VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Belagavi-590018

A Project Report on

E-CHAIN: BLOCKCHAIN BASED SECURED SUPPLY CHAIN MANAGEMENT

Submitted by

RAJATH ACHARI	**4SN20CS055**
RAKSHIT SHETTY	**4SN20CS057**
RITIK RANJAN	**4SN20CS060**

In partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING

Under the Guidance of

Dr. SANDEEP BHAT

Professor

Department of Computer Science & Engineering

SRINIVAS INSTITUTE OF TECHNOLOGY

(NAAC ACCREDITED)

MANGALURU-574143, KARNATAKA.

2023 - 2024

SRINIVAS INSTITUTE OF TECHNOLOGY

(NAAC ACCREDITED)

MANGALURU - 574143, KARNATAKA.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

Certifiedthat the project work entitled ***" E-CHAIN: BLOCKCHAIN BASED SECURED SUPPLY CHAIN MANAGEMENT"*** *is a bonafide work carried out by*

RAJATH ACHARI	**4SN20CS055**
RAKSHIT SHETTY	**4SN20CS057**
RITIK RANJAN	**4SN20CS060**

in partial fulfillment for the award of ***BACHELOR OF ENGINEERING*** *in* ***COMPUTER SCIENCE & ENGINEERING*** *of the* ***VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI*** *during the year 2023-2024.It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Engineering Degree.*

Dr. Sandeep Bhat **Project Guide**	**Prof. Ravishankara K** **Head of the Department**	**Dr. Shrinivasa Mayya D** **Principal**

Name of the Examiners **Signature with Date**

1.

2.

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got their support all along the completion of our project.

We are highly grateful and would like to express our wholehearted thankfulness to our Project Guide and Coordinator, **Dr. Sandeep Bhat**, Professor, Department of Computer Science and Engineering, who has been our source of inspiration. He has communicated various ideas for improving the project and has been especially enthusiastic in giving his opinions and critical reviews in a constructive manner. We will remember his contribution forever.

We express a deep sense of gratitude to **Prof. Ravishankara K**, Head of the Department, Computer Science and Engineering, for his cordial support, valuable information and guidance, which helped us in completing this project through various stages.

We also express our heartfelt gratitude to our Principal **Dr. Shrinivasa Mayya D.** for his kind co-operation and encouragement which helped us in the completion of this project.

We also thank our **Management** who helped us directly and indirectly for the successful completion of our project.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all the **Teaching and Non-teaching staff** of Department of Computer Science and Engineering who helped us in successfully completing our project.

Lastly, we would like to thank our **Parents** for their moral support and our **Friends** with whom we shared our day-to-day experiences and received lots of suggestions that improved our quality of work.

- RAJATH ACHARI
- RAKSHIT SHETTY
- RITIK RANJAN

ABSTRACT

In today's global economy, supply chain management (SCM) plays a pivotal role in ensuring the smooth flow of goods and services from manufacturers to consumers. However, traditional SCM systems are plagued by challenges such as lack of transparency, fraud, inefficiencies, and data silos. Blockchain technology offers a promising solution to these issues by providing a decentralized, immutable, and transparent ledger. This project report presents the development of a blockchain-based supply chain management system using Ethereum and a stack comprising React, Ganache, Solidity, Truffle, Metamask, and Web3.js. The system enables seamless registration of entities, transparent product ordering, and real-time tracking of product flow from raw material suppliers to retailers. Utilizing smart contracts written in Solidity, the system ensures secure and automated execution of supply chain processes. Key benefits include enhanced transparency, traceability, security, and efficiency in supply chain operations. Case studies such as IBM Food Trust and Maersk TradeLens illustrate successful implementations of blockchain in SCM, showcasing tangible improvements in traceability, efficiency, and customer trust. While challenges such as technological complexity, regulatory compliance, and cost implications persist, ongoing developments in blockchain infrastructure and industry standards are paving the way for wider adoption. The integration of blockchain with emerging technologies like IoT and AI holds promise for further innovation in supply chain management, enabling predictive analytics, autonomous decision-making, and adaptive supply chain optimization. This project demonstrates the transformative potential of blockchain in revolutionizing supply chain management, offering a scalable and sustainable solution to address the evolving needs of modern businesses in an increasingly interconnected world.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

In the dynamic landscape of global commerce, supply chain management (SCM) plays a crucial role in orchestrating the flow of raw materials, production, distribution, and delivery to meet consumer demands. Traditional SCM systems face numerous challenges such as opacity, inefficiency, and susceptibility to fraud and counterfeiting. As industries evolve and consumer expectations rise, innovative solutions to streamline and secure supply chain processes become increasingly urgent. Blockchain technology, with its principles of decentralization, immutability, and transparency, offers a transformative solution by providing a tamper-proof ledger that records every transaction in a distributed network, addressing key pain points and unlocking efficiencies in the SCM ecosystem. This project report explores the development and implementation of a blockchain-based SCM system using Ethereum, React, Ganache, Solidity, Truffle, Metamask, and Web3.js, focusing on seamless entity registration, transparent product tracking, and secure smart contract execution. Real-world case studies, such as IBM Food Trust and Maersk TradeLens, highlight the tangible benefits of blockchain, including enhanced traceability, fraud prevention, streamlined logistics, and increased consumer trust. The report also candidly addresses the technological, regulatory, and cost challenges of integrating blockchain into SCM, while envisioning future innovations through the convergence of blockchain with IoT and AI for predictive analytics, autonomous decision-making, and adaptive optimization. Ultimately, our project demonstrates the transformative potential of blockchain in SCM, paving the way for a more resilient and efficient future in the digital age.

1.1 Problem Statement

Supply chain management (SCM) faces significant challenges, including lack of transparency, susceptibility to fraud and counterfeiting, operational inefficiencies, fragmented data silos, and security vulnerabilities. Traditional SCM systems often fail to provide visibility into the product journey, leading to mistrust among stakeholders and consumers. Manual processes and siloed data hinder efficiency and timely decision-making, while centralized databases are prone to cyberattacks and data breaches. These issues necessitate a transformative solution that enhances transparency, traceability, security, and operational efficiency. Blockchain technology offers a promising remedy by providing a decentralized, immutable, and transparent ledger. This project aims to develop a blockchain-based SCM system using Ethereum and tools like React, Ganache, Solidity, Truffle, Metamask, and Web3.js to create a secure, efficient, and transparent supply chain network, addressing the critical challenges of traditional systems and paving the way for innovative SCM practices.

1.2 Existing system

Traditional supply chain management (SCM) systems rely heavily on centralized databases and legacy technologies, which often result in significant inefficiencies and vulnerabilities. These systems, such as Enterprise Resource Planning (ERP) systems, involve multiple stakeholders operating within their own silos, leading to fragmented data and poor real-time visibility. The lack of transparency in these systems makes it difficult to track the origin and journey of products, increasing the risk of fraud and counterfeiting, particularly in critical industries like pharmaceuticals and luxury goods. Moreover, manual processes and reliance on paperwork contribute to operational delays and errors, while centralized architectures are susceptible to cyberattacks and data breaches, compromising the integrity and security of supply chain information.

Advanced SCM solutions like IBM’s Sterling Supply Chain Suite attempt to address some of these issues by offering cloud-based analytics and enhanced data visibility. However, they still operate within a centralized framework that requires significant trust in third parties and can be expensive to implement and maintain. Overall, traditional SCM systems fall short in providing the transparency, traceability, and security needed for modern supply chains, highlighting the need for a transformative approach like blockchain technology to revolutionize supply chain operations.

1.3 Proposed System

The proposed system leverages blockchain technology to create a transparent, secure, and efficient supply chain management (SCM) system using the Ethereum platform and a tech stack including React, Ganache, Solidity, Truffle, Metamask, and Web3.js. By decentralizing the ledger and using immutable records, the system ensures transparency and trust among all stakeholders. It includes features such as entity registration, where each participant in the supply chain (raw material suppliers, manufacturers, distributors, and retailers) is assigned a unique identifier, and smart contracts that automate and enforce agreements, reducing errors and accelerating processes. Product ordering and tracking are made transparent, with every transaction from procurement to delivery recorded on the blockchain. Each product is assigned a QR code for real-time tracking and authenticity verification. The system enhances security by eliminating single points of failure and using cryptographic techniques to safeguard data. These improvements address the major issues of traditional SCM systems, such as lack of transparency, fraud, inefficiencies, and security vulnerabilities, thereby revolutionizing supply chain operations and building greater trust among consumers and partners.

1.4 Objective

The primary objective of this project is to develop a blockchain-based supply chain management (SCM) system that enhances transparency, traceability, security, and efficiency within the supply chain ecosystem. By utilizing the Ethereum blockchain and a suite of technologies including React, Ganache, Solidity, Truffle, Metamask, and Web3.js, the system aims to address critical challenges faced by traditional SCM systems, such as lack of visibility, fraud, operational inefficiencies, and data fragmentation. Specific objectives include implementing a decentralized ledger to record all transactions immutably, enabling real-time tracking of products from raw material suppliers to end consumers, automating processes through smart contracts to reduce errors and improve speed, ensuring secure and authorized access to supply chain data, and providing stakeholders with the tools to verify the authenticity and provenance of products. Ultimately, the project seeks to demonstrate how blockchain technology can revolutionize supply chain practices, fostering greater trust and collaboration among all participants.

Chapter 2

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

The proposed plan involves an in-depth review of existing research and developments in the field of blockchain-based supply chain management. Blockchain technology has been increasingly recognized for its potential to enhance transparency, security, and efficiency in supply chains. Various studies have demonstrated how blockchain can address common issues such as fraud, counterfeiting, and lack of visibility by providing a decentralized, immutable ledger for tracking products and transactions. The advantages of blockchain in improving traceability and accountability are essential for maintaining the integrity of supply chains. The application of blockchain ensures the authenticity of products, showcasing real-world implementations that validate the technology's effectiveness. Moreover, the use of smart contracts—self-executing contracts with terms directly written into code—has been explored extensively for automating processes and reducing the need for intermediaries. These contracts ensure that all supply chain activities are executed as per predefined rules, enhancing operational efficiency, and reducing the scope for errors.

VeChain (VET)

VeChain, established in 2015, leverages blockchain technology to revolutionize supply chain management by integrating smart contracts. This innovative approach involves tokenizing products and tracking each step of their journey through Radio-Frequency Identification (RFID) labels. These labels serve as digital footprints, meticulously recording the history and movement of products throughout the supply chain. This system ensures unparalleled transparency, allowing stakeholders to view comprehensive historical details of a product at any point in its lifecycle. The practical applications of VeChain span various industries including cold-chain logistics, where temperature-sensitive goods can be monitored in real-time, ensuring they remain within required temperature ranges. [2]

In the automotive industry, VeChain's technology can track the provenance of parts, providing an auditable trail that enhances trust and reduces fraud. The medical and healthcare sector benefits from ensuring the integrity and authenticity of pharmaceuticals, preventing counterfeit drugs from entering the supply chain.

WaltonChain (WTC)

WaltonChain, founded in 2016, employs proprietary RFID technology to offer an advanced solution for tracking objects through the supply chain. This system provides detailed information about the locations a product traverses, the entities that handle it, and the various steps involved in the supply chain process. WaltonChain's technology enhances traceability and accountability, crucial for industries where product history and integrity are paramount. By integrating RFID tags with blockchain, WaltonChain ensures that every interaction and movement of a product is recorded on an immutable ledger. [3]

This transparency not only reduces the risk of fraud but also improves operational efficiency by providing real-time data and analytics. Industries such as retail, logistics, and manufacturing can leverage WaltonChain to optimize their supply chain operations, ensure compliance with regulatory standards, and enhance customer trust by providing verifiable proof of product authenticity and provenance. [3]

Ambrosus (AMB)

Ambrosus, launched in 2017, is an innovative supply chain management system that integrates advanced Internet of Things (IoT) devices with Ethereum blockchain technology. This dual approach offers a robust and scalable solution for ensuring product safety and authenticity across various industries. At the core of Ambrosus's technology are high-tech sensors that monitor and record essential data such as temperature, humidity, and other environmental conditions that can affect product quality. These sensors are strategically placed throughout the supply chain to continuously collect data, which is then securely transmitted and stored on the blockchain, creating an immutable and transparent ledger that can be accessed and verified by all stakeholders. The use of Ethereum smart contracts adds another layer of security and automation to the process, ensuring that the data remains accurate and trustworthy.

In the food industry, Ambrosus addresses the critical need for monitoring perishables by ensuring that temperature-sensitive items are stored and transported within required temperature ranges. This real-time monitoring helps prevent spoilage and reduces food waste, ultimately leading to cost savings and increased consumer trust. By providing a transparent history of the product's journey from farm to table, Ambrosus enables consumers to verify the freshness and safety of their food. In the pharmaceutical industry, Ambrosus's technology ensures the integrity of medications throughout the supply chain by monitoring conditions crucial for maintaining the efficacy of many drugs. [1]

OriginTrail (TRAC)

OriginTrail, established in 2013, is a decentralized protocol dedicated to sharing data throughout the supply chain. It offers businesses an efficient and effective method to exchange information both internally and across borders. OriginTrail’s transparent approach aims to increase accountability and efficiency within supply chains. By facilitating seamless data exchange, OriginTrail helps organizations streamline operations, reduce costs, and enhance collaboration. The protocol ensures that data is not only securely shared but also remains interoperable across different systems and platforms. This interoperability is crucial for creating a cohesive and connected supply chain network. OriginTrail’s technology is particularly beneficial for industries such as agriculture, where traceability of products from farm to table is essential, and manufacturing, where it enhances visibility into the sourcing and production processes. By providing a robust framework for data sharing, OriginTrail fosters trust among supply chain participants and improves overall efficiency.

This protocol is particularly beneficial for industries such as agriculture, where traceability of products from farm to table is essential for ensuring food safety and quality. Additionally, in the manufacturing sector, OriginTrail enhances visibility into the sourcing and production processes, helping companies to comply with regulatory requirements and reduce risks associated with supply chain disruptions. OriginTrail's focus on transparency and data integrity fosters a more accountable and efficient supply chain, ultimately contributing to better business practices and increased consumer confidence in the products they purchase.

Tael

Tael, launched in 2017, is a pioneering project dedicated to ensuring the authenticity of products through the development of advanced anti-counterfeit solutions. Utilizing anti-counterfeit QR codes, Tael empowers consumers to verify the genuineness of products before purchase, ensuring they are valid and untampered. This technology is particularly popular in regions with lower regulatory standards, such as China, where counterfeit goods are a significant issue. By simply scanning the QR code on a product, consumers can access detailed information about its origin, journey, and handling, providing a transparent view of the product's lifecycle and building consumer trust. .

Chapter 3

SOFTWARE REQUIREMENT ANALYSIS

CHAPTER 3

SOFTWARE REQUIREMENT ANALYSIS

Systems engineering and software engineering are responsible for the methods used to identify the needs or prerequisites that must be satisfied for a new or modified product or project, taking into account the potentially conflicting needs of the various owners. These methods also include examining, listing, maintaining, and controlling software or system prerequisites.

3.1 Feasibility Study

The major goal of the feasibility study is to determine whether expanding the application is technically, operationally, and financially feasible. A project's viability is determined by its feasibility. A feasibility study is the procedure utilized to make that decision. Given infinite resources and time, all buildings are possible. The project's feasibility assessment entails the following:

- Technical Feasibility
- Social Feasibility
- Economic Feasibility

3.1.1 Technical Feasibility

The E-Chain app must be technically feasible, ensuring that it doesn't place heavy demands on the available technical resources. It should be designed with efficiency in mind, requiring minimal adjustments for implementation. The app should utilize technologies that are readily available and compatible with the existing infrastructure to minimize technical strain.

3.1.2 Social Feasibility

User acceptance is crucial for the success of E-Chain. The app should be user-friendly, with intuitive navigation and clear instructions to ensure that users feel comfortable and empowered while using it. Efforts should be made to educate and inform users about the benefits of the app, emphasizing its role in addressing food insecurity and waste within their communities.

3.1.3 Economic Feasibility

The economic impact of the E-Chain app on the organization should be carefully evaluated. Development costs should be justified, with a focus on utilizing cost-effective technologies and resources. Open-source or freely available technologies should be prioritized to keep development costs within budget constraints, with any necessary customized products being purchased judiciously.

Chapter 4

SYSTEM REQUIREMENT SPECIFICATION

CHAPTER 4

SYSTEM REQUIREMENT SPECIFICATION

The fundamental goal of a system requirement specification is to turn the objectives that a client has in mind into a document that has been authorized. System requirement specifications help both the customer and the developer understand exactly what they need from the created method and what level of expertise is needed to build the system. It consists of a number of components that attempt to describe the expected functionality needed by the client to fulfil their various users.

4.1 Functional Overview

- Our system uses blockchain technology for decentralized access control management.
- Fair Access introduces new transaction types for granting, obtaining, delegating, and revoking access.
- Transactions are verified by a distributed network, minimizing human error.
- Each transaction is securely recorded on the blockchain, ensuring transparency and tamper-proof records. Real - time processing of donation transactions.
- Errors are isolated to a single blockchain copy, making widespread errors nearly impossible.
- The system reduces administrative overhead and costs through automated processes.
- Fair Access provides a robust, secure solution for managing access rights.
- Blockchain technology enhances security by preventing unauthorized alterations.
- The system supports seamless access management for multiple users and entities.

4.2 Operating Environment

The operating environment requires the system to have minimum software and hardware requirements.

4.2.1 Software Requirements

- Operating System : Windows
- Tools used : Visual Studio Code, Metamask, Ganache

4.2.2 Hardware Requirements

- Processor : Intel Core i3
- Input device : Standard Keyboard and Mouse
- RAM : 4GB or above
- Hard Disk : 80GB or above
- Output device : Monitor.

4.3 Functional Requirements

A device's or its component feature's functional criteria determine it. A function is described as a collection of inputs, procedures, and outputs. System-specific outcomes are specified by functional requirements. Functional requirements shape a system's application architecture. The following list summarizes the functional requirements used in the project.

- **Business Rules:** To work effectively, the commercial center empowered by blockchain requires the various members to concede to the principles. To have the option to do that, specialists from every taking part element must have the option to comprehend the rationale actualized in the keen agreements.

- **Transaction, correction adjustments and cancellations:** Transactions on the blockchain are verified by a network of thousands or even millions of computers. This minimizes human involvement in the verification process, resulting in fewer errors and a more accurate record of information. Even if a computer on the network were to make a computational mistake, the error would only affect one copy of the blockchain. For the error to propagate across the rest of the blockchain, it would need to be replicated by at least 51% of the network’s computers, which is nearly impossible.

- **Administrative function:** Similarly to banks, governments can leverage blockchain technology for their key record keeping and verification functions, resulting in significant administrative cost savings.

- **Authorization levels:** To implement our model, we adapt blockchain technology into a decentralized access control manager. Unlike financial Bitcoin transactions, fair access introduces new types of transactions that are used to grant, obtain, delegate, and revoke access.

- **External interfaces:** The brilliant agreement is kind of code conjured by an outside customer application The API interface uncovered the administrations gave by the blockchain arrange spaces, for example, flexibly chain, vitality exchanging, and information commercial center.

4.4 Non-Functional Requirements

Non-functional requirements specify norms rather than specific behaviors that can be utilized to determine how a system operates. System quality characteristics are frequently referred to as non-functional requirements. The application's non-functional requirements are listed below.

- **Performance:**
  The system should be responsive and able to handle concurrent user interactions, ensuring smooth user experience even during peak usage times.

- **Reliability:**
  Data ownership reliability could be enhanced through a collaborative, decentralized approach, with information sourced from the communities that use it and stored on a blockchain. The emergence of blockchain as a shared, immutable ledger has the potential to provide the platform for this innovation.

- **Security:**
  Blockchain is a distributed database that provides a secure, yet transparent way to make record and verify any type of transaction. Blockchain eliminates the need for centralized control instead all transactions are decentralized, and verified by the blockchain database itself in the distributed ledger.

- **Scalability:**
  On Availability for Blockchain-Based Systems. Conceptual: Blockchain has as of late picked up force. Reliability properties, similar to accessibility, are basic for a large number of these applications, however the transactions offered by the blockchain innovation stay indistinct, particularly from an application point of view.

- **Maintainability:**
  The bigger the arrangement of members in the system, the more troublesome it is to make changes to the standard or convention of the appropriated record. The equivalent applies to keen agreements.

4.5 Performance Requirements

The performance requirements for the E-Chain app entail swift response times, with user interactions like posting donations or requesting assistance completing within 2 seconds. Real-time transaction processing, such as confirming donation requests, should be executed within 5 seconds. The system must accommodate a minimum of 100 simultaneous users without performance degradation, supporting at least 50 donation transactions per minute during peak periods. Location tracking of volunteers and donation pickups must maintain an accuracy of at least 10 meters. Notifications for donation requests and suspicious activities should be delivered within 1 second. Additionally, the system should ensure a minimum uptime of 99.5% and undergo load testing to verify performance under peak loads.

Chapter 5

SYSTEM DESIGN

CHAPTER 5

SYSTEM DESIGN

The design phase's requirement is to develop a solution to the issue outlined in the requirement document. The design of a system is perhaps the most challenging issue determining the software standards, and it has a significant impact on the following stages, particularly testing and maintenance. The design document is the product of this step. Two distinct phases of the design process are usually separated. System design and detailed design are process.

5.1 High Level Design

System design, also known as high-level design, seeks to define the modules that should be in the system, the identifications of these modules, and how they interact with one another to create the desired results. The principal data structures, file formats, output formats, main system modules, and their needs are all listed at the conclusion of the system design process.

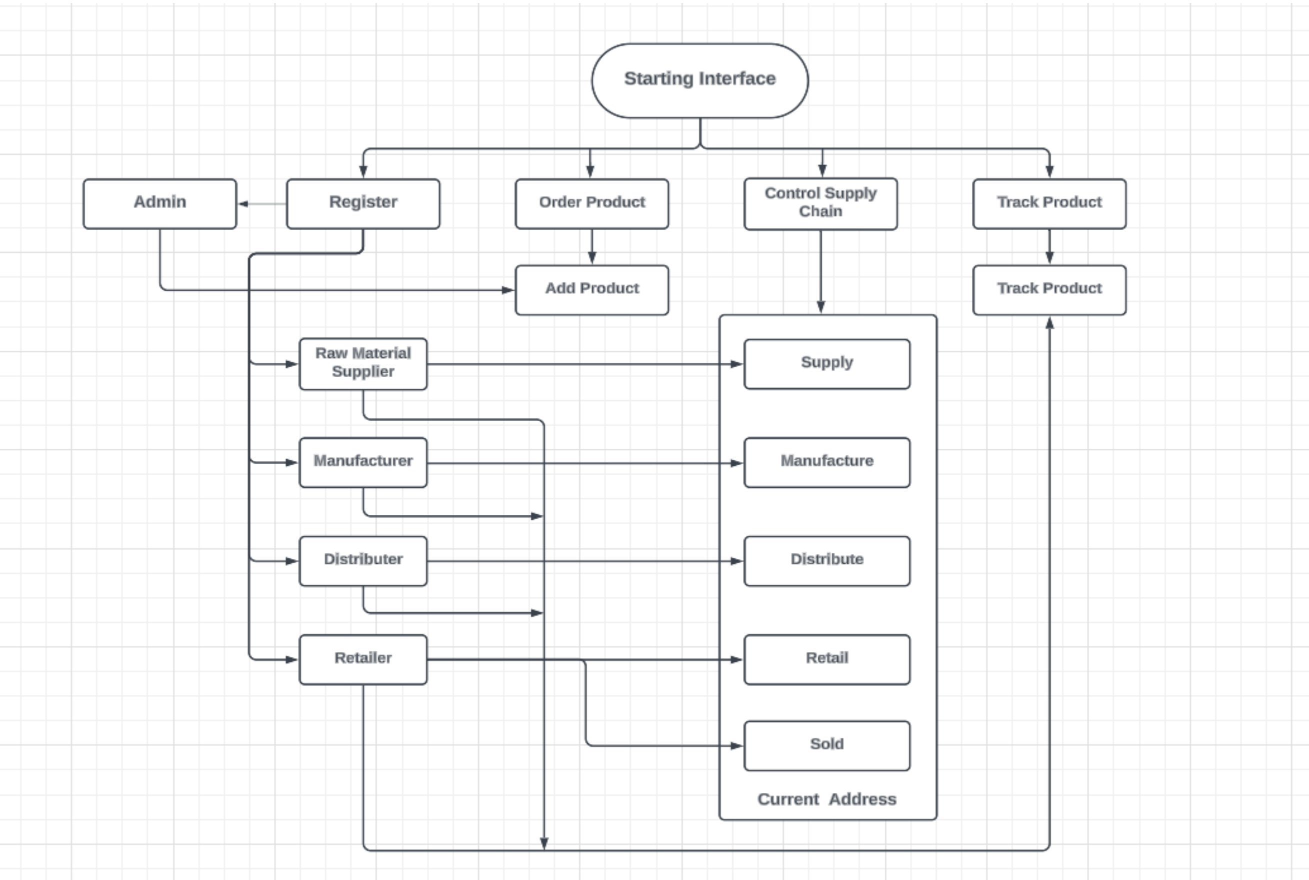


Figure 5.1: *System Architecture of E-Chain*

Frontend Design (React):

The E-Chain app's frontend is built using React, a robust JavaScript library known for its component-based architecture and efficient rendering, creating an intuitive and user-friendly interface. This design facilitates effortless interaction for stakeholders such as raw material suppliers, manufacturers, distributors, and retailers. Key features include a dashboard providing an overview of supply chain status, recent transactions, and alerts; registration forms for the admin to add new entities to the blockchain; an order management interface for placing, tracking, and managing orders; and product tracking using QR codes to scan and retrieve real-time product information throughout the supply chain.

Blockchain Layer:

The blockchain layer of the E-Chain app, implemented on the Ethereum platform, leverages its decentralized nature and smart contract capabilities to ensure transparency and security. Smart contracts, written in Solidity, form the core of the application, managing business logic and rules. Key contracts include the Entity Registration Contract for adding and verifying supply chain participants, the Product Order Contract for creating and tracking product orders, and the Tracking Contract for maintaining a record of product movements, ensuring traceability and accountability. Integration points include Metamask, which bridges the user's browser and the Ethereum blockchain for secure account management and transaction signing; Ganache, used for local development and testing by simulating an Ethereum environment; and QR code generation and scanning, which facilitate real-time product tracking by encoding product details into QR codes that can be scanned at various supply chain checkpoints.

Security Consideration:

Security considerations for the E-Chain app focus on data integrity, user authentication, and smart contract security. Data integrity is ensured by the immutable nature of blockchain transactions, which prevents unauthorized modifications. User authentication is managed via Metamask, along with secure backend practices to protect user accounts and interactions. Smart contract security is prioritized by conducting rigorous audits and thorough testing of the contracts to prevent vulnerabilities and exploits, ensuring the system remains robust and secure.

5.2 Detailed Design

The internal logic of each and every unit of the modules defined in the system design is established during the detailed design phase. More information on the data structures and algorithmic design of each module is stated in this phase. The logic of each unit is often definedusing a high-level design description language, which is separate from the target language in which the programme will finally be implemented.

5.2.1 Data Flow Diagram

A data flow diagram (DFD) illustrates the flow of data within a system, showing how information moves between processes, data stores, and external entities. Here's a high-level data flow diagram for the E-Chain app.

DFD Level-0

DFD Level 0 illustrates how data flows between external entities, processes, and data stores within the E- Chain app to facilitate Secured supply chain management with the organization and consumers.

DFD Level-1

In the DFD Level 1 for the E-Chain app, each process identified in the Level 0 is further broken down into subprocesses, along with their associated data flows and data stores.

1. **Registration Data Flow:**
    - **Data Flow:** Admin inputs the entity details into the system.
    - **Process:** The system verifies the input and registers the entity in the blockchain.
    - **Outcome:** Registered entity data is stored securely in the blockchain, accessible for supply chain interactions.

2. **Order Data Flow:**
    - **Data Flow:** Stakeholders input order details (product, quantity, delivery) into the system.
    - **Process:** The system validates the order and records it on the blockchain.
    - **Outcome:** Order data is stored securely, enabling efficient order management

and tracking throughout the supply chain.

3. **Status Update Data Flow:**

- **Data flow:** Stakeholders update the status of products (received, in transit, delivered) in the system.

- **Process:** The system validates the order and records it on the blockchain.

- **Outcome:** Order data is stored securely, enabling efficient order management and tracking throughout the supply chain.

4. **Tracking Request Data Flow:**

- **Data flow:** Users scan QR codes to request product tracking information.

- **Process:** The system retrieves product data from the blockchain based on the QR code.

- **Outcome:** Product tracking information is displayed to users, enabling them to monitor product movements and status.

In summary, the DFD provides a detailed breakdown of the processes identified, illustrating how data flows within each process to support secured supply chain management within the E-Chain app.

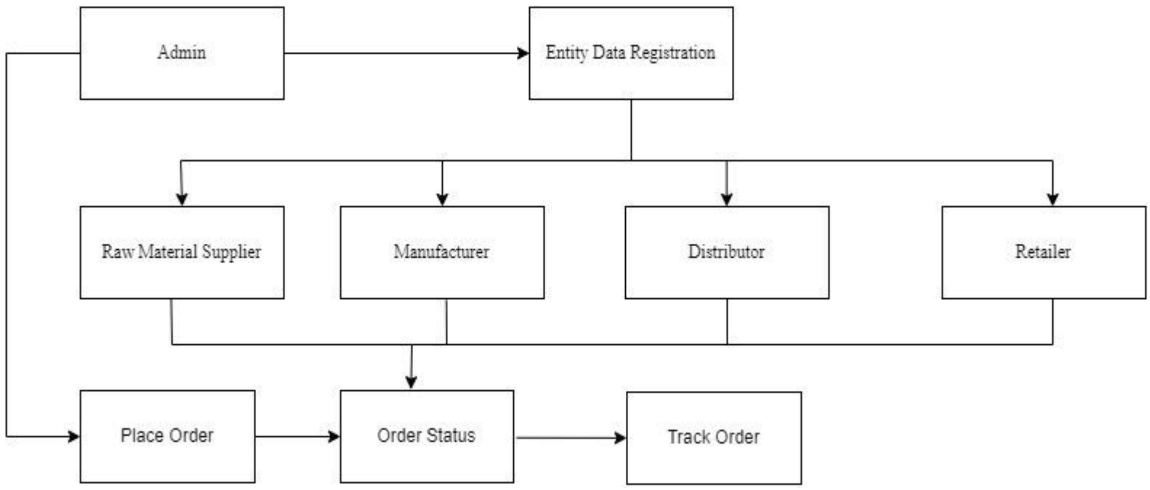


Figure 5.2: *DFD Diagram for E-Chain*

5.2.2 Use Case Diagram of E– Chain App

A use case diagram shows the way a user interacts with the system and shows the use case specifications. The many user types and the varied methods in which they interact with a system can be displayed in a case diagram of use.

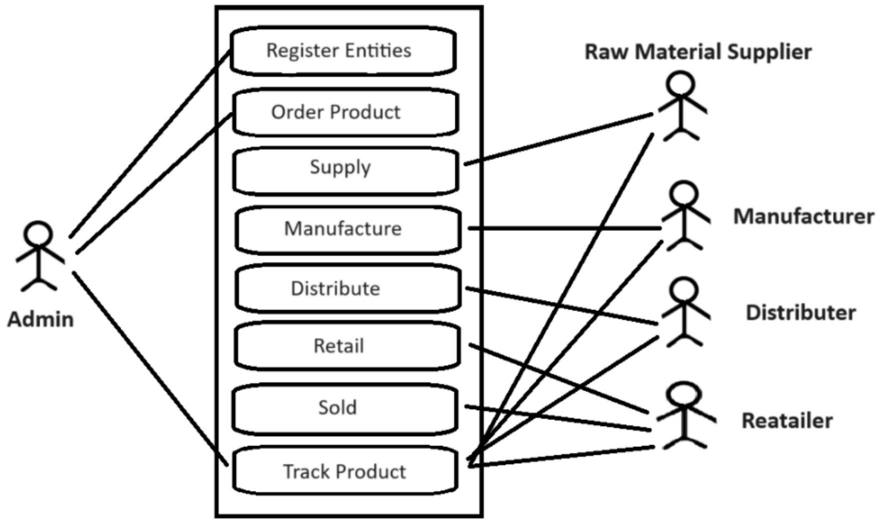


Figure 5.3: *Use Case Diagram for E-Chain App*

Figure 5.3 shows the use case diagram of E-Chain App. Admin and all Registered entities the functions and features are mapped here.

Chapter 6

SYSTEM IMPLEMENTATION

CHAPTER 6

SYSTEM IMPLEMENTATION

The blockchain-based supply chain management system is implemented using a combination of Ethereum, Solidity, Truffle, Ganache, Metamask, Web3.js, and React. Smart contracts, written in Solidity, handle entity registration, product ordering, and real-time tracking, ensuring transparency and security through an immutable ledger. The backend integrates these smart contracts using Web3.js, enabling seamless communication with the blockchain, while the frontend, developed with React, provides a user-friendly interface for administrators and users to manage and track supply chain activities. QR codes are generated and scanned to link physical products to their blockchain records, facilitating easy tracking and authenticity verification. The entire system is deployed and tested on the Ethereum test network using Truffle, with a focus on security and performance optimization to minimize gas costs and ensure efficient transaction processing. This implementation enhances transparency, traceability, and efficiency, addressing the key challenges of traditional supply chain management systems.

Additionally, the system's deployment involves extensive testing to ensure robustness and reliability under various conditions. Automated and manual test cases are executed to validate functionalities such as entity registration, order placement, status updates, and QR code tracking. Security measures, including secure handling of private keys and encrypted communication channels, are implemented to protect against unauthorized access and cyber threats. Performance optimizations are conducted to handle high transaction volumes and reduce latency. By integrating blockchain technology with modern web development frameworks, the system not only improves operational efficiency but also builds trust among supply chain stakeholders, paving the way for more resilient and transparent supply chain practice.

6.1 Technology Stack Selection

The technology selection for the blockchain-based supply chain management (SCM) system focuses on transparency, security, efficiency, and user-friendliness. Ethereum was chosen as the blockchain platform for its robust smart contract capabilities and widespread adoption, with Solidity as the smart contract language due to its ease of use and compatibility with the Ethereum Virtual Machine (EVM). Truffle serves as the development framework, offering tools for compiling, testing, and deploying smart contracts, while Ganache provides a personal blockchain for local development and testing. Metamask, a browser extension, bridges web applications and the Ethereum blockchain, enhancing usability. Web3.js, a comprehensive JavaScript library, facilitates blockchain interactions within web applications. React was selected for the frontend framework due to its component-based architecture and efficient rendering, creating a dynamic and responsive user interface. This tech stack collectively supports a secure, transparent, and efficient SCM system, addressing the limitations of traditional supply chain methods.

6.2 Development Environment Setup

The development environment for the E-Chain app, a blockchain-based supply chain management system, is meticulously set up to ensure efficient development, seamless integration, and robust testing. The primary blockchain platform used is Ethereum, selected for its well-established ecosystem and support for smart contract functionality. To simulate an Ethereum network locally, Ganache is utilized. Ganache provides a personal blockchain environment that allows developers to deploy and test smart contracts without the need for constant interaction with the live Ethereum network. This setup not only accelerates the development process but also enables comprehensive testing in a controlled environment, ensuring that all functionalities perform as expected before deployment to a live network.

For smart contract development, Truffle is employed as the framework of choice. Truffle offers a suite of tools that streamline the process of writing, compiling, testing, and deploying smart contracts written in Solidity. The integration of Truffle with Ganache creates a powerful development environment where contracts can be iteratively developed and tested. This setup is complemented by Node.js and NPM, which are used to manage project dependencies and run JavaScript code necessary for both backend and frontend development. Node.js, with its asynchronous event-driven architecture, is particularly well-suited for handling the real-time requirements of a blockchain-based application.

On the frontend, the E-Chain app utilizes React, a popular JavaScript library known for its component-based architecture and efficient rendering. React facilitates the creation of a dynamic and responsive user interface, essential for the interactive elements of the E-Chain app such as the admin dashboard and tracking interfaces. To bridge the frontend with the Ethereum blockchain, Web3.js is incorporated. Web3.js is a comprehensive JavaScript library that enables communication between the frontend application and the Ethereum network, allowing for seamless interactions with smart contracts. Additionally, Metamask, a browser extension, is used to manage users' Ethereum accounts and facilitate transactions. Metamask's user-friendly interface and security features make it an essential tool for interacting with the Ethereum blockchain directly from the web browser, enhancing the overall usability and security of the E-Chain app. Together, these tools and frameworks create a cohesive development environment that supports the building of a secure, transparent, and efficient supply chain management system.

6.3 Backend Development

The backend development of the E-Chain app is designed to create a robust, secure, and efficient system that effectively interacts with the Ethereum blockchain. At the core of the backend are the smart contracts written in Solidity, which handle essential functionalities such as entity registration, product ordering, and tracking. These smart contracts define the rules and logic governing the supply chain processes, ensuring that all transactions are transparent, immutable, and secure. Key contracts include the Entity Registration Contract, which manages the registration of all supply chain participants, the Product Order Contract, which facilitates the creation and management of product orders, and the Tracking Contract, which records the movement of products through the supply chain, ensuring each product's journey is transparent and traceable.

Integration with the Ethereum blockchain is achieved using Web3.js, a powerful JavaScript library that allows the frontend to interact seamlessly with the blockchain. Web3.js handles essential operations such as sending transactions to the blockchain, querying smart contract data, and listening for events emitted by the contracts. This integration enables functionalities such as placing orders, updating statuses, and tracking products to be executed seamlessly, ensuring real-time synchronization between the user interface and the blockchain. The backend leverages Web3.js to bridge the gap between the user-facing frontend and the decentralized Ethereum network.

6.4 Integration and Testing

Integration and testing are critical phases in the development lifecycle of the E-Chain app, ensuring that all components work seamlessly together and meet the project requirements. The integration process involves combining the frontend, backend, and blockchain components to create a cohesive and functional application. This includes integrating the user interface developed with React with the backend Node.js server, which interacts with the Ethereum blockchain using Web3.js. Through this integration, users can access the E-Chain app through their web browsers, interact with smart contracts deployed on the Ethereum network, and track the movement of products across the supply chain in real-time. Additionally, integration testing is conducted to verify that the various modules and components of the application communicate correctly, handle data securely, and respond appropriately to user inputs. This ensures that the application functions as expected across different devices, browsers, and network conditions.

Once integration is complete, rigorous testing is conducted to validate the functionality, performance, and security of the E-Chain app. Testing encompasses a range of methodologies, including unit testing, integration testing, and end-to-end testing, to identify and address any issues or defects. Unit tests are performed on individual components, such as smart contracts, API endpoints, and React components, to verify their behavior and functionality in isolation. Integration tests validate the interactions between different modules and components, ensuring that data flows correctly between the frontend, backend, and blockchain layers. End-to-end tests simulate user interactions with the application from start to finish, validating the entire workflow of placing orders, updating statuses, and tracking products across the supply chain.

In addition to functional testing, security testing is conducted to identify and mitigate potential vulnerabilities and threats. This includes testing for common security risks such as input validation, authentication, authorization, and data encryption. Penetration testing may also be performed to assess the resilience of the application against malicious attacks and intrusion attempts. Furthermore, performance testing is conducted to evaluate the responsiveness, scalability, and reliability of the E-Chain app under various load conditions. This ensures that the application can handle a high volume of concurrent users and transactions without experiencing performance degradation or downtime. By conducting thorough testing throughout the development process, the E-Chain app can deliver a secure, reliable, and user-friendly experience to its stakeholders, fostering trust and confidence in its ability to streamline supply chain management processes effectively.

6.5 Deployment

The deployment of the E-Chain app to a local environment involves configuring and running all components on a local machine, enabling developers to thoroughly test the application in a controlled setting before considering broader deployment. The deployment process begins with the finalization of the smart contracts, which are deployed to the Ethereum mainnet to ensure immutability, security, and global accessibility. This involves careful management of gas fees and ensuring that the contracts are optimized for performance and cost-efficiency. The backend Node.js server, which handles business logic and interacts with the Ethereum blockchain via Web3.js, is deployed on a reliable cloud platform such as AWS or Heroku. This ensures high availability, scalability, and robust performance. Additionally, the server configuration includes setting up secure connections using HTTPS, and integrating a robust database system for managing auxiliary data such as user profiles and session data.

On the frontend, the React application is built and deployed on a local host. The frontend is configured to interact seamlessly with the backend server and the Ethereum blockchain, ensuring real-time updates and interactions. Metamask integration ensures that users can securely manage their Ethereum accounts and perform transactions directly from their browsers. Post-deployment, continuous monitoring and maintenance are crucial to ensure the system remains secure, performs well under varying loads, and can be updated as needed. This includes setting up automated monitoring tools to track application performance, error logging, and security alerts, ensuring that any issues can be promptly addressed to maintain the integrity and reliability of the E-Chain app in a live production environment.

6.6 Maintenance and Updates

Maintenance and updates are critical components of the E-Chain app's lifecycle, ensuring its long-term reliability, security, and efficiency. Regular maintenance involves monitoring the application's performance, identifying and fixing bugs, and ensuring that all components remain up-to-date with the latest security patches and enhancements. This includes monitoring the smart contracts deployed on the local blockchain using tools like Ganache, as well as the backend server and frontend application. Logs and analytics are continuously reviewed to detect any anomalies or performance issues, allowing for proactive resolution before they impact users.

Updates to the E-Chain app are managed through a structured development workflow, incorporating version control with Git to track changes and facilitate collaboration among developers.

Chapter 7

TESTING

CHAPTER 7

TESTING

Software testing is that the accustomed method helps to assess the correctness, completeness, protection and consistency of the built computer applications. This includes the way the appliance or scheme is executed with the intention of finding errors.

The first step in identifying program errors is testing. It is evident that the effectiveness of testing relies significantly on how well the test cases reveal program flaws. Testing is crucial because it is the stage where any remaining errors from previous processes must be detected, as the code is the only product that can be executed and have its actual behavior observed. A series of test cases are used to run the programme that needs to be verified, and the software's performance for the test cases is then examined to determine whether the programming is unmistakably executed.

7.1 Testing Methodologies

7.1.1 Unit Testing: It is the primary step of the test; the varied modules or components are individually tested, often dispensed by the coder himself. Unit testing for our supply chain management system involves verifying entity registration, supply chain control, and product tracking. We validate that owners can register suppliers, manufacturers, distributors, and retailers, and that only registered entities can manage the supply chain. Additionally, we ensure consumers can track products accurately using QR codes, reflecting real-time supply chain updates.

7.1.2 Integration Testing: Most Test the interaction between different modules, such as blockchain nodes, smart contracts, and supply chain management interfaces. Ensure seamless data flow and proper functioning of integrated components.

7.1.3 System Testing: Here Evaluate the entire system’s performance, including the blockchain network and supply chain management application. Ensure the system meets specified requirements and functions correctly under various conditions.

7.1.4 Acceptance Testing: It's It is conducted using real client data to showcase satisfactory software performance, testing against specifications to ensure meeting business requirements and user expectations.

7.2 Testing Criteria

Table 7.2.1: Test cases for registration screen

	Test Cases	Expected Output	Observed Output	Result
1	Register Raw Material Supplier	Supplier successfully added to blockchain with unique identifier	As expected	Pass
2	Register Manufacturer	Manufacturer registered and recorded on blockchain with unique ID	As expected	Pass
3	Register Distributor	Distributor registration completed, recorded on blockchain	As expected	Pass
4	Register Retailer	Retailer registration successful, reflected on blockchain	As expected	Pass
5	Manage Supply Chain by Registered Entities	Actions by registered entities recorded accurately on blockchain.	As expected	Pass
6	Prevent Unauthorized Access	Unauthorized attempts rejected, system logs unauthorized access.	As expected	Pass
7	Track Product Using QR Code	QR code scan displays accurate product journey and status.	As expected	Pass
8	Verify Product Information Accuracy	Product tracking interface shows up-to-date, accurate information	As expected	Pass

Chapter 8

RESULTS AND DISCUSSION

CHAPTER 8
RESULT AND DISCUSSION

8.1 Landing Page

Upon launching the E-Chain, users are greeted by an intuitive and visually appealing landing page. This page serves as the gateway to the entire supply chain management system.

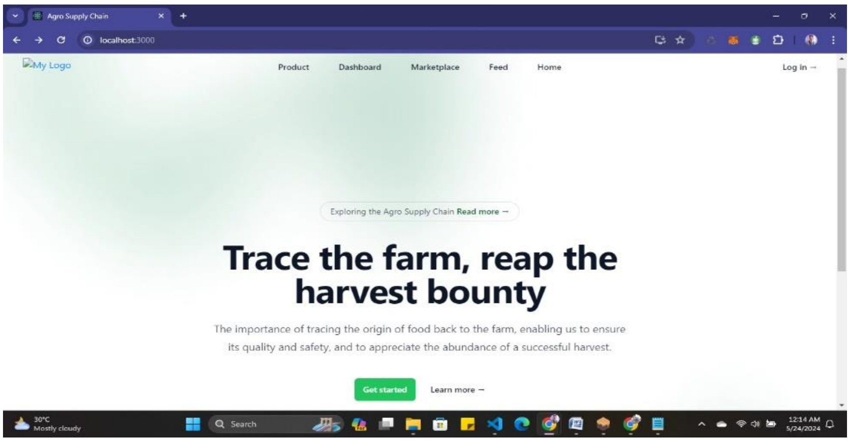


Figure 8.1: *Landing Page*

8.2 Home Page

The Home page of the E-Chain app is the central hub for managing the supply chain. It features sections for "Entity Registration" where admins can add entities, "Order Management" for placing and managing orders, "Status Update" for tracking product statuses, and "Product Tracking" using QR codes.

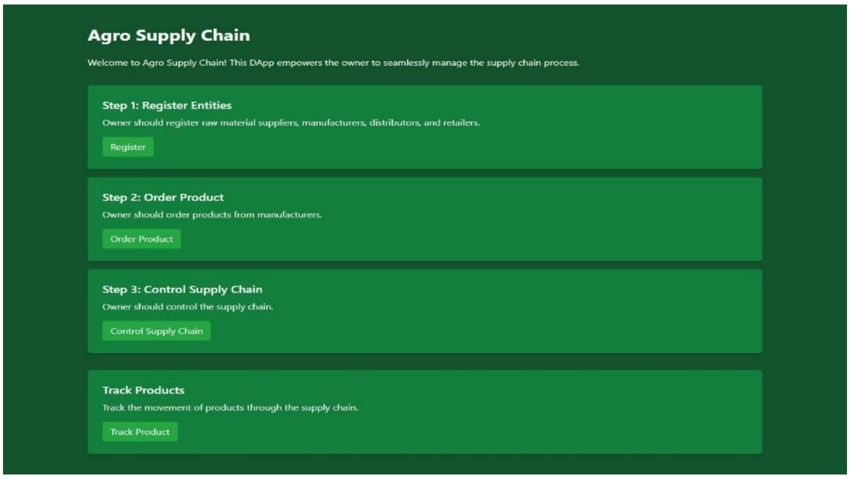


Figure 8.2: *Home Page*

8.3 Registering Entities

Register Entities section helps the admin to register various supply chain participants such as raw material suppliers, manufacturers, distributors, and retailers. By inputting essential details and assigning unique IDs to each entity. Admin ensures that all participants are securely added to the blockchain.

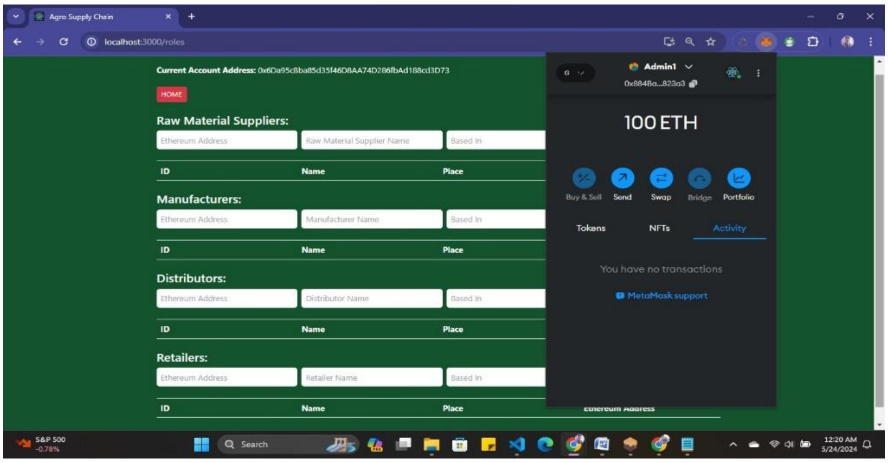


Figure 8.3: *Registering Entities*

8.4 Order Product

Using the Order Product feature, users can initiate new orders. Once an order is placed, it is recorded on the blockchain, ensuring transparency and traceability throughout the supply chain. This process helps maintain an organized and transparent supply chain network.

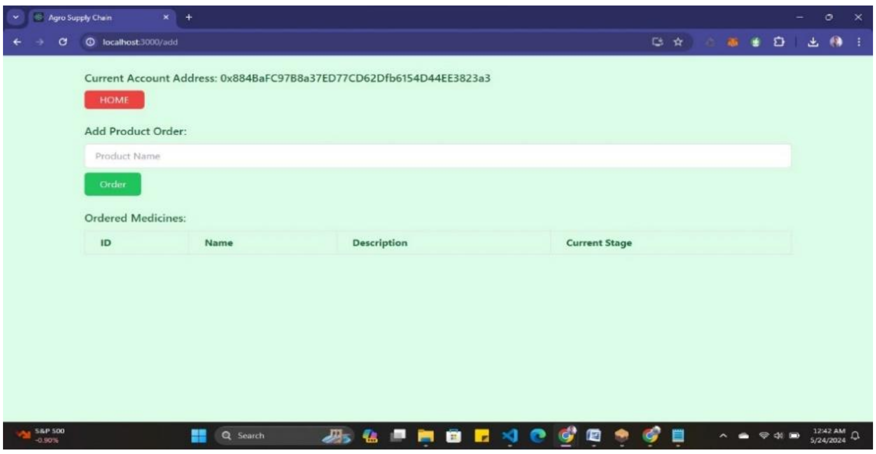


Figure 8.4: *Order Product*

8.5 Control Supply Chain

In the Control Supply Chain section, only registered participants can manage the movement of products. Registered raw material suppliers initiate the process by supplying products to registered manufacturers. Each stage requires participants to update the status of products, ensuring real-time visibility and accountability.

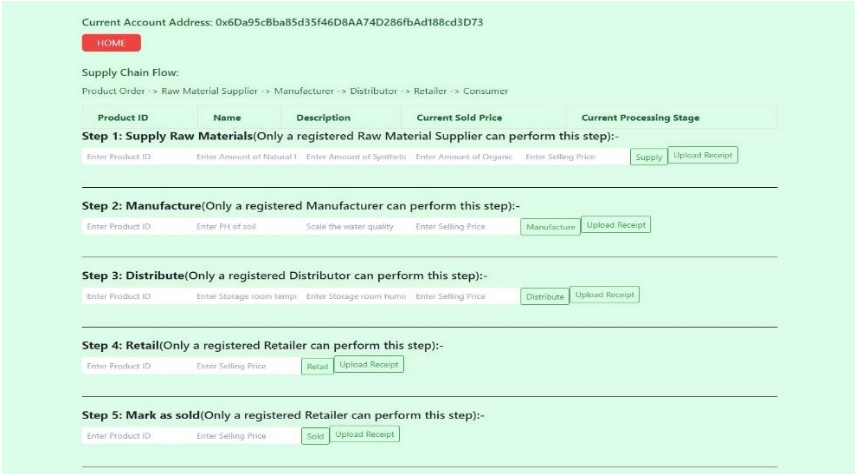


Figure 8.5: *Control Supply Chain*

8.6 Track Products

Track Products feature helps users to retrieve essential information about each product's current stage in the supply chain using its unique product ID. Users can generate a QR code associated with each product ID for quick access to relevant details, streamlining the tracking process.

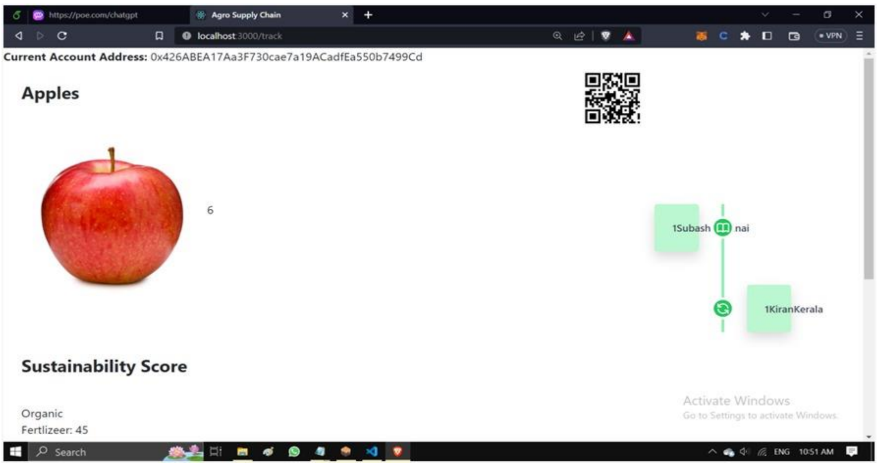


Figure 8.6: *Track Products*

Chapter 9

CONCLUSION
AND
FUTURE SCOPE

CONCLUSION AND FUTURE SCOPE

The E-Chain app effectively leverages blockchain technology to enhance transparency, security, and efficiency in supply chain management. By integrating Ethereum smart contracts, a Node.js backend, and a React frontend, the app provides a robust solution for tracking products and managing supply chain entities. The deployment on a local blockchain environment with tools like Ganache ensures a controlled and efficient development process. Looking ahead, the future scope of E-Chain includes scaling the application to a live Ethereum network, incorporating advanced features such as predictive analytics for supply chain optimization, and integrating IoT devices for real-time data collection and enhanced tracking accuracy. Additionally, exploring interoperability with other blockchain platforms and enhancing user interface capabilities will further solidify E-Chain's position as a comprehensive tool for modern supply chain management.

REFERENCES

REFERENCES

[**1**]. Mansfield-Devine S. Beyond Bitcoin: using blockchain technology to provide assurance in the commercial world. Computer Fraud & Security, 2017; 5: 14-18.

[**2**]. Jangga R, Ali N, Ismail M, Sahari N. Effect of Environmental Uncertainty and Supply Chain Flexibility Towards Supply Chain Innovation: An exploratory Study. Procedia Econ. Financ. 2015; 31: 262-268.

[**3**] Chen G, Xu B, Lu M, Chen NS. Exploring blockchain technology and its potential applications for education. Smart Learning Environments, 2018; 5(1): 1-10.

[4] Lu Y. Blockchain and the related issues: A review of current research topics, J. Manag. Anal., 2018; 5: 231-255.

[5] Wang Z, Wang T, Hu H, Gong J, Ren X, Xiao Q. Blockchain-based framework for improving supply chain traceability and information sharing in precast construction. Autom. Constr., 2020; 111: 103063.

[6] Wang Y, Singgih M, Wang J, Rit M. Making sense of blockchain technology: How will it transform supply chains? Int. J. Prod. Econ., 2019; 211: 221-236.

[7] Huddiniah ER, Er M. Product Variety, Supply Chain Complexity and the Needs for Information Technology: A Framework Based on Literature Review, Operations and Supply Chain Management: An International Journal, 2019; 12(4): 245-255.

DrillBit

The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

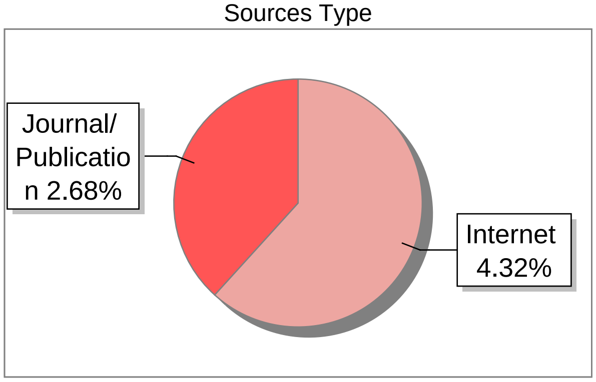
Author Name	RAJATH ACHARI, RAKSHIT SHETTY, RITIK RANJAN
Title	E-CHAIN: BLOCKCHAIN BASED SECURED SUPPLY CHAIN MANAGEMENT Submitted
Paper/Submission ID	1861306
Submitted by	library@sitmng.ac.in
Submission Date	2024-05-24 12:19:52
Total Pages, Total Words	47, 8185
Document type	Project Work

Result Information

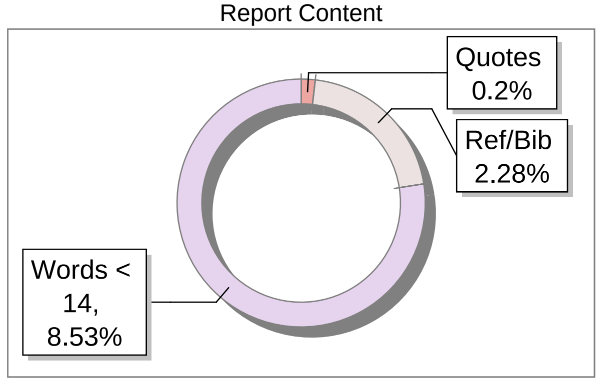
Similarity **7 %**

1 10 20 30 40 50 60 70 80 90

Sources Type



Report Content



Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Source: Excluded < 14 Words	Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

7	18	A	A-Satisfactory (0-10%) B-Upgrade (11-40%) C-Poor (41-60%) D-Unacceptable (61-100%)
SIMILARITY %	MATCHED SOURCES	GRADE	

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	www.linkedin.com	1	Internet Data
2	qdoc.tips	1	Internet Data
3	www.ilink-digital.com	1	Internet Data
4	information-science-engineering.newhorizoncollegeofengineering.in	1	Publication
5	journal.literasisainsnusantara.com	1	Publication
6	qdoc.tips	1	Internet Data
7	naac.mictech.edu.in	1	Publication
8	dochero.tips	<1	Internet Data
9	journal.oscm-forum.org	<1	Publication
10	www.frontiersin.org	<1	Internet Data
11	bgsit.ac.in	<1	Publication
12	www.linkedin.com	<1	Internet Data
13	www.deskera.com	<1	Internet Data
14	www.linkedin.com	<1	Internet Data

15	naac.atme.in	<1	Publication
16	www.deskera.com	<1	Internet Data
18	www.jucs.org	<1	Publication
20	www.mdpi.com	<1	Internet Data