VACCINE TRACKING-TRANSPARENT

PROJECT DOCUMENTATION

Submitted by

Team ID:	NM2023TMID03802
Team Leader:	VINOTHINI S
Team Member:	VASANTHAPRIYA S
Team Member:	UDHAYA K
Team Member:	SHIYAMALDEVI S

INDEX

1. INTRODUCTION

1.1Project Overview:

Objective: The primary goal of a vaccine tracking project is to enhance public health outcomes by ensuring widespread and equitable access to vaccines, monitoring vaccination coverage rates, and assessing the impact on disease prevention.

Components: Data Collection: Gathering and recording data related to vaccines, including information about vaccine types, batch numbers, expiration dates, recipients' demographics, and administration dates.

Inventory Management: Tracking vaccine stock levels, storage conditions, and distribution logistics to prevent wastage and ensure a steady supply chain.

Monitoring Adverse Events: Recording and investigating any adverse events or side effects reported after vaccination.

Reporting and Analytics: Generating reports and analyzing data to track progress, identify trends, and make informed decisions regarding vaccination strategies.

Communication and Outreach: Engaging with healthcare providers, the public, and stakeholders to disseminate information about vaccination schedules, safety, and benefits.

Key Stakeholders: Public Health Authorities: Government agencies responsible for overseeing vaccination programs, setting policies, and providing guidance to healthcare providers.

Healthcare Providers: Including hospitals, clinics, pharmacies, and other facilities responsible for administering vaccines.

Pharmaceutical Companies: Manufacturers and distributors of vaccines play a critical role in the supply chain. Technology Providers: Companies or organizations providing software, tools, or platforms for data management and analytics.

Technological Infrastructure: Database and Information Systems: Utilizing databases and information systems to store, organize, and retrieve vaccine-related data securely.

Integration with Healthcare Systems: Ensuring compatibility with existing electronic health record (EHR) systems and **health information exchanges.**

Mobile Applications or Portals: Providing user-friendly interfaces for data entry, reporting, and access to vaccination information.

Data Privacy and Security: Implementing robust security measures to protect sensitive information, ensuring compliance with data protection regulations, and safeguarding against unauthorized access or breaches.

Monitoring and Evaluation: Regularly assessing the project's performance against predefined metrics, such as vaccination coverage rates, timeliness of reporting, and adverse event monitoring.

1.2Project Purpose:

Monitoring Vaccine Distribution: Tracking projects help in keeping a record of where vaccines have been distributed. This is crucial for ensuring that vaccines are reaching the communities that need them most, and that they are being distributed equitably.

Ensuring Adequate Supply and Demand Management: By tracking the distribution and administration of vaccines, authorities can identify areas where there may be a surplus or shortage of vaccines. This helps in managing the supply chain effectively.

Optimizing Vaccine Administration: It helps in streamlining the vaccination process. By tracking which types of vaccines are available at which locations, healthcare providers can efficiently schedule appointments and avoid waste. Monitoring

Adverse Reactions: Tracking projects also collect data on any adverse reactions to vaccines. This information is essential for monitoring vaccine safety and making any necessary adjustments to vaccination protocols.

Tracking Immunization Coverage: It helps in keeping tabs on the percentage of the population that has been vaccinated. This is crucial for achieving herd immunity, which is a significant milestone in controlling the spread of infectious diseases.

Identifying High-Risk Groups: Through tracking, it's possible to identify specific demographics or geographic areas that may be under-vaccinated. This allows for targeted outreach efforts to ensure that everyone has access to vaccines.

Research and Analysis: The data collected from vaccine tracking projects can be used for research purposes. This can include studies on vaccine efficacy, the long-term effects of vaccination, and the effectiveness of different vaccination strategies.

Emergency Response Planning: In the event of an outbreak or resurgence of a disease, having accurate and up-to-date vaccine tracking data is critical for planning an effective emergency response.

Building Trust and Transparency: Transparently sharing information about vaccine distribution and administration can help build trust in the vaccination process. It allows the public to see that efforts are being made to ensure fairness and efficiency.

Compliance with Regulatory Requirements: Many regulatory bodies and health organizations require accurate and up-to-date vaccine tracking data to ensure compliance with their guidelines and protocols.

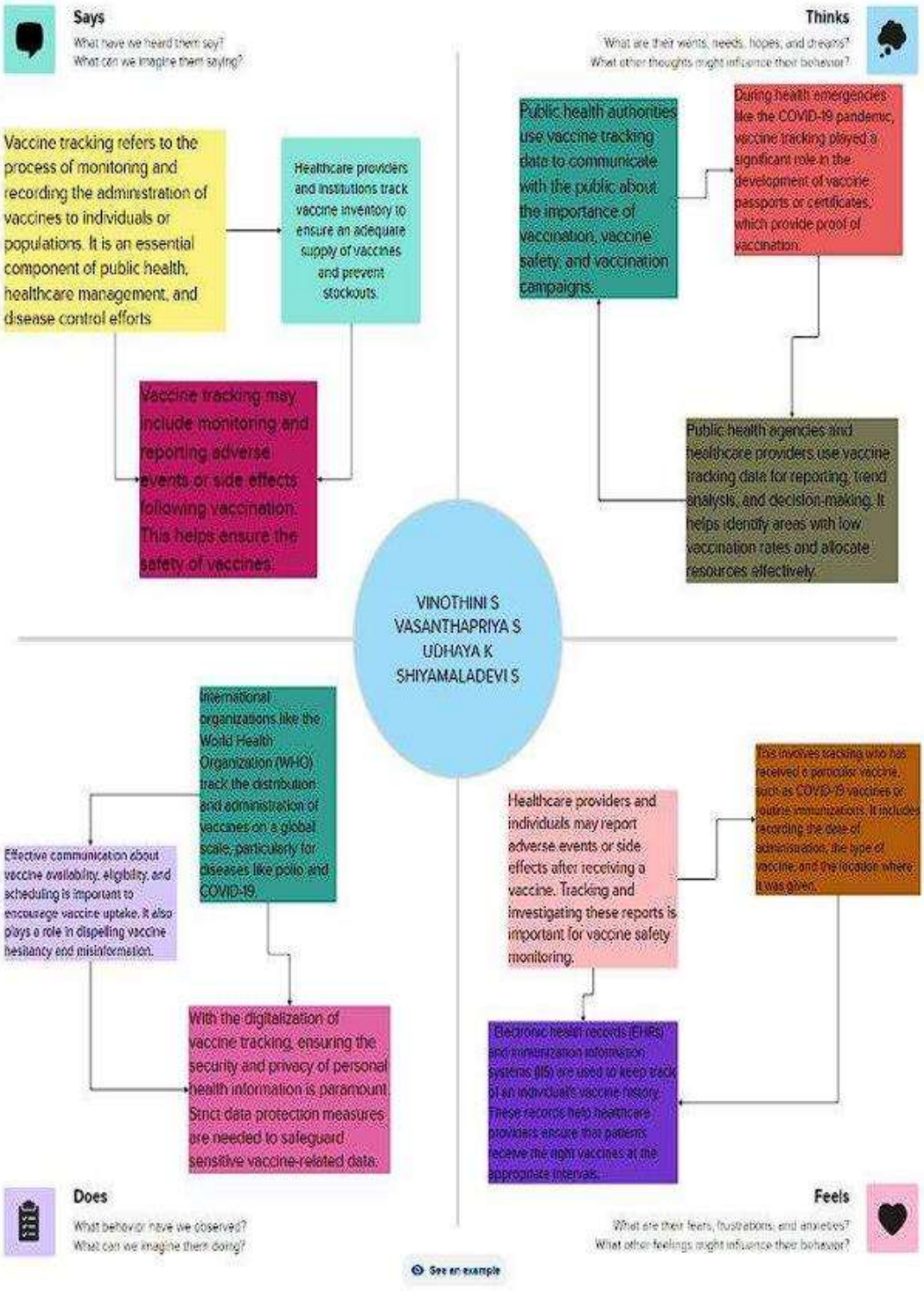
Global Health Surveillance: For international diseases, vaccine tracking projects contribute to global health surveillance efforts. This helps in understanding the global spread of diseases and enables coordinated responses.

2. IDEATION & PROPOSED SOLUTION

2.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user behavior and Attitudes.

It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the use

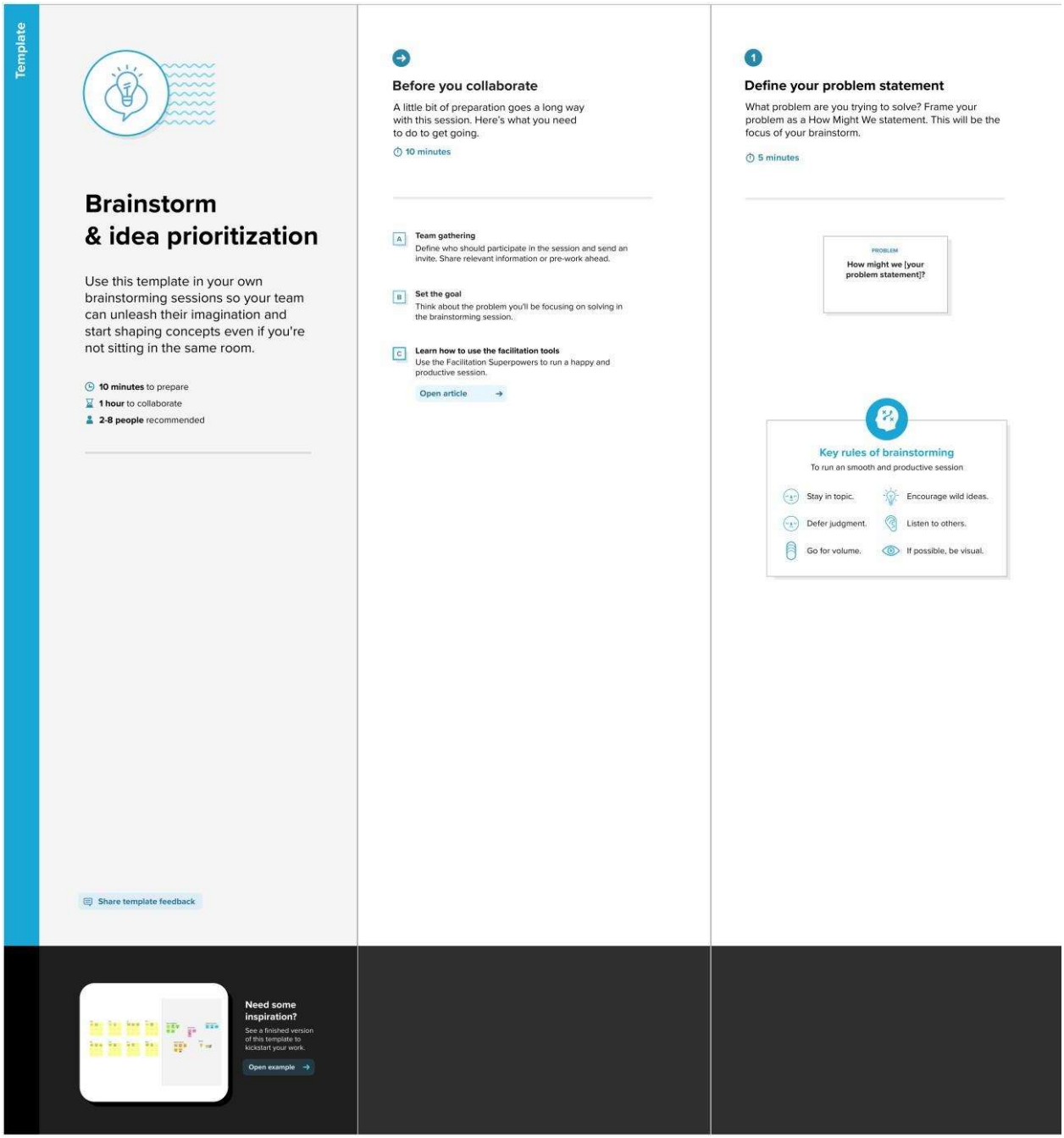

Says
What have we heard them say?
What can we imagine them saying?
Vaccine tracking refers to the process of monitoring and recording the administration of vaccines to individuals or populations. It is an essential component of public health, healthcare management, and disease control efforts
Healthcare providers and institutions track vaccine inventory to ensure an adequate supply of vaccines and prevent stockouts.
Vaccine tracking may include monitoring and reporting adverse events or side effects following vaccination. This helps ensure the safety of vaccines.
Thinks
What are their wants, needs, hopes, and dreams?
What other thoughts might influence their behavior?
Public health authorities use vaccine tracking data to communicate with the public about the importance of vaccination, vaccine safety, and vaccination campaigns.
During health emergencies like the COVID-19 pandemic, vaccine tracking played a significant role in the development of vaccine passports or certificates, which provide proof of vaccination.
Public health agencies and healthcare providers use vaccine tracking data for reporting, trend analysis, and decision-making. It helps identify areas with low vaccination rates and allocate resources effectively.
VINOTHINI S
VASANTHAPRIYA S
UDHAYA K
SHIYAMALADEVI S
International organizations like the World Health Organization (WHO) track the distribution and administration of vaccines on a global scale, particularly for diseases like polio and COVID-19.
Effective communication about vaccine availability, eligibility, and scheduling is important to encourage vaccine uptake. It also plays a role in dispelling vaccine hesitancy and misinformation.
With the digitalization of vaccine tracking, ensuring the security and privacy of personal health information is paramount. Strict data protection measures are needed to safeguard sensitive vaccine-related data.
Healthcare providers and individuals may report adverse events or side effects after receiving a vaccine. Tracking and investigating these reports is important for vaccine safety monitoring.
This involves tracking who has received a particular vaccine, such as COVID-19 vaccines or routine immunizations. It includes recording the date of administration, the type of vaccine, and the location where it was given.
Electronic health records (EHRs) and immunization information systems (IIS) are used to keep track of an individual's vaccine history. These records help healthcare providers ensure that patients receive the right vaccines at the appropriate intervals.
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

2.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team participate in the creative thinking process thatleads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

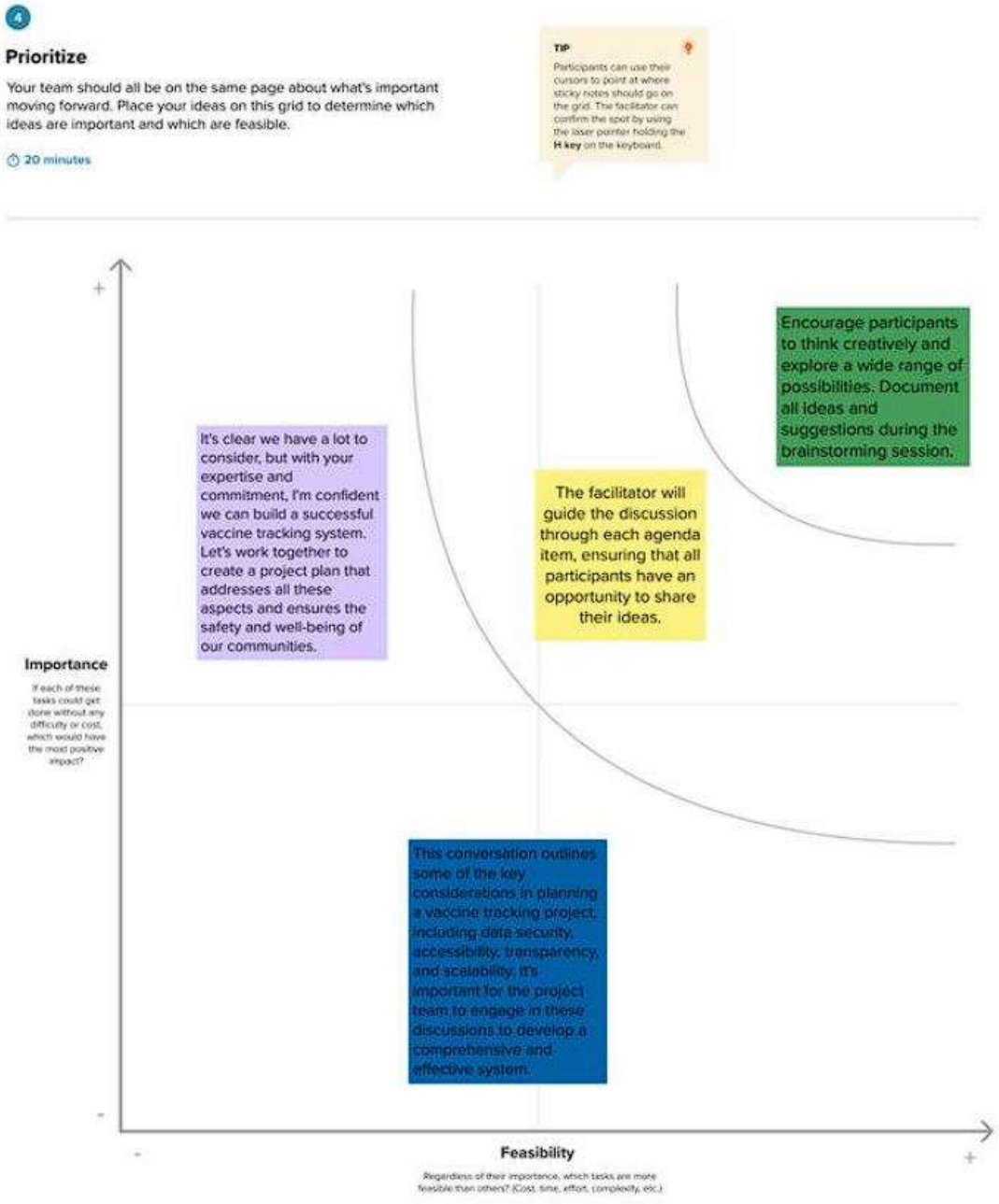
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the dame room.

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



Step-2: Brainstorm, Idea Listing and grouping

Step-3: Idea Prioritization



3. REQUIREMENT ANALYSIS

3.1Functional Requirements:

User Authentication and Authorization: Users should be able to create accounts, log in, and have appropriate access levels based on their roles (e.g., administrator, healthcare provider, general user).

Vaccine Registration: Healthcare providers should be able to register vaccines into the system, including details like name, manufacturer, batch number, expiration date, and dosage schedule.

Patient Registration: Users should be able to register patients, including personal information, medical history, and any known allergies or adverse reactions.

Vaccination Record Management: The system should record each administered dose, including date, time, location, healthcare provider, and vaccine details.

Dosage Schedule Reminder: The system should be able to send reminders to patients and healthcare providers for upcoming doses based on the recommended schedule.

Inventory Management: The system should keep track of vaccine inventory, including current stock levels, restocking alerts, and expiring doses.

Adverse Event Reporting: Users should be able to report any adverse events or side effects related to vaccination, with a mechanism for further investigation and follow-up.

Data Analytics and Reporting: Generate reports and statistics on vaccination rates, coverage, and demographic distribution. This can help in identifying trends and making informed decisions.

Integration with Health Records: The system should integrate with existing electronic health record (EHR) systems to ensure seamless data exchange and continuity of care.

3.2Non-functional Requirements:

Performance: The system should be able to handle a large number of simultaneous users and process transactions efficiently without significant delays.

Scalability: The system should be designed to scale with increasing demand, ensuring it can accommodate more users and data without a drop in performance.

Security: Data should be stored securely and encrypted to protect patient privacy. Access controls and authentication mechanisms should be in place to prevent unauthorized access.

Reliability: The system should be available and reliable 24/7 to ensure that vaccines can be administered and tracked without disruption.

Compliance: The system should comply with relevant healthcare regulations and standards to ensure legal and ethical use of patient data.

User Interface and Experience: The user interface should be intuitive, user-friendly, and accessible to a diverse range of users, including healthcare professionals and patients.

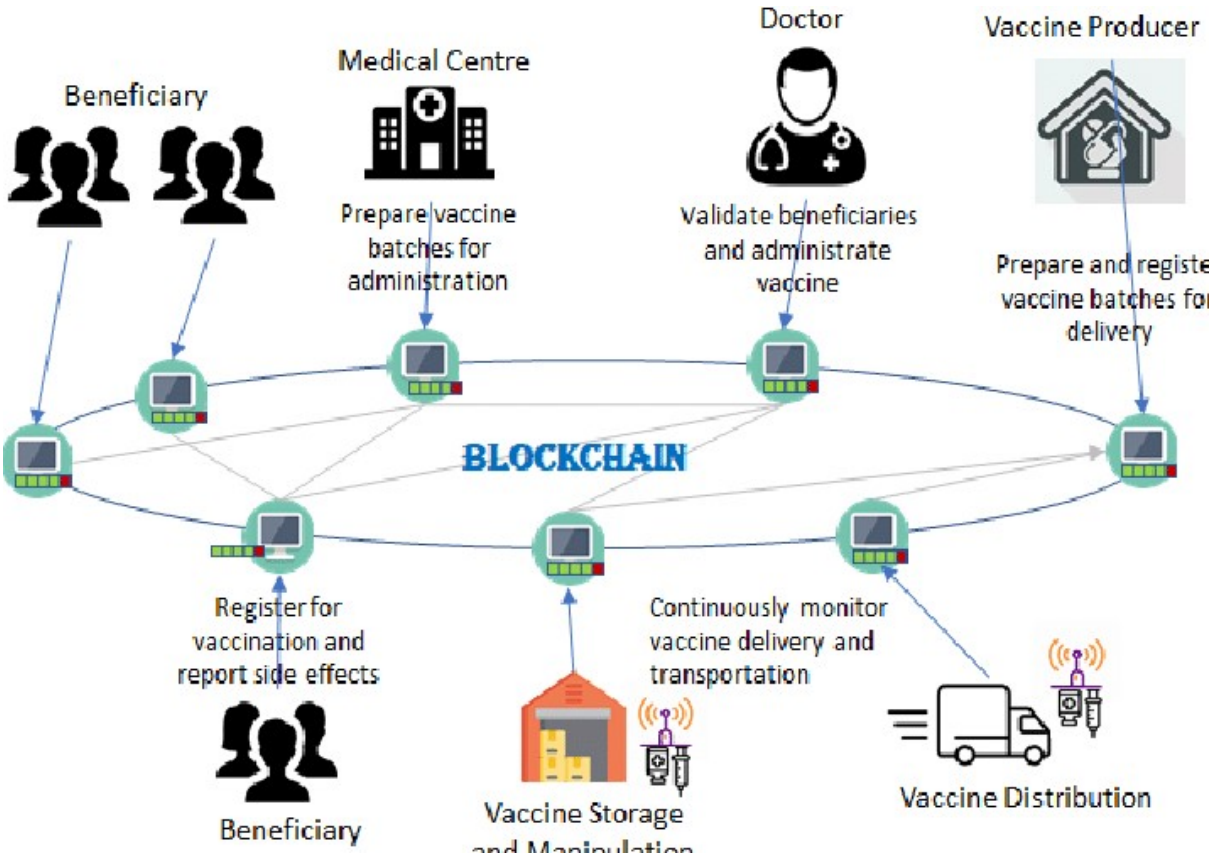
Data Backup and Recovery: Regular data backups should be performed to prevent data loss in case of system failures or unexpected events.

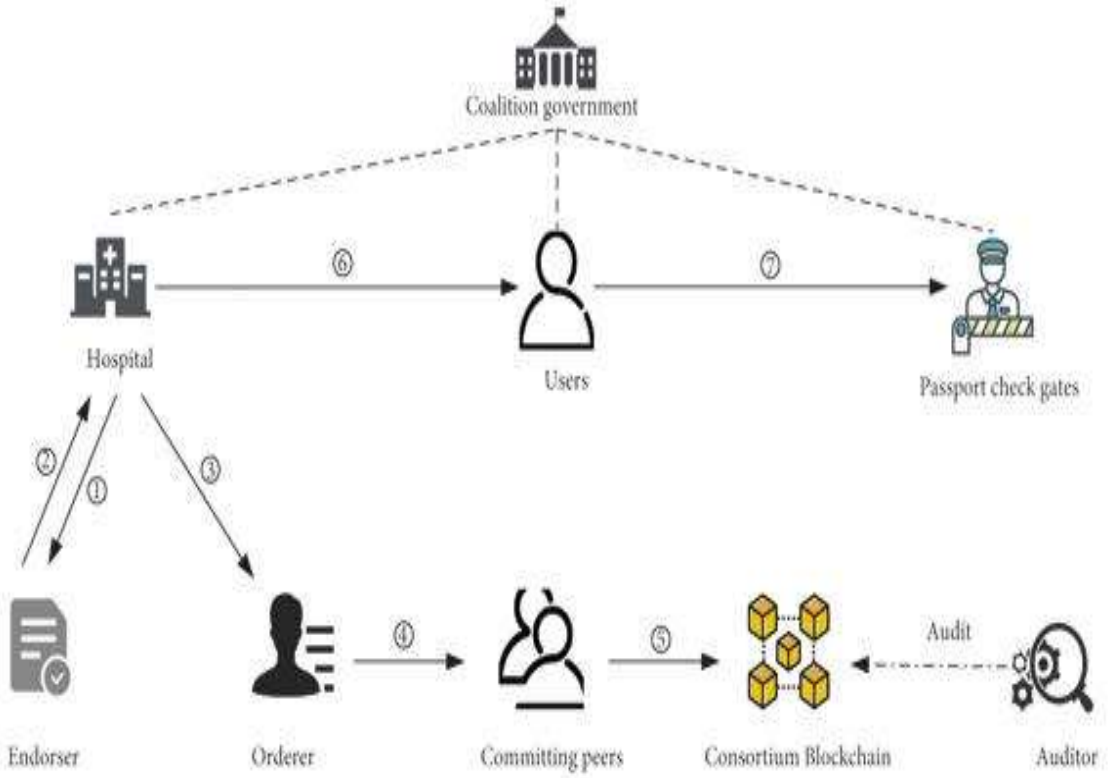
Audit Trail: The system should maintain an audit trail of all interactions, allowing for traceability and accountability.

4. PROJECT DESIGN

4.1Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of thesystem requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



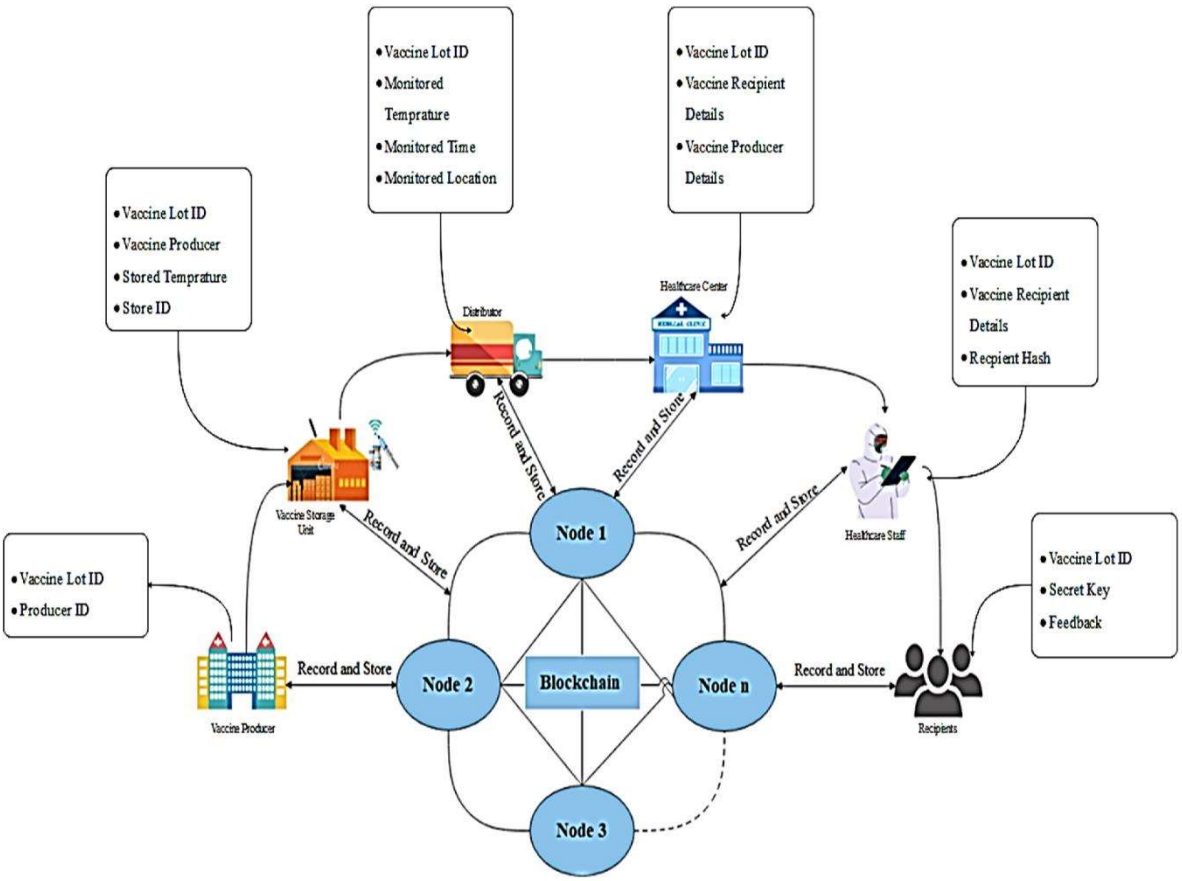

Coalition government
Hospital
⑥
Users
⑦
Passport check gates
②
①
③
Endorser
Orderer
④
Committing peers
⑤
Consortium Blockchain
Audit
Auditor

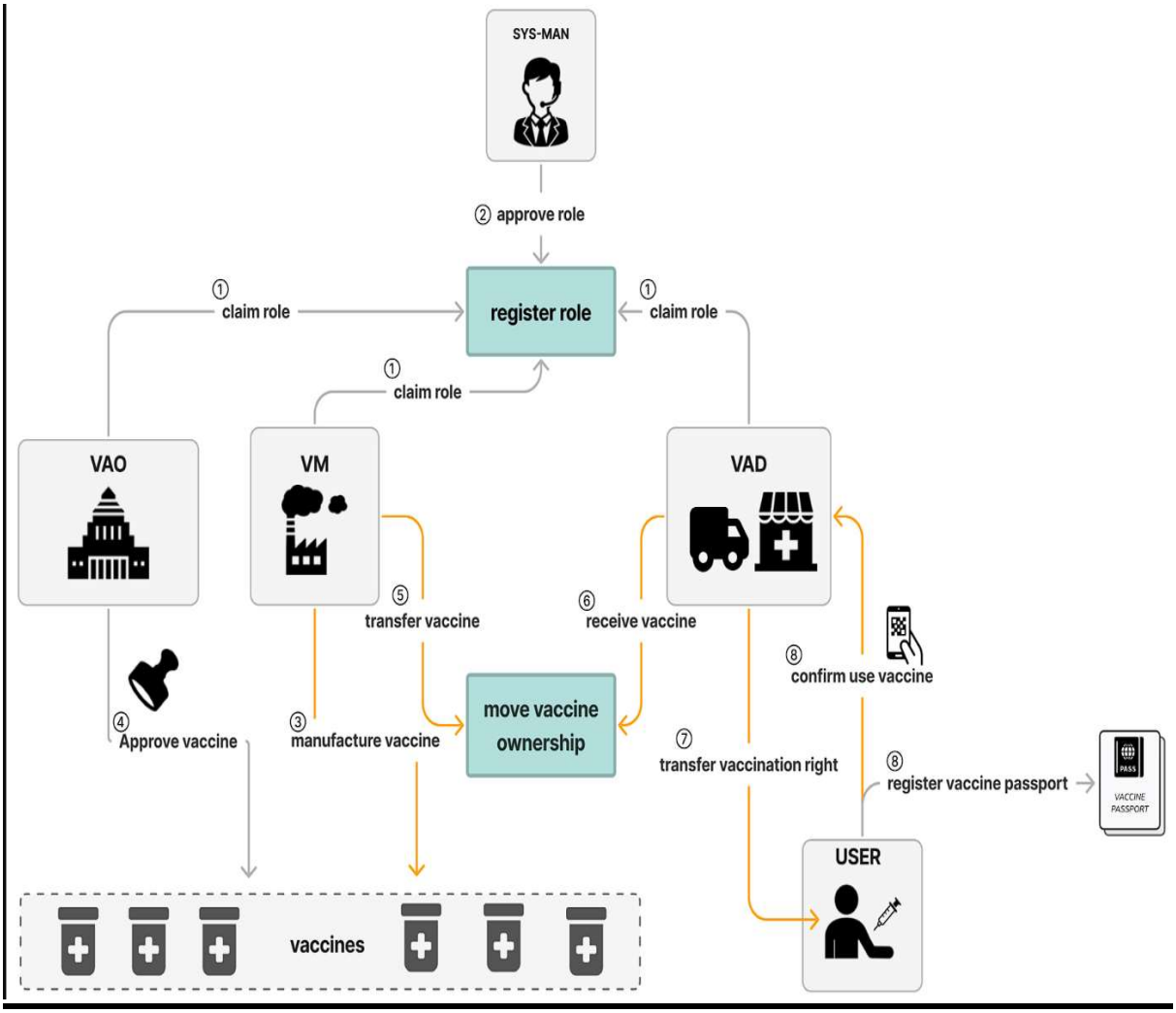
4.2Solution Architecture:

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

Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed and delivered.




SYS-MAN
② approve role
register role
① claim role
① claim role
① claim role
VAO
VM
VAD
⑤ transfer vaccine
⑥ receive vaccine
⑧ confirm use vaccine
④ Approve vaccine
③ manufacture vaccine
move vaccine ownership
⑦ transfer vaccination right
⑧ register vaccine passport
PASS
VACCINE PASSPORT
USER
vaccines

4.3User Stories:

User Registration: As a new user, I want to be able to create an account with my personal information, including name, contact details, and identification. As a healthcare provider, I want to be able to register my clinic or facility and provide necessary credentials for verification.

Vaccination Record Creation: As a healthcare provider, I want to be able to create a new vaccination record for each individual that receives a vaccine, including details like vaccine type, dosage, and date administered. As a user, I want to receive a confirmation of my vaccination record after receiving a vaccine.

Appointment Scheduling: As a user, I want to be able to schedule an appointment for my vaccination at a convenient location and time. As a healthcare provider, I want to have the ability to view and manage upcoming appointments.

Dose Tracking: As a user, I want to receive reminders for my second dose and information on which vaccine I received for the first dose. As a healthcare provider, I want to track which individuals have received one or both doses of the vaccine.

Vaccine Availability and Inventory Management: As a healthcare administrator, I want to be able to monitor the available vaccine inventory and request restocks as needed. As a user, I want to be able to check for vaccine availability at different locations.

Reporting and Analytics: As a public health official, I want to generate reports on vaccination coverage by demographics, location, and vaccine type. As a healthcare provider, I want to see statistics on the number of vaccines administered by my clinic.

Verification and Certificates: As a user, I want to be able to download a digital certificate as proof of my vaccination status. As a third-party, I want to be able to verify the authenticity of a vaccination certificate.

Adverse Event Reporting: As a user, I want to be able to report any adverse events or side effects I experience after receiving a vaccine. As a healthcare provider, I want to be able to document and report adverse events to the relevant authorities.

Multi-language Support and Accessibility: As a user, I want the system to be accessible to individuals who speak different languages or have disabilities. Privacy and Security: As a user, I want assurance that my personal health information is kept secure and confidential. As a healthcare provider, I want to ensure that only authorized personnel have access to sensitive patient data.

5.CODING & SOLUTIONING

Coding refers to the process of translating a software design into a functional program using programming languages like JavaScript, Python, or Java. It involves writing code, debugging, and optimizing algorithms to create a solution for a specific problem or requirement.

Solutioning involves designing a comprehensive solution strategy before coding. It includes problem analysis, architectural design, selecting appropriate technologies, and planning for scalability and security. Solutioning ensures that the software addresses the problem effectively and aligns with the project's goals.

5.1 FEATURE 1

Feature 1: User Authentication

```
function authenticateUser(username, password) {
// Code to validate username and password
        if (isValidCredentials(username, password)) {
                return "Authentication successful";
        }
        else {
                return "Authentication failed";
        }
}
function isValidCredentials(username, password) {
// Code to check credentials against database or backend
system
// Return true if valid, false otherwise
}
```

Explanation:

The code snippet checks the provided username and password against a database or backend system. If the credentials are valid, the function returns "Authentication successful"; otherwise, it returns "Authentication failed." This feature ensures secure user access to the system.

5.2FEATURE 2

Feature 2: Data Encryption

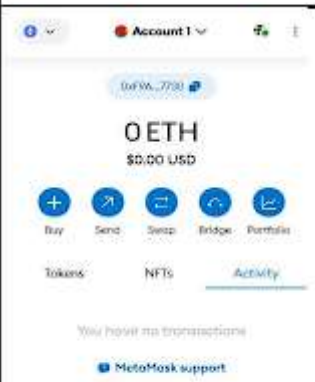
```
const crypto = require('crypto');
function encryptData(data, key) {
    const cipher = crypto.createCipher('aes-256-cbc', key);
    let encryptedData = cipher.update(data, 'utf-8', 'hex');
    encryptedData += cipher.final('hex');
    return encryptedData;
}
function decryptData(encryptedData, key) {
    const decipher = crypto.createDecipher('aes-256-cbc', key);
    let decryptedData = decipher.update(encryptedData, 'hex',
    'utf-8');
    decryptedData += decipher.final('utf-8');
    return decryptedData;
}
```

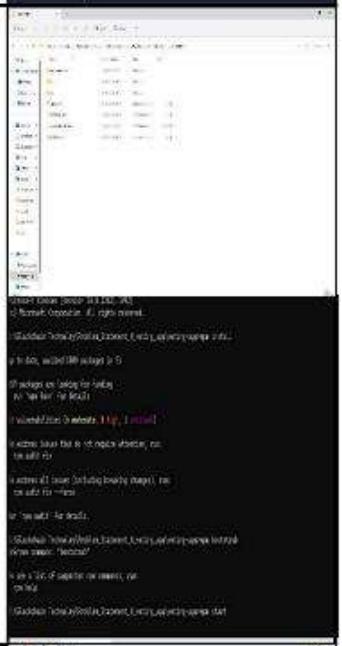
Explanation:

This code snippet demonstrates data encryption and decryption using the AES encryption algorithm. encryptData function takes data and a key, encrypts it, and returns the encrypted data in hexadecimal format. decryptData function reverses the process, decrypting the data using the same key.

6.RESULTS

6.1 Performance Metrics:

S.No.	Parameter	Values	Screenshot
1.	Information gathering	Setup all the Prerequisite:	
2.	Extract the zip files	Open to vs code	

3.	Remix Ide platform exploring	Deploy the smart contract code Deploy and run the transaction. By selecting the environment - inject the MetaMask.	
4	Open file explorer	Open the extracted file and click on the folder. Open src, and search for utiles. Open cmd enter commands 1.npm install 2. npm bootstrap 3. npm start	
5	{LOCALHOST IP ADDRESS	copy the address and open it t chrome so you can see the front en of your project.	

7. ADVANTAGES & DISADVANTAGES

Advantages:

Disease Control and Eradication: Vaccine tracking helps public health officials monitor vaccination rates and coverage. This information is crucial for controlling and, in some cases, eradicating vaccine-preventable diseases.

Timely Interventions: It allows for the identification of pockets of low vaccination coverage, enabling public health authorities to take timely action to increase vaccination rates in those areas.

Individual Health Management: For healthcare providers, it ensures that they have an accurate record of a patient's vaccination history. This helps in making informed decisions about future vaccinations and ensuring that individuals receive the appropriate vaccines at the right time.

Research and Development: Aggregated and anonymized vaccine data can be used for research purposes. It can aid in the development of new vaccines, the improvement of existing ones, and the understanding of vaccine effectiveness and safety.

Safety Monitoring: It facilitates the monitoring of vaccine safety by enabling the identification of any unusual patterns of adverse events following immunization (AEFIs).

Supply Chain Management: It helps in managing vaccine supply chains, ensuring that there are enough vaccines available to meet the demand.

Disadvantages:

Privacy Concerns: Maintaining accurate vaccine records involves storing personal health information, which raises privacy concerns. Safeguards must be in place to protect this sensitive data from unauthorized access or breaches.

Data Accuracy: There may be instances of inaccurate or incomplete vaccine records, especially when relying on manual data entry. This can lead to potential issues in public health decision-making and individual patient care.

Digital Divide: In areas with limited access to technology or healthcare infrastructure, implementing electronic vaccine tracking systems may be challenging, leading to disparities in **tracking capabilities.**

Resource Intensiveness: Establishing and maintaining a robust vaccine tracking system can require significant financial and human resources. This may be a barrier, especially for lower-income or resource-constrained regions.

Resistance and Trust Issues: Some individuals may be hesitant about sharing their vaccination data due to concerns about privacy or mistrust of healthcare systems. Building and maintaining public trust is crucial for the success of any vaccine tracking initiative.

Potential for Stigmatization: In some cases, vaccine tracking may inadvertently lead to the stigmatization of individuals who are not fully vaccinated, potentially causing social divisions.

8. CONCLUSION

Effective Disease Control: Vaccine tracking plays a pivotal role in controlling the spread of diseases. It enables health authorities to identify and respond to outbreaks quickly, ensuring that appropriate measures are taken to protect public health.

Improved Immunization Rates: By monitoring vaccine distribution and administration, health agencies can identify underserved populations and implement targeted interventions to improve immunization rates, reducing the overall burden of vaccine-preventable diseases.

Data-Driven Decision Making: The data collected through vaccine tracking provides valuable insights for policymakers and healthcare professionals. It helps in making informed decisions regarding vaccine allocation, distribution strategies, and public health campaigns.

Safety Monitoring: Continuous tracking allows for real-time monitoring of vaccine safety. Adverse events can be promptly identified and investigated, leading to timely interventions to ensure the safety of the population.

Boosting Confidence: Transparent and robust vaccine tracking systems can help build public trust in vaccination programs. It provides transparency regarding the safety and efficacy of vaccines, addressing concerns and misinformation.

9. FUTURE SCOPE

Integration of Digital Technologies: Future vaccine tracking systems will likely leverage advanced technologies such as blockchain, artificial intelligence, and Internet of Things (IoT) for secure and efficient data management. This will enhance traceability and reduce the risk of data tampering.

Global Collaboration and Standardization: There is a need for international cooperation and standardization of vaccine tracking systems. This will facilitate seamless sharing of data across borders, particularly in the context of global health emergencies.

Personalized Vaccination Strategies: Advanced analytics and personalized medicine approaches will enable tailored vaccination strategies based on individual risk factors, demographics, and health status. This can optimize vaccine allocation and improve overall immunization outcomes.

Enhanced Surveillance for New Pathogens: Vaccine tracking systems will need to be adaptable to emerging infectious diseases. Rapid deployment and tracking of vaccines for novel pathogens will be crucial in future pandemics.

Public-Private Partnerships: Collaboration between governments, healthcare providers, pharmaceutical companies, and technology firms will be essential for the development and implementation of robust vaccine tracking systems. This will ensure access to the latest technologies and expertise.

Behavioral Insights and Education Campaigns: Future vaccine tracking efforts should include behavioral research to understand factors influencing vaccine acceptance and hesitancy. This will inform targeted education and awareness campaigns to address public concerns.

10. APPENDIX

Vaccine Tracking: Vaccine tracking involves keeping a record of vaccines from production to administration. This is crucial for ensuring the authenticity, safety, and efficacy of vaccines.

Blockchain Technology: Blockchain is a decentralized, distributed ledger technology that records transactions across multiple computers. It ensures transparency, security, and immutability of data.

Transparent Vaccine Tracking: Using blockchain for vaccine tracking can provide transparency at every step of the process. Each transaction (in this case, a vaccine-related event like production, shipping, administration, etc.) is recorded on the blockchain, making it accessible to all relevant parties.

Production: When a batch of vaccines is produced, the details (e.g., manufacturer, batch number, production date) are recorded on the blockchain.

Shipping and Handling: As the vaccines move through the supply chain, each step is recorded. This includes when they leave the factory, enter a distribution center, and arrive at a healthcare facility.

Administration: When a vaccine is administered, details such as the patient's information, the administering healthcare professional, and the date are recorded on the blockchain.

The benefits of using blockchain for vaccine tracking include**:**

Transparency: All relevant parties (manufacturers, distributors, healthcare providers, and even patients) can access the information, ensuring trust and accountability.

Security: Data on a blockchain is secured through cryptographic techniques, making it highly resistant to tampering.

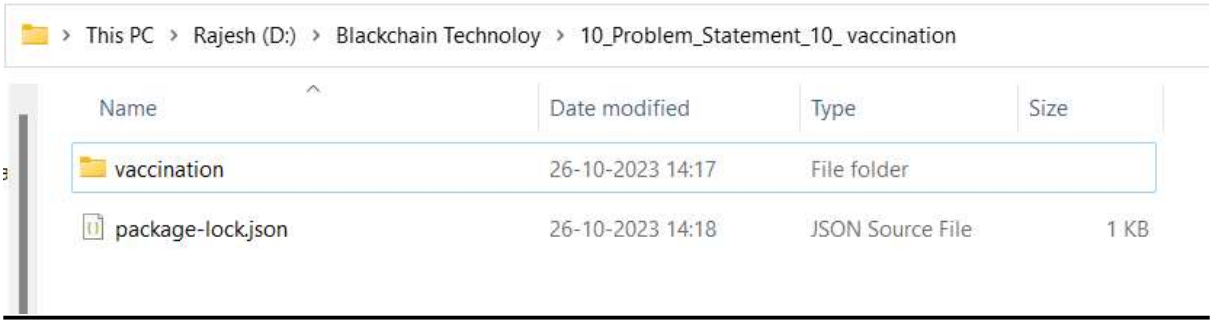
Immutability: Once information is recorded, it cannot be altered, providing a reliable record of events.

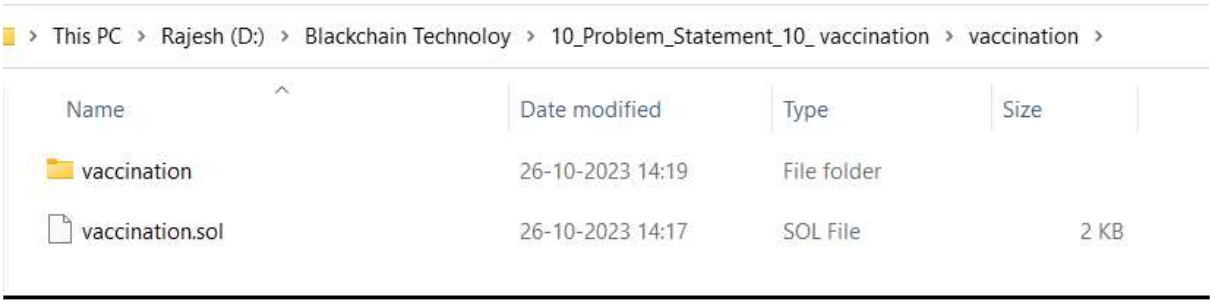
Traceability: It allows for precise tracking of each vaccine's journey, which is critical for quality control and recalls if necessary.

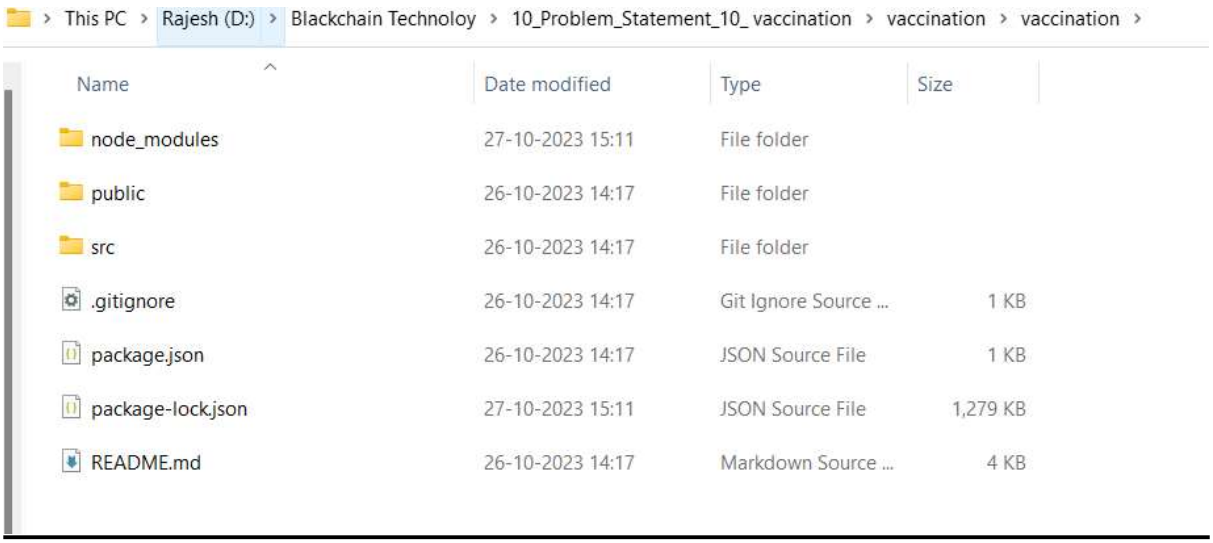
Reduced Fraud: With transparent records, it becomes harder for counterfeit vaccines to enter the supply chain.

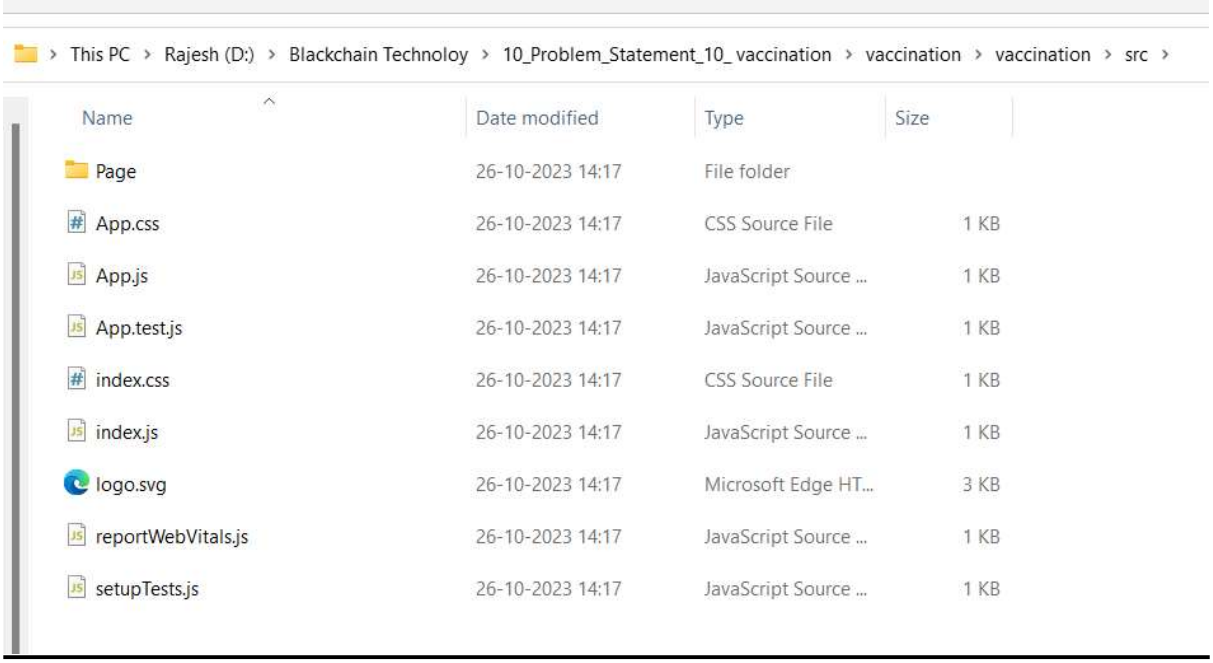
11.SOURCE CODE

Folder Structure:






This PC
Rajesh (D:)
Blackchain Technoloy
10_Problem_Statement_10_ vaccination
vaccination
vaccination
Name
Date modified
Type
Size
node_modules
27-10-2023 15:11
File folder
public
26-10-2023 14:17
File folder
src
26-10-2023 14:17
File folder
.gitignore
26-10-2023 14:17
Git Ignore Source ...
1 KB
package.json
26-10-2023 14:17
JSON Source File
1 KB
package-lock.json
27-10-2023 15:11
JSON Source File
1,279 KB
README.md
26-10-2023 14:17
Markdown Source ...
4 KB


This PC
Rajesh (D:)
Blackchain Technoloy
10_Problem_Statement_10_ vaccination
vaccination
vaccination
src
Name
Date modified
Type
Size
Page
26-10-2023 14:17
File folder
App.css
26-10-2023 14:17
CSS Source File
1 KB
App.js
26-10-2023 14:17
JavaScript Source ...
1 KB
App.test.js
26-10-2023 14:17
JavaScript Source ...
1 KB
index.css
26-10-2023 14:17
CSS Source File
1 KB
index.js
26-10-2023 14:17
JavaScript Source ...
1 KB
logo.svg
26-10-2023 14:17
Microsoft Edge HT...
3 KB
reportWebVitals.js
26-10-2023 14:17
JavaScript Source ...
1 KB
setupTests.js
26-10-2023 14:17
JavaScript Source ...
1 KB

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>
```

Manifest.json

```
{
  "short_name": "React App",
  "name": "Create React App Sample",
  "icons": [
    {
      "src": "favicon.ico",
      "sizes": "64x64 32x32 24x24 16x16",
      "type": "image/x-icon"
    },
    {
      "src": "logo192.png",
      "type": "image/png",
      "sizes": "192x192"
    },
    {
      "src": "logo512.png",
      "type": "image/png",
      "sizes": "512x512"
    }
  ],
  "start_url": ".",
  "display": "standalone",
  "theme_color": "#000000",
  "background_color": "#ffffff"
}
```

App.css

```css
.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}
```

App.js

```
import './App.css';
import Home from './Page/Home'

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <Home />
      </header>
    </div>
  );
}

export default App;
```

Home.js

```
import React, { useState } from "react";
import { Button, Container, Row, Col } from 'react-bootstrap';

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 Home;
```

App.test.js

```
import { render, screen } from '@testing-library/react';
import App from './App';

test('renders learn react link', () => {
  render(<App />);
  const linkElement = screen.getByText(/learn react/i);
  expect(linkElement).toBeInTheDocument();
});
```

Index.css

```
body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto',
'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}
```

```
code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
    monospace;
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

Reportwebvitals.js

```
const reportWebVitals = onPerfEntry => {
  if (onPerfEntry && onPerfEntry instanceof Function) {
    import('web-vitals').then(({ getCLS, getFID, getFCP, getLCP, getTTFB }) =>
{
      getCLS(onPerfEntry);
      getFID(onPerfEntry);
      getFCP(onPerfEntry);
      getLCP(onPerfEntry);
      getTTFB(onPerfEntry);
    });
  }
```

```
};

export default reportWebVitals;
```

Setuptest.js

```
// jest-dom adds custom jest matchers for asserting on DOM nodes.
// allows you to do things like:
// expect(element).toHaveTextContent(/react/i)
// learn more: https://github.com/testing-library/jest-dom
import '@testing-library/jest-dom';
```

Package.json

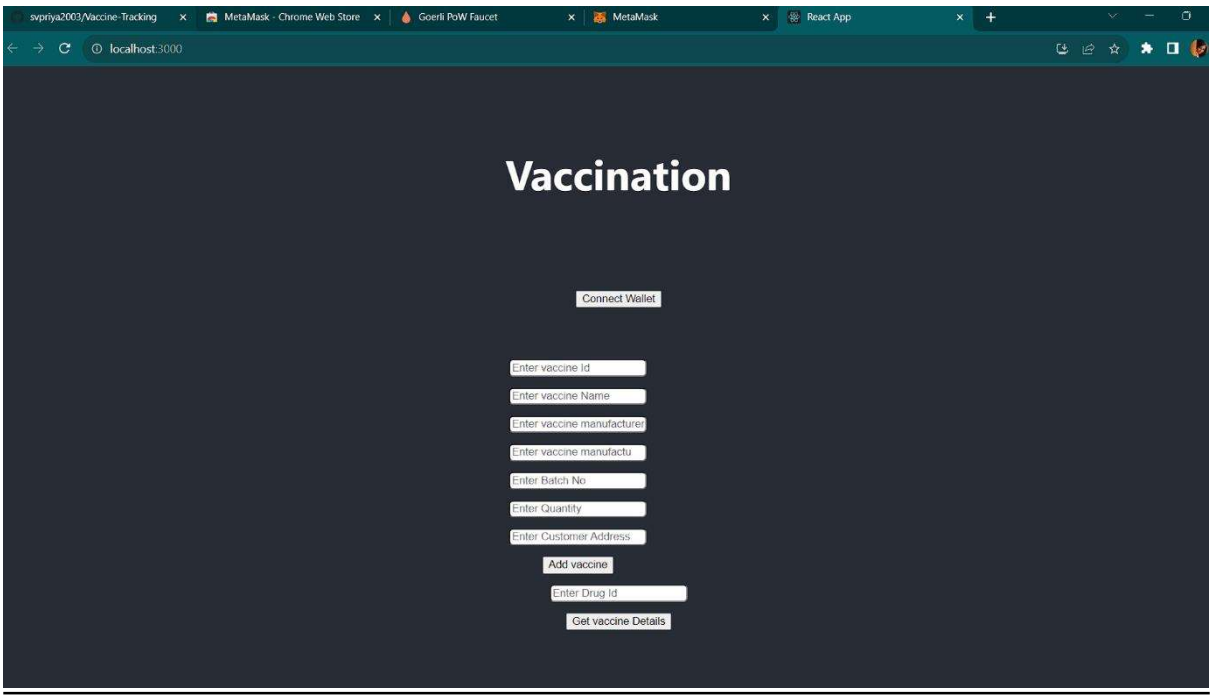
```
{
  "name": "vaccination",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@testing-library/jest-dom": "^5.17.0",
    "@testing-library/react": "^13.4.0",
    "@testing-library/user-event": "^13.5.0",
    "ethers": "^5.6.6",
    "react": "^18.2.0",
    "react-bootstrap": "^2.8.0",
    "react-dom": "^18.2.0",
    "react-scripts": "5.0.1",
    "web-vitals": "^2.1.4"
  },
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },
```

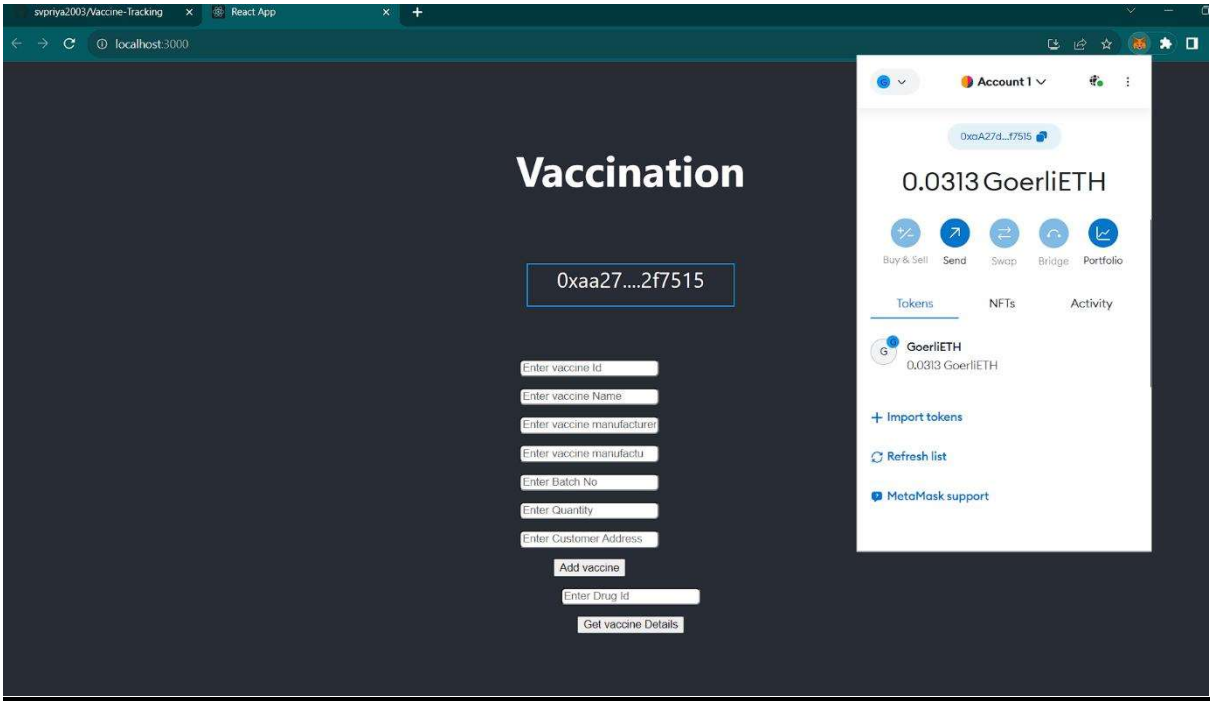
```
  "browserslist": {
    "production": [
      ">0.2%",
      "not dead",
      "not op_mini all"
    ],
    "development": [
      "last 1 chrome version",
      "last 1 firefox version",
      "last 1 safari version"
    ]
  }
}
```

GitHub Link: https://github.com/svpriya2003/Vaccine-Tracking

Demo Link: https://drive.google.com/file/d/1XWBRSA3chqWMIsTUsQRezY7gWhntQ6zX/view?usp=drive_link

11.OUTPUT SCREENSHOTS




svpriya2003/Vaccine-Tracking
React App
localhost:3000
Vaccination
0xaa27....2f7515
Enter vaccine Id
Enter vaccine Name
Enter vaccine manufacturer
Enter vaccine manufactu
Enter Batch No
Enter Quantity
Enter Customer Address
Add vaccine
Enter Drug Id
Get vaccine Details
Account 1
0xaA27d...f7515
0.0313 GoerliETH
Buy & Sell
Send
Swap
Bridge
Portfolio
Tokens
NFTs
Activity
GoerliETH
0.0313 GoerliETH
Import tokens
Refresh list
MetaMask support