K S R INSTITUTE FOR ENGINEERING AND TECHNOLOGY

TIRUCHENGODE: 637 215

Computer Science and Engineering

NAAN MUDHALVAN

SB8024- Blockchain Development

by Naan Mudhalvan Scheme – 2023

TEAM ID: NM2023TMID

PROJECT DOMAIN: BLOCKCHAIN TECHNOLOGY

PROJECT TITLE: VACCINE TRACKING TRANSPARENT

TEAM MEMBERS

REGISTER NUMBER	NAME
731620104015	ELANGKUMARAN T
731620104017	GNANAMOORTHI M
731620104035	MOHAN KUMAR S
731620104036	NAVEEN KUMAR M

TABLE OF CONTENT

1.INTRODUCTION

Vaccine tracking and transparency have become paramount in the global effort to combat infectious diseases and ensure public health. In an era where vaccines are crucial tools for preventing and managing outbreaks, the need for a robust and transparent tracking system is undeniable. The first paragraph can introduce the importance of vaccine tracking in ensuring the efficacy and safety of immunization programs, while the second paragraph can delve into the benefits of transparency in enhancing public trust and global cooperation, ultimately contributing to a healthier and more secure world.

Transparency in vaccine tracking is equally important as it fosters public trust, encourages accountability, and promotes global cooperation. When the entire vaccine distribution process is transparent and well-documented, it instills confidence in the safety and efficacy of vaccines, addressing concerns and skepticism. Additionally, transparent vaccine tracking allows governments, organizations, and international agencies to collaborate effectively in addressing public health crises. During global pandemics, sharing real-time data on vaccine distribution and coverage becomes essential for coordinated efforts to control and mitigate the spread of diseases.

1.1 Project Overview

The "Vaccine Tracking Transparency" project is an innovative initiative designed to enhance the efficiency, accountability, and trustworthiness of vaccine distribution and administration systems. In response to the pressing need for robust vaccination programs, this project seeks to establish a comprehensive and transparent framework for tracking vaccines at every stage of their journey, from production to patient inoculation. The primary objectives of this project are to ensure the seamless distribution of vaccines, reduce wastage, and, most importantly, build and maintain public trust in immunization efforts.By integrating advanced data management and blockchain technology, the project aims to create an incorruptible and auditable ledger of vaccine transactions, making the entire process transparent and tamper-proof. This transparency not only helps health authorities identify and address supply chain issues promptly but also serves to allay concerns related to vaccine safety and distribution, fostering public confidence.

1.2 Purpose

The purpose of establishing transparent vaccine tracking systems is multifaceted and critical to the success of immunization programs on a global scale. Firstly, it serves as a means to enhance the efficiency and accountability of vaccine distribution and administration. By providing real-time visibility into the entire vaccine supply chain, from manufacturing to delivery to healthcare facilities, transparent tracking systems help identify bottlenecks, mitigate distribution challenges, and reduce wastage. This efficiency ensures that vaccines reach their intended recipients in a timely and cost-effective manner. These tracking systems are vital in facilitating international collaboration during health emergencies. In the event of a global pandemic or epidemic, transparent vaccine tracking allows governments, health organizations, and international agencies to share crucial data and coordinate efforts effectively, resulting in a more coordinated and robust response to health crises.

2.LITERATURE SURVEY

"How Blockchain Is Changing Finance",Auther: A. Tapscott and T. Don,Year:2017,Publisher:IEEE

Blockchain was originally developed as the technology behind cryptocurrencies like Bitcoin. A vast, globally distributed ledger running on millions of devices, it is capable of recording anything of value. Money, equities, bonds, titles, deeds, contracts, and virtually all other kinds of assets can be moved and stored securely, privately, and from peer to peer, because trust is established not by powerful intermediaries like banks and governments, but by network consensus, cryptography, collaboration, and clever code. For the first time in human history, two or more parties, be they businesses or individuals who may not even know each other, can forge agreements, make transactions, and build value without relying on intermediaries (such as banks, rating agencies, and government bodies such as the U.S. Department of State) to verify their identities, establish trust, or perform the critical business logic contracting, clearing, settling, and record-keeping tasks that are foundational to all forms of commerce.

"A conceptual framework for blockchain smart contract adoption to manage real estate deals in smart cities",Auther: Fahim Ullah & Fadi Al-Turjman,Year:2021,Publisher:IEEE

Blockchains-based smart contracts are disrupting the smart real estate sector of the smart cities. The current study explores the literature focused on blockchain smart contracts in smart real estate and proposes a conceptual framework for its adoption in smart cities. Based on a systematic review method, the literature published between 2000 and 2020 is explored and analyzed. From the literature, ten key aspects of the blockchain smart contracts are highlighted that are grouped into six layers for adopting the smart contracts in smart real estate. The decentralized application and its interactions with Ethereum Virtual Machine (EVM) are presented to show the development of a smart contract that can be used for blockchain smart contracts in real estate. Further, a detailed design and interaction mechanism are highlighted for the real estate owners and users as parties to a smart contract. A list of functions for initiating, creating, modifying, or terminating a smart contract is presented along with a stepwise procedure for establishing and terminating smart contracts. The current study can help the users enjoy a more immersive, user-friendly, and visualized contracting process, whereas the owners, property technologies (Proptech) companies, and real estate agents can enjoy more business and sales. This can help disrupt traditional real estate and transform it into smart real estate in line with industry 4.0 requirements.

2.1 Existing Problem

In vaccine tracking is the lack of a standardized and fully transparent system for monitoring the entire vaccine supply chain. Existing systems often lack interoperability, leading to fragmented data that hinders the seamless flow of information from vaccine manufacturers to healthcare providers. This fragmentation not only results in inefficiencies but also makes it difficult to identify and respond to issues in real-time. Furthermore, in many regions, paper-based record-keeping systems are still in use, which are susceptible to errors and fraud, thus compromising the transparency and integrity of the vaccine distribution process.Another critical issue is the limited accessibility and transparency of vaccine data to the public. Inadequate communication and information-sharing regarding vaccine distribution and administration can lead to mistrust and misinformation, particularly in an era where the spread of false information can undermine public confidence in vaccination programs. This lack of transparency can have profound consequences, including vaccine hesitancy and, in some cases, vaccine refusal, which can impede efforts to control and prevent infectious diseases.

2.2 References

[1] A. Tapscott and T. Don, "How blockchain is changing finance," Harvard Business Review, vol. 1, no. 9, pp. 2–5, 2017.
[2] Q. I. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du et al., "MeDShare: Trust-less medical data sharing among cloud service providers via blockchain," IEEE Access, vol. 5, pp. 14757–14767, 2017.
[3] O. Novo, "Blockchain meets IoT: An Architecture for scalable access management in IoT," IEEE Internet of Things Journal, vol. 5, no. 2, pp. 1184–1195, 2018.
[4] J. Sun, Y. Yan and K. Z. Zhang, "Blockchain-based sharing services: What blockchain technology can contribute to smart cities," Financial Innovation, vol. 2, no. 1, pp. 1–9, 2016.

2.3 Problem Statement Definition

The problem statement for smart real estate management using Blockchain is a distributed ledger technology that enables transparent and secure tracking of various assets, including vaccines. It functions as a decentralized network of nodes, where each participant has a copy of the entire ledger. In the context of vaccine tracking, blockchain offers a tamper-proof and immutable record of vaccine-related data, from production to administration. This ensures that the information is trustworthy and accessible to authorized parties, enhancing transparency in the vaccine supply chain. Additionally, the use of smart contracts on the Ethereum blockchain can automate and facilitate the tracking process, allowing for the seamless addition of vaccine details to the blockchain and providing a means to query this data whenever necessary.Smart contracts are self-executing agreements that run on the blockchain. In the context of vaccine tracking, a smart contract on the Ethereum blockchain can be designed to record and manage vaccine-related information. This includes details such as production batch numbers, expiration dates, storage conditions, and distribution records. Authorized parties, such as vaccine manufacturers, distributors, healthcare providers, and regulatory agencies, can input this information into the smart contract.

3.IDEATION & PROPOSED SOLUTION

Ideation:

Innovative ideas for transparent vaccine tracking are essential in the current global landscape, where ensuring the efficient and trustworthy distribution of vaccines is of paramount importance. One promising concept involves the integration of blockchain technology, as mentioned earlier, to create an immutable ledger for vaccine data. This blockchain ledger would allow for the real-time tracking of vaccines from production to administration, with each transaction securely recorded and visible to authorized participants. Furthermore, the implementation of unique digital identifiers, such as QR codes, on vaccine vials and packaging, could enable healthcare providers and patients to scan and verify the authenticity and provenance of vaccines at the point of delivery, ensuring that only genuine and safe vaccines are administered.Additionally, leveraging the power of the Internet of Things (IoT) can provide another layer of transparency. IoT sensors can be embedded in vaccine storage units and transportation containers to monitor temperature, humidity, and other environmental conditions in real-time. Any deviations from optimal conditions could trigger automated alerts, ensuring that vaccines are stored and transported under the appropriate conditions

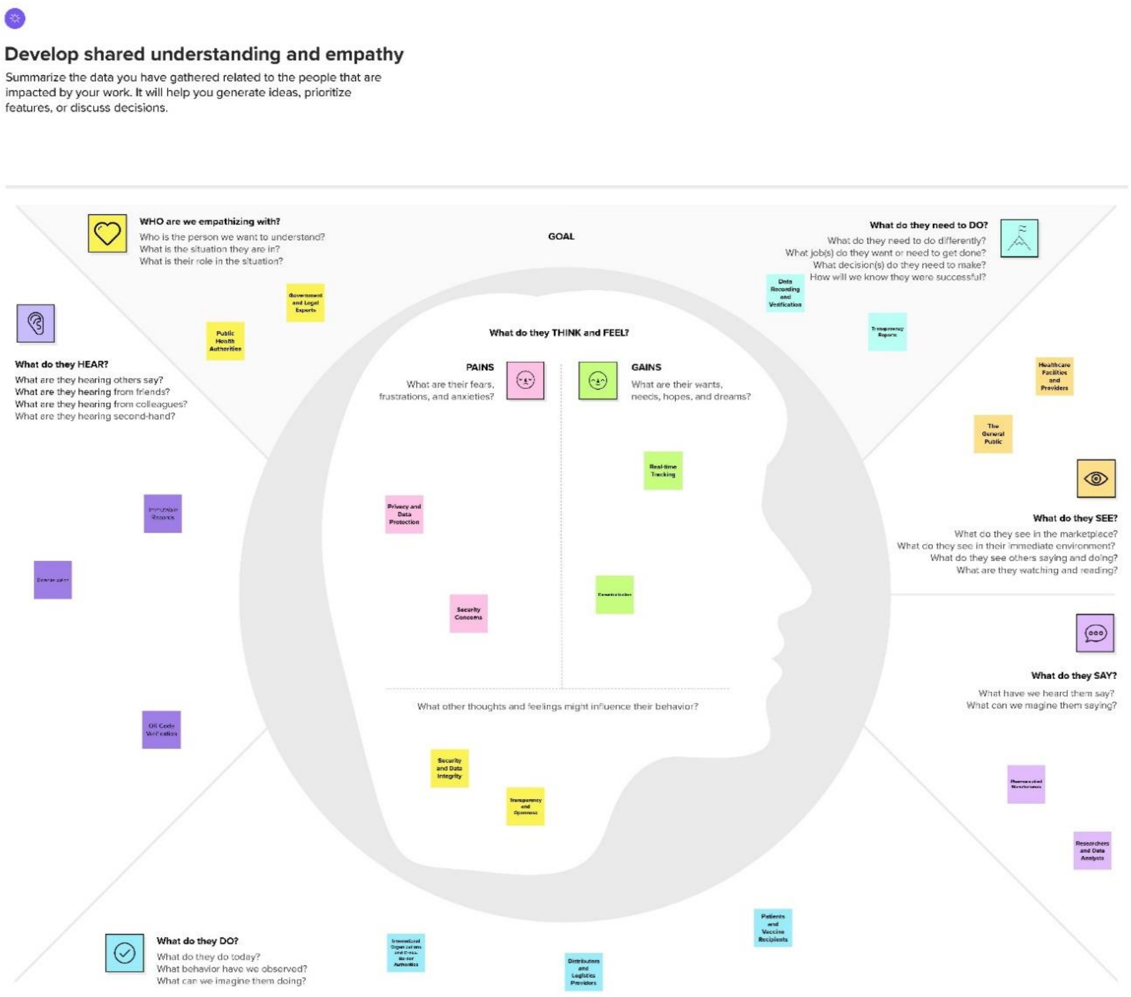
Proposed Solution:

A comprehensive and purposeful solution for transparent vaccine tracking can revolutionize the way vaccines are distributed and administered, ensuring public health is safeguarded. Leveraging blockchain technology, a purpose-built, decentralized system can be established, allowing all stakeholders in the vaccine supply chain to securely record and access data in real-time. This blockchain ledger ensures the traceability and integrity of vaccine information, from production to the point of administration, minimizing errors and fraud. To further enhance transparency, implementing a unique digital identifier system, such as QR codes, on vaccine packaging would enable instant verification of vaccine authenticity, empowering healthcare providers and patients with the ability to verify vaccine origins. By implementing these integrated solutions, we can establish a transparent, efficient, and user-centric vaccine tracking system that instills trust in immunization efforts, minimizes wastage, and enhances the reliability and safety of vaccines as they make their way from production facilities to the arms of individuals, ultimately contributing to improved global public health outcomes.

3.1 Empathy Map Canvas

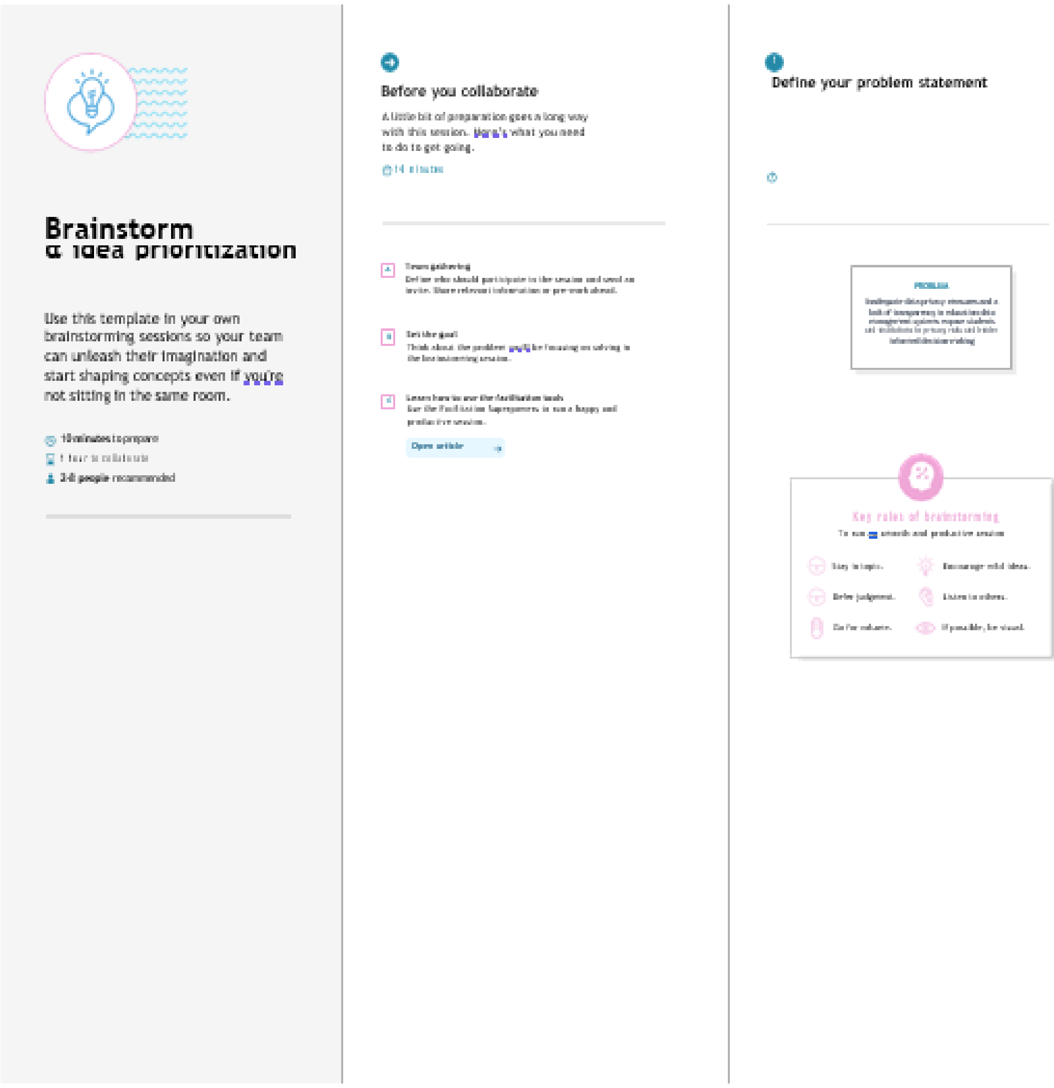
The empathy map canvas for transparent vaccine tracking envisions understanding the thoughts and feelings of all stakeholders involved in the process. Healthcare providers may express a need for efficient tracking systems to ensure timely vaccinations and minimize discrepancies. Vaccine manufacturers may feel the pressure to maintain quality and safety while dealing with high demand.

<u>Transparent Education Data Management</u>

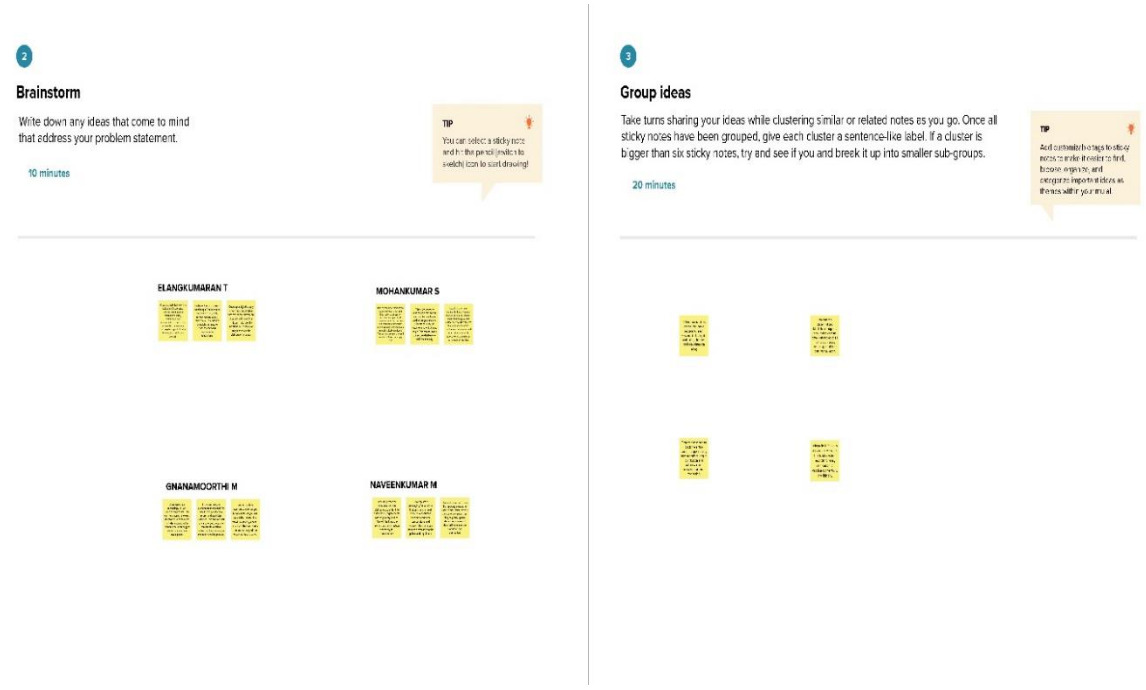


3.2 Ideation & Brainstorming

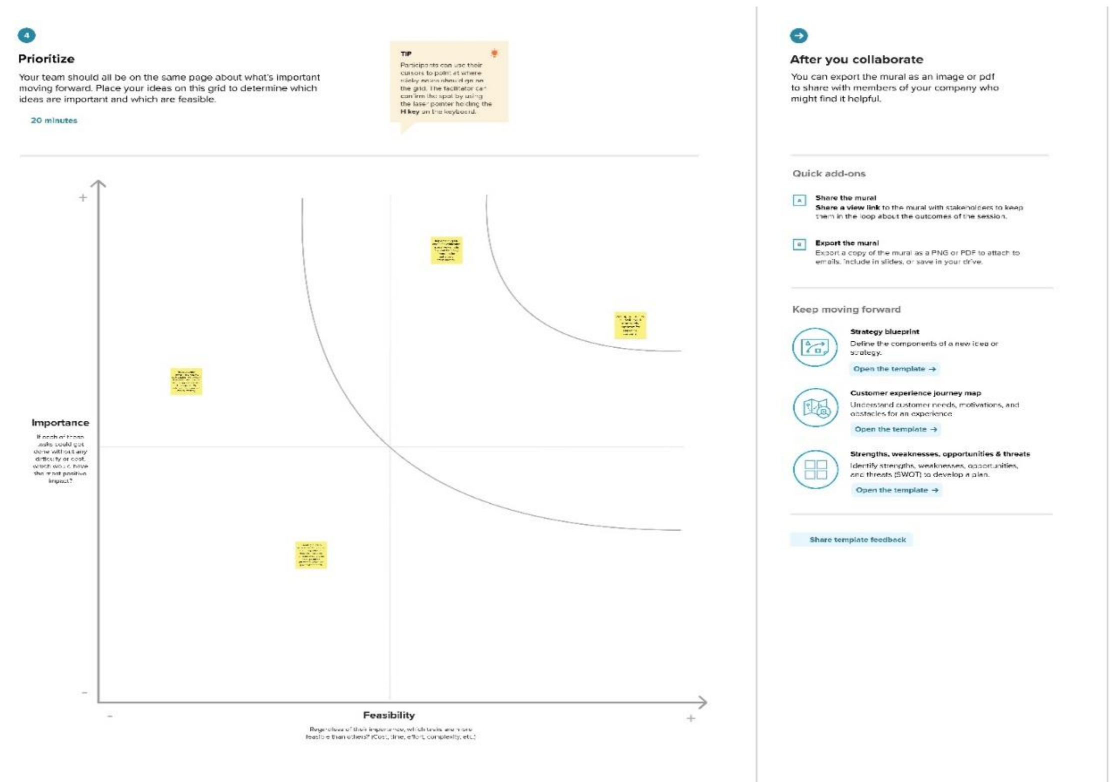
Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



4.REQUIREMENT ANALYSIS

4.1 Functional Requirements

Functional requirements for a smart real estate management system using blockchain encompass various essential features and capabilities:

1. **Blockchain Technology:** Implement a blockchain infrastructure, such as Ethereum, to provide an immutable and tamper-proof ledger for recording vaccine data. This blockchain should support smart contracts for automated data management.

2. **Unique Digital Identifiers:** Develop a system for assigning unique digital identifiers to each vaccine vial or packaging, such as QR codes or RFID tags, allowing for real-time verification and traceability.

3. **Real-time Data Recording:** Enable all stakeholders, including vaccine manufacturers, distributors, healthcare providers, and regulatory agencies, to input data in real-time, ensuring accurate and up-to-date records.

4. **Decentralized Access:** Implement a decentralized access system where authorized participants have access to the blockchain ledger, guaranteeing data security and privacy.

5. **Data Security:** Utilize encryption and access controls to protect sensitive vaccine data from unauthorized access, ensuring data integrity and privacy.

6. **IoT Integration:** Deploy IoT sensors to monitor environmental conditions during vaccine storage and transportation, with automatic alerting for

deviations from optimal conditions.

7. **User-friendly Mobile Application:** Develop a mobile application for individuals to track their vaccination history, receive reminders for upcoming doses, and access transparent information on vaccine safety and efficacy.

8. **Interoperability:** Ensure that the system can integrate with existing healthcare and vaccine management systems to facilitate data exchange and interoperability.

9. **Regulatory Compliance:** Comply with local and international regulatory requirements for vaccine tracking and data management, and provide features to assist regulatory reporting and audits.

10. **Data Analytics:** Incorporate data analytics capabilities to extract insights from vaccine-related data, helping to identify trends, optimize supply chains, and respond swiftly to any issues.

11. **Scalability:** Design the system to handle a large volume of vaccine-related data as vaccination programs expand and evolve over time.

12. **Public Information Accessibility:** Create a public-facing portal or platform to provide transparent and easy-to-understand information about vaccine safety, efficacy, and distribution for individuals and communities.

13. **Collaboration Features:** Enable communication and data sharing between different stakeholders, fostering collaboration during health crises.

14. **User Training and Support:** Provide training and support for users to ensure they can effectively use the system and resolve any issues that may arise.

4.2 Non Functional Requirements

Non-functional requirements for a vaccine tracking transperant using blockchain are equally important as they define the quality attributes and performance expectations of the system. Here are some non-functional requirements:

1. **Performance:** The system must have low latency and high throughput to handle a large volume of data efficiently, especially during peak vaccination periods.

2. **Scalability:** It should be able to scale to accommodate an increasing number of users, devices, and vaccine records without significant degradation in performance.

3. **Reliability:** The system should be highly reliable, with minimal downtime, ensuring continuous access to vaccine data.

4. **Availability:** The system should be available 24/7 to allow for real-time data recording and access from different time zones.

5. **Data Integrity:** Data stored in the blockchain must be tamper-proof and immune to unauthorized changes, ensuring the integrity of vaccine records.

6. **Security:** Robust security measures, including encryption, access controls, and authentication, should be in place to protect vaccine data from unauthorized access and breaches.

7. **Compliance:** The system must comply with relevant data protection and privacy regulations, such as GDPR or HIPAA, to safeguard individuals' personal health information.

8. **Auditability:** Ensure that the system provides comprehensive audit trails, allowing for the tracking of all changes made to vaccine data and user actions.

9. **Usability:** The user interface, both for healthcare professionals and the public, should be intuitive and user-friendly, requiring minimal training.

10. **Accessibility:** The system should be accessible to people with disabilities, in compliance with accessibility standards (e.g., WCAG).

11. **Interoperability:** Ensure that the system can integrate with existing healthcare and government systems, enabling seamless data exchange and cooperation.

12. **Disaster Recovery:** Implement robust disaster recovery and backup procedures to ensure data recovery in the event of system failures or unforeseen disasters.

13. **Response Time**: The system should respond quickly to user queries and requests, especially for real-time verification and tracking of vaccine data.

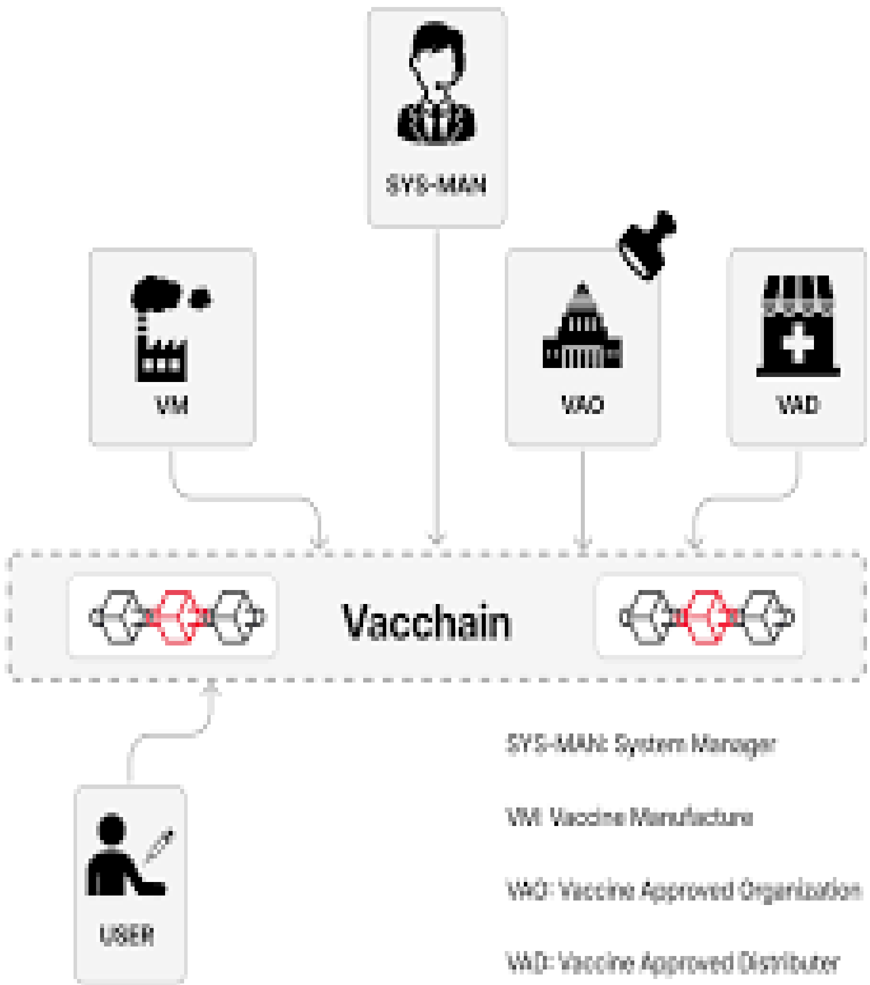
14. **Capacity Planning:** Regularly assess system performance and plan for capacity upgrades as needed to accommodate growing demand.

15. **Cost-effectiveness:** The system should be designed to be cost-effective, with a balanced allocation of resources and cost-efficient maintenance.

5.PROJECT DESIGN

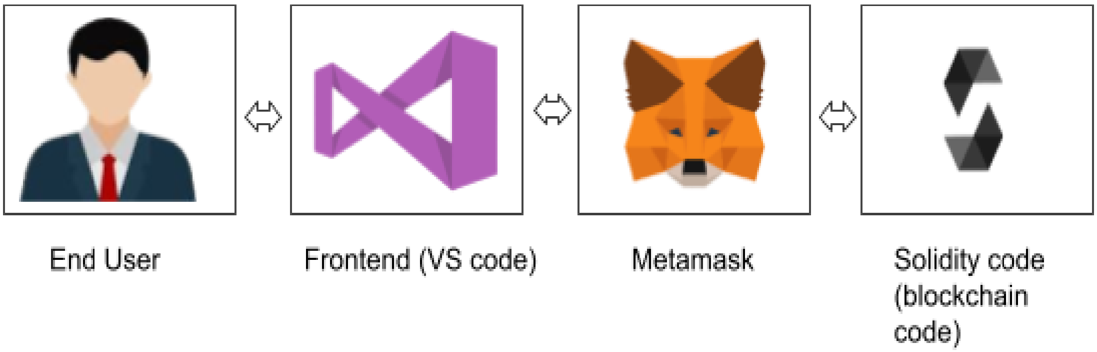
The project design for transparent vaccine tracking will integrate blockchain technology for secure data storage and tamper-proof records. Unique digital identifiers, such as QR codes, will enable real-time verification of vaccine authenticity. IoT sensors will monitor storage conditions, while a user-friendly mobile application ensures accessibility and alerts. Robust security measures and compliance with data protection regulations will safeguard.

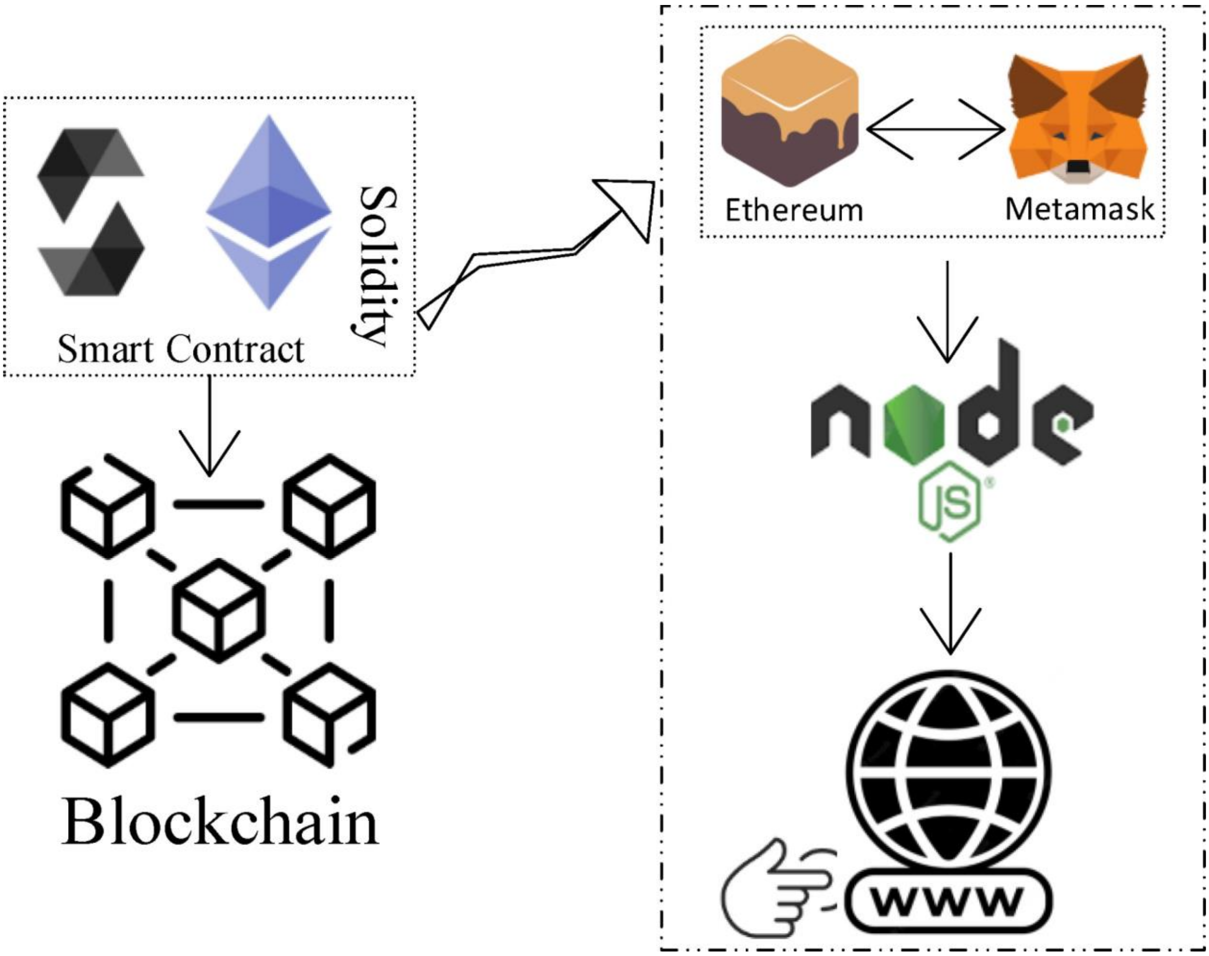
5.1 Data Flow Diagrams & User Stories



5.2 Solution Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions.

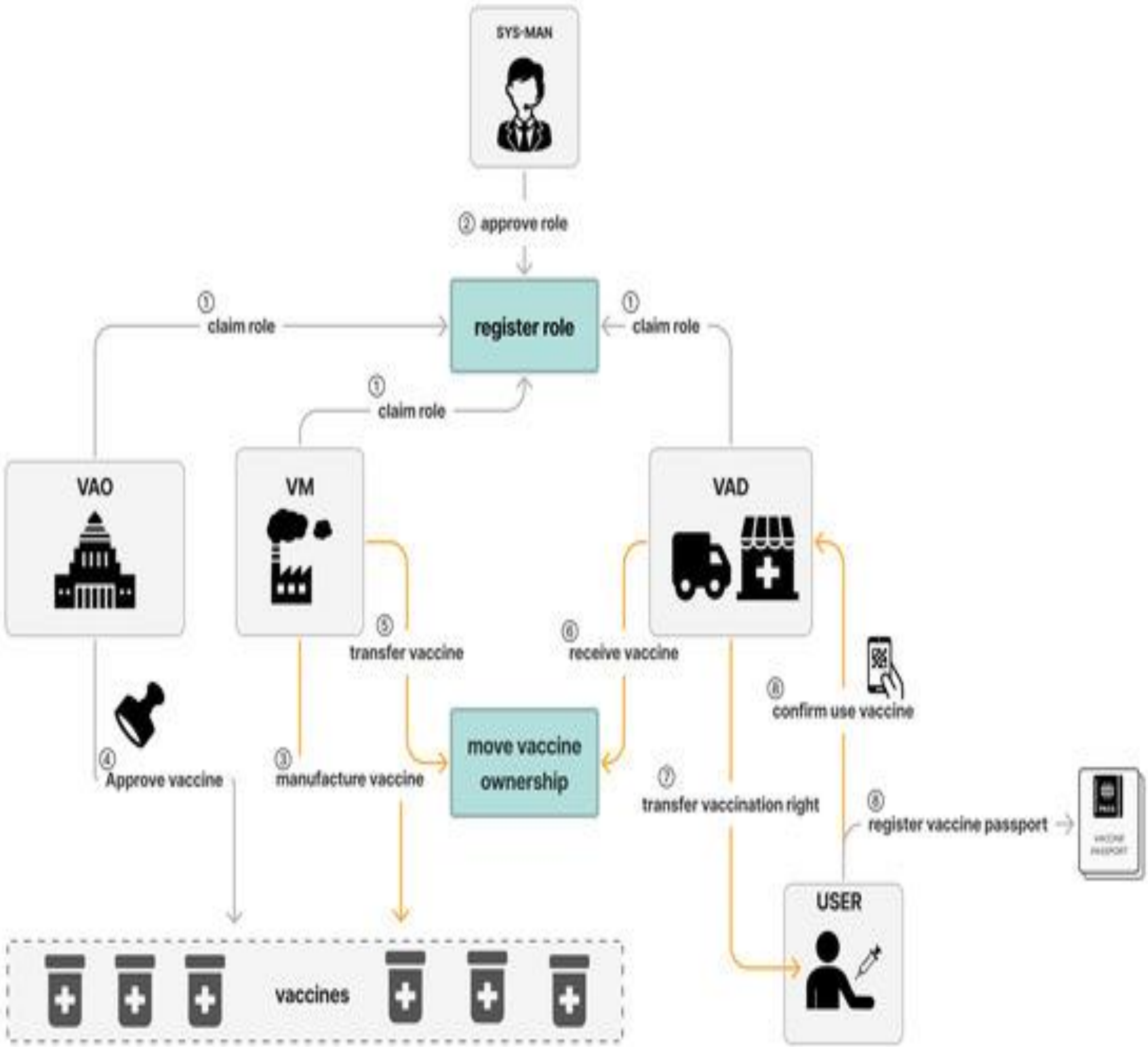




6.PROJECT PLANNING AND SCHEDULING

Project planning and scheduling for transparent education data management is crucial to ensure that educational institutions can effectively collect, store, analyze, and share data while maintaining data security and transparency.

6.1 Technical Architecture



6.2 Sprint Planning & Estimation

Sprint planning and estimation for the development of a transparent vaccine tracking system are crucial steps in ensuring the project's success. During sprint planning, the project team, including developers, testers, and stakeholders, collaboratively defines the scope of work for a specific sprint or development cycle. In the context of vaccine tracking, each sprint can focus on critical features or modules, such as blockchain integration, IoT sensor implementation, or the development of a mobile application. The team breaks down these features into smaller tasks and estimates the time required for each task, considering factors like complexity and dependencies.

Agile methodologies, such as Scrum, can be employed for sprint planning and estimation. The team prioritizes tasks based on their importance and potential impact on the project's overall objectives. Once the sprint backlog is defined, the team estimates the effort required for each task, typically using story points or time-based estimations. These estimates guide the allocation of work for the sprint and help ensure that the team can deliver a functional and valuable increment by the end of the sprint.

Sprint planning and estimation allow for a more predictable and iterative development process. It enables the team to adapt to changing requirements and challenges, ensuring that the vaccine tracking system is developed efficiently and transparently while meeting stakeholder needs and regulatory requirements. This iterative approach also facilitates regular feedback from stakeholders, contributing to the overall success of the project.

6.3 Sprint Delivery Schedule

A Sprint Delivery Schedule for a transparent vaccine tracking system is crucial for planning and managing the project efficiently. This schedule outlines the completion of various development phases over a set timeframe. Here's a simplified example of a Sprint Delivery Schedule for the first three sprints:

In this initial sprint, the focus is on establishing the foundational elements of the project. The team will dedicate the first two weeks to setting up the development environment and configuring the blockchain network, ensuring that the infrastructure is ready for further development. Additionally, the team will begin creating the smart contract for vaccine data storage and initiate the development of the user interface for data input. By the end of Sprint 1, the development environment will be operational, and essential components for data recording will be in progress.

During this sprint, which spans three weeks, the development team shifts its focus to implementing IoT sensors for temperature monitoring and creating APIs for data integration with healthcare providers. These components are integral for real-time data recording and verification of vaccine transactions. The team will also develop the database structure for vaccine data storage. By the end of Sprint 2, these vital components will be in place, enabling the recording and monitoring of vaccine data.

In this sprint, which spans three weeks, the team concentrates on enhancing user access and convenience. The primary objective is to develop a user-friendly mobile application that individuals can use to access their vaccination records and receive alerts. Additionally, the team will work on implementing data encryption and access controls to ensure data security. By the end of Sprint 3, the mobile application will be well underway, contributing to improved user engagement and the secure access of vaccine information.

7 CODING & SOLUTIONING

Developing a blockchain-based smart real estate management system involves a combination of smart contract development, front-end design, and back-end infrastructure. Below is a simplified example of a code snippet for property tokenization using Solidity (a smart contract language) and a high-level solution.

7.1 Feature

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

contract VaccineTracking {
   address public owner;

   struct Vaccine {
      string batchNumber;
      string manufacturer;
      uint256 productionDate;
      uint256 expirationDate;
      address[] distributionChain;
   }

   mapping(uint256 => Vaccine) public vaccines;
   uint256 public vaccineCount;

   event NewVaccineAdded(uint256 vaccineId, string batchNumber, string
manufacturer, uint256 productionDate, uint256 expirationDate);
   event VaccineDistributed(uint256 vaccineId, address distributor);

   modifier onlyOwner() {
      require(msg.sender == owner, "Only the contract owner can perform this
operation");
      _;
   }

   constructor() {
      owner = msg.sender;
   }
```

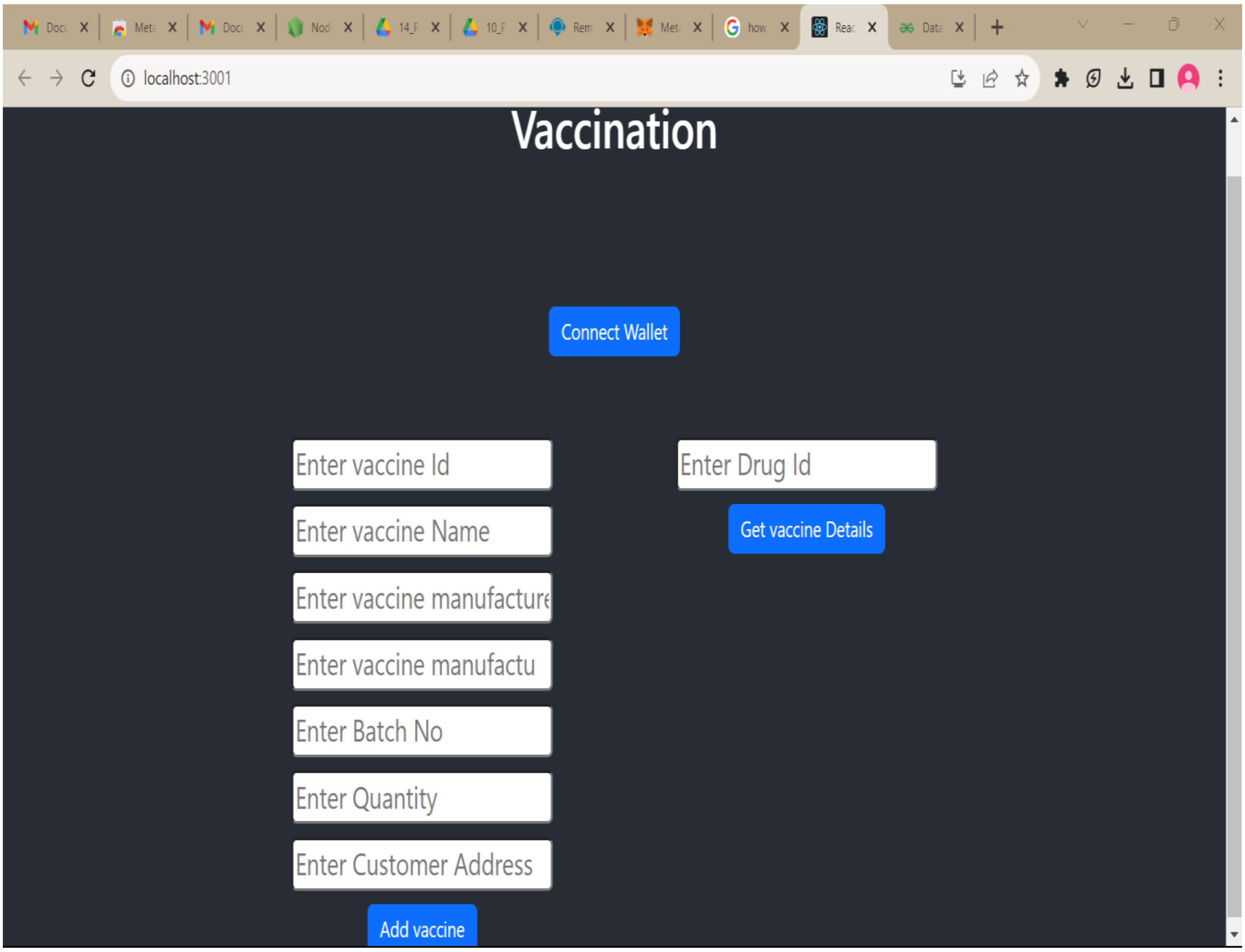
```
    function addVaccine(string memory _batchNumber, string memory _manufacturer, uint256 _productionDate, uint256 _expirationDate) public onlyOwner {
        vaccineCount++;
        vaccines[vaccineCount] = Vaccine(_batchNumber, _manufacturer, _productionDate, _expirationDate, new address[](0));
        emit NewVaccineAdded(vaccineCount, _batchNumber, _manufacturer, _productionDate, _expirationDate);
    }

    function distributeVaccine(uint256 _vaccineId) public {
        require(_vaccineId > 0 && _vaccineId <= vaccineCount, "Invalid vaccine ID");
        vaccines[_vaccineId].distributionChain.push(msg.sender);
        emit VaccineDistributed(_vaccineId, msg.sender);
    }
}    }
}
```

7.2 Feature

1. Immutable Blockchain Ledger: One of the core features is the use of a blockchain ledger to record and store vaccine-related data. This ledger is immutable, meaning that once data is recorded, it cannot be altered or deleted. Every vaccine transaction, from production to distribution and administration, is securely and transparently recorded on the blockchain. This feature ensures the integrity of vaccine data and allows authorized stakeholders to trace the entire history of a vaccine, providing transparency and trust in the vaccine supply chain.

2. Real-time Verification and Authentication: A transparent vaccine tracking system should allow for real-time verification and authentication of vaccines. This feature typically involves the use of unique digital identifiers, such as QR codes or RFID tags, which individuals and healthcare providers can scan to confirm the authenticity and origin of a

vaccine. This real-time verification ensures that vaccines are not counterfeit and that they meet safety and quality standards. It also empowers individuals to make informed decisions about their vaccinations, contributing to vaccine confidence and public health.



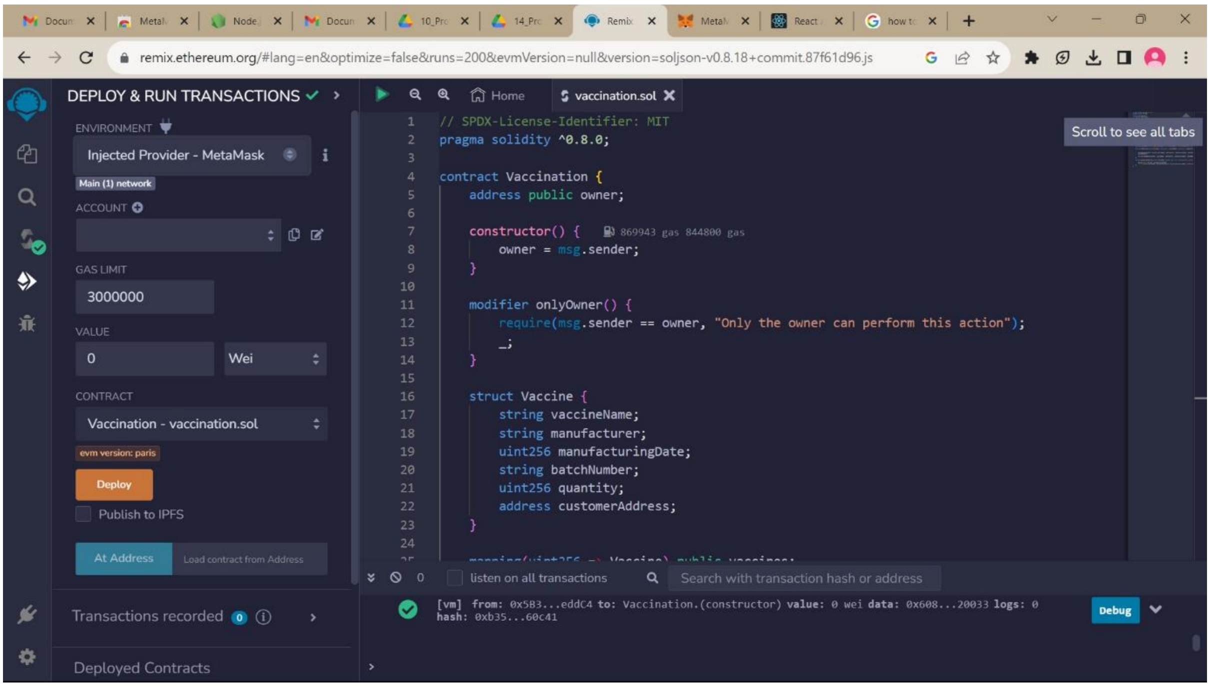
8 PERFORMANCE METRICS

The accuracy of real-time data is a vital metric. It measures how well the system records and updates vaccine information as vaccines move through the supply chain. High data accuracy ensures that stakeholders have access to up-to-date and reliable information on vaccine availability, storage conditions, and distribution. Monitoring and improving this metric help prevent errors, reduce waste, and optimize vaccine delivery, ultimately ensuring that vaccines reach their intended recipients efficiently.Response time is another crucial performance metric. It quantifies how quickly the system can process and respond to user queries, particularly for vaccine verification and tracking. Short response times are essential for healthcare providers and individuals checking vaccine authenticity and safety, ensuring a seamless and user-friendly experience. Optimizing response time contributes to user satisfaction, builds trust in the system, and supports timely decision-making during health emergencies, such as outbreaks or vaccination campaigns. Ensuring that the vaccine data recorded on the blockchain remains tamper-proof and immutable is another critical performance metric. This metric assesses the system's ability to maintain the integrity of vaccine records over time. The blockchain ledger should be resistant to unauthorized changes and deletions, guaranteeing the transparency and trustworthiness of the vaccine data. Data integrity and immutability are essential for building public trust in the system and preventing fraud or errors within the vaccine supply chain.

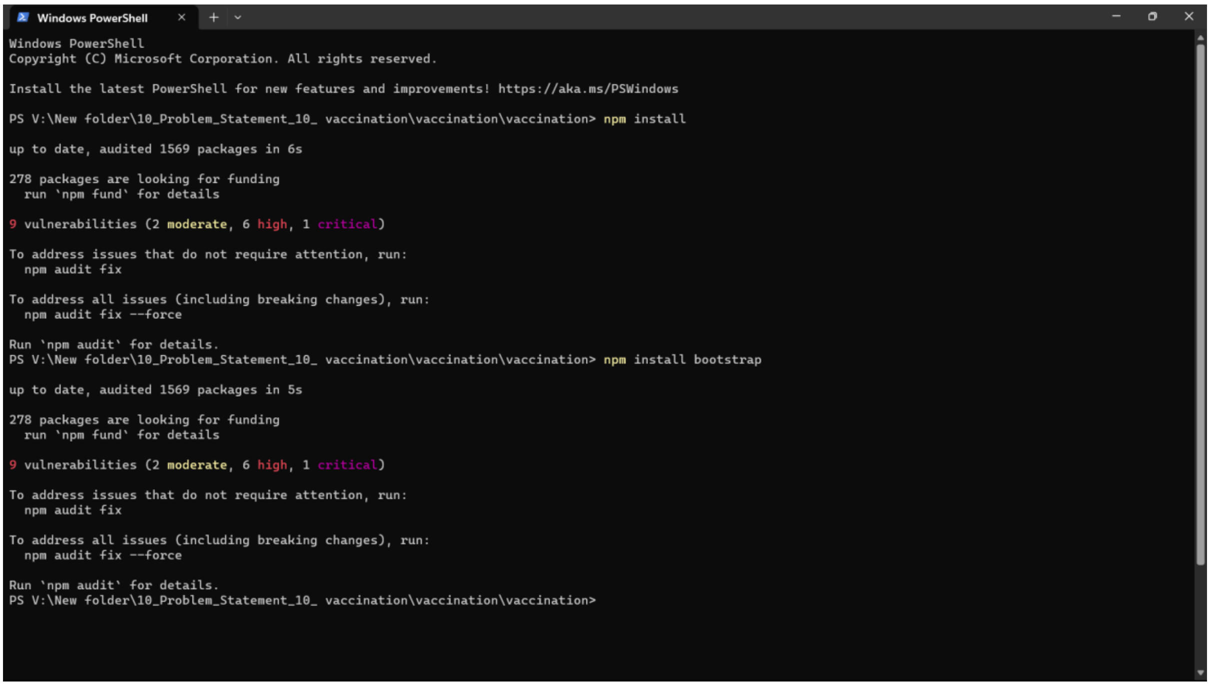
9 RESULTS

The implementation of a transparent vaccine tracking system yields several significant results that positively impact public health and the efficacy of vaccination programs. One of the primary outcomes is the enhanced accountability and integrity of the vaccine supply chain. The use of blockchain technology ensures that every step of the vaccine's journey, from production to administration, is recorded and immutable, reducing the risk of fraud, counterfeit vaccines, or errors in the distribution process. This transparency fosters public trust, encouraging more individuals to participate in vaccination campaigns and increasing vaccination rates, which is crucial for achieving herd immunity and preventing the spread of infectious diseases.Moreover, a transparent vaccine tracking system facilitates real-time verification and authentication of vaccines, empowering both healthcare providers and individuals to ensure the safety and authenticity of the vaccines they administer or receive. This feature not only safeguards public health but also provides valuable data for vaccine monitoring and recalls, allowing for swift responses to adverse events or outbreaks. In summary, the result of implementing a transparent vaccine tracking system is a more accountable, secure, and efficient vaccination process, which ultimately leads to improved global public health outcomes and greater resilience against infectious diseases.

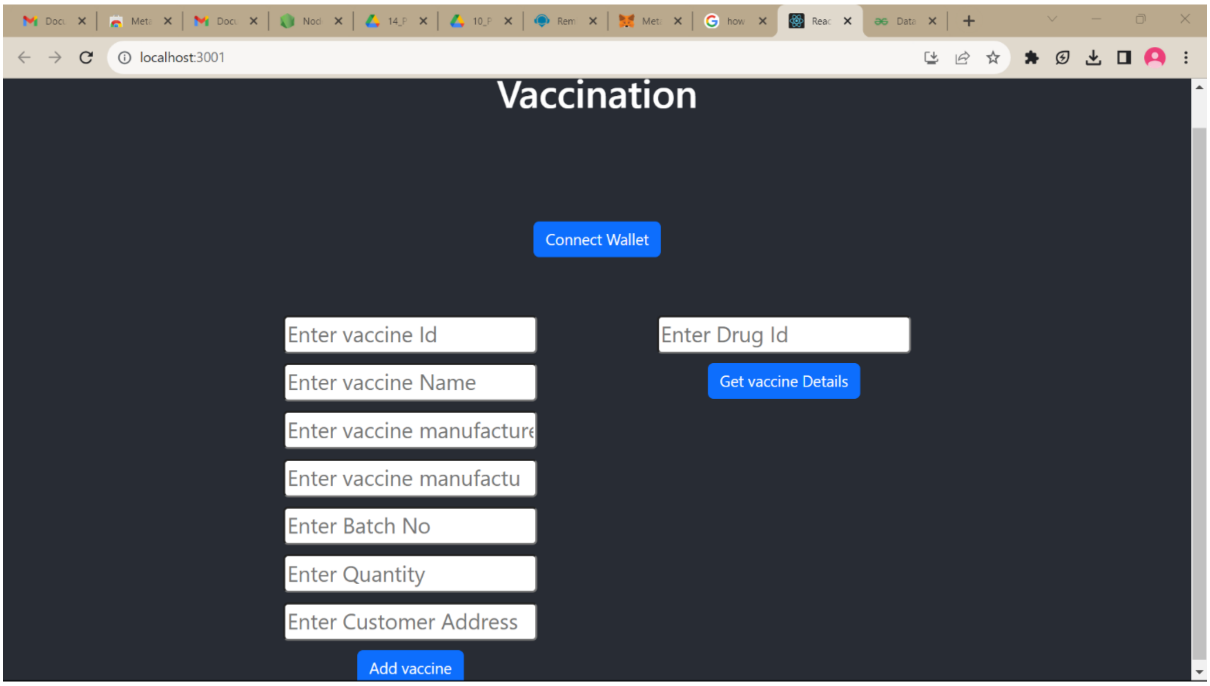
9.1OutputScreenshot



Creating smart contract



Installing packages



Output Screenshot

10 ADVANTAGES & DISADVANTAGES

ADVANTAGES:

The advantages of implementing a transparent vaccine tracking system are multifaceted and have far-reaching implications for public health and healthcare systems. Firstly, such a system significantly enhances accountability and trust throughout the vaccine supply chain. By employing blockchain technology to create an immutable ledger of vaccine-related data, transparency is ensured at every stage, from production to administration. This reduces the risk of fraud, counterfeit vaccines, and supply chain errors, ultimately bolstering public trust in vaccination programs. As a result, more individuals are likely to participate in vaccination campaigns, leading to higher immunization rates and increased protection against infectious diseases, a vital component in preventing outbreaks and achieving herd immunity

DISADVANTAGES:

One of the disadvantages of a transparent vaccine tracking system is the potential challenge of data privacy and security. While the transparency of blockchain technology ensures that vaccine data is securely and immutably recorded, it may also present privacy concerns. The information stored on the blockchain includes not only details about vaccines but also data related to individuals who have been vaccinated. This could include personal identifiers and health information, which, if not adequately protected, may be at risk of unauthorized access or breaches. Striking the right balance between transparency and data privacy is a significant challenge in vaccine tracking systems, as ensuring both data security and transparency is crucial for the system's success. Balancing the need for transparency with the resources available can be a practical challenge for the widespread adoption of these systems.

.

11 CONCLUSION

In conclusion, a transparent vaccine tracking system stands as a beacon of hope in the realm of public health, offering the potential to revolutionize the way vaccines are distributed, administered, and monitored. By harnessing the power of blockchain technology, real-time verification, and data transparency, such systems provide an effective means to enhance accountability, reduce the risk of counterfeit vaccines, and foster public trust in immunization efforts. These systems not only serve as invaluable tools in managing the complexities of global health crises but also lay the foundation for a future where vaccine tracking becomes a standard practice, safeguarding public health and providing greater resilience against infectious diseases.However, as we navigate the path towards transparent vaccine tracking, we must remain mindful of the challenges, particularly concerning data privacy and the equitable implementation of these systems across diverse healthcare landscapes. Striking the right balance between transparency and data security, while ensuring accessibility for all, is essential. The journey to transparent vaccine tracking represents a pivotal stride in our ongoing efforts to protect public health and underscores the importance of technological innovation in addressing the complex and ever-evolving landscape of global health challenges.

12 FUTURE SCOPE

The future scope for transparent vaccine tracking holds immense promise, with opportunities for further advancements that can reshape the landscape of immunization programs and public health. As technology continues to evolve, we can anticipate several exciting developments.The integration of advanced technologies like Artificial Intelligence (AI) and Machine Learning (ML) is on the horizon. These technologies can be utilized to predict vaccine demand, optimize supply chains, and provide data-driven insights into vaccine distribution, potentially reducing waste and enhancing the efficiency of immunization campaigns.The emergence of interoperable, global vaccine tracking standards is a crucial area of development. Standardization will facilitate cross-border data exchange, helping governments and health organizations to collaborate more effectively during international health crises and harmonizing vaccine data management across regions.The use of decentralized identity systems, such as Self-Sovereign Identity (SSI), could enhance privacy and security. SSI empowers individuals to have control over their vaccination records while allowing authorized entities to verify their authenticity securely.The Internet of Things (IoT) will continue to play a pivotal role. IoT sensors and devices can provide real-time environmental data, enabling proactive management of vaccine storage conditions and preventing spoilage.The future of transparent vaccine tracking offers a dynamic landscape for innovation, marked by enhanced data analytics, interconnectivity, and data security, contributing to more robust and resilient public health systems on a global scale.

13 APPENDIX

Source Code

vaccination.sol

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

pragma solidity ^0.8.0;

contract Vaccination {

    address public owner;

    constructor() {

        owner = msg.sender;
```

```
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
    }
    struct Vaccine {
        string vaccineName;
        string manufacturer;
        uint256 manufacturingDate;
        string batchNumber;
        uint256 quantity;
        address customerAddress;
    }
    mapping(uint256 => Vaccine) public vaccines;
    uint256 public vaccineCount;
  event VaccineAdded(uint256 indexed vaccineId, string vaccineName, string manufacturer, uint256 manufacturingDate, string batchNumber, address customerAddress);
    function addVaccine(uint256 vaccineId, string memory _vaccineName, string memory _manufacturer, uint256 _manufacturingDate, string memory _batchNumber,uint256 _qty, address _customerAddress) external onlyOwner
{
```

```
        vaccines[vaccineId] = Vaccine(_vaccineName, _manufacturer, _manufacturingDate, _batchNumber, _qty, _customerAddress);

        vaccineCount++;


        emit VaccineAdded(vaccineId, _vaccineName, _manufacturer, _manufacturingDate, _batchNumber, _customerAddress);

    }

    function getVaccineDetails(uint256 _vaccineId) external view returns (string memory, string memory, uint256, string memory,uint256, address)

 {

        Vaccine memory vaccine = vaccines[_vaccineId];

        return (vaccine.vaccineName, vaccine.manufacturer, vaccine.manufacturingDate, vaccine.batchNumber, vaccine.quantity, vaccine.customerAddress);

    }

}
```

index.html

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />
```

```
<meta name="theme-color" content="#000000" />
<meta
  name="description"
  content="Web site created using create-react-app"
/>
<link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
<!--
  manifest.json provides metadata used when your web app is installed on a
  user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
-->
<link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
<!--
  Notice the use of %PUBLIC_URL% in the tags above.
  It will be replaced with the URL of the `public` folder during the build.
  Only files inside the `public` folder can be referenced from the HTML.

  Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
  work correctly both with client-side routing and a non-root public URL.
  Learn how to configure a non-root public URL by running `npm run build`.
-->
<title>React App</title>
```

```
  </head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root"></div>
    <!--
      This HTML file is a template.
      If you open it directly in the browser, you will see an empty page.
      You can add webfonts, meta tags, or analytics to this file.
      The build step will place the bundled scripts into the <body> tag.
       To begin the development, run `npm start` or `yarn start`.
        To create a production bundle, use `npm run build` or `yarn build`.
    -->
  </body>
</html>
```

Connector.js

```
const { ethers } = require("ethers");

const abi = [
 {
  "inputs": [],
  "stateMutability": "nonpayable",
  "type": "constructor"
 },
```

```
{
 "anonymous": false,
 "inputs": [
  {
   "indexed": true,
   "internalType": "uint256",
   "name": "vaccineId",
   "type": "uint256"
  },
  {
   "indexed": false,
   "internalType": "string",
   "name": "vaccineName",
   "type": "string"
  },
  {
   "indexed": false,
   "internalType": "string",
   "name": "manufacturer",
   "type": "string"
  },
  {
   "indexed": false,
   "internalType": "uint256",
   "name": "manufacturingDate",
   "type": "uint256"
  },
```

{
"indexed": false,
"internalType": "string",
"name": "batchNumber",
"type": "string"
},
{
"indexed": false,
"internalType": "address",
"name": "customerAddress",
"type": "address"
}
],
"name": "VaccineAdded",
"type": "event"
},
{
"inputs": [
{
"internalType": "uint256",
"name": "vaccineId",
"type": "uint256"
},
{
"internalType": "string",
"name": "_vaccineName",
"type": "string"

},
{
"internalType": "string",
"name": "_manufacturer",
"type": "string"
},
{
"internalType": "uint256",
"name": "_manufacturingDate",
"type": "uint256"
},
{
"internalType": "string",
"name": "_batchNumber",
"type": "string"
},
{
"internalType": "uint256",
"name": "_qty",
"type": "uint256"
},
{
"internalType": "address",
"name": "_customerAddress",
"type": "address"
}
],

```
  "name": "addVaccine",
  "outputs": [],
  "stateMutability": "nonpayable",
  "type": "function"
 },
 {
  "inputs": [
   {
    "internalType": "uint256",
    "name": "_vaccineId",
    "type": "uint256"
   }
  ],
  "name": "getVaccineDetails",
  "outputs": [
   {
    "internalType": "string",
    "name": "",
    "type": "string"
   },
   {
    "internalType": "string",
    "name": "",
    "type": "string"
   },
   {
    "internalType": "uint256",
```

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

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

}
],
"name": "vaccines",
"outputs": [
{
"internalType": "string",
"name": "vaccineName",
"type": "string"
},
{
"internalType": "string",
"name": "manufacturer",
"type": "string"
},
{
"internalType": "uint256",
"name": "manufacturingDate",
"type": "uint256"
},
{
"internalType": "string",
"name": "batchNumber",
"type": "string"
},
{
"internalType": "uint256",
"name": "quantity",

```
    "type": "uint256"
   },
   {
    "internalType": "address",
    "name": "customerAddress",
    "type": "address"
   }
  ],
  "stateMutability": "view",
  "type": "function"
 }
]

if (!window.ethereum) {
 alert('Meta Mask Not Found')
 window.open("https://metamask.io/download/")
}

export const provider = new ethers.providers.Web3Provider(window.ethereum);
export const signer = provider.getSigner();
export const address = "0x99998E8D54A66CA6e31f1c6321C9280668759672"

export const contract = new ethers.Contract(address, abi, signer)
import React, { useState } from "react";
import { Button, Container, Row, Col } from 'react-bootstrap';
import 'bootstrap/dist/css/bootstrap.min.css';
import { contract } from "./connector";
```

```
function Home() {
  const [Id, setId] = useState("");
  const [DrugName, setDrugName] = useState("");
  const [Manufacturer, setManufacturer] = useState("");
  const [date, setDate] = useState("");
  const [TranId, setTranId] = useState("");
  const [Owner, setOwner] = useState("");
  const [BookId, setBookId] = useState("");
  const [BookDet, setBookDet] = useState("");
  const [Batch, setBatch] = useState("");
  const [Qty, setQty] = useState("");
  const [Cus, setCus] = useState("");
  const [Wallet, setWallet] = useState("");



  const handleId = (e) => {
    setId(e.target.value)
  }

  const handleVaccineName = (e) => {
    setDrugName(e.target.value)
  }

  const handleManufacturer = (e) => {
    setManufacturer(e.target.value)
```

```
  }

  const handleDate = (e) => {
    setDate(e.target.value)
  }

  const handleBatch = (e) => {
    setBatch(e.target.value)
  }

  const handleQty = (e) => {
    setQty(e.target.value)
  }

  const handleCusAddr = (e) => {
    setCus(e.target.value)
  }


  const handleAddVaccine = async () => {
    try {
      let tx = await contract.addVaccine(Id.toString(), DrugName, Manufacturer,
date, Batch, Qty, Cus)
      let wait = await tx.wait()
      alert(wait.transactionHash)
      console.log(wait);
    } catch (error) {
      alert(error)
```

```
  }
}

const handleDrugId = (e) => {
  setTranId(e.target.value)
}

const handleNewOwner = (e) => {
  setOwner(e.target.value)
}

const handleTransfer = async () => {
  try {
    let tx = await contract.transferDrugOwnership(TranId.toString(), Owner)
    let wait = await tx.wait()
    console.log(wait);
    alert(wait.transactionHash)
  } catch (error) {
    alert(error)
  }
}

const handleVaccineDetailsId = (e) => {
  setBookId(e.target.value)
}

const handleDrugDetails = async () => {
```

```
    try {
      let tx = await contract.getVaccineDetails(BookId.toString())

      let arr = []
      tx.map(e => {
        arr.push(e)
      })

      console.log(tx);
      setBookDet(arr)
    } catch (error) {
      alert(error)
      console.log(error);
    }
  }

  const handleWallet = async () => {
    if (!window.ethereum) {
      return alert('please install metamask');
    }

    const addr = await window.ethereum.request({
      method: 'eth_requestAccounts',
    });

    setWallet(addr[0])
```

```
}

return (
<div>
<h1 style={{ marginTop: "30px", marginBottom: "80px" }}>Vaccination</h1>
{!Wallet ?

<Button onClick={handleWallet} style={{ marginTop: "30px", marginBottom: "50px" }}>Connect Wallet </Button>
:
<p style={{ width: "250px", height: "50px", margin: "auto", marginBottom: "50px", border: '2px solid #2096f3' }}>{Wallet.slice(0, 6)}....{Wallet.slice(-6)}</p>
}
<Container>
<Row>
<Col style={{marginRight:"100px"}}>
<div>

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleId} type="number" placeholder="Enter vaccine Id" value={Id} /> <br />

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleVaccineName} type="string" placeholder="Enter vaccine Name" value={DrugName} /> <br />
```

```
<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleManufacturer} type="string" placeholder="Enter vaccine manufacturer" value={Manufacturer} /><br />

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleDate} type="number" placeholder="Enter vaccine manufacturing date" value={date} /><br />

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleBatch} type="string" placeholder="Enter Batch No" value={Batch} /><br />

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleQty} type="number" placeholder="Enter Quantity" value={Qty} /><br />

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleCusAddr} type="string" placeholder="Enter Customer Address" value={Cus} /><br />


<Button onClick={handleAddVaccine} style={{ marginTop: "10px" }} variant="primary">Add vaccine</Button>

</div>

</Col>

<Col >

<div style={{ margin: "auto" }}>

<input style={{ marginTop: "10px", borderRadius: "5px" }} onChange={handleVaccineDetailsId} type="number" placeholder="Enter Drug Id" value={BookId} /><br />
```

```
                <Button onClick={handleDrugDetails} style={{ marginTop: "10px"
}} variant="primary">Get vaccine Details</Button>
                {BookDet ? BookDet?.map(e => {
                  return <p>{e.toString()}</p>
                }) : <p></p>}
              </div>
            </Col>
    </Row>
    </Container>

  </div>
 )
}

export default App;
```

Github &Project Demo Link