PROJECT REPORT

NAAN MUDHALVAN

PROJECT NAME: Food Tracking System

COURSE NAME: Block Chain Technology

This Project done by:

Team Leader Name: S.Saminadhan

Team Member 01 : Potta Venkata Mahesh

Team Member 02 : Senthilkumaran R

Team Member 03 : Naresh B

Department: Electronics and Communication Engineering

College Name & Address:

Prathyusha Engineering College
Poonamallee-Tiruvallur Road, Aranvoyal Kuppam,

Tiruvallur Taluk

PINCODE: 602025

1. INTRODUCTION

Blockchain for the food supply chain provides 100% traceability of the food-related data and multi-party transactions, enables backtracking the food provenance in seconds rather than days, facilitates food safety and quality compliance verification, enhances protection of supply chain data.

The global blockchain market for the food supply chain was estimated at $128.87 million in 2020. It is expected to reach $886.18 million by 2025 at a CAGR of 47.1%. The solid growth is mainly driven by the demand across the food industry for supply chain transparency and facilitated traceability of food provenance to assure product authenticity, safety and high quality. The need to prove sustainable food manufacturing practices and ethical sourcing are also among the important factors contributing to the growing popularity of the blockchain-based food supply chain solutions.

1.1. Project Overview

Allow your users to search for different restaurants, cafes by location, and cuisines. Using the search filter, users can easily find their favourite eating places, list menu, offers, etc.

Order placement: The user can place an order of selected dishes and food. They just need to cross-verify their preferred dish, delivery time, and proceed check-out.

Tracking Delivery Partners: With real-time tracking features, it becomes easy for users to track delivery drivers and know their real-time location information. Users can check the time taken by the food delivery executive to deliver their parcel.

Payment gateway integration: You provide the users with multiple payment options like credit/debit cards, different wallets like Google Pay, Paytm, Phonepe, UPI, etc

Core Features of Delivery Partner

1) Delivery Partner’s profile: Through this feature, a driver can keep his profile update. It contains his full name, address, email, contact number, photo, and other personal information.
2) Notification for orders: Through push notifications, drivers can get constant updates & alerts for new orders. It will help in the accurate delivery service of your restaurant.

3) Map for the delivery route: Integrate Google Map or other providers and allow drivers to choose the shortest and fastest routes to reach the location.

Core Features of Food Partners/ Restaurants

1)Restaurant Profile/Menu: Through this feature, a restaurant owner can add their restaurant details, menu and its availability, price, preparation times, etc

2)Notification for orders: Through push notifications, orders. It will help in the accurate delivery service of your restaurant.

3)Notifications for Pickup Partners: They will get alerts about delivery partners, their location when they will pick up, etc.

1.2. Purpose

Food traceability systems are tools that help you record, track, and verify a product's history, and the location it has reached through your supply chain. Food traceability will allow you to monitor your perishable inventory, giving you the tools to keep your business compliant and to perform product recalls.

2.IDEATION & PROPOSED SOLUTION

2.1.Problem Statement Definition

Imagine you meet a small startup company planning to launch an Online Food Delivery System like Swiggy/Zomato that allows consumers & service providers to interact in real-time. They want to support both Mobile App and Web-based applications. Like many small start-ups, they are confident that they will be the next big thing and expect significant, rapid growth in the next few months. With this in mind, they are concerned about the following:

Scaling to meet the demand, but they are not sure when and how the demand will grow — they are very concerned about buying too much infrastructure too soon or not enough too late!

Effective distribution of load.

The ability for Service Providers to send notifications to consumer.

Configurable database/s and data access layer to yield high performance and throughput.

Allocate food delivery personal efficiently.

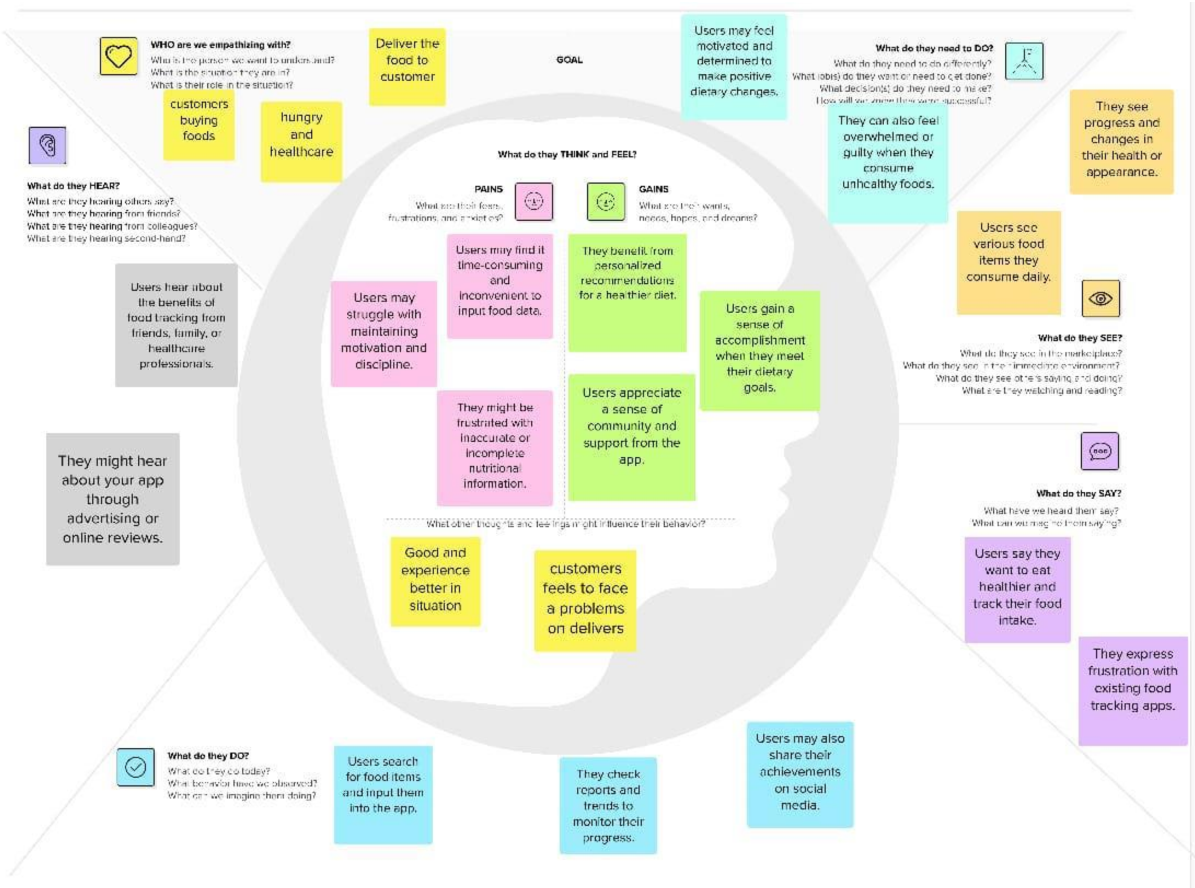
Design for easy onboarding and searchability of restaurants
Prediction of the food delivery time for order

2.2.Empathy Map Canvas

Improving Communication: Teams can use empathy maps to facilitate better communication and understanding among team members. It's a common reference point that ensures everyone has a shared understanding of the user's needs and perspectives.

1. **Identifying Opportunities**: Empathy maps help identify opportunities for improvement or innovation by revealing unmet needs or unspoken desires of the target audience.
2. **Tailoring Marketing and Messaging**: In marketing, empathy maps can be used to better understand the target audience and tailor messages and content that resonate with their emotions, thoughts, and behaviours.
3. **Reducing Assumptions**: Using an empathy map encourages teams

to rely on real data and user insights rather than making assumptions about what the users want or need.



2.3.Ideation & Brainstorming

To prioritize these ideas, you can create a matrix based on two key dimensions:

1. **Impact:** How much will the idea positively influence the effectiveness and
security of the electronic voting system? Rate it on a scale of 1-5, with 5 being the highest impact.
2. **Feasibility:** How feasible is it to implement the idea considering budget,
technology, and resources? Rate it on a scale of 1-5, with 5 being the most

Rank each idea based on these dimensions and calculate a combined score for each idea (Impact x Feasibility). Prioritize the ideas with the highest combined scores

2.4. Proposed Solution

1.Proposed below is a high-level reference architecture for Online Food Delivery systems.

2.This proposed architecture is generic and it can be deployed to any of cloud provider like AWS/GCP/Azure

Assumptions:

Delivery partners are provided with devices that have inbuilt GPS. This will help to locate their current location accurately.

Integration with map provider is there and we get details about routes, traffic and commute time.

3.REQUIREMENT ANALYSIS

3.1.Functional Requirements

1. User Registration and Profile Management:

Users should be able to create accounts, set up profiles, and manage their personal information.

2. Food Logging:

Users can log their daily food intake, including details like quantity, preparation method, and mealtime.

3. Food Database:

The system must have a comprehensive database of foods, ingredients, and recipes, allowing users to easily select and log items.

4. Nutritional Analysis:

The system should calculate and display nutritional information, such as calories, macronutrients (carbohydrates, fats, proteins), and micronutrients (vitamins and minerals) for each food item and overall meals.

5. Barcode Scanner:

Implement a feature for users to scan barcodes on packaged food items for quick input of nutritional data.

6. Meal Planning:

Users should be able to plan meals in advance, including breakfast, lunch, dinner, and snacks, with the ability to set portion sizes and nutritional targets.

7. Personalized Goals:
Allow users to set personalized nutritional goals based on factors like age, weight, gender, and activity level.
8. Recommendations and Insights:
Provide personalized recommendations and insights to help users make healthier food choices based on their goals and dietary restrictions.

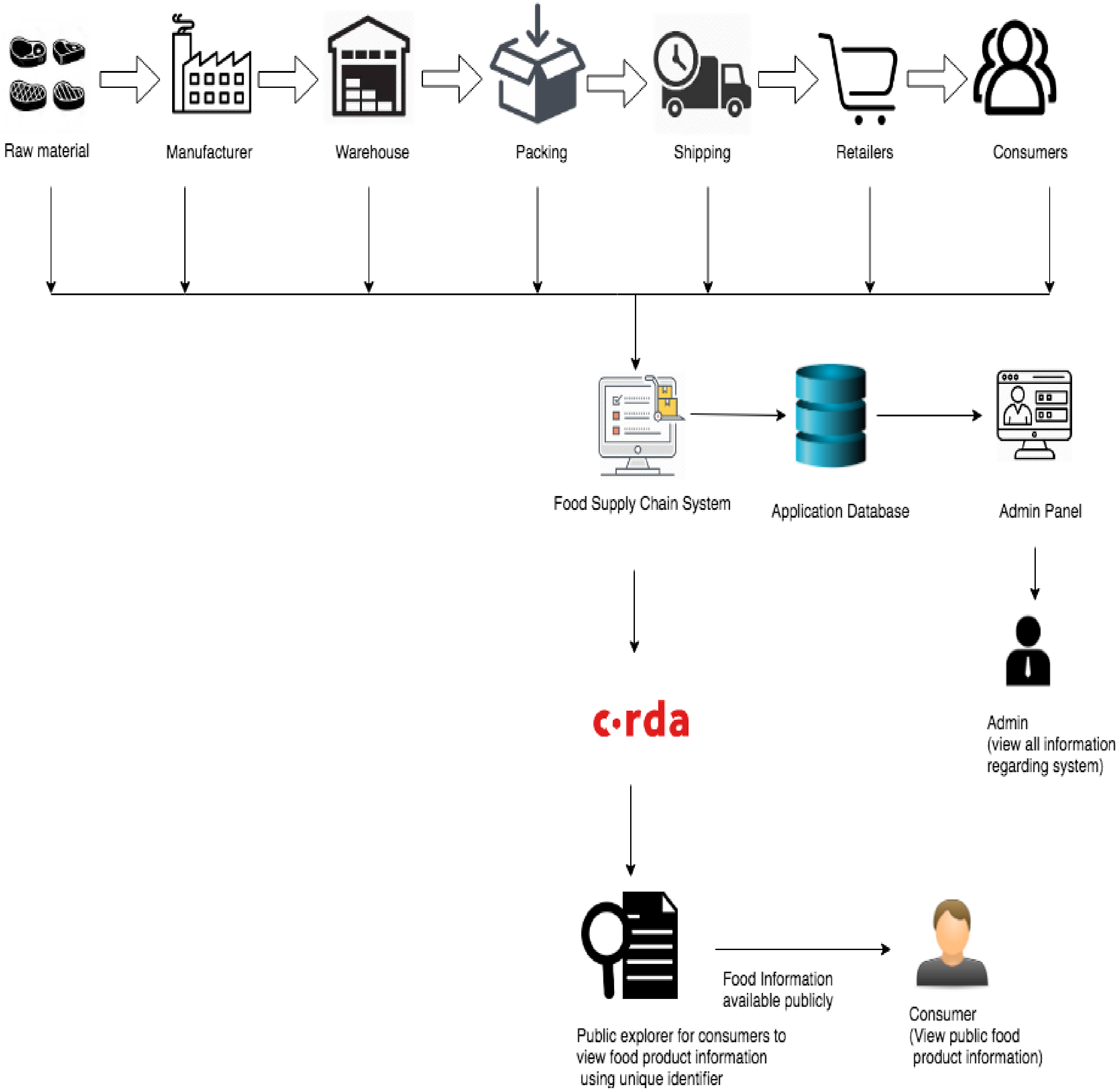
3.2.Non-Functional Requirements

1. Data Privacy and Security:
Ensure that user data, including personal information and dietary records, is securely stored and protected from unauthorized access.
2. Scalability:
Design the system to handle a growing user base and increasing data volumes without performance degradation.
3. Usability and User Experience:
The system should have an intuitive and user-friendly interface to encourage regular use by a diverse user base.
4. Reliability:
The system must be available and operational 24/7, minimizing downtime for maintenance.
5. Performance:
Ensure quick response times for data retrieval, calculations, and other core functions, even during peak usage.
6. Compatibility:
The system should be compatible with various devices and platforms, such as web browsers, mobile devices, and wearable tech.

4.PROJECT DESIGN

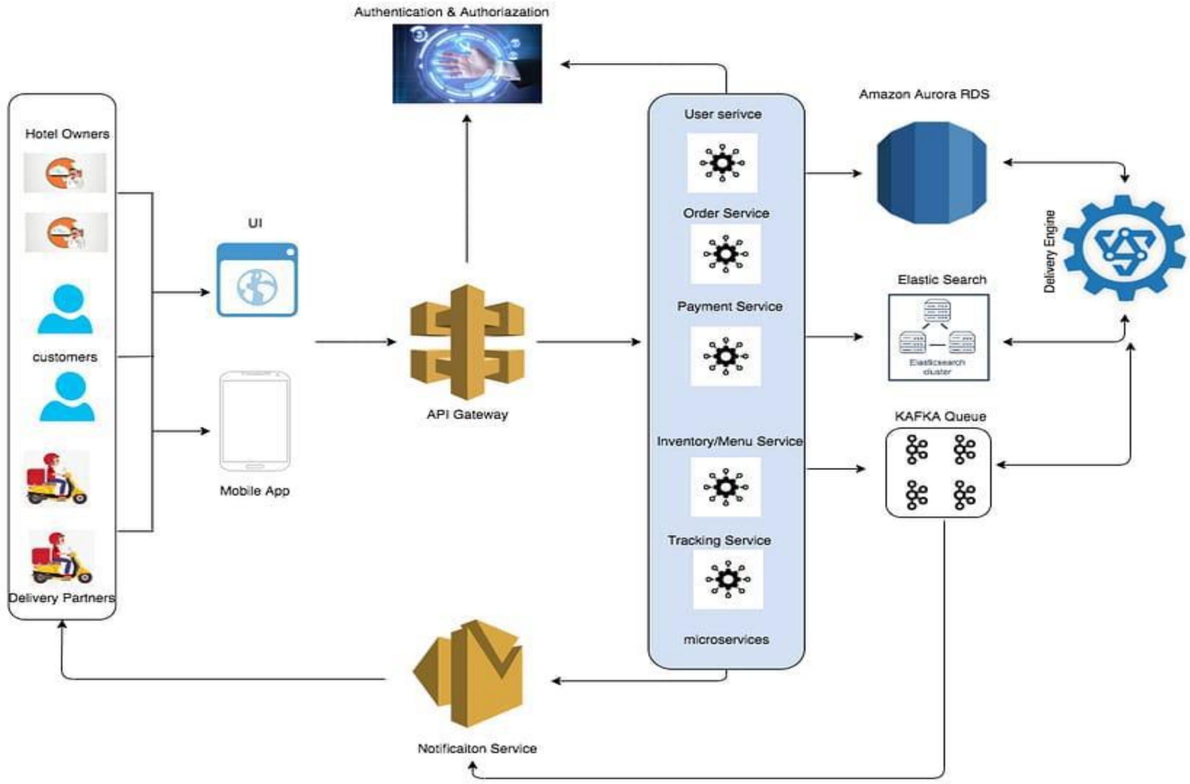
4.1.Data Flow Diagrams

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 the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

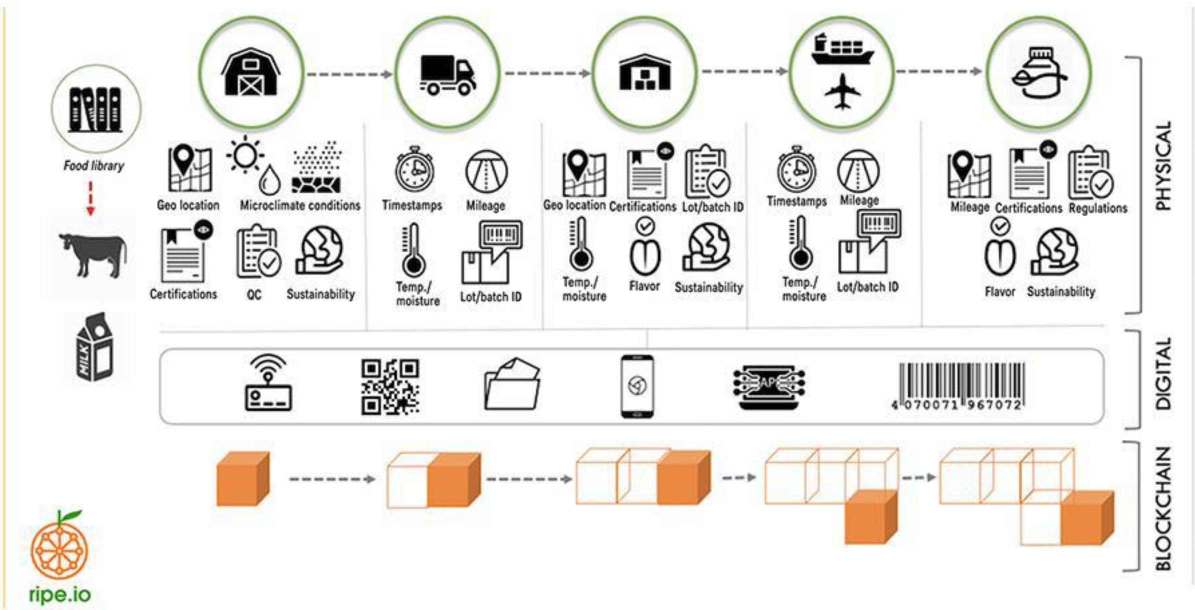


4.2.Solution & Technical Architecture

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



4.3.User Stories

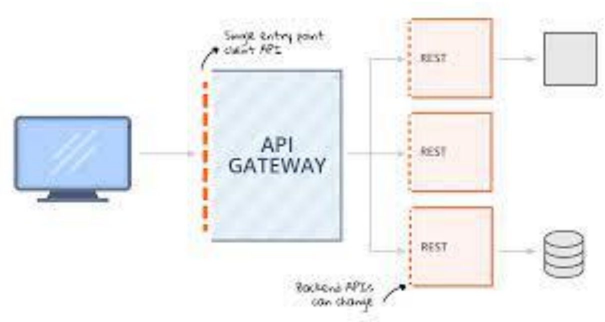


5.CODING & SOLUTION

5.1.Feature 1

API Gateway:

An API gateway is a data-plane entry point for API calls that represent client requests to target applications and services. It typically performs request processing based on defined policies, including authentication, authorization, access control, SSL/TLS offloading, routing, and load balancing.



5.2.Feature 2

Authentication,Authorization & Payment:

Authentication is knowing the identity of the user. For example, Alice logs in with her username and password, and the server uses the password to authenticate Alice. Authorization is deciding whether a user is allowed to perform an action. For example, Alice has permission to get a resource but not create a resource.

Code:

```json
{
  "name": "food-tracking",
  "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",
    "bootstrap": "^5.3.1",
    "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"
    ]
  }
}
```

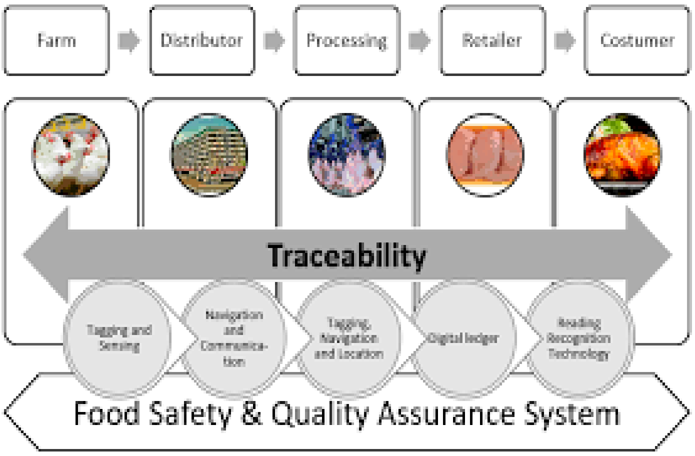
5.3.Data Base Scheme

Here we used Block Chain Technology for storing the data for more security for the data.

6.RESULTS

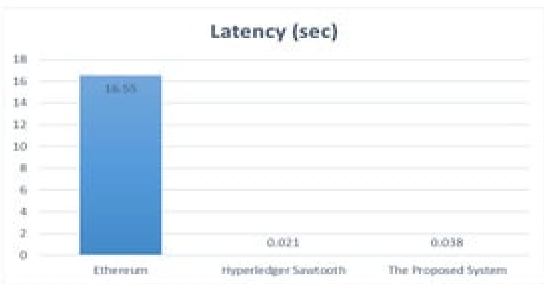
Before using the blockchain-based food tracking system, the performance data of the system were obtained. In this way, it will be necessary to prevent problems such as scalability and to stop the work if it is foreseen that the blockchain-based system to be used will not reach the desired performance values. The performance values of Ethereum and Hyperledger Sawtooth are used to benchmark the values obtained from the proposed system. A simulation environment has been set up to collect and compare these data using Matlab. The latency (s), Net Tx (bytes), Net Rx (bytes), and CPU load (%) values are the variables that keep the data obtained in this simulation environment. With the data obtained in this simulation environment, the aim is to reveal the difference with other platforms clearly and concretely.

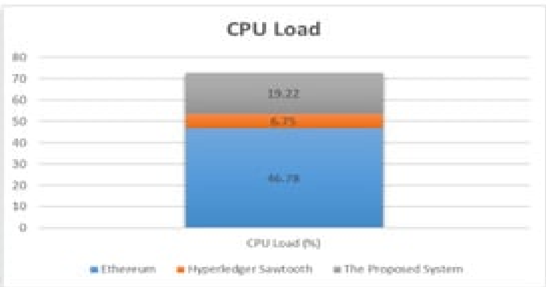
The latency (s) value in the proposed system was obtained as 0.038. The transmission per second value is 285, the reception per second value is 335, and the CPU load rate value is 19.22. Especially when we evaluate the latency times, the obtained value is at a very good level compared to Ethereum. When it is compared with Hyperledger Sawtooth, it is seen that there is a little more delay. The main reason for this is that the system architecture is more complicated, and the data size obtained is high. This is also evident from the fact that the transmission per second and reception per second values are much higher than Hyperledger Sawtooth. It has been observed that a rate of 19.22 was achieved in the CPU usage rate As a result, it is seen that the performance data obtained have a serious advantage over Ethereum, especially in terms of latency, and it has started to converge in other values. Considering that the real-time operation of the installed system is extremely important, the choice of Hyperledger Fabric has once again emerged as the right decision.

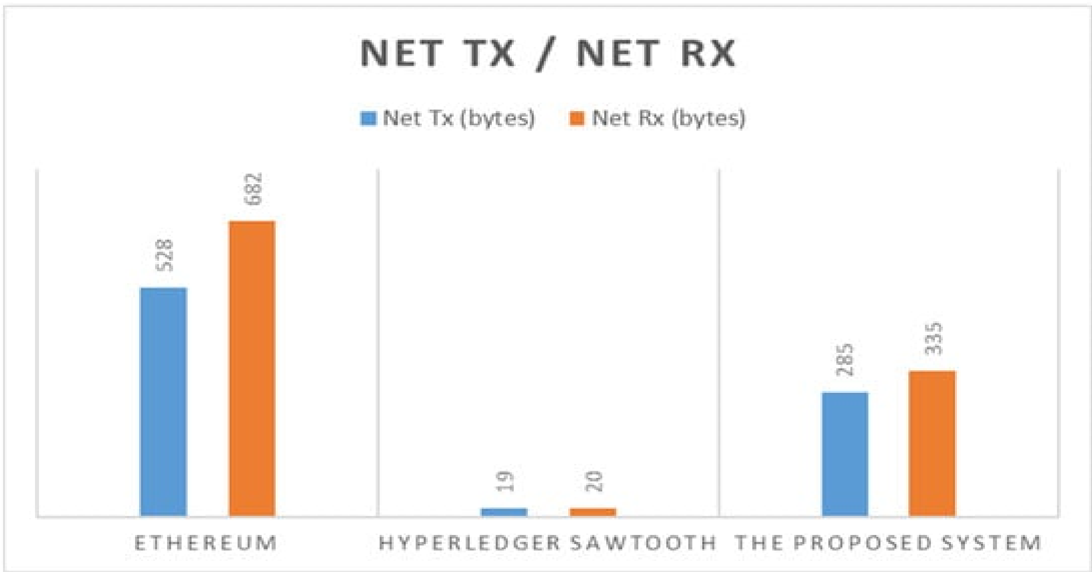


6.1.Performance Metrics

Graphical Representation:







7.ADVANTAGES & DISADVANTAGES

Advantages:

- Improved Food Safety
- Rapid Recall
- Quality Control
- Transparency
- Regulatory Compliance

Disadvantages:

- Implementation Costs
- Data Management
- Resistance to Change
- Complexity
- Error Potential

8.CONCLUSION

In conclusion, while food traceability systems offer numerous advantages in terms of food safety, quality, transparency, and efficiency, they also come with challenges related to cost, complexity, and data management. To maximize the benefits and overcome the disadvantages, businesses must carefully plan and implement traceability systems that align with their specific needs and goals.

9.FUTURE SCOPE

The future scope for food traceability systems is promising, with several trends and developments driving their continued growth and evolution. Here are some key areas of future scope for food traceability systems:

1. **Blockchain Technology**: Blockchain offers a secure, transparent, and decentralized way to record and verify food supply chain data. It can provide an immutable ledger of every step in the supply chain, enhancing transparency and trust. Many companies are exploring blockchain-based traceability solutions.
2. **Internet of Things (IoT)**: IoT devices can be used to track and monitor food products throughout the supply chain. Sensors can collect data on temperature, humidity, and other environmental conditions to ensure food quality and safety. This real-time data can be integrated into traceability systems.
3. **Artificial Intelligence (AI)**: AI can analyze large datasets generated by traceability systems to identify patterns and anomalies. Machine learning algorithms can help predict and prevent food safety issues and optimize supply chain operations.
4. **Smart Packaging**: Packaging with embedded RFID (Radio-Frequency Identification) tags or QR codes can provide consumers with information about the product's source, expiration date, and handling instructions. Smart packaging can also enable easier tracking throughout the supply chain.
5. **Consumer Demand for Transparency**: Increasing consumer demand for transparency and information about the food they consume will drive the adoption of more comprehensive and user-friendly traceability systems. Consumers may expect to access detailed information via mobile apps or websites.
6. **Regulatory Requirements**: Governments and regulatory bodies are likely to continue strengthening food safety regulations, including traceability requirements. This will drive the adoption of more robust and standardized traceability systems.
7. **Globalization and Supply Chain Complexity**: As supply chains become more global and complex, the need for traceability systems to manage and secure these supply chains will increase. This includes traceability of both raw materials and finished products.

8. **Sustainability and Ethical Sourcing**: Traceability systems can help track and verify sustainable and ethical sourcing practices, addressing growing consumer concerns about environmental and social responsibility in the food industry.
9. **Third-Party Verification**: Independent third-party organizations specializing in food traceability verification may play a larger role in certifying the accuracy and reliability of traceability data, further enhancing trust in the system.
10. **Integration with Emerging Technologies**: Traceability systems will likely integrate with emerging technologies like 5G, edge computing, and augmented reality, enabling faster data transfer and real-time insights throughout the supply chain.
11. **Food Waste Reduction**: Traceability systems can be used to reduce food waste by providing data to optimize inventory management, reduce spoilage, and ensure products are used before their expiration dates.
12. **Customization and Scalability**: Future traceability systems will need to be customizable and scalable to fit the specific needs of different sectors within the food industry, from small-scale farms to large food conglomerates.

10.APPENDIX

1. **Sample Reports**:
    - Mock food consumption reports generated by the system to demonstrate its capabilities.
2. **User Guides**:
    - User manuals or guides explaining how to use the food tracking system.
    - Troubleshooting guides for common issues.
3. **Data Tables**:
    - Tables of food items with nutritional information.
    - Sample data sets for testing and understanding data input and output.
4. **Database Schema**:

- An overview of the database structure used in the system, including tables, fields, and relationships.

5. **Flowcharts**:
   - Visual representations of the data flow and user interactions within the system.
6. **Sample Input Screens**:
   - Screenshots or wireframes illustrating the data entry forms or interfaces for users.
7. **Terms and Definitions**:
   - A glossary of key terms related to the food tracking system, such as "calories," "macros," "serving size," etc.
8. **Privacy and Security Policies**:
   - Information on how user data is collected, stored, and protected in compliance with relevant laws and regulations.
9. **API Documentation**:
   - If the system offers an API for developers, include documentation on endpoints, request/response formats, and authentication methods.
10. **Feedback Forms**:
    - Templates for collecting user feedback and suggestions for improvement.
11. **Legal Disclaimers**:
    - Legal information regarding terms of use, copyright, and liability.
12. **System Architecture**:
    - Diagrams or explanations of the system's architecture and components.
13. **Scalability and Performance Information**:
    - Details on how the system handles increased loads and performance benchmarks.
14. **Support Contact Information**:
    - Contact details for user support, including email addresses, phone numbers, or a support ticket system.
15. **Release Notes**:

- A history of system updates and changes, including new features, bug fixes, and improvements.

16. **References**:

- Citations and references to academic studies, nutritional databases, or other sources used in developing the system.

GitHub Project Link:

https://github.com/Saminadhan/Food-Tracking-System.git