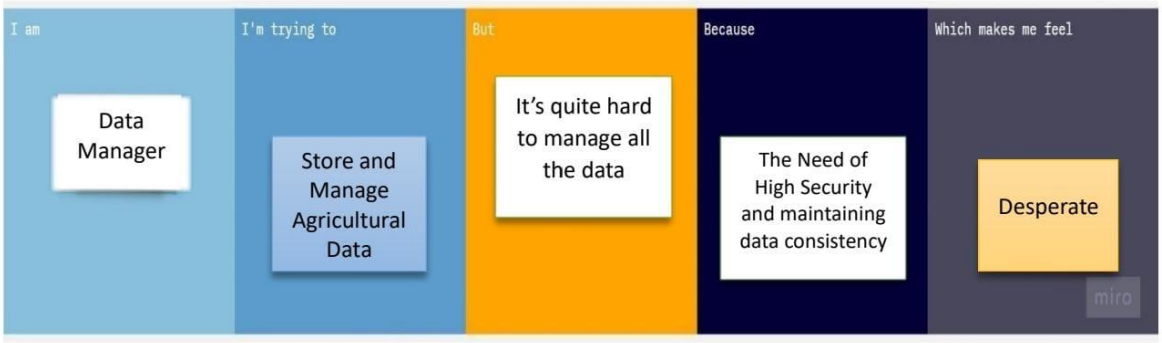


Proposed Solution:

Based on the empathy map insights and ideation, proposing a solution. This involve using blockchain technology for document security, digital platforms for accessibility, or automation for efficiency.

Define the key features and benefits of our solution:

The proposed solution should aim to address the pain points and provide gains for the stakeholders involved in the agriculture document chain



4. REQUIREMENT ANALYSIS:

4.1 Functional Requirement:

Requirement Analysis:

This involves identifying, analyzing, and documenting the requirements for a system or project. In the case of agriculture document chain, this would involve understanding what functionalities and features the system needs to have.

Functional Requirement:

These are specific functionalities or capabilities that a system must possess. In the context of agriculture document chain, this might include features like document storage, retrieval, authentication, version control, etc.

Recruitment:

In the context of requirements, this could refer to the process of identifying and hiring individuals or teams with the necessary skills and expertise to develop and implement the agriculture document chain system.

For an agriculture document chain, the requirement analysis would involve determining what functionalities (functional requirements) the system needs to have, as well as considering non-functional requirements such as security measures, system performance, and user experience. Additionally, recruitment would be necessary to assemble a team with the expertise to develop and implement the system based on these requirements.

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Data collection	Define how data is collected,whether through manual entry, sensors,or other methods.
FR-2	Data storage	Specify where and how data will be stored,such as in a database or cloud platform.
FR-3	Data processing	Describe the steps for data processing and analysis,including algorithms and software tools.

4.2 Non-Functional Requirements

Non-Functional Requirement:

These are criteria that are not related to specific functionalities, but are important for the overall performance and usability of the system. This could include aspects like security, performance, scalability, user experience, etc. In your context, this might involve things like data security, response times, system availability, etc.

Recruitment:

In the context of requirements, this could refer to the process of identifying and hiring individuals or teams with the necessary skills and expertise to develop and implement the agriculture document chain system.

Non-Functional Requirements:

Following are the non-functional requirements of the proposed solution.

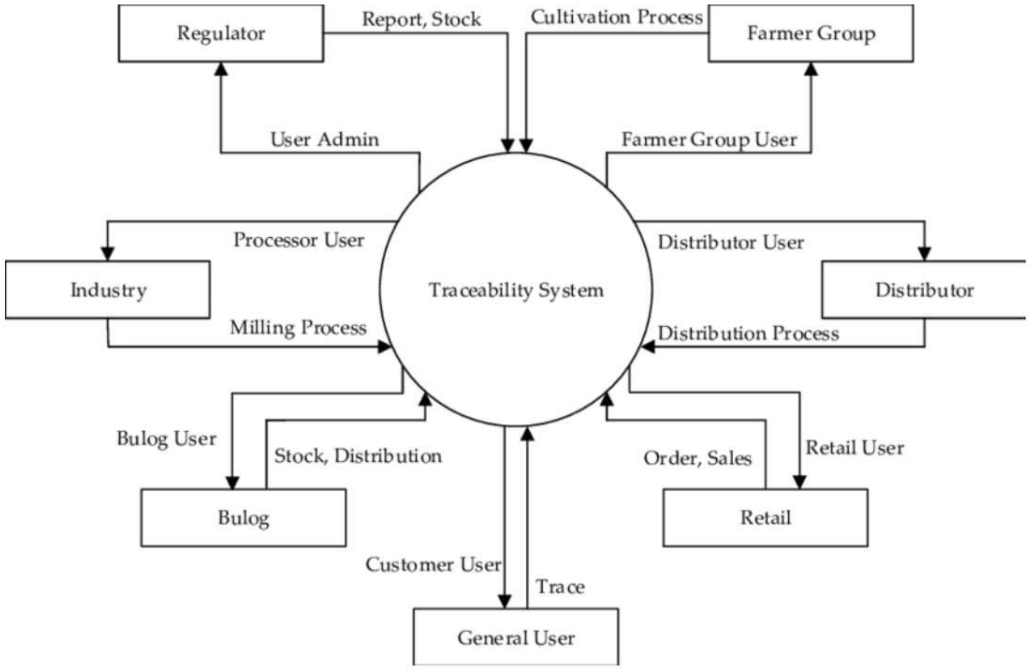
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Describe user experience expectations,such as "The system should have an intuitive user interface".
NFR-2	**Security**	Specify security measures like data encryption, access control,and authentication requirements,eg.,"Data must be encrypted both in transit and at rest".
NFR-3	**Reliability**	Determine the systems uptime requirement, such as "The system should be available 99.9% of the time ".
NFR4	**Performance**	Specify how quickly documents should be retrieved or updated within the chain .For example,"Documents should be retrievable within 2 seconds".
NFR-5	Auditability	Ensure the ability to track changes and access to track changes and access to documents,like "The system should maintain an audit log of all documents changes".
NFR-6	**Scalability**	Define how the system should handle an increasing number of documents or users.For instance, "The system should support a minimum of 10,000 documents and 100 concurrent users".

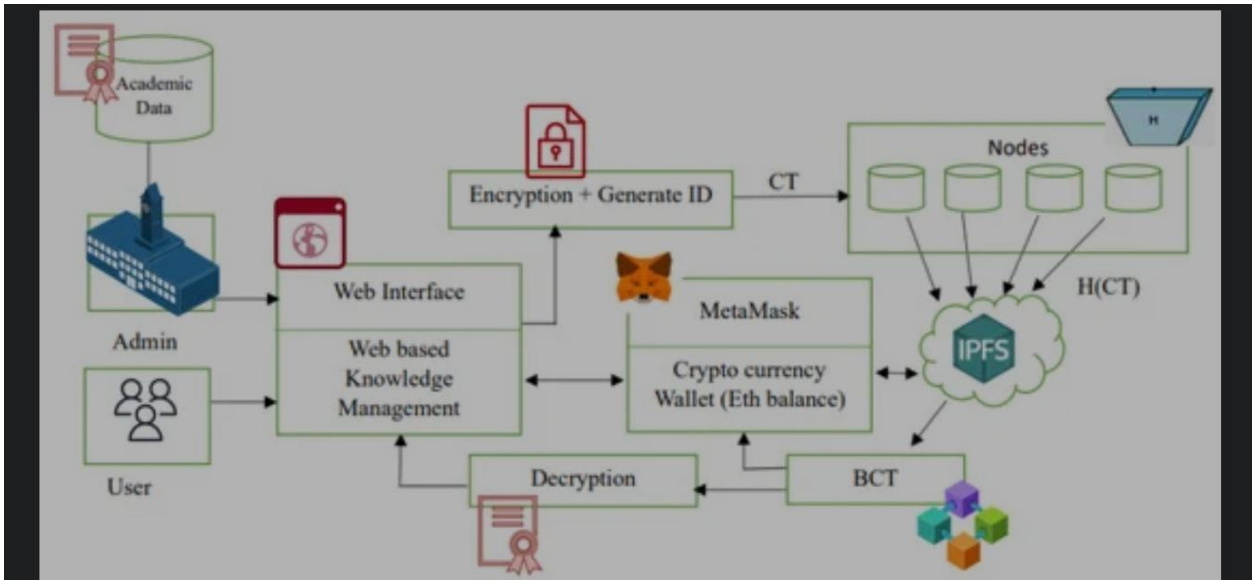
5. PROJECT DESIGN:

5.1 Data Flow Diagrams & Users Stories:

Data Flow Diagram (DFD):

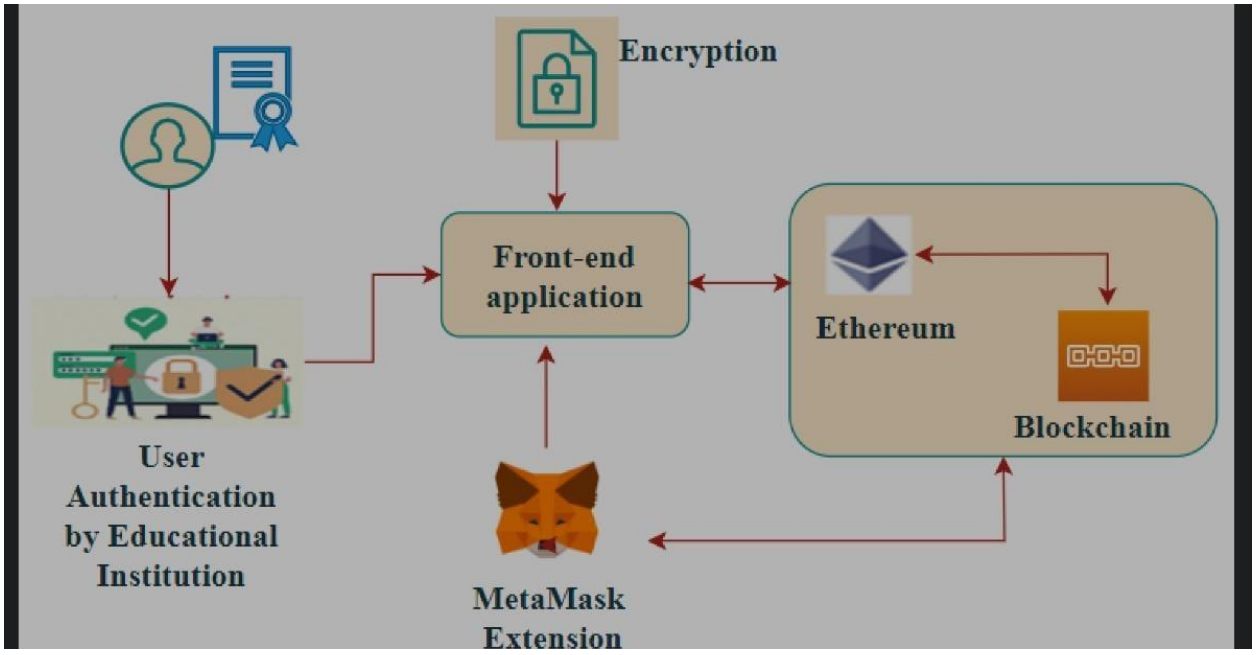
Start with a context diagram that shows the system as a single process and its interactions with external entities (like users, databases, etc.).Break down the context diagram into lower-level DFDs, showing more detail about processes and data flows.Clearly label processes, data stores, data flows, and external entities.





User Stories:

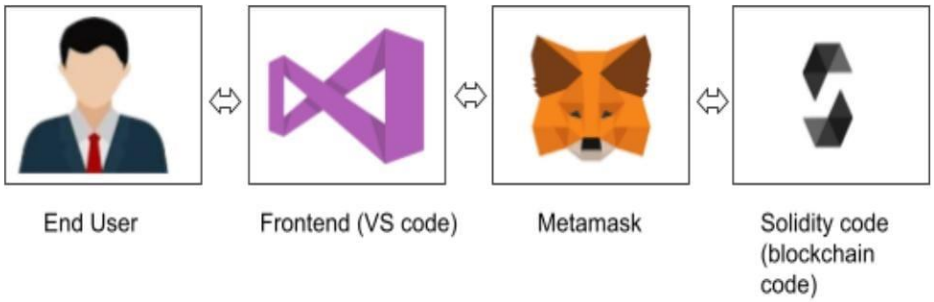
Write user stories from the perspective of different stakeholders (farmers, regulators, processors, etc.)."Prioritize stories based on importance and dependencies.

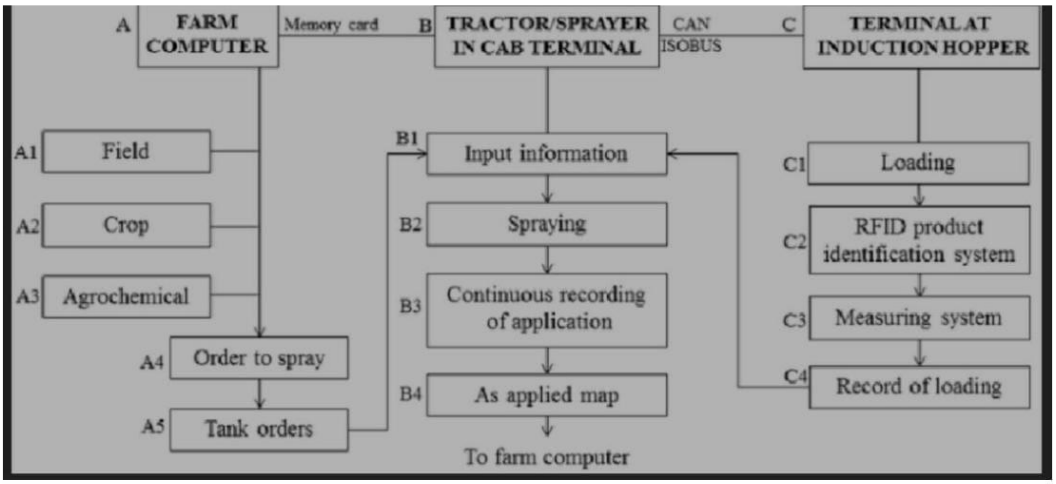


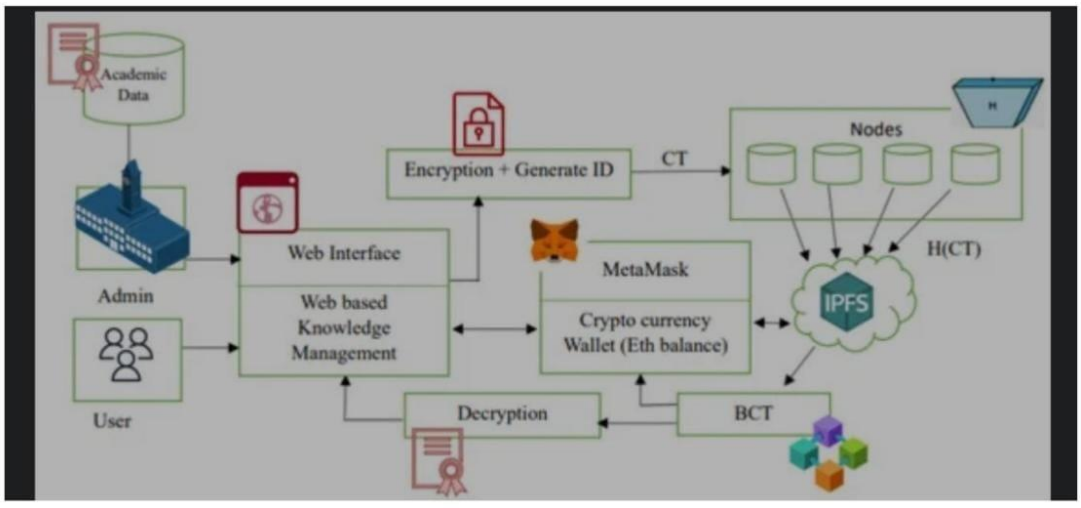
5.2 Solution Architecture

Define the system components (e.g., frontend, backend, database, external APIs).Choosing the appropriate technologies and frameworks for each component. Considering scalability, security, and data management strategies. Designing interfaces between different components and plan for data storage and retrieval mechanisms.

Solution Architecture:




A
FARM COMPUTER
Memory card
B
TRACTOR/SPRAYER IN CAB TERMINAL
CAN ISOBUS
C
TERMINAL AT INDUCTION HOPPER
A1
Field
A2
Crop
A3
Agrochemical
A4
Order to spray
A5
Tank orders
B1
Input information
B2
Spraying
B3
Continuous recording of application
B4
As applied map
To farm computer
C1
Loading
C2
RFID product identification system
C3
Measuring system
C4
Record of loading


Academic Data
Admin
User
Web Interface
Web based Knowledge Management
Encryption + Generate ID
CT
Nodes
H(CT)
MetaMask
Crypto currency Wallet (Eth balance)
IPFS
Decryption
BCT

6. PROJECT PLANNING & SCHEDULING:

6.1 Technical Architecture:

Architectural Design:

Design the technical architecture for the project, including data storage, software components, and communication systems.

Technology Selection:

Choose the appropriate technologies and tools for implementation.

Security Considerations:

Ensure that the architecture addresses security concerns, especially if sensitive data is involved.

6.1 project plan & scheduling :

Define Project Scope:

Clearly outline the objectives and scope of your agriculture documentation project, specifying the paper documents and information are collected.

Stakeholder Identification:

Identify key stakeholders and their roles in the project.

Create a Project Team:

A team with the necessary skills for the project, knowing about agriculture, Metamask, vs-code, node-js, remix plateform and basic blockchain concepts.

Set Goals and Objectives:

Our goals and objectives are:

1.To protect the agricultural data and making it accurate for further uses.

2.Adding each and every updates and steps , and keenly monitoring the agricultural field .

3. Traceability and scalability will increase linearly and so nutrition flow in the chain also determined.

4.Increasing food safety and standard of the harvested , cultivated foods and so there is no health issues in the society.

6.2 Sprint Planning & Estimation:

Breakdown Tasks:

Divide the project into smaller tasks or user stories that can be completed in sprints.

Prioritize Tasks:

Rank tasks based on importance and dependencies.

Sprint Planning:

Select a set of tasks to be completed in the upcoming sprint.

Estimation:

Estimate the time and effort required for each task. Consider using methods like story points or time-based estimation.

6.3 Sprint Delivery Schedule:

Sprint Execution:

Carry out the tasks as planned in each sprint, including agriculture documentation and technical development.

Monitoring and Control:

Continuously monitor progress, and address any issues that arise during sprints.

Review and Testing:

Review and test the work completed during each sprint to ensure quality and functionality.

Sprint Review and Retrospective:

Evaluate the sprint's outcomes and identify areas for improvement.Throughout the flow , it's important to maintain clear communication with the team, stakeholders, and adjust plans as necessary.

7.CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

ABI Code:

```
[
    {
      "anonymous": false,
      "inputs": [
            {
```

```
        "indexed": false,
        "internalType": "uint256",
        "name": "productId",
        "type": "uint256"
    },
    {
        "indexed": false,
        "internalType": "string",
        "name": "name",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "string",
        "name": "description",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "uint256",
```

```
            "name": "quantity",
            "type": "uint256"
        },
        {
            "indexed": false,
            "internalType": "address",
            "name": "owner",
            "type": "address"
        }
    ],
    "name": "ProductAdded",
    "type": "event"
},
{
    "anonymous": false,
    "inputs": [
        {
            "indexed": false,
            "internalType": "uint256",
            "name": "productId",
```

```
        "type": "uint256"
    },
    {
        "indexed": false,
        "internalType": "string",
        "name": "name",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "string",
        "name": "description",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "uint256",
        "name": "quantity",
        "type": "uint256"
    }
```

```
  ],
  "name": "ProductUpdated",
  "type": "event"
},
{
  "inputs": [
        {
              "internalType": "uint256",
              "name": "ProductId",
              "type": "uint256"
        },
        {
              "internalType": "string",
              "name": "_name",
              "type": "string"
        },
        {
              "internalType": "string",
              "name": "_description",
              "type": "string"
```

```
        },
        {
            "internalType": "uint256",
            "name": "_quantity",
            "type": "uint256"
        }
    ],
    "name": "addProduct",
    "outputs": [],
    "stateMutability": "nonpayable",
    "type": "function"
},
{
    "inputs": [
        {
            "internalType": "uint256",
            "name": "_productId",
            "type": "uint256"
        }
    ],
```

```
"name": "getProductDetails",
"outputs": [
        {
                "internalType": "string",
                "name": "name",
                "type": "string"
        },
        {
                "internalType": "string",
                "name": "description",
                "type": "string"
        },
        {
                "internalType": "uint256",
                "name": "quantity",
                "type": "uint256"
        },
        {
                "internalType": "address",
                "name": "owner",
```

```
                "type": "address"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [],
        "name": "productCount",
        "outputs": [
            {
                "internalType": "uint256",
                "name": "",
                "type": "uint256"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
```

```
"inputs": [
        {
                "internalType": "uint256",
                "name": "",
                "type": "uint256"
        }
],
"name": "products",
"outputs": [
        {
                "internalType": "string",
                "name": "name",
                "type": "string"
        },
        {
                "internalType": "string",
                "name": "description",
                "type": "string"
        },
        {
```

```
                "internalType": "uint256",
                "name": "quantity",
                "type": "uint256"
            },
            {
                "internalType": "address",
                "name": "owner",
                "type": "address"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [
            {
                "internalType": "uint256",
                "name": "_productId",
                "type": "uint256"
            },
```

```
        {
            "internalType": "string",
            "name": "_name",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "_description",
            "type": "string"
        },
        {
            "internalType": "uint256",
            "name": "_quantity",
            "type": "uint256"
        }
],
"name": "updateProduct",
"outputs": [],
"stateMutability": "nonpayable",
"type": "function"
```

```
    }
][
    {
      "anonymous": false,
      "inputs": [
            {
                  "indexed": false,
                  "internalType": "uint256",
                  "name": "productId",
                  "type": "uint256"
            },
            {
                  "indexed": false,
                  "internalType": "string",
                  "name": "name",
                  "type": "string"
            },
            {
                  "indexed": false,
                  "internalType": "string",
```

```
            "name": "description",
            "type": "string"
        },
        {
            "indexed": false,
            "internalType": "uint256",
            "name": "quantity",
            "type": "uint256"
        },
        {
            "indexed": false,
            "internalType": "address",
            "name": "owner",
            "type": "address"
        }
    ],
    "name": "ProductAdded",
    "type": "event"
},
{
```

```
"anonymous": false,
"inputs": [
        {
                "indexed": false,
                "internalType": "uint256",
                "name": "productId",
                "type": "uint256"
        },
        {
                "indexed": false,
                "internalType": "string",
                "name": "name",
                "type": "string"
        },
        {
                "indexed": false,
                "internalType": "string",
                "name": "description",
                "type": "string"
        },
```

```
        {
            "indexed": false,
            "internalType": "uint256",
            "name": "quantity",
            "type": "uint256"
        }
    ],
    "name": "ProductUpdated",
    "type": "event"
},
{
    "inputs": [
        {
            "internalType": "uint256",
            "name": "ProductId",
            "type": "uint256"
        },
        {
            "internalType": "string",
            "name": "_name",
```

```
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "_description",
            "type": "string"
        },
        {
            "internalType": "uint256",
            "name": "_quantity",
            "type": "uint256"
        }
    ],
    "name": "addProduct",
    "outputs": [],
    "stateMutability": "nonpayable",
    "type": "function"
},
{
    "inputs": [
```

```
        {
            "internalType": "uint256",
            "name": "_productId",
            "type": "uint256"
        }
    ],
    "name": "getProductDetails",
    "outputs": [
        {
            "internalType": "string",
            "name": "name",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "description",
            "type": "string"
        },
        {
            "internalType": "uint256",
```

```
            "name": "quantity",
            "type": "uint256"
        },
        {
            "internalType": "address",
            "name": "owner",
            "type": "address"
        }
    ],
    "stateMutability": "view",
    "type": "function"
},
{
    "inputs": [],
    "name": "productCount",
    "outputs": [
        {
            "internalType": "uint256",
            "name": "",
            "type": "uint256"
```

```
        }
    ],
    "stateMutability": "view",
    "type": "function"
},
{
    "inputs": [
        {
            "internalType": "uint256",
            "name": "",
            "type": "uint256"
        }
    ],
    "name": "products",
    "outputs": [
        {
            "internalType": "string",
            "name": "name",
            "type": "string"
        },
```

```
        {
            "internalType": "string",
            "name": "description",
            "type": "string"
        },
        {
            "internalType": "uint256",
            "name": "quantity",
            "type": "uint256"
        },
        {
            "internalType": "address",
            "name": "owner",
            "type": "address"
        }
    ],
    "stateMutability": "view",
    "type": "function"
},
{
```

```
"inputs": [
        {
                "internalType": "uint256",
                "name": "_productId",
                "type": "uint256"
        },
        {
                "internalType": "string",
                "name": "_name",
                "type": "string"
        },
        {
                "internalType": "string",
                "name": "_description",
                "type": "string"
        },
        {
                "internalType": "uint256",
                "name": "_quantity",
                "type": "uint256"
```

```
            }
        ],
        "name": "updateProduct",
        "outputs": [],
        "stateMutability": "nonpayable",
        "type": "function"
    }
]
```

7.2 Feature 2:

Program:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract AgricultureRegistry {
    struct foodProduct {
        string name;
        string description;
        uint256 quantity;
        address owner;
```

```
}

mapping(uint256 => foodProduct) public products;
uint256 public productCount;

event ProductAdded(uint256 productId, string name, string description, uint256 quantity, address owner);
event ProductUpdated(uint256 productId, string name, string description, uint256 quantity);

modifier onlyOwner(uint256 _productId) {
    require(products[_productId].owner == msg.sender, "Only the owner can perform this action");
    _;
}

function addProduct(uint256 ProductId, string memory _name, string memory _description, uint256 _quantity) external {

    products[ProductId] = foodProduct(_name, _description, _quantity, msg.sender);
```

```
        productCount++;
        emit ProductAdded(productCount, _name, _description, _quantity, msg.sender);
    }

    function updateProduct(uint256 _productId, string memory _name, string memory _description, uint256 _quantity) external onlyOwner(_productId) {
        foodProduct storage product = products[_productId];
        product.name = _name;
        product.description = _description;
        product.quantity = _quantity;
        emit ProductUpdated(_productId, _name, _description, _quantity);
    }

    function getProductDetails(uint256 _productId) external view returns (string memory name, string memory description, uint256 quantity, address owner) {
        foodProduct memory product = products[_productId];
        return (product.name, product.description, product.quantity, product.owner);
    }
```

}

7.3 Database Schema (if Applicable)

Byte Code:

60806040523480156100105760008fd5b50610f0b8061002060003960
00f3fe608060405234801561001057600080fd5b506004361061005757
60003560e01c80636813b53b1461005c5780637acc0b201461008f5780
639c083284146100c2578063a227379c146100de578063e0f6ef8714610
0fa575b600080fd5b6100766004803603810190610071919061006eb565
b610118565b604051610086949392919061007f8565b60405180910390f
35b6100a960048036038101906100a491906106eb565b6102e7565b60
40516100b9949392919061007f8565b60405180910390f35b6100dc6004
8036038101906100d79190610980565b610447565b005b6100f860048
036038101906100f39190610980565b61056f565b005b61010261069b5
65b60405161010f9190610a1f565b60405180910390f35b60608060008
060008060008781526020019081526020016000206040518060800160
405290816000820180546014d90610a69565b80601f01602080910402
602001604051908101604052809291908181526020018280546101799
0610a69565b80156101c65780601f1061019b576101008083540402835
291602001916101c6565b820191906000526020600020905b81548152
906001019060200180831161a957829003601f168201915b50505050
508152602001600182018054610a69565b80601f0160208091
040260200160405190810160405280929190818152602001828054610
20b90610a69565b80156102585780601f1061022d57610100808354040
283529160200191610258565b820191906000526020600020905b8154
815290600101906020018083116102301b5782900360f168201915b5050
505050815260200160028201548152602001600382016000905490610

1000a900473ffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffff
ffffffff1673ffffffffffffffffffffffffffffffffffff168152505090508060000151
816020015182604001518360600151945094509450945050919350919
3565b60006020528060005260406000206000915090508060000180541
61030a90610a69565b80601f016020809104026020016040519081016040
405280929190818152602001828054610336906l0a69565b801561038
35780601f10610358576101008083540402835291602001916103835651
b8201919060005260206000209051b8154815290600101906020018083
116103665782900360lf168201915b505050505090806001018054610 3
9890610a69565b80601f016020809104026020016040519081016040521
809291908181526020018280546103c490610a69565b8015610411578
0601f106103e6576101008083540402835291602001916104115651820
191906000526020600020905b8154815290600101906020018083116l
03f457829003601f168201915b505050505090806002015490806003 01
600090549061010a900473ffffffffffffffffffffffffffffffffffff169050845
65b604051806080016040528084815260200183815260200182815260
20013373ffffffffffffffffffffffffffffffffffff16815250600080868152602 00
190815260200160002060008201518160000190816104a59190610c46
565b506020820151816001019081610 4bb9190610c46565b506040820
151816002015560608201518160030160006101000a81548173ffffffffff
ffffffffffffffffffffffffff021916908373ffffffffffffffffffffffffffffffffffff1602
179055509050506001600081548092919061052390610d47565b91905
055507fffed2a30aabbaef379429a9bcd823ae1a613502b5e867747d603
13ab0d8388ee6001548484843360405161056195949392919061 0d8f56
5b60405180910390a150505050565b833373ffffffffffffffffffffffffffffffff
ffff166000808381526020019081526020016000206003016000905490 6
101000a900473ffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffff
ffffffffffff161461061357604051 7f08c379a0000000000000000000000

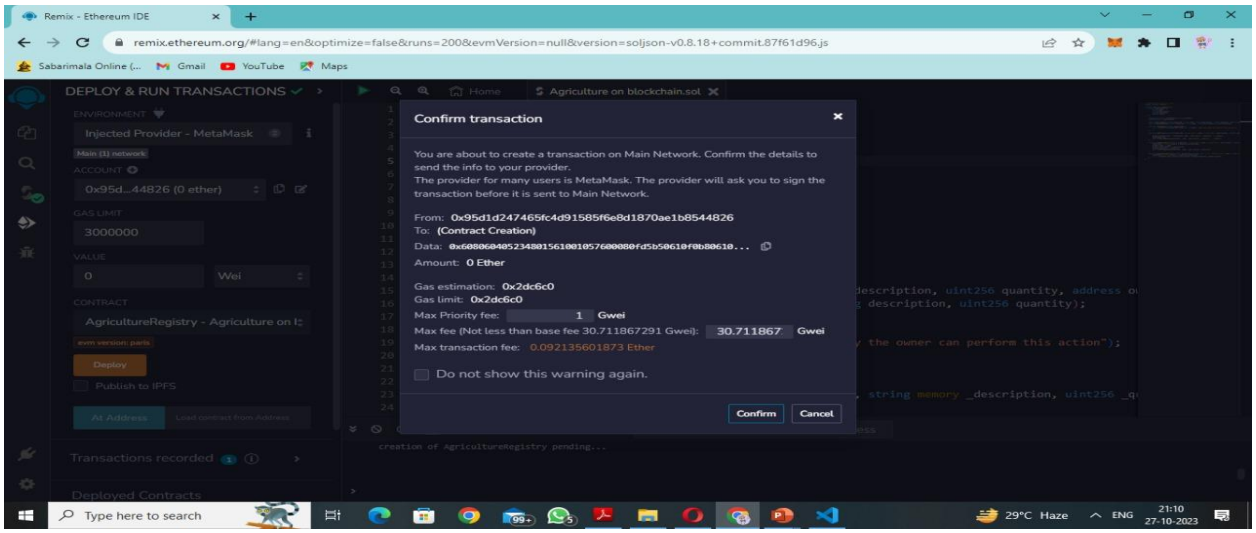
0000000000000000000000000000000081526040161060a90610e6
2565b60405180910390fd5b60008060008781526020019081526020016000209050848160000190816106 3a9190610c46565b5083816001019081610 64c9190610c46565b5082816002018190555 07fa912cc40a5dbbdbdf6a8e9b2bcdac1717b6af3df58c19d3f5d911bca01216f4b868686866040516106 8b9493929190610e82565b60405180910390a150505050505565b60015481565b600060405190509056 5b600080fd5b600080fd5b600081905091905056 5b6106c8816106b5565b81146106d357600080fd5b50565b600081359050610 6e5816106bf565b92915050565b6000602082840312156107015761070 6106ab565b5b600061070f84828501610 6d6565b91505092915050565b600081519050919050565b6000828252602082019050929150505 65b60005b838110156107525780820151818401526020810190506107 37565b60008484015250505050565b600060 1f19601f8301169050919050565b600061077a82610718565b6107848185610723565b935061079481856020860161073 4565b61079d8161075e565b840191505092915050565b6107b1816106b5565b82525050565b600073ffffffffffffffffffffffffffffffffffffffff8216905091905056 5b60006107e2826107b7565b905091905056 5b6107f2816107d7565b82525050565b600060808201905081810360008301526108128761076f565b9050818103602083015261082681866107 6f565b90506108356040830185610 7a8565b61084260608301846107e9565b9594505050505056 5b600080fd5b600080fd5b7f4e487b7100000000000000000000000000000000000000000000000000 0000000000000000600052604160045260246000fd5b61088d8261075e565b810181811067ffffffffffffffff821117156108ac576108ab61085565b5b8060405250505056 5b60006108bf6106a1565b90506108cb828261088456 5b919050565b600067ffffffffffffffff8211156108eb576108ea610855565b5b6108f48261075e565b905060208101905091905056 5b828183376000838301525050565b600061092361091e846108d0565b6

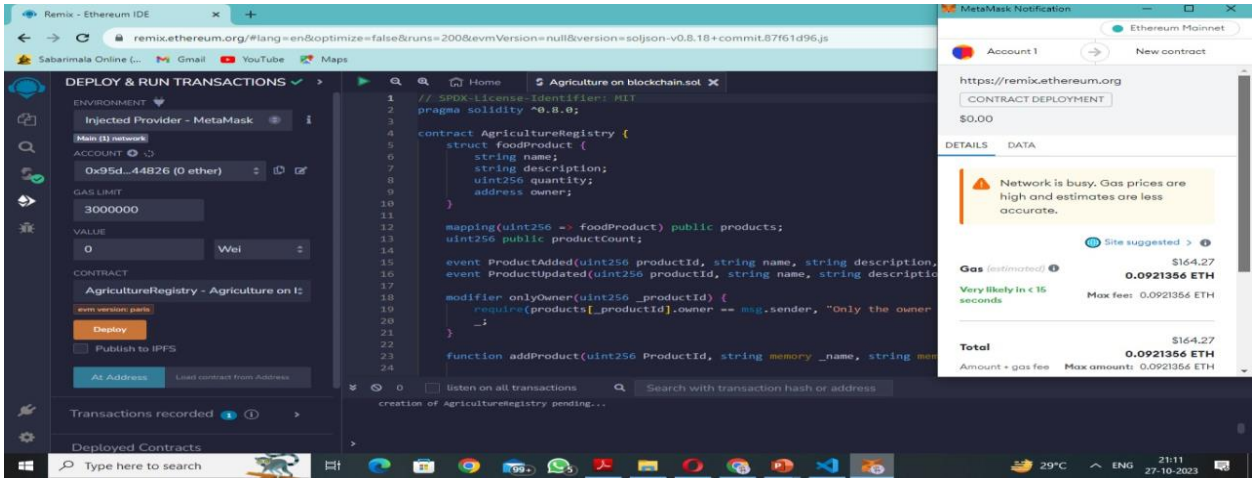
108b5565b905082815260208101848484011115610 93f5761093e61085
0565b5b61094a848285610901565b5093925050505 65b600082601f830
112610967576109666 1084b565b5b8135610977848260208601610910
565b915050929150505 65b600080600080608085870312156109 9a576
109996106ab565b5b60006109a8878288016106d6565b945050602085
013567ffffffffffffffff8111156109c9576109c86106b0565b5b6109d58782
8801610952565b935050604085013567ffffffffffffffff8111156109f65761
09f56106b0565b5b610a028782880161095256 5b925050606061 0a1387
8288016106d6565b915050929591945092505 65b600060208201905 06
10a34600083018461 07a8565b92915050565b7f4e487b7100000000000
0000000000000000000000000000000000000000000600052602260
045260246000fd5b600060028204905060018216 80610a8157607f8216
91505b602082108103610a9457610a93610a3a565b5b50919050565b6
000819050816000526020600020905091905056 5b60006020601f83010
490509190505 65b600082821b905092915050565b600060088302610af
c7ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff82610abf565b610b0
68683610abf565b9550801984169350808616841792505050939250505
0565b60008190509190505 65b6000610b43610b3e610b39846106b556
5b610b1e565b6106b5565b9050919050565b60008190509190505 65b6
10b5d83610b28565b610b71610b6982610b4a565b848454610acc565b
825550505050565b600090565b610b86610b79565b610b91818484610
b54565b505050565b5b81811015610bb557610baa600082610b7e565b
600181019050610b97565b5050565b601f821115610bfa57610bcb8161
0a9a565b610bd484610aaf565b810160208510156 10be3578190505b61
0bf7610bef85610aaf565b830182610b96565b50505b505050565b6000
82821c905092915050565b6000610c1d60001984600802610bff565b19
808316915050929150505 65b6000610c368383610c0c565b9150826002
028217905092915050565b610c4f82610718565b67ffffffffffffffff811115

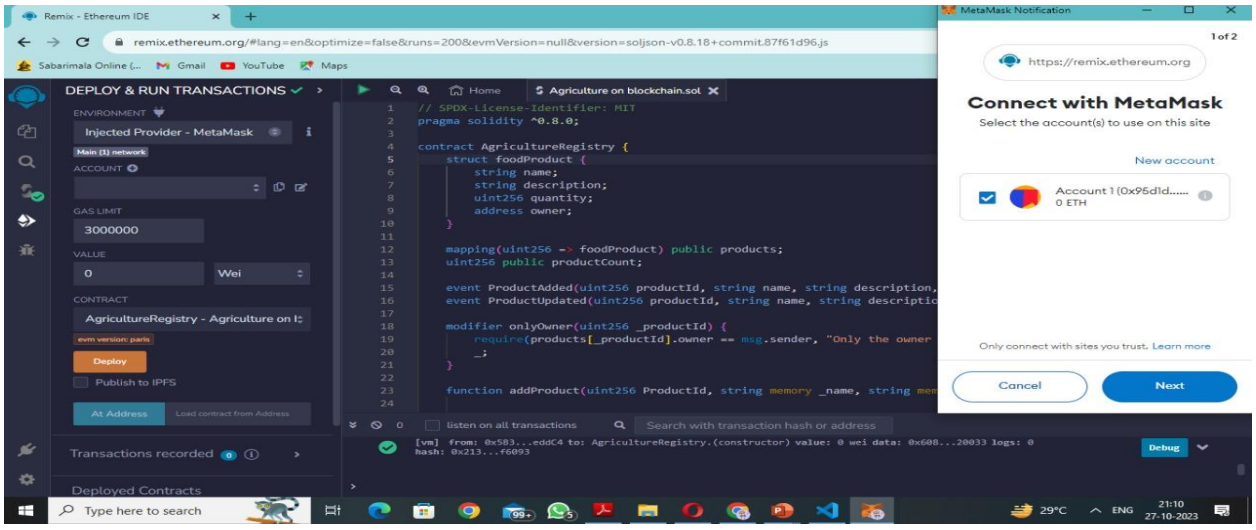
610c6857610c67610855565b5b610c728254610a69565b610c7d828285
610bb9565b600060209050601f831160018114610cb05760008415610c
9e578287015190505b610ca88582610c2a565b865550610d10565b601f
198416610cbe86610a9a565b60005b82811015610ce65784890151825 5
60018201915060208501945060208101905 0610cc1565b86831015610
d035784890151610cff601f891682610c0c565b8355505b600160028802
0188555050505b505050505050565b7f4e487b7100000000000000000
0000000000000000000000000000000000000600052601160045260 2
46000fd5b6000610d52826106b5565b91507fffffffffffffffffffffffffffffffffffffff
ffffffffffffffffffffffffffff8203610d8457610d83610d18565b5b6001820190
50919050565b600060a082019050610da46000830188610 7a8565b818
1036020830152610db6818761076f565b9050818103604083 0152610dc
a818661076f565b9050610dd960608301856107a8565b610de660808301
846107e9565b9695505050505050565b7f4f6e6c7920746865206f776e
65722063616e20706572666f726d207468697320600082015 27f616374
696f6e0000000000000000000000000000000000000000000000000000
6020820152505 65b6000610e4c602683610723565b9150610e5782610
df0565b60408201905091905 0565b60006020820190505 0818103600083
0152610e7b81610e3f565b9050919050565b60006080820190 50610e97
60008301876107a8565b8181036020830152610ea9818661076f565b90
5081810360408301 52610ebd818561076f565b9050610ecc6060830184
6107a8565b9594505050505056fea2646970667358221 2209e97f0cf998
36326f2bd97f5e166dd2c4c289a9b3f8c288a11bfc94ce8347c2164736f6
c63430008120033

8. PERFORMANCE TESTING

8.1 Performance Metrics:

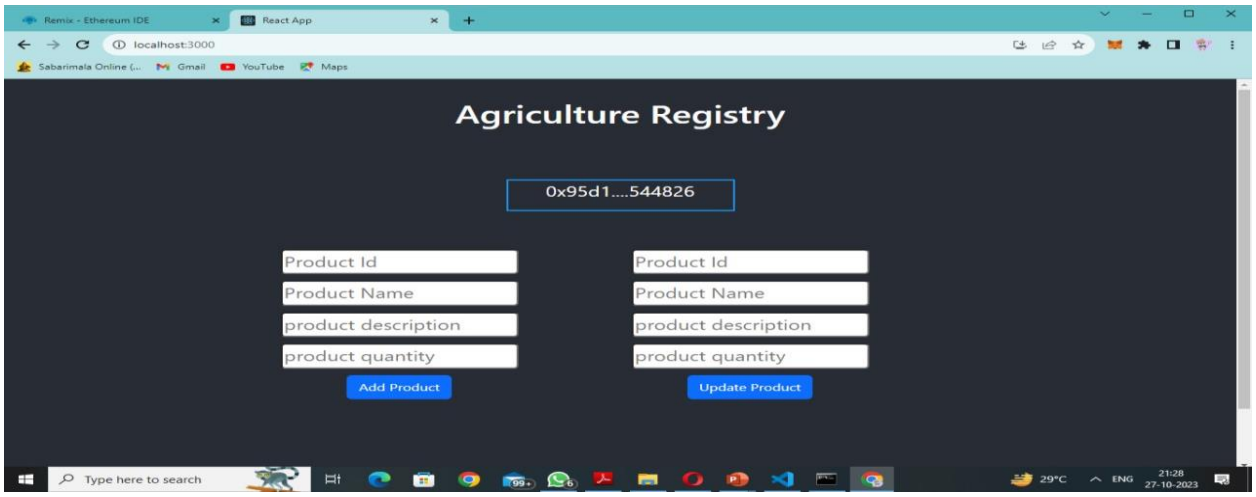

Remix - Ethereum IDE
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS
Injected Provider - MetaMask
Confirm transaction
You are about to create a transaction on Main Network. Confirm the details to send the info to your provider.
The provider for many users is MetaMask. The provider will ask you to sign the transaction before it is sent to Main Network.
From: 0x95d1d247465fc4d91585f6e8d1870ae1b8544826
To: (Contract Creation)
Amount: 0 Ether
Gas estimation: 0x2dc6c0
Gas limit: 0x2dc6c0
Max Priority fee: 1 Gwei
Max fee (Not less than base fee 30.711867291 Gwei): 30.711867 Gwei
Max transaction fee: 0.092135601873 Ether
Do not show this warning again.
Confirm
Cancel
Transactions recorded
Deployed Contracts
Type here to search
29°C Haze
ENG
21:10
27-10-2023

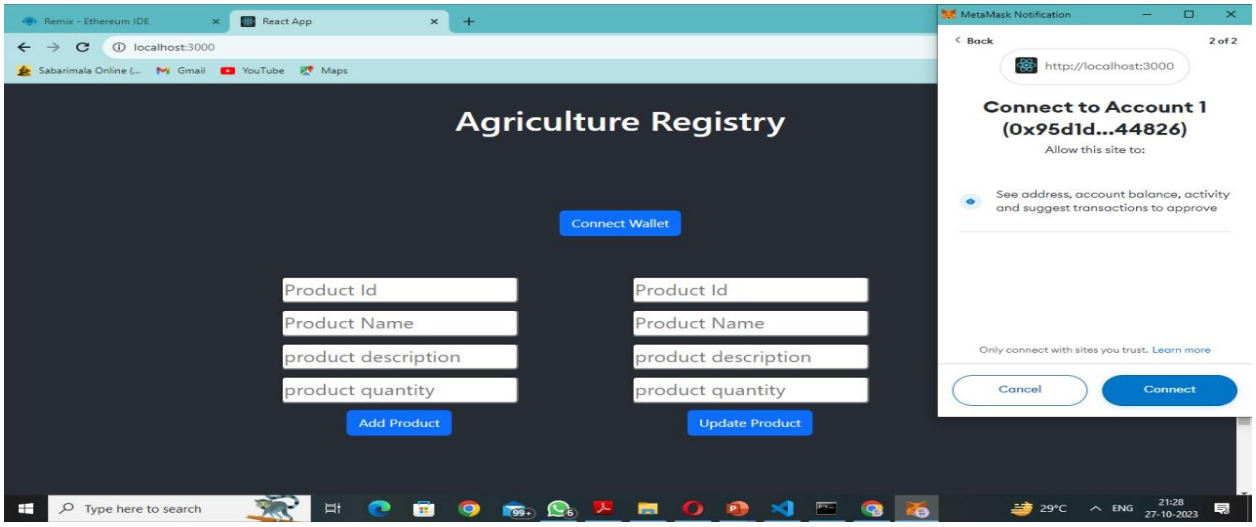

Remix - Ethereum IDE
DEPLOY & RUN TRANSACTIONS
Injected Provider - MetaMask
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract AgricultureRegistry {
struct foodProduct {
string name;
string description;
uint256 quantity;
address owner;
}
mapping(uint256 => foodProduct) public products;
uint256 public productCount;
event ProductAdded(uint256 productId, string name, string description,
event ProductUpdated(uint256 productId, string name, string descriptio
modifier onlyOwner(uint256 _productId) {
require(products[_productId].owner == msg.sender, "Only the owner
_;
}
function addProduct(uint256 ProductId, string memory _name, string mem
MetaMask Notification
Ethereum Mainnet
Account 1
New contract
https://remix.ethereum.org
CONTRACT DEPLOYMENT
$0.00
DETAILS
DATA
Network is busy. Gas prices are high and estimates are less accurate.
Site suggested
Gas (estimated)
$164.27
0.0921356 ETH
Very likely in < 15 seconds
Max fee: 0.0921356 ETH
Total
$164.27
0.0921356 ETH
Amount + gas fee
Max amount: 0.0921356 ETH
Transactions recorded
Deployed Contracts
Type here to search
29°C
ENG
21:11
27-10-2023

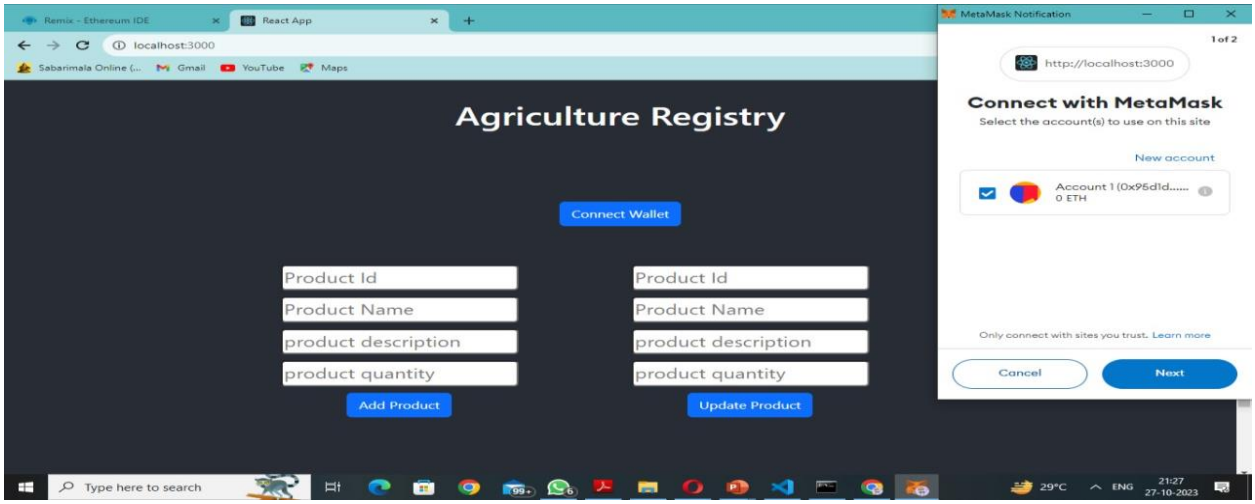

Remix - Ethereum IDE
DEPLOY & RUN TRANSACTIONS
Injected Provider - MetaMask
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract AgricultureRegistry {
struct foodProduct {
string name;
string description;
uint256 quantity;
address owner;
}
mapping(uint256 => foodProduct) public products;
uint256 public productCount;
event ProductAdded(uint256 productId, string name, string description,
event ProductUpdated(uint256 productId, string name, string descriptio
modifier onlyOwner(uint256 _productId) {
require(products[_productId].owner == msg.sender, "Only the owner
_;
}
function addProduct(uint256 ProductId, string memory _name, string mem
MetaMask Notification
1 of 2
https://remix.ethereum.org
Connect with MetaMask
Select the account(s) to use on this site
New account
Account 1 (0x95d1d.....
0 ETH
Only connect with sites you trust. Learn more
Cancel
Next
[vm] from: 0x583...eddC4 to: AgricultureRegistry.(constructor) value: 0 wei data: 0x608...20033 logs: 0
hash: 0x213...f6093
Debug
Transactions recorded
Deployed Contracts
Type here to search
29°C
ENG
21:10
27-10-2023

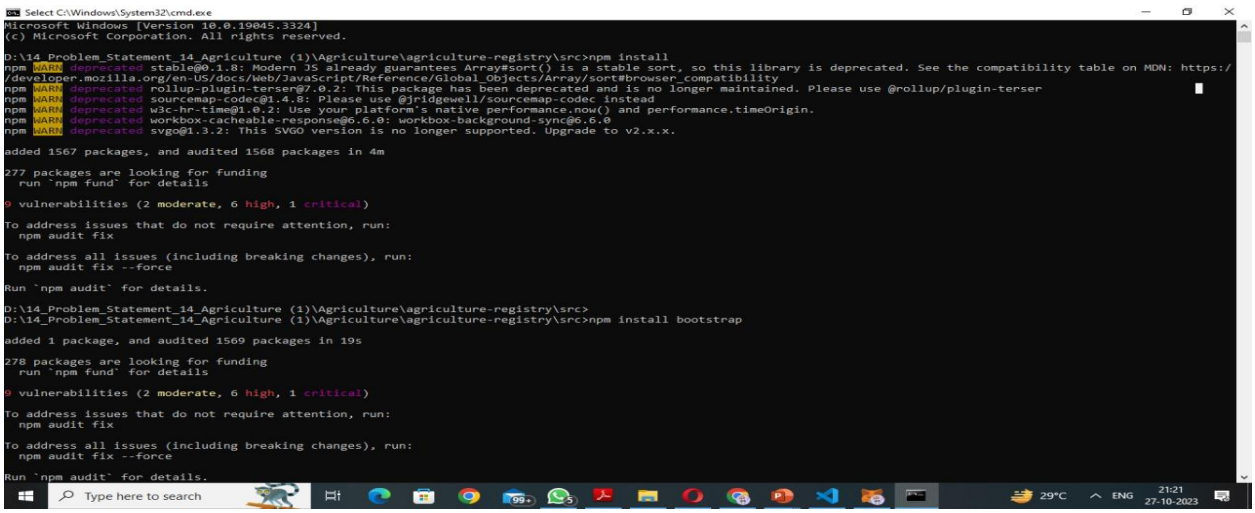
9.RESULTS

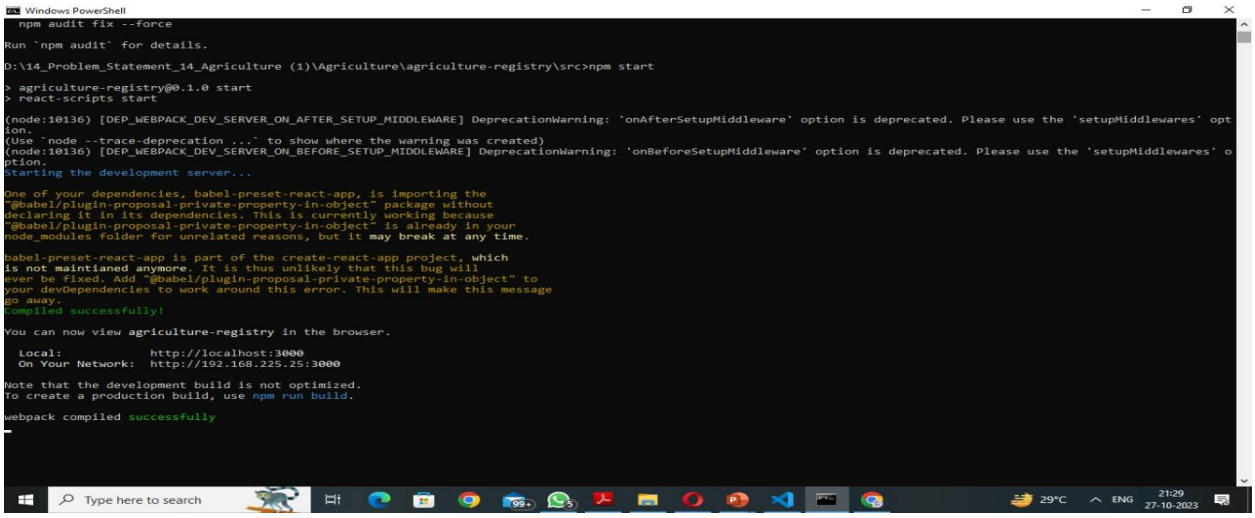
9.1 Output Screenshots






Remix - Ethereum IDE
React App
localhost:3000
Sabarimala Online (...
Gmail
YouTube
Maps
Agriculture Registry
Connect Wallet
Product Id
Product Name
product description
product quantity
Add Product
Product Id
Product Name
product description
product quantity
Update Product
MetaMask Notification
1 of 2
http://localhost:3000
Connect with MetaMask
Select the account(s) to use on this site
New account
Account 1 (0x96d1d.....
0 ETH
Only connect with sites you trust. Learn more
Cancel
Next
Type here to search
29°C
ENG
21:27
27-10-2023


Select C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.
D:\14_Problem_Statement_14_Agriculture (1)\Agriculture\agriculture-registry\src>npm install
npm WARN deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the compatibility table on MDN: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort#browser_compatibility
npm WARN deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plugin-terser
npm WARN deprecated sourcemap-codec@1.4.8: Please use @jridgewell/sourcemap-codec instead
npm WARN deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm WARN deprecated workbox-cacheable-response@6.6.0: workbox-background-sync@6.6.0
npm WARN deprecated svgo@1.3.2: This SVGO version is no longer supported. Upgrade to v2.x.x.
added 1567 packages, and audited 1568 packages in 4m
277 packages are looking for funding
run `npm fund` for details
9 vulnerabilities (2 moderate, 6 high, 1 critical)
To address issues that do not require attention, run:
npm audit fix
To address all issues (including breaking changes), run:
npm audit fix --force
Run `npm audit` for details.
D:\14_Problem_Statement_14_Agriculture (1)\Agriculture\agriculture-registry\src>
D:\14_Problem_Statement_14_Agriculture (1)\Agriculture\agriculture-registry\src>npm install bootstrap
added 1 package, and audited 1569 packages in 19s
278 packages are looking for funding
run `npm fund` for details
9 vulnerabilities (2 moderate, 6 high, 1 critical)
To address issues that do not require attention, run:
npm audit fix
To address all issues (including breaking changes), run:
npm audit fix --force
Run `npm audit` for details.
Type here to search
29°C
ENG
21:21
27-10-2023


Windows PowerShell
npm audit fix --force
Run `npm audit` for details.
D:\14_Problem_Statement_14_Agriculture (1)\Agriculture\agriculture-registry\src>npm start
> agriculture-registry@0.1.0 start
> react-scripts start
(node:10136) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationWarning: 'onAfterSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
(Use `node --trace-deprecation ...` to show where the warning was created)
(node:10136) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] DeprecationWarning: 'onBeforeSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' option.
Starting the development server...
One of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it may break at any time.
babel-preset-react-app is part of the create-react-app project, which
is not maintianed anymore. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.
Compiled successfully!
You can now view agriculture-registry in the browser.
Local: http://localhost:3000
On Your Network: http://192.168.225.25:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
webpack compiled successfully
Type here to search
29°C
ENG
21:29
27-10-2023

10. ADVANTAGES & DISADVANTAGES

Advantages:

Traceability:

It allows for the tracing of products back to their origin, ensuring accountability in case of contamination or quality issues.

Efficient Management:

Proper documentation helps in the efficient management of resources, including land, water, and fertilizers.

Compliance and Regulation:

It helps in ensuring compliance with legal and regulatory requirements, such as organic certifications or environmental standards.

Data-Driven Decision Making:

Access to historical data can aid in making informed decisions about crop selection, planting times, and resource allocation.

Research and Development:

Comprehensive documentation can be invaluable for research purposes, enabling the development of new farming techniques and technologies.

Disadvantages:

Administrative Burden:

Maintaining detailed documentation can be time-consuming and may divert resources from actual farming activities.

Costs:

Setting up and maintaining a robust documentation system can involve financial investments in software, training, and personnel.

Privacy Concerns:

Depending on the extent of the documentation, there may be concerns about privacy and the security of sensitive information.

Technology Dependency:

If the documentation system relies heavily on technology, there could be vulnerabilities to consider, such as data loss or cyber threats.

Resistance to Change:

Implementing a new documentation system may face resistance from farmers who are accustomed to traditional methods. It's worth noting that the specific advantages and disadvantages can vary depending on the scale of the farm, the type of crops, and the local regulatory environment.

11. CONCLUSION:

The integration of blockchain in agriculture can provide benefits such as increased transparency, traceability, and efficiency in the supply chain. It helps in tracking the origin of produce, ensuring food safety, and reducing fraud. However, successful implementation requires addressing challenges like access to technology, data privacy, and standardization. Overall, combining agriculture with blockchain holds potential for improving various aspects of the industry.

12. APPENDIX

Source Code:

```
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;


contract AgricultureRegistry {

   struct foodProduct {

      string name;
```

```
        string description;
        uint256 quantity;
        address owner;
    }

    mapping(uint256 => foodProduct) public products;
    uint256 public productCount;

    event ProductAdded(uint256 productId, string name, string description, uint256 quantity, address owner);
    event ProductUpdated(uint256 productId, string name, string description, uint256 quantity);

    modifier onlyOwner(uint256 _productId) {
        require(products[_productId].owner == msg.sender, "Only the owner can perform this action");
        _;
    }

    function addProduct(uint256 ProductId, string memory _name, string memory _description, uint256 _quantity) external {
```

```
        products[ProductId] = foodProduct(_name, _description, _quantity, msg.sender);
        productCount++;
        emit ProductAdded(productCount, _name, _description, _quantity, msg.sender);
    }

    function updateProduct(uint256 _productId, string memory _name, string memory _description, uint256 _quantity) external onlyOwner(_productId) {
        foodProduct storage product = products[_productId];
        product.name = _name;
        product.description = _description;
        product.quantity = _quantity;
        emit ProductUpdated(_productId, _name, _description, _quantity);
    }

    function getProductDetails(uint256 _productId) external view returns (string memory name, string memory description, uint256 quantity, address owner) {
        foodProduct memory product = products[_productId];
```

```
    return (product.name, product.description, product.quantity,
product.owner);
  }
}
```

GitHub & Project Demo Link:

Github link : https://github.com/mmkeerthana2002/Agriculture-docs

Demo link: https://youtu.be/2ne7ASlq1j8?si=ekr40bPHKRUs-p_6