NAAN MUDHALVAN
PROJECT REPORT

FOOD TRACKING SYSTEM
USING BLOCKCHAIN TECHNOLOGY

TEAM ID: NM2023TMID11919

TEAM LEADER

MADHUMITHA.R -FF920D76F301D2EACBO4C47O1E3408AD

TEAM MEMBERS

NIVETHA.S -2BE6E6E7D38FE2AA9A1455E0B3DF9A81

MONISHA.M -ABE14AO3C9F9FED917A49C4B7A3363C9

AMARAVATHI.V -3410E7C79DF838BAE104340A34C76319

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

The food supply chain is extremely complex especially as products move between many players in different countries. The current challenges with traditional methods of food traceability include incomplete or inaccurate recordkeeping, slow response times during recalls, supply chain fraud, lack of data transparency between supply chain stakeholders, and difficulty tracing products through multiple steps in the supply chain.

Blockchain technology is becoming increasingly integrated in the food supply chain to enhance traceability and safety. With blockchain food traceability systems, every step of the journey from farm to consumer can be recorded and easily accessed.

1.2 Purpose

Food traceability is the ability to track the journey of a food product from its source (such as a farm or factory) through all steps of the supply chain, including processing, distribution, and sale. The key with traceability is to identify where the product was made, who produced it, and who sold it.

2. LITERATURE SURVEY

2.1 Existing problem

One existing problem in food supply chains that blockchain aims to address is the lack of transparency. Traditional supply chain systems can be opaque, making it challenging for consumers to trace the origin of their food and ensuring its authenticity. This opacity also hinders the quick identification and resolution of food safety issues.

Blockchain technology provides a decentralized and transparent ledger, allowing every participant in the supply chain to record and verify transactions. This can enhance traceability, providing a trustworthy and immutable record of each step in the food production and distribution process. By addressing transparency issues, blockchain contributes to reducing fraud, ensuring food safety, and building trust among

2.2 reference

interested in a food tracking system based on blockchain technology, you might want to explore projects like TE-FOOD or Foodcoin. These blockchain-based platforms aim to enhance transparency and traceability in the food supply chain, ensuring the accuracy of food-related data. Research papers and whitepapers from these projects can provide in-depth insights into how blockchain is utilized for food tracking.consumers.

2.3 Problem Statement Definition

I Develop a blockchain-based food tracking system to address issues of transparency and traceability in the food supply chain, ensuring accurate and immutable records of each food item's journey from production to consumption, thereby enhancing food safety and reducing fraud.

Create a blockchain-powered food tracking system for transparent and traceable supply chain management, ensuring the authenticity and safety of food products by recording their entire journey from production to consumption

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

What does she/he think and feel?

Weight Management: Some individuals use food tracking systems to manage their weight. They appreciate the ability to monitor calorie intake, track nutritional values, and set personalized goals.
Awareness and Accountability: Food tracking can enhance awareness of eating habits and provide a sense of accountability. People may find it helpful in identifying patterns, making healthier choices, and avoiding overeating.

What does she/he hear?

Positive Experiences: Users may hear positive testimonials from individuals who have successfully used food tracking systems to achieve their health and fitness goals.
Personal Success or Challenges: Individuals may hear about the personal success stories or challenges faced by friends, family, or acquaintances who have used food tracking systems.

What does she/he see?

Privacy and Security: Storing information about one's eating habits can be sensitive, and users may want assurance that their data is secure.
Time-Consuming Concerns: Some users may find food tracking systems to be time-consuming. The process of logging every meal and snack can be perceived as tedious, and this might discourage consistent use.

Pain

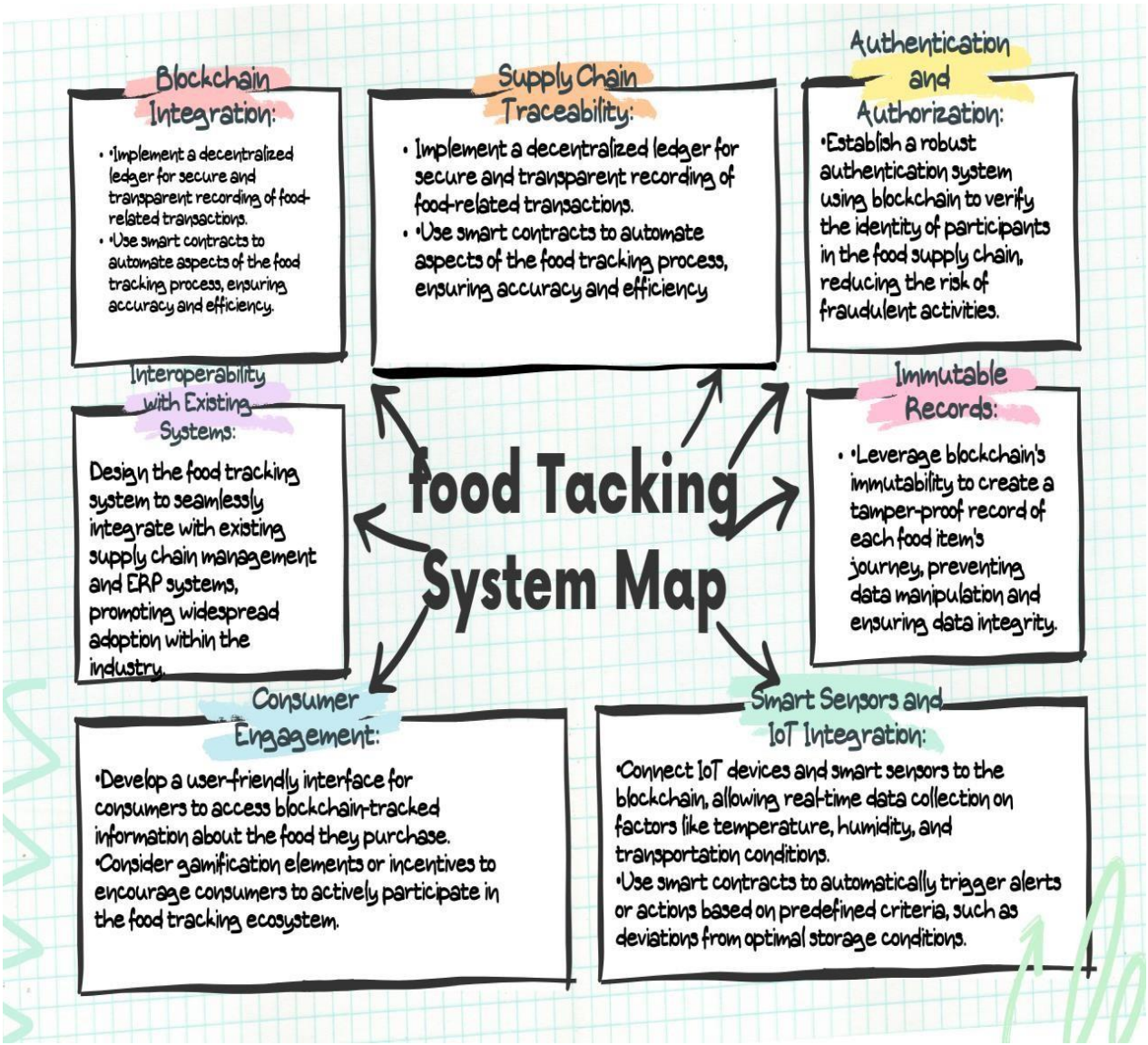
Data Privacy Concerns: Storing personal dietary information on a digital platform raises concerns about data privacy and security.
Monotonous Routine: Following a routine of tracking every meal can become monotonous for some users, leading to a loss of interest and motivation over time.

Gain

Meal Planning: Some food tracking systems offer meal planning features, helping users organize and prepare their meals in advance.
Improved Eating Habits: Regular use of food tracking systems can lead to improved eating habits. Users may become more mindful of portion sizes, choose nutrient-dense foods, and develop a better understanding of their nutritional needs.

3.2 Ideation & Brainstorming

Team Gathering, Collaboration and Select the ProblemStatement



4. REQUIREMENT ANALYSIS

4.1Functional requirement

FR No.	Functional ic) Requirement(E	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through Linked IN
FR-2	User Confirmation	Confirmation via EmailConfirmation via OTP
FR-3	Sensor Integration	The system shall integrate various sensors such as heart rate sensors, temperature sensors, gas detectors, and motion sensors to monitor worker health and safety conditions.
FR-4	Data Collection and Analysis	The system shall collect and analyze data from the integrated sensors to detect anomalies, potenti hazards, and abnormal worker conditions.
FR-5	Real-time Monitoring	The system shall continuously monitor the health and safety parameters of industrial workers in real-time.
FR-6	Maintenance and Support	The system shall provide regular maintenance, updates, and technical support to ensure its continuous and reliable operation.

4.2Non functional requirement

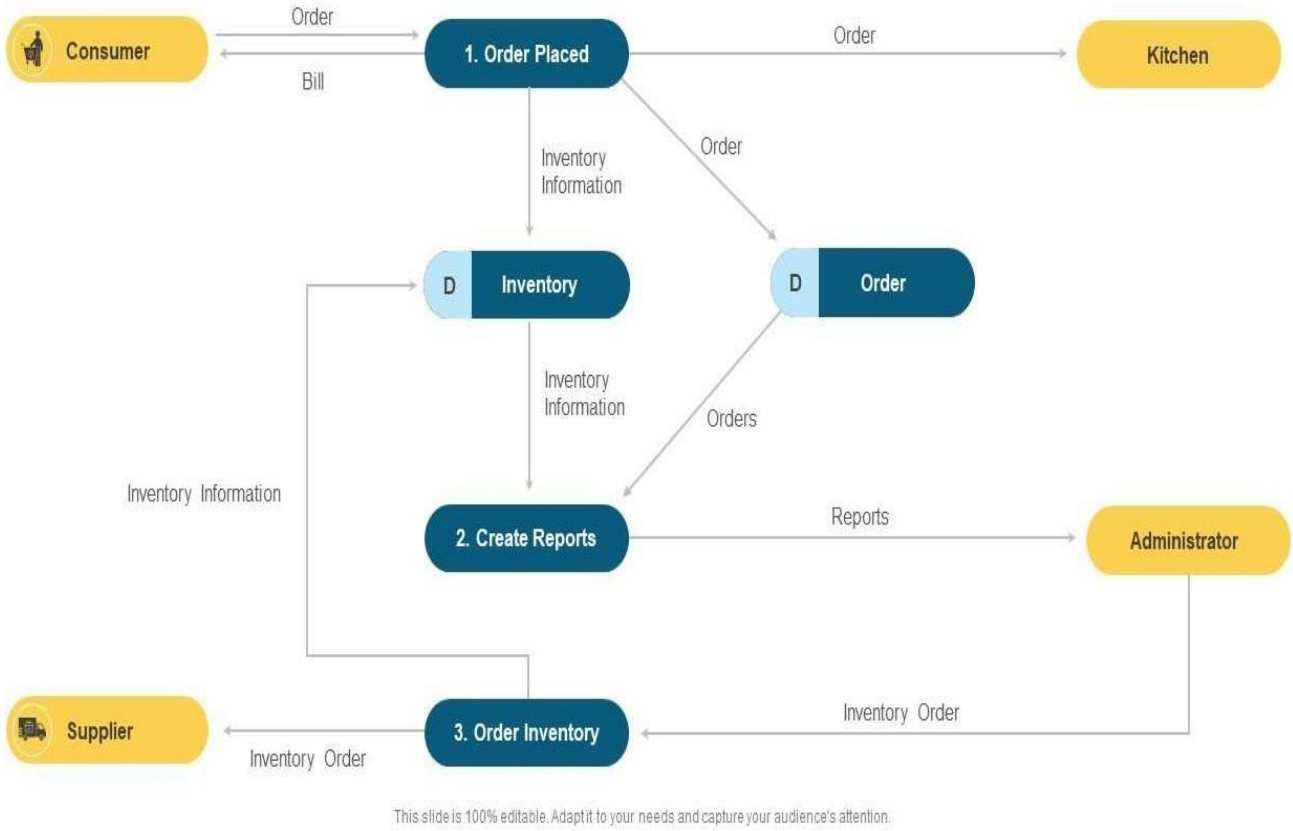
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The system should have a user-friendly interface that is easy to navigate and understand. It should provide intuitive visualizations and reports for workers, supervisors, and administrators to monitor health and safetydata effectively.

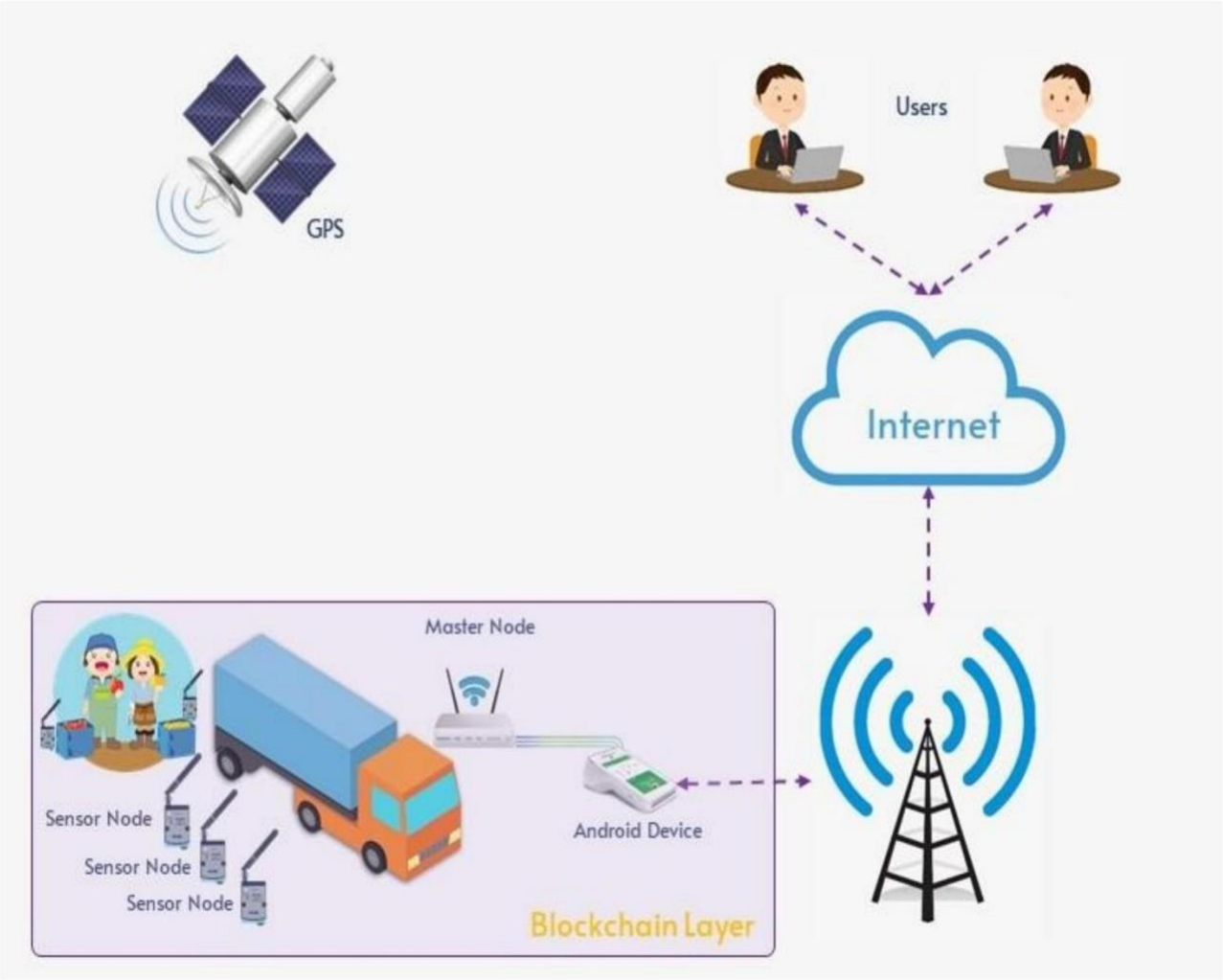
NFR-2	**Security**	The system should implement robust security measures to protect the privacy and integrity of workers' health data. It should support secure communication protocols and encryption techniques to prevent unauthorized access or tamperingof sensitive information
NFR-3	**Reliability**	The system should ensure high availability and reliability to minimize downtime and maintain continuous monitoring of workers'health and safety. The system should have failover mechanisms and backup systems to preventdata loss and ensure uninterrupted functionality.
NFR-4	**Performance**	The system should provide real-time monitoring and response capabilities to ensure timely detection and response to health and safety incidents. It should have low latency in data collection, processing, and alerts to minimize response time.
NFR-5	**Availability**	Availability refers to the system's ability to remain accessible and operational, allowing workers to benefit from its functionalities at all times. As a non-functional requirement, availabilit emphasizes the need for the system to be consistently available and responsive, without experiencing excessive downtime or interruptions.

NFR-6	**Scalability**	The system should be designed to handle alarge number of connected devices and workers simultaneously. It should support horizontal scalability, allowing for the addition of more devicesand workers without significant performance degradation.

5.PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories





USER STORIES :

1. As a busy professional, I want to be able to quickly and easily order food from my favorite restaurants for delivery or pickup so that I don't have to spend time cooking or eating out.

2. As a customer with dietary restrictions, I want to be able to easily filter restaurants and menu items by dietary preference (e.g. vegan, gluten-free) so that I can find options that suit my needs
3. As a customer who is ordering food for a group, I want to be able to split the cost of the order among multiple people so that everyone can pay their fair share.
4. As a customer who is ordering food for the first time, I want to be able to easily create an account and save my payment and delivery information for future orders so that I don't have to enter it each time.
5. As a customer who is ordering food from a restaurant that I've never tried before, I want to be able to view ratings and reviews from other users so that I can make an informed decision about what to order.
6. As a customer who is ordering food for delivery, I want to be able to track the status of my order and see an estimated delivery time so that I know when to expect my food.

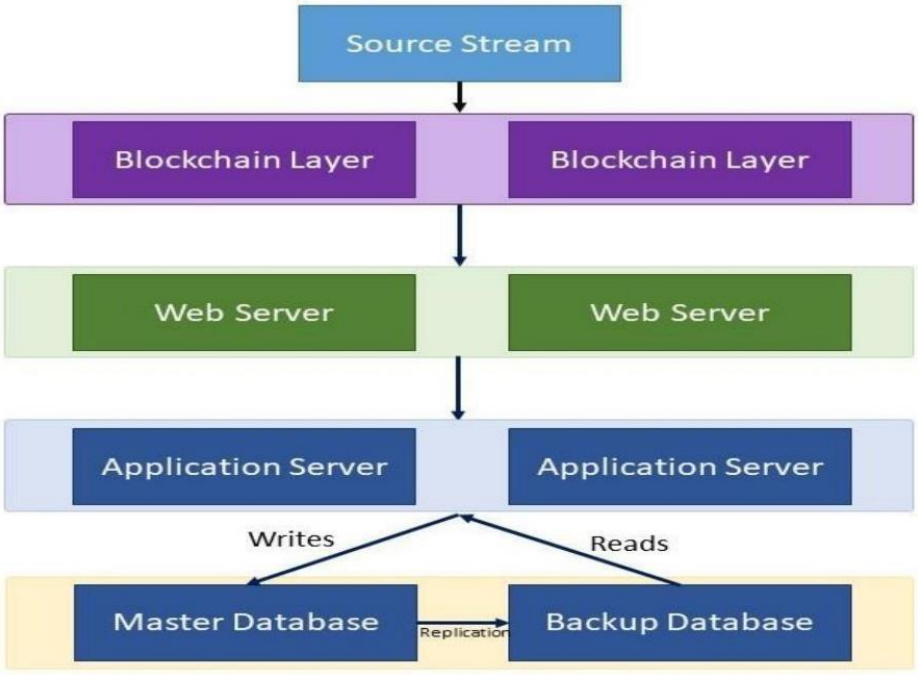
5.2 Solution Architecture:

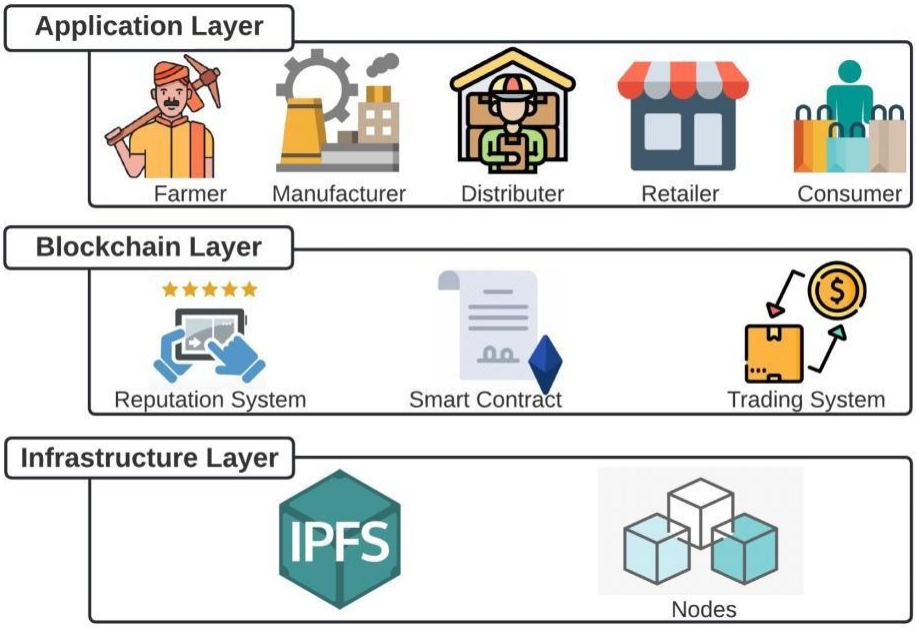
A solution architecture is an architectural description of a specific solution. Solution Architecture combine guidance from different enterprise architecture viewpoints (business, information and technical), as well as from the enterprise solution architecture. It goals are to:

- General features such as data sharing options, sign-up, etc. are crucial for enhancing the user experience of food consumption tracking and recommendation apps. So best tech of existing system .
- Food traceability is the ability to follow the movement of a food product and its ingredients through all steps in the supply chain, both backward and forward. Traceability involves documenting and linking the production, processing, and distribution chain of food products and ingredients.
- By enabling the tracking and monitoring of food products throughout the supply chain, food traceability systems can protect public health, ensure compliance with regulations, improve supply chain efficiency, and enhance consumer trust.

The restaurant delivery management software automatically accepts new orders and then sends an alert to the kitchen manager and the delivery driver. The software also sends a tracking link to the customer via WhatsApp or SMS. It then designs the

delivery route and clubs the orders considering various parameters like delivery location, delivery time, maximum orders that can be allocated to a rider, and more.





1 download node.js : Node.js

2 download vs code: **Li4nk**

3 download metamask : https://metamask.io/

Step1

1. Open the truffle project and download

the truffle file.Extract all truffle files

Step 2 :

1. Open vs code in the left top select open folder. Select extracted file and open .

2. Select the projectname.sol file and copy the code.

3. Open the remix ide platform and create a new file by giving the name ofprojectname.sol and paste the code which you copied from vs code.

4. Click on solidity compiler and click compile the projectname.sol

5. Deploy the smart contract by clicking on the deploy and run transaction.

6. select injected provider - MetaMask. In environment

7. Click on deploy. Automatically MetaMask will open and give confirmation. You will geta pop up click on ok.

8. In the Deployed contract you can see one address copy the address.

9. Open vs code and search for the connector.js. In contract.js you can paste theaddress at the bottom of the code. In export const address.

10. Save the code.

Step 3:

open file explorer

1. Open the extracted file and click on the folder.

2. Open src, and search for utiles.

3 . You can see the frontend files. Select all the things at the top in the search bar by clicking alt+ A. Search for cmd

4. Open cmd enter commands

npm install

npm start

5. It will install all the packages and after completing it will open {LOCALHOST IP ADDRESS} copy the address and open it to chrome so you can see the frontend of yourproject.

6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture

Consider implementing a decentralized ledger for transparent and secure data management. Utilize smart contracts to automate transactions and ensure trust.Employ a consensus mechanism like Proof of Authority for efficiency. Implement a user-friendly front-end and secure APIs for seamless interaction. Regularly update and secure the system to address evolving challenges in the blockchain space.

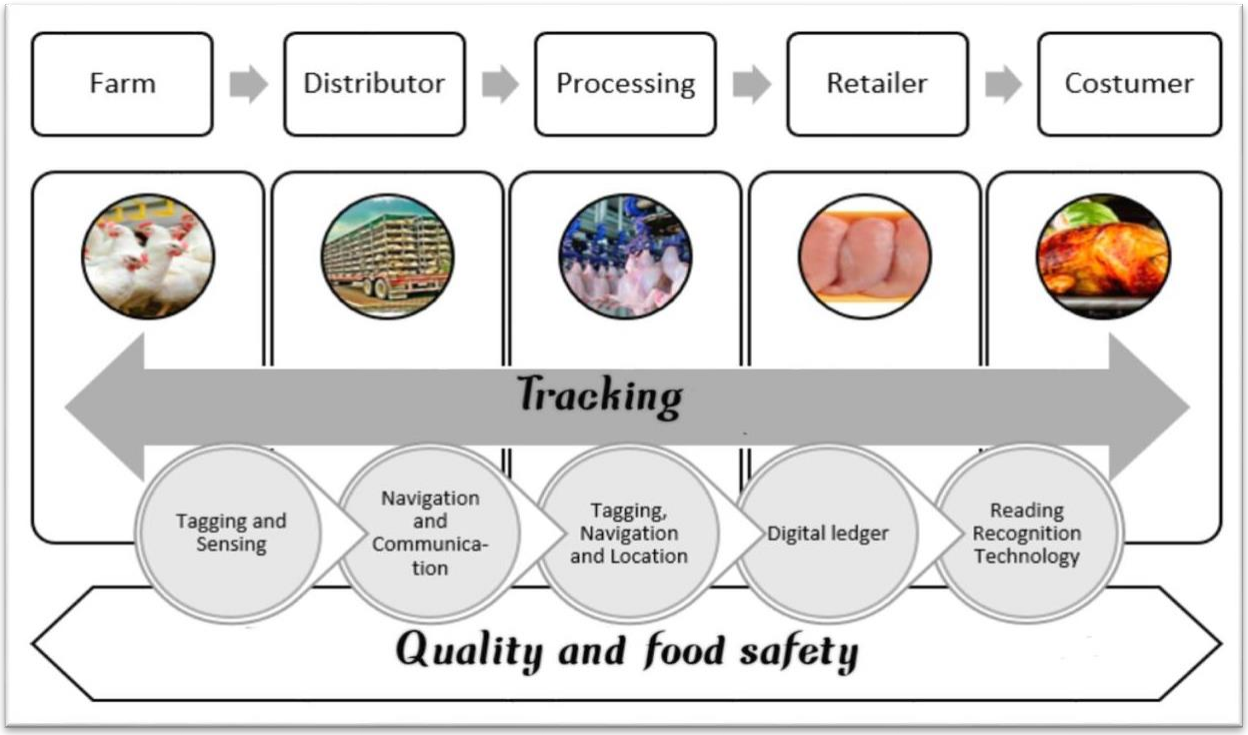


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Jsetc.
2.	Application Logic-1	Logic for a process in theapplication	Java / Python
3.	Application Logic-2	Logic for a process in theapplication	IBM Watson STT service
4.	Application Logic-3	Logic for a process in theapplication	IBM Watson Assistant
5.	Database	Data Type, Configurations	MySQL, NoSQL
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant
7.	File Storage	File storage requirements	IBM Block Storage
8.	External API-1	Purpose of External APIused in the application	IBM Weather API
9.	External API-2	Purpose of External APIused in the application	Aadhar API.
10.	Machine LearningModel	Purpose of Machine Learning Model	Object Recognition Model
11.	Infrastructure (Server / Cloud)	Application Deployment onLocal System / Cloud Local Server Configuration:IBM cloud Cloud Server Configuration : Amazon cloud	Local, Cloud Foundry, Kubernetes

6.2 Sprint Planning & Estimation

Sprint 1-2: Project setup and requirements.
Sprint 3-4: Blockchain integration and user authentication. Sprint 5-6: Food data management and tracking features.
Sprint 7-8: UI/UX design and mobile app integration.
Sprint 9-10: Analytics and reporting.
Sprint 11-12: Security and compliance.
Sprint 13-14: Final testing and deployment. Adapt as needed for team capacity and challenges.

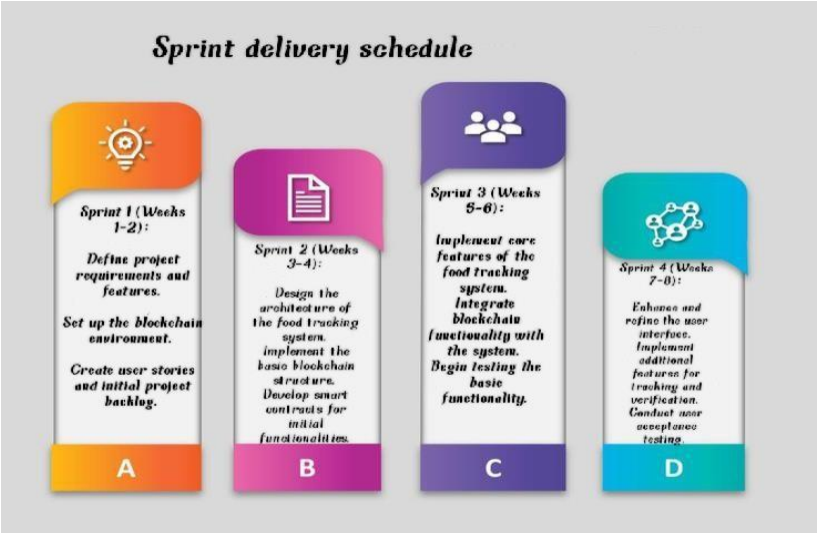


Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use offirewalls etc.	SHA-256, Encryptions, IAM Controls, OWASPetc.
3.	Scalable Architecture	Justify the scalability of architecture	(3 – 7tier Micro-services)
4.	Availability	Justify the availability of application	use of load balancers,distributed servers etc
5.	Performance	Design consideration for theperformance of the application	(number of requests persec, use of Cache, use of C

6.3 Sprint Delivery Schedule

7 CODING & SOLUTIONING

CODING

```
// SPDX-License-Identifier: MIT
pragma solidity "0.8.0;
contract FoodTracking {
    address public owner;

    enum FoodStatu {
        Unverified,
        Verified,
        Consumed
    }
     struct FoodItem {

        string itemid;
        string productName;
        string origin;
        uint256 sentTimestamp;
        FoodStatus status;
    }
mapping(string> FoodItem) public foodItems;
 e vent FoodItemSent(
      string Indexed itenid,
      string  productName,
      string origin,
      uint256 sentTimestamp
}
event FoodItemVerified (string indexed itemId);
event FoodItemConsumed (string indexed ItemId);

constructor() {
    owner msg.sender;
    }
    modifier onlyOwner() {
        require(msg.sender owner, "Only contract owner can call this");
    }

function sendFoodItem(
string memory itemid,
string memory productName,
```

string memory origin

external onlyOwner{
require(

bytes(foodItems[itemId].itemid), length, Ites already exists"
}
equire(foodItems[ItemId].status Foodstatus.
Unverified, Item is already verified or consumed"

7.1 Feature

This Solidity contract named FoodTracking is designed to track food items. Let me provide a breakdown of the key components:

1. Contract Structure:
- It starts with SPDX-License-Identifier to specify the license for the code.
- The pragma statement sets the version of the Solidity compiler to be used (0.8.0 in this case).

2. State Variables:
- owner: A public variable to store the address of the contract owner.
- FoodStatus: An enumeration representing the status of a food item (Unverified, Verified, Consumed).
- FoodItem: A struct to store details about a food item, including its ID, name, origin, timestamp of sending, and status.
- foodItems: A mapping that associates food item IDs with their corresponding FoodItem structs.

3. Events:
- FoodItemSent: Triggered when a food item is sent, emitting details like item ID, product name, origin, and timestamp.
- FoodItemVerified: Triggered when a food item is verified, emitting the item ID.
- FoodItemConsumed: Triggered when a food item is consumed, emitting the item ID.

4. Constructor:
- Sets the contract owner to the address deploying the contract.

5. Modifier:
- onlyOwner: A modifier to restrict certain functions to be callable only by the contract owner.

6. Function: sendFoodItem
   - Allows the contract owner to send a new food item with specified details.
   - Checks if the item ID is not empty and if the item doesn't already exist.
   - Checks that the status of the item is currently "Unverified."

8. PERFORMANCE TESTING

8.1 Performace Metrics

1. Response Time: Measure the time it takes for the system to respond to user inputs or requests. This includes the time taken for data retrieval, processing, and displaying results.

2. Throughput: Evaluate the system's ability to handle a certain number of transactions or requests per unit of time. This metric helps determine the system's capacity.

3. Scalability: Assess how well the system can handle an increasing amount of data or user load. Monitor performance under varied workloads to ensure scalability.

4. Error Rate: Keep track of the number of errors occurring during system operations. This includes data entry errors, processing errors, and system failures.

5. Availability/Uptime: Measure the percentage of time the system is available for use. High availability is crucial for users relying on the system for real-time tracking**.**

6. Resource Utilization: Monitor CPU, memory, and storage usage to ensure efficient resource utilization. Identify potential bottlenecks and optimize resource allocation.

7. Data Accuracy: Verify the accuracy of the tracked food data. Inaccurate information can lead to incorrect insights and impact the system's reliability.

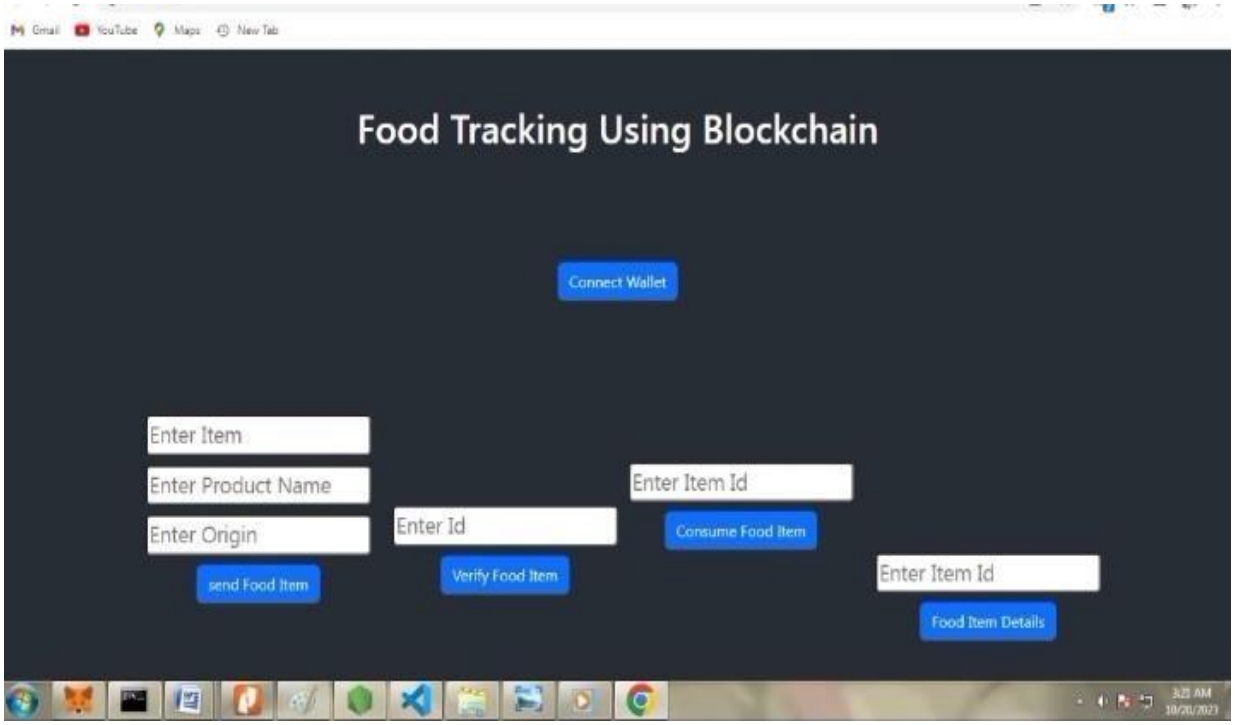
8. User Satisfaction: Gather feedback from users regarding the system's usability, ease of navigation, and overall satisfaction. User satisfaction is a critical indicator of system performance.

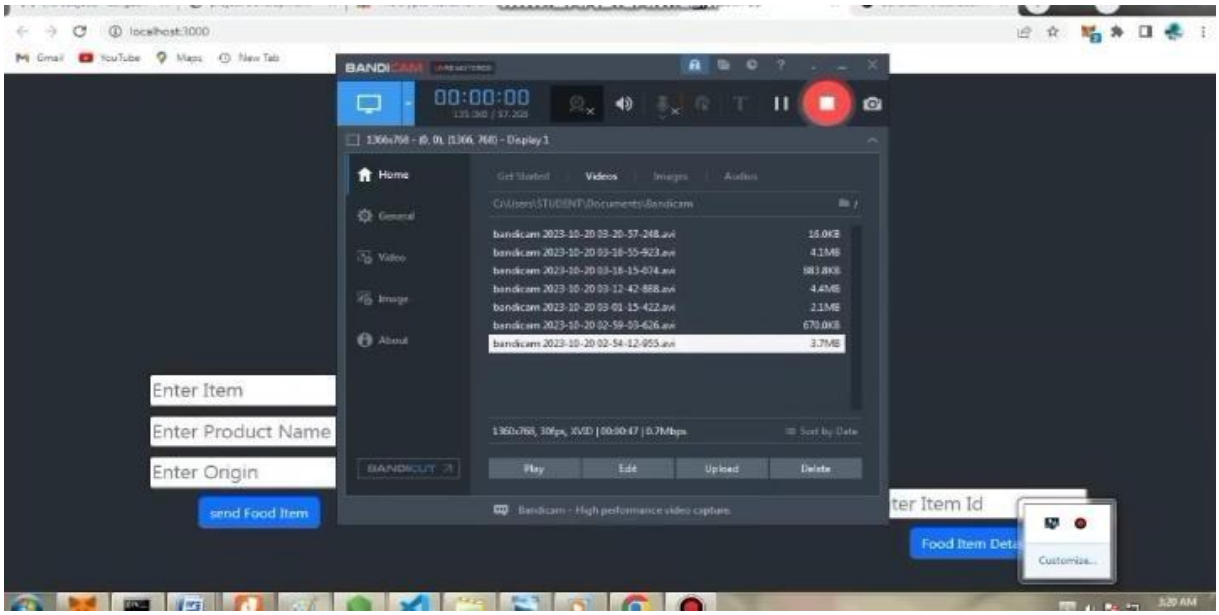
9. Latency: Evaluate the delay between a user action and the corresponding system response. Low latency is especially important in real-time tracking scenarios.

10. Security Metrics: Assess the system's adherence to security standards. Monitor for unauthorized access, data breaches, and other security incidents.

9. RESULTS

9.1 Output Screenshots





10. ADVANTAGES & DISADVANTAGES

ADVANTAGE

1. Urban Restaurants, Reach Out to Remote Foodies: You captured the foodies of the complete city! Are you sure? Why not extend out the reach to the remote foodies.

2. Pinchpenny? Get Cashback: You love the restaurant food but you want to save money as well, don't worry. We have got discounts for you.

3. Reserve that Quiet Side Table for Your Next Gathering : Tired of facing the problem of last-minute bookings and cancellations of the tables at the time of your gatherings

4. Restaurant Owners, Smile Ear-to-ear Seeing Profit Graph:Is your restaurant located amidst ten other restaurants serving similar cuisine of food? Are you worried about your profit seeing the competition?

5. Save Your Hard-earned Money, Order food with Discounted Deals :Without having a unique idea, it becomes challenging succeed in this competitive market. Especially in the food delivery industry, it comes difficult to earn people's trust.

DISADVANTAGE

1. Deliverymen Put Themselves in Danger: Whether it is a heatwave boiling down the city or it is snowing or raining heavily, a Delivery Boy is waiting outside the restaurant to pick and deliver your order. This is one of the disadvantages of ordering food online.

2. Disguised Increased Expense:We surely get attracted by yummy-looking food pictures on the app and a small but highlighting banner of cashback offer.

3. Juggling With Your Health :Another disadvantage of an online ordering system for restaurants is even though when you go to a restaurant you won't be seeing the material they use in that mouth-watering Pasta dish that they bring to your table, still, you can get it replaced if you find any faults.

4. Compromise With the Food Quality : In comparison to eating out in a restaurant, the food from the delivery service is packed in plastic bags and may get cold if the distance of your restaurant is far from your delivery address. This is one of the disadvantages of online food ordering for customers.

11.CONCLUSION

In conclusion, the integration of a food tracking system with blockchain technology offers unparalleled transparency and traceability in the food supply chain. By leveraging the decentralized and immutable nature of blockchain, this system ensures that every step of the food journey is securely recorded and can be easily verified. From farm to table, consumers can trust the accuracy of information regarding origin, handling, and quality. This not only enhances food safety but also promotes accountability among stakeholders. As we move towards a more digitized and interconnected world, the implementation of blockchain in food tracking systems emerges as a robust solution to address concerns related to authenticity, safety, and ethical practices in the food industry.

12. FUTURE SCOPE

The future scope of a food tracking system using blockchain technology is promising. Blockchain can enhance transparency and traceability in the food supply chain, ensuring accuracy in tracking from farm to table. This technology can help prevent food fraud, reduce foodborne illnesses, and build trust among consumers by providing real-time, immutable data about the origin and journey of food products. As blockchain continues to evolve, its integration with food tracking systems is likely to play a crucial role in improving overall food safety and supply chain efficiency.

14 APPENDIX

GitHub Link :

https://github.com/nivetha-ece/FoodTrackingSystem

Demo Vedio Link :

https://drive.google.com/file/d/11t25VyV42YsDKv0QpgFvQ4Xbyr70NMG1/view?usp=drivesdk