UNDERGRADUATE FINAL YEAR PROJECT REPORT

Department of Telecommunications Engineering
NED University of Engineering and Technology

ADAPTATION OF SMART CONTRACT IN SUPPLY CHAIN MANAGEMENT USING IoT

Group Number: 01 **Batch: 2018**

Group Member Names:

Fatima Haider Naqvi TC-18020
Binish Haseeb TC-18024
Soomal Qureshi TC-18016
Namra Khan TC-18006

Approved by

…….

Dr. Sundus Ali
Assistant Professor of Department of Telecommunications NEDUET
Project Advisor

Sir Taha Sajid
5G security lead, Comcast, USA
Industrial Advisor

AUTHOR'S DECLARATION

We declare that we are the sole authors of this project. It is the actual copy of the project that was accepted by our advisor(s) including any necessary revisions. We also grant NED University of Engineering and Technology permission to reproduce and distribute electronic or paper copies of this project.

Signature and Date	Signature and Date	Signature and Date	Signature and Date
.....	.....	.....	.....
Fatima Haider Naqvi TC-18020 naqvi4105572@cloud.neduet.edu.pk	Binish Haseeb TC-18024 haseeb4102235@cloud.neduet.edu.pk	Soomal Qureshi TC-18016 Soomal4102751@cloud.neduet.edu.pk	Namra Khan TC-18006 khan4130114@clooud.neduet.edu

STATEMENT OF CONTRIBUTIONS

1. **Ms. Fatima Haider Naqvi** has developed a code for smart contract and has done coding in solidity programming also she has done her research in block chain and smart contract development.

2. **Ms. Binish Haseeb** has done her research in IoT (internet of things) she sent IoT sensor data (temperature and humidity) from Node MCU to firebase cloud server to store all data in cloud. She has also designed an User Interface (UI) for front-end development of the project using Figma designing tool.

3. **Ms. Namra Khan** has done backend integration of smart contract with IoT module where she has used database server (firebase) to interconnect IoT data, smart contract and front-end of the project.

4. **Ms. Soomal Qureshi** has done her research in block chain as well in IoT and she has done frontend app development work to analyse the integration of IoT and Blockchain and to check the conditions of shipment of vaccine from seller to customer using web application developed in react.js.

5. All authors has equally contributed to write the final year design project (FYDP) report.

EXECUTIVE SUMMARY

In today's time, new viruses and diseases are hitting the world and have a significant impact on practically every aspect of people's life, as well as many economic sectors and global regions. In a cases like these, vaccination, injections and medicines may be considered essential, but its quality will be determined by the availability and transparency of the distribution process, which will be shared among the stakeholders.

Numerous complicated problems affect the supply chain, such as blind spots, inadequate data compatibility and predictability between parties, high monitoring expenses, and dynamic coordination between numerous parties. To change supply chain operations between suppliers, merchants, and customers, it is anticipated and preferred that the block chain will be able to replace complicated, error-prone processes with straightforward smart contracts. As technology develops and matures, it will be used to enable new supply chain business models and offer chances for cross-organizational collaboration. Different distributed ledger technologies (DLTs), including block chain, offer an open, public, unchangeable, tamper-proof system with significant potential. In this study, we suggest a block chain-based technique to increase the safety and accountability of shipment tracing. For more than a decade, the block chain has been regarded as a technique in which a distributed database keeps track of all transactions that take place in a peer-to-peer network. It is regarded as a distributed computing model that masterfully addresses the centralized party trust problem. As a result, many nodes cooperate in a distributed manner in a block chain network to secure and maintain a set of shared transaction records without the need for a centralized authority.

Smart contracts have been built to track vaccination of supply distribution by the use of IoT systems. The proposed approach will aid in the creation of a tamper-proof and secure environment for viruses and disease immunization vial distribution. The proposed solution uses proof of delivery as a consensus mechanism. We constructed and evaluated the suggested solution on the Ethereum test network. The suggested framework has promising performance and scalability.

ACKNOWLEDGMENTS

First and foremost, All praise and gratitude to Almighty Allah, by the grace of whom we are able to reach here at this sd like to express our special thanks of gratitude to our project supervisor Dr Sundus Ali as well as our project evaluator Sir Muneeb Ahmed who gave us the golden opportunity to do this project on the Adaptation of Smart contract in supply chain in IoT, which also helped us in doing research on this new technology integration topic and we came to explore and learn about many new things which would help us in our professional careers in ICT.

We would like to extend our gratitude to our Co-supervisor Sir Taha Sajid who has guided us tremendously in this project not only by sharing his knowledge but also has continuously motivated us to work consistently on this project. Because of their prompt and continuous support and guidance, we accomplished our tasks.

Also we would like to thank our parents for their continuous encouragement and unwavering support and belief in our abilities.

Last but not the least, the guidance of all other faculty members of the Telecommunications Engineering Department has also been of much assistance. We would also like to thank the whole Final Year Project Committee for their cooperation throughou*t* the semester.

TABLE OF CONTENTS

AUTHOR'S DECLARATION ii

STATEMENT OF CONTRIBUTIONS iii

EXECUTIVE SUMMARY iv

ACKNOWLEDGMENTS v

TABLE OF CONTENTS vi

LIST OF FIGURES x

LIST OF ABBREVIATIONS xi

UNITED NATIONS SUSTAINABLE DEVELOPMENT GOALS xii

SIMILARITY INDEX REPORT xiii

CHAPTER 1 1

INTRODUCTION 1

1.1 Background Information 2

1.2 Significance and Motivation 3

1.3 Aims and Objectives 4

1.4 Methodology 4

1.4.1 Node MCU: 4

1.4.2 DHT11 Sensor: 4

1.4.3 IoT Module: 4

1.4.4 Firebase Database: 5

1.4.5 Design Flow For IoT Module 6

1.4.6 Smart Contract: 7

1.4.7 Design Flow For Smart Contract: 8

1.4.8 Integration of Smart Contract And IoT Module: 9

1.4.9 Flowchart of Integration of IoT Module And Smart Contract: 10

1.4.10 Front End: 11

1.5 Report Outline 12

1.5.1 Scope of the Project 12

CHAPTER 2 14

LITERATURE REVIEW 14

2.1 Introduction 14

2.2 Smart Contract 14

2.2.1 IoT Enabled Container 14

2.2.2 Ethereum Smart Contract (BlockChain Applications) 15

2.2.3 Integration of IoT and Blockchain 15

2.3 Supply Chain 16

2.3.1 Supply Chain Management and Cold Chain 16

2.4 Blockchain Applications 17

2.4.1 Blockchain and Smart Contract 17

2.4.2 Blockchain, IoT and 4th Industrial Revolution (Industry 4.0) 18

2.5 Conclusion 19

CHAPTER 3 20

SYSTEM DESIGN 20

3.1 Introduction 20

3.2 Hardware 21

3.2.1 NodeMCU ESP8266 Microcontroller 21

3.2.2 DHT11 Humidity and Temperature Sensor 22

3.2.3 IoT Enabled Container 23

3.3 Softwares 23

3.3.1 MetaMask 23

3.3.2 Etherscan 24

3.3.3 FireBase 25

3.3.4 Arduino IDE 25

3.3.5 Remix IDE 26

3.3.6 Figma Design Tool 26

3.4 Languages and Libraries 26

3.4.1 Web3.js 26

3.4.2 Solidity Programming Language 26

3.4.3 React. Js 27

3.4.4 Node.Js 28

3.5 Conclusion 28

CHAPTER 4 29

RESULTS 29

4.1 IoT module: 29

4.2 Firebase Data 30

4.3 Smart contract Output: 31

4.4 Integration of Smart Contract and IoT Module Via Frontend 32

CHAPTER 5 38

SUMMARY 38

5.1 CONCLUSION 38

5.2 RECOMMENDATIONS FOR FUTURE WORK 39

REFERENCES 40

LIST OF FIGURES

Figure 1: IoT Module Connection Diagram 5

Figure 2: Firebase Project 5

Figure 3: Flow diagram 20

Figure 4: Node MCU 22

Figure 5:DHT11 Sensor 22

Figure 11: Frontend Landing page 32

Figure 12: Seller's Log-in 33

Figure 13: Order is created 33

Figure 14: Seller's active Order Id list 34

Figure 15: User side with refund claim 34

Figure 16: Seller side with refund 34

Figure 17: Amount refunded 35

Figure 18: MetaMask wallet after refund 35

Figure 19: User side with successful case 36

Figure 20: Seller side with successful case 36

Figure 21: Order Approved 36

Figure 22: Data saved in decentralized storage 37

LIST OF ABBREVIATIONS

IoT: Internet of Things

BDA: Big Data analytics

WEF: World Economic Forum

AI: Artificial Intelligence

SCM: Supply chain management

SC: Smart Contract

DLTs: Distributed ledgers

UNITED NATIONS SUSTAINABLE DEVELOPMENT GOALS

The Sustainable Development Goals (SDGs) are the blueprint to achieve a better and more sustainable future for all. They address the global challenges we face, including poverty, inequality, climate change, environmental degradation, peace and justice. There are a total of 17 SDGs as mentioned below. Check the appropriate SDGs related to the project.

□ No Poverty

□ Zero Hunger

□ Good Health and Well being ✓

□ Quality Education

□ Gender Equality

□ Clean Water and Sanitation

□ Affordable and Clean Energy

□ Decent Work and Economic Growth

□ Industry, Innovation and Infrastructure

□ Reduced Inequalities

□ Sustainable Cities and Communities ✓

□ Responsible Consumption and Production ✓

□ Climate Action

□ Life Below Water

□ Life on Land ✓

□ Peace and Justice and Strong Institutions

□ Partnerships to Achieve the Goals

SIMILARITY INDEX REPORT

Following students have compiled the final year report on the topic given below for partial fulfillment of the requirement for Bachelor's degree in Telecommunication Engineering.

Project Title: ADAPTATION OF SMART CONTRACT IN SUPPLY CHAIN MANAGEMENT USING IoT

S. No.	Student Name	Seat Number
1	Fatima Haider Naqvi	TC-18020
2	Binish Haseeb	TC-18024
3	Soomal Qureshi	TC-18016
4	Namra Khan	TC-18006

This is to certify that the Plagiarism test was conducted on complete report, and overall similarity index was found to be less than 20%, with maximum 5% from single source, as required.

Signature and Date

.....

Dr: Sundus Ali

Plagiarism Test must be conducted on the project report by the project advisor using Turnitin Software, or any other software approved by the university. Project Advisor is required to sign the Plagiarism Certificate if the similarity is found as per rules defined in the guidelines. Plagiarism reports must be included (only the first and last page) generated by Turnitin. A sample of the plagiarism test report is provided with the FYDP template package.

ORIGINALITY REPORT

19%	11%	9%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	4%
2	www.ncbi.nlm.nih.gov Internet Source	2%
3	Yue Wu, Junxiang Li, Jiru Zhou, Shichang Luo, Liwei Song. "Evolution Process and Supply Chain Adaptation of Smart Contracts in Blockchain", Journal of Mathematics, 2022 Publication	1%
4	dokumen.pub Internet Source	1%
5	Submitted to Kaplan Professional Student Paper	1%
6	med.neduet.edu.pk Internet Source	1%
7	"Blockchain Technology for Industry 4.0", Springer Science and Business Media LLC, 2020 Publication	<1%

51	research.library.mun.ca Internet Source	<1%
52	www.journaltocs.ac.uk Internet Source	<1%
53	www.x-mol.com Internet Source	<1%
54	Clement Nartey, Eric Tutu Tchao, James Dzisi Gadze, Eliel Keelson, Griffith Selorm Klogo, Benjamin Kommey, Kwasi Diawuo. "On Blockchain and IoT Integration Platforms: Current Implementation Challenges and Future Perspectives", Wireless Communications and Mobile Computing, 2021 Publication	<1%

Exclude quotes	On	Exclude matches	< 1 words
Exclude bibliography	On		

CHAPTER 1

INTRODUCTION

Supply chain management has a tremendous impact on the global economy in today's market. The transfer of goods from producer to consumer is a common definition of SCM. It involves the manufacturer, distributor, and retailer, and is divided into several phases, beginning with the delivery of raw materials and ending with the client. It's a worldwide process that involves sourcing components from a single location, packaging them, and shipping them all over the world. With each passing day, technological advancements and their benefits invade more of our lives; more government, economic, and social functions, among other things, are accessible via the internet. We bank online, shop online, and log into applications and services to create our digital selves and send data back and forth. Because of its decentralized, peer-to-peer transaction, distributed consensus, and anonymity qualities, block chain technology has become a popular research topic in the recent decade. Regulatory issues and technical hurdles are overshadowed by blockchain technology [20].

A smart contract is a simply a collection of self-verifying, self-executing, and tamper-proof programs. Smart contracts incorporates blockchain technology and are capable to perform tasks in real time at a low cost and with a higher level of security [20].

In blockchain technology, smart contracts are implemented by digital nodes rather than people, and so do not require human intervention to remain in effect. As a result, smart contracts may run independently and precisely as planned, with no risk of downtime, censorship, fraud, or third-party interference. Smart contracts eliminate the need for a middleman to oversee transactions, allowing equitable transactions to take place directly between participants. Block chain has the ability to solve the single point of failure problem in cloud-based IoT infrastructure, resulting in a more resilient ecosystem. IoT solutions can enable communications between devices in an IoT network by leveraging smart contracts, which model agreement between the two devices, by leveraging blockchain [20].

1.1 Background Information

Bartering was the mechanism of transacting between individuals in the early commerce; this system had been in existence for centuries before money was invented in 1100 BC, when people began utilizing miniature metal reproductions. The MEDICI family brought the globe to the official banking system, which included the Florin as a currency that was widely acknowledged throughout Europe. New currencies have been developed since then, but until Satoshi Nakamoto introduced Bitcoin, the global banking system had not seen a revolutionary technology that might transform the procedures. The major question is what would have happened if Bitcoin had never been in the first place?

Bitcoin is more than money; it is the first to introduce blockchain technology to the world, which has already had an influence on various industries and has the ability to change the way businesses are operated today.

Consider blockchain as a historical fabric that records everything that happens, including every digital transaction, data exchange, the exchange of products and services, and the exchange of private data, in real time. After that, the chain stitches the data into encrypted blocks that can never be changed or amended, then disperses the bits throughout a global network of scattered computers known as "Nodes."

Consider a blockchain to be a distributed database that manages a shared list of records. Each encrypted block holds the history of every block that came before it, with time stamped data transactions down to the second, eventually chaining all of those blocks together. As a result, block chain was developed [20].

Blockchain initiatives give an alternative method of transferring money that does not require the use of banks or clearing houses, has an average fee of 1% (which will decrease), and allows for real-time transfers. The most significant component that blockchain brings to remittance is transparency.

But is that it? No.

Blockchain is not just limited to transactions, it was never just related to transactions, it always had the potential for integration with other technologies.

After the introduction of IoT i.e. Internet of things in 1999, when for the first time sensors were used with physical things in order to access data from it and this data was then shared on the internet for further assessment and use, this new technological concept started playing an important role in the world of data, information, and internet. Although its popularity did not reach the masses till the 2010 and 2011, it has been flourishing significantly since then.

1.2 Significance and Motivation

Since technology is progressing at a rapid pace, and in order to make things easier for people, IT has developed a number of techniques that are not only simple but also efficient and reliable, we decided to work on smart contracts, its goal is to let two unknown parties that might lacks trust on each other to conduct business without the middle man involvement. It is also claimed to be protective and secure than conventional contract law because they are mathematically proved. A smart contract is a two- or more-party self-executing agreement. It enables transactions to be completed faster and more securely than traditional methods, while simultaneously reducing the expenses of third-party enforcement. While smart contracts are decentralized on the block chain, there is no single point of failure or vulnerability to attack or hacking. All parties have complete control over their cash at all times, reducing fraud and protecting both buyers and sellers in the event of a dispute. We aimed to design a supply chain management system based on smart contracts to ensure that the temperature sensitive medicines, vaccines or chemical reach its intended destination in the required temperature.

As a result, smart contracts may be used to develop effective supply chain management in the fields of finance and banking, healthcare, public management, insurance, real estate, energy, and even gaming goods.

1.3 Aims and Objectives

The specific objectives being addressed by the project are:

1. Implementation of blockchain feature in IoT devices to establish a smart contract enabling supply chain management.

2. Analysis of performance of smart contract enabled IoT devices.

1.4 Methodology

1.4.1 Node MCU:

The work begins with the use of Node MCU. It is an IoT platform which is open source. It uses ESP8266 wifi module as a firmware and ESP-12 module is its hardware.

1.4.2 DHT11 Sensor:

A device which senses an event is a sensor. To understand a physical phenomenon, a sensor sends an output signal. Here we have used a DHT11 sensor which is very frequently used for the purpose of sensing Temperature and humidity.

1.4.3 IoT Module:

We have connected the DHT11 sensor to the ESP8266 node MCU and then placed it in the vaccination container. Now this IoT module will be able to detect the temperature of the container and send it to Arduino IDE where we will view it in Arduino IDE terminal.

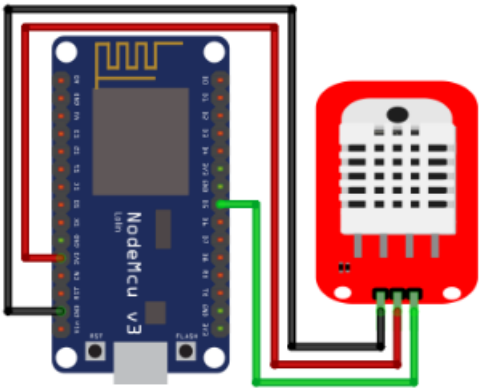


Figure 1: IoT Module Connection Diagram

1.4.4 Firebase Database:

Firebase is a permanent database hosted by cloud. This data is synchronized in real time with the clients that are connected.

Now what we have done is that we have pushed data from the Arduino to the firebase in real time.

This is done by first creating a project in firebase. Then we link this project database to Arduino using the secret key and host address. Thus in this way, we share the real time data to firebase.

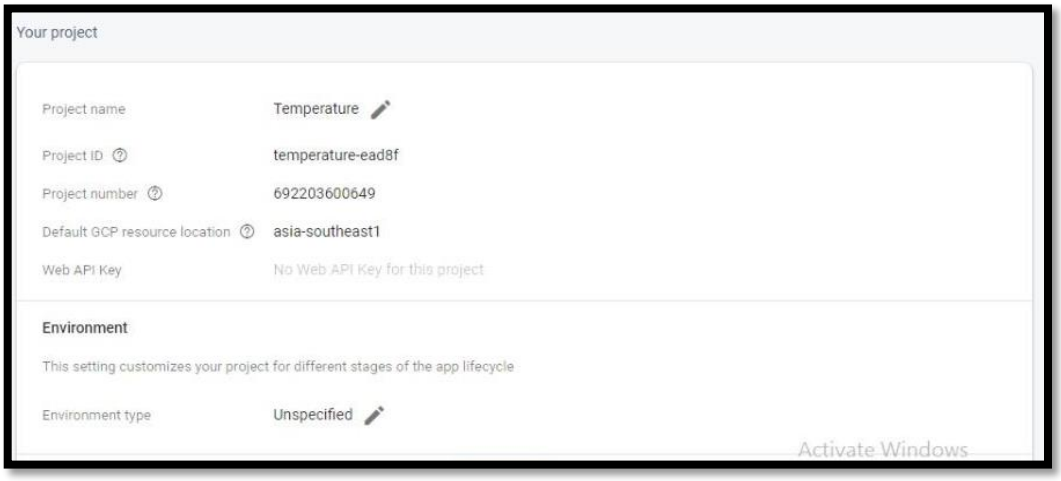
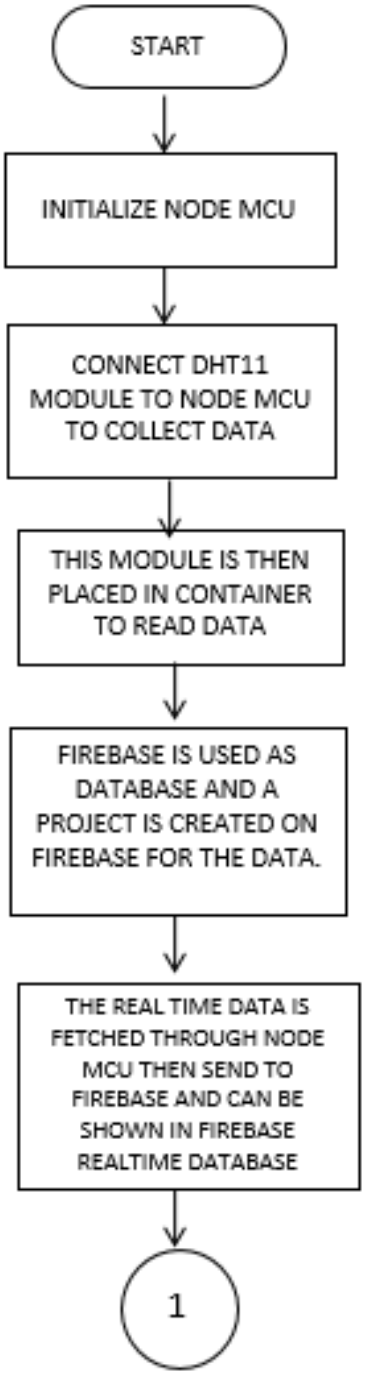


Figure 2: Firebase Project

1.4.5 Design Flow For IoT Module

To fetch real time data connect DHT11 module to the node MCU microcontroller and placed this IoT module inside the container.

Then send this data to the firebase and can be shown in firebase real-time database.



1.4.6 Smart Contract:

We then create a smart contract on remix ide and deploy it through remix IDE through test net.

Smart Contract has three functions.

- Deposit Transaction
- Refund Transaction
- Mapping Function

Once the shipment is started, at the start money will be deposited by the purchaser. For this purpose, a deposit transaction function is created. During the shipment the temperature will be monitored throughout to make sure the temperature of the vaccine is within the required range. At the end of shipment, we take 20 data values of temperature that were taken throughout the shipment and check if the temperature goes out of range, if it does then the refund transaction function will be triggered and the entire money will be refunded to the purchaser as the conditions are not met. If the temperature stays within the required range, then Shipment is Successful.

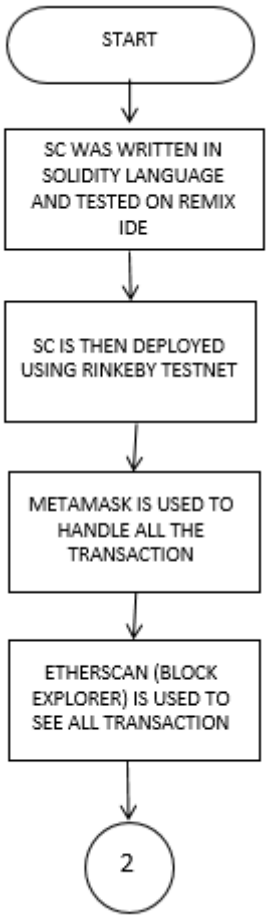
The purpose of the mapping function is to keep the record for all transactions and shipments that are done through this smart contract.

We use MetaMask as our wallet for transactions and ether scan is used for the purpose of viewing the transactions.

1.4.7 Design Flow For Smart Contract:

Smart contract was written in solidity language and tested on Remix IDE then deployed through rinkeby testnet.

Metamask wallet is used for the transactions and transactions can be shown in Etherscan.



1.4.8 Integration of Smart Contract And IoT Module:

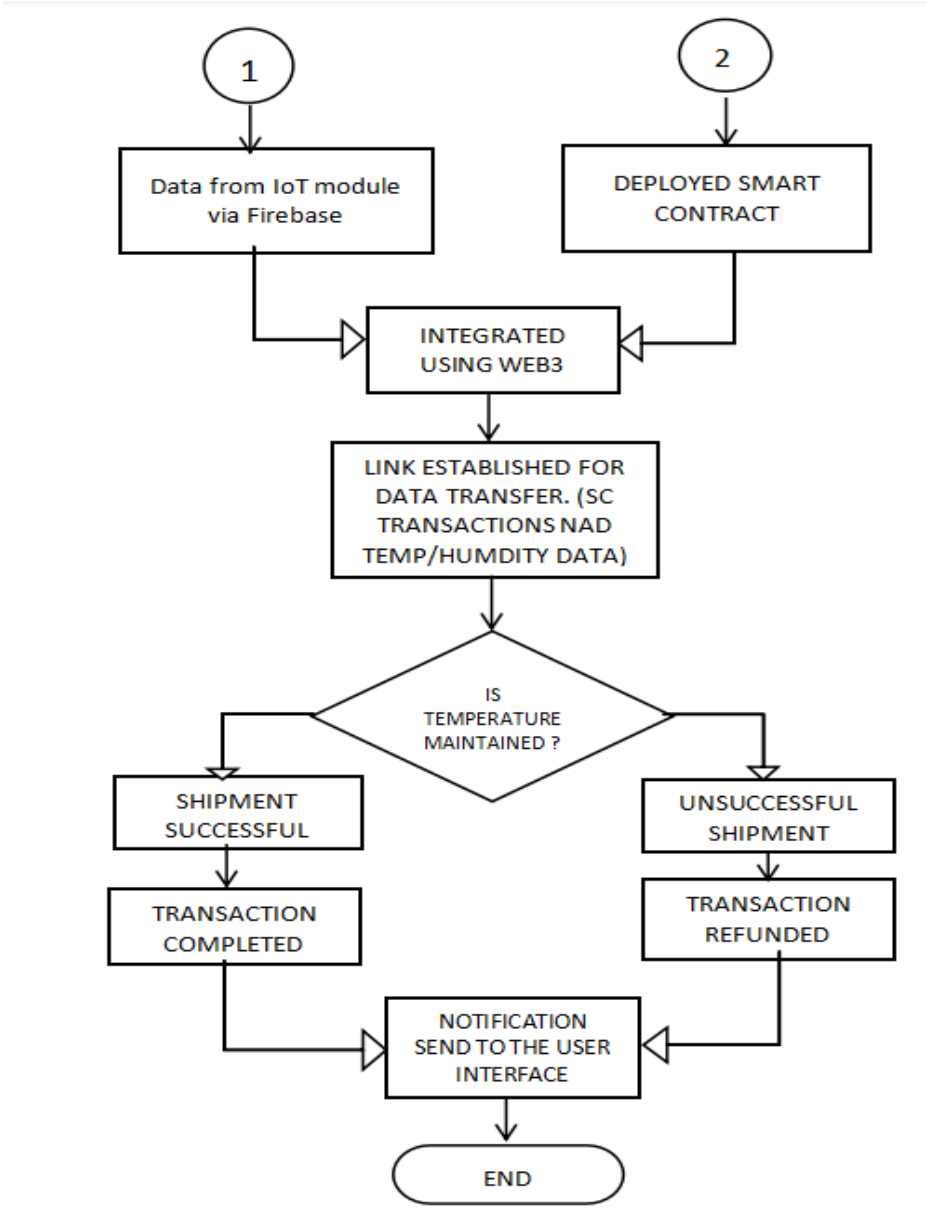
For the purpose of integration, we have used React js. A Java script library whose purpose is to create a user interface that is reusable is React JS. We call the data from firebase using a secret key and host link to React JS. Once data from is called then the next step is to call the already deployed Smart contract to React Js. For this we have used Web3.Now Web 3.0 has a protocol which is called decentralized Block chain which actually enables users to connect technology over the internet.

After calling both the data and the smart contract, we will integrate the data from the IOT module and Smart contract, the conditions for temperature are tested and based on whether the conditions are met or not the functions in smart contract are called. This is further explained in the flowchart below.

1.4.9 Flowchart of Integration of IoT Module And Smart Contract:

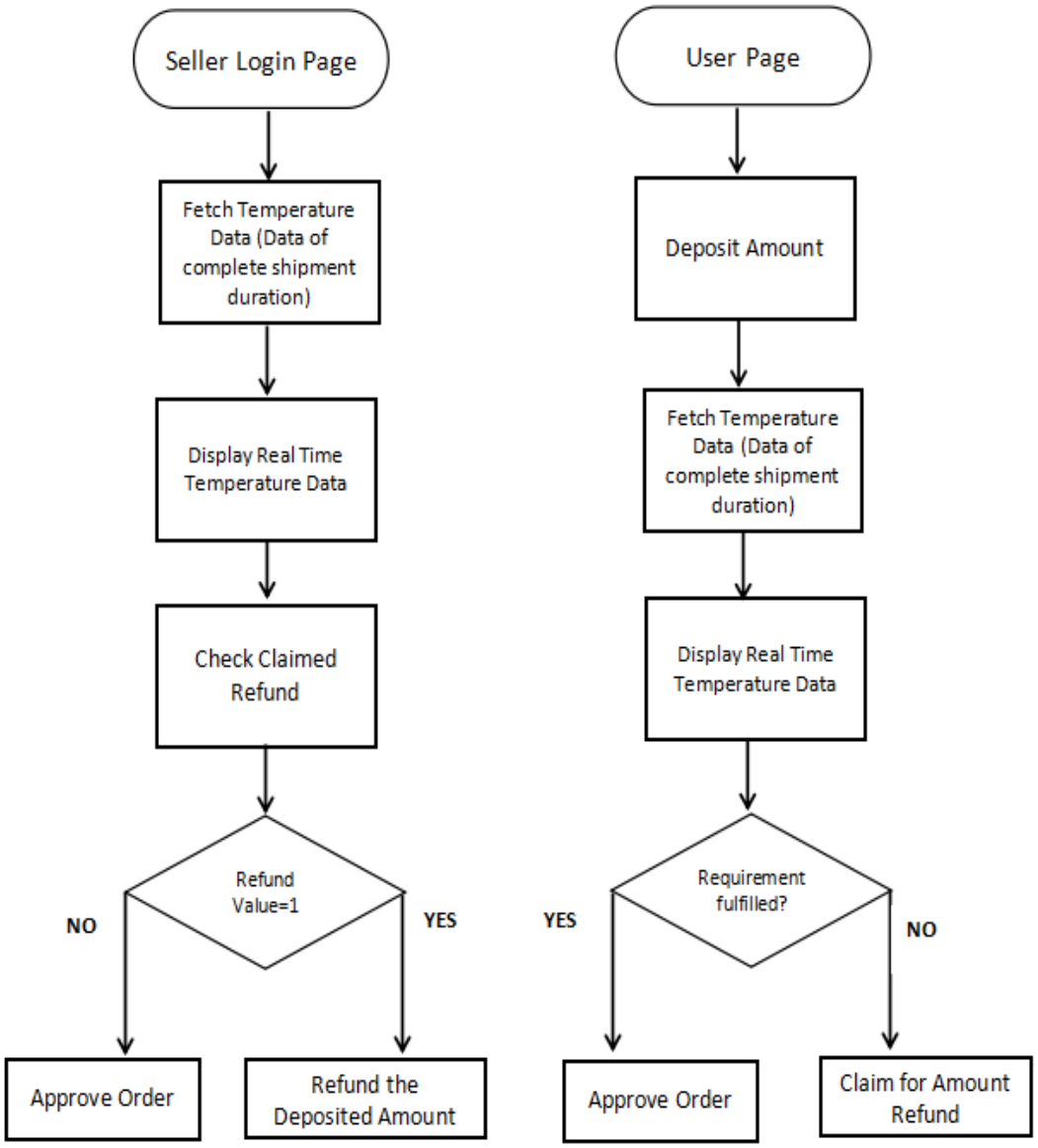
Real time data from firebase and deployed smart contract integrated using web3 on react.js link established for monitoring the temperature on frontend which is monitor by client and seller.

In case of temperature maintained throughout the shipment as per client's demand, the shipment will be successful and payment will be released otherwise unsuccessful shipment and payment refunded to the client and notification sent to the user interfaces.



1.4.10 Front End:

The front end is created on React JS. The Flow for both seller and purchaser are shown below.



1.5 Report Outline

1.5.1 Scope of the Project

Blockchain, the digital ledger technology that underpins many online currencies including Bitcoin and other crypto currency networks, has the strength to revolutionize the financial industry. Whereas, supply chain management is another area of potential by enabling high speed, more cost-effective delivery of product, improvingthe traceability of supply chain as well as product, boost the parties coordination, and facilitates access to funds, blockchain can bring significant improvement in supply chains.

The blockchain has been considered as a technology where a decentralize and distributed database manages all transactions that have occurred in a peer-to-peer network for quite a time. It's considered a distributed, efficient computing model that brilliantly solves the issue of trust between centralized party. As a result, in a block chain network, number of nodes work together to secure and maintain a set of distributed transaction records in a decentralize manner, without relying on one authority.

Network automation and the capacity to convert paper contracts into digital contracts were made possible by smart contracts.

Smart contracts, as opposed to conventional contracts, allowed users to form their agreements and trust relationships through automated transactions that were not supervised by a one central authority. Smart contracts are copied to every node in the blockchain network to mitigate contract manipulation. Man error could be eliminated by allowing computers to perform tasks and using services supplied by blockchain platforms to avoid contract disputes.

The supply chain has a number of complex issues, including dynamic coordination among various parties, low predictability, data compatibility between parties, high commodity monitoring costs, and blind spots. The capacity of block chain to replace complex, error-prone processes with simple smart contracts is predicted to

alter supply chain activities between suppliers, merchants, and consumers. Technology will be leveraged to provide new opportunities for cross-organizational collaboration and enable new supply chain business models as it evolves and matures. The application scenarios continue to increase, and the functionality of applications continue to be developed, thanks to the constant improvement of smart contract technology. New requirements, on the other hand, encourage smart contract technology to continue to evolve.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Extensive research work has been done on the smart contract. Some of them helped us understand, and helped build our project. This is an exploratory kind of research which involves the Implementation of block chain features in IoT devices to establish a smart contract enabling supply chain management.

To collect the data, we have used some secondary sources such as some research papers and websites and the primary sources that involved guidance from our internal and external.

2.2 Smart Contract

2.2.1 IoT Enabled Container

By 2025, the number of Internet of Things (IoT) devices is predicted to reach 75 billion, an increase of more than 50 billion devices in just five years [15]. Traditional computer-aided industries are being transformed into smart sectors that use Internet of Things (IoT) and Big Data Analytics (BDA) principles to make data-driven decisions [15].

Smart contracts on block chains can offer decentralized authentication rules and logic to IoT devices, allowing for single and multi-party authentication. When compared to standard authorization protocols, smart contracts can give more effective authorization access rules to linked IoT devices with less complexity [15].

2.2.2 Ethereum Smart Contract (BlockChain Applications)

Smart contracts, which are Turing complete programme that run on the blockchain, are a feature of Ethereum. These smart contracts are immutable, self-executing, and decentralized thanks to blockchain technology. As a result, smart contracts can't be changed once they've been uploaded on the block chain, and their whole execution is based on this immutable code. The expense of running these smart contracts across a large number of servers is "gas," which costs money. These features provide smart contracts their credibility and make the technology appealing to developers and users. Developers may quickly create Decentralized Applications (DApps) using smart contracts, which have been used in a variety of fields including IoT, finance, gaming, and data security [3].

Because of the large variety of blockchain platforms and programming languages, we have chosen to concentrate on Ethereum smart contracts for the following reasons:

(1) Ethereum was the first blockchain-based application to implement Turing-complete smart contracts.

(2) It is one of the most famous and frequently used smart contract platforms;

(3) Transaction throughput is always first.

(4) the most widely used programming language is Solidity, which was created specifically for Ethereum contracts [2].

Ethereum is the most widely used block chain platform for IoT applications. Ethereum has more features than Bitcoin, and the addition of smart contracts dramatically widens the range of conceivable uses [5].

2.2.3 Integration of IoT and Blockchain

Blockchain and the Internet of Things, on their own, have proved to provide significant advances and benefits to the areas and industries in which they have been used. Over the last several years, blockchain technology has grown at an exponential rate. According to a report by Statista, startup investment capital in blockchain climbed to

almost 550 million dollars in 2016 and is expected to reach well over 2.3 billion dollars by 2021 [18].

The Internet of Things (IoT) is a concept that has acquired a lot of attention and popularity in recent years. Smart healthcare solutions, smart and linked agriculture, smart homes, wearables, augmented reality, and transportation, among other areas, have all seen applications. Physical equipment and items that previously had no interconnectivity can now communicate with other devices and networks, allowing them to become smarter and share information and make decisions [18].

2.3 Supply Chain

2.3.1 Supply Chain Management and Cold Chain

Cold chain and logistics systems for vaccines, medicines and chemicals are critical in overcoming some of these issues. While a number of vaccines, medicines and chemicals have been introduced in underdeveloped countries (with more on the way), cold chain systems are finding it difficult to sustain national immunization campaigns. As a result, there is a possibility of (a) vaccines, medicines and chemicals potency being diminished (e.g. due to poor temperature control, nonfunctional equipment) on this regard smart contract and block chain technology is developed to overcome such issues by enabling IoT sensor data monitoring conditions in a decentralized manner [1].

Modern enterprises' supply chains have been continuously expanded in recent years as a result of the strengthening of the global division of labour. This expansion has caused fragmentation, complexity, geographical dispersion, and other characteristics, all of which have posed serious challenges to supply chain management. It is a possible alternative to employ block chain technology to handle the aforementioned issues; in other words, supply chain and blockchain go hand in hans[10].

2.4 Blockchain Applications

2.4.1 Blockchain and Smart Contract

The blockchain technology, which was originally successfully used in cryptocurrencies, may have the potential to develop into a highly secure and privacy-preserving solution for IoT applications. Blockchain is a term used to describe a decentralized, tamper-proof, and transactional database that offers a safe way to store and process data across a significant number of network participants. In order for a system to achieve high availability and reliability, it is not desired for any component to be a point of failure because it may stop the entire network from operating if it were to fail [14].

By utilizing blockchain technology, IoT may avoid single points of failure and provide a sufficient method for processing and storing IoT data in a secure and effective manner [14].

Decentralization is the primary benefit of blockchain-based smart contracts. When a specific system is integrated with block chain-based smart contracts, it is feasible to do away with the need for trustworthy middlemen like brokers, agents, or service providers. With the removal of a trusted third party, centralised institutions' transaction fees and levels of power will be reduced[13].

Transparency is one of the most distinctive traits that smart contracts have inherited from the block chain. There are two main ways that the smart contract is transparent. First of all, the code written in smart contracts is visible to both the general public and interested parties. Second, the collection of transactions that make up the blocks is likewise accessible to the general public. As a result, the intermediaries of the block chain network can trust in its transactions and logic [13].

Prior to deployment on block chain network nodes, the smart contract's pre-programmed conditions are verified and unchangeable. Once the condition is satisfied, the execution is automated. There won't be any human or other executional errors, which is a guarantee of correctness. Transparent accurate execution through

autonomous accurate execution reduces biased operation and increases confidence. [7]

2.4.2 Blockchain, IoT and 4th Industrial Revolution (Industry 4.0)

Technology is advancing quickly, and individuals need to be able to not only trust while transacting but also to verify or authenticate. In the twenty-first century, block chain technology is expected to be another source of disruptive technology, particularly important to all time-stamped activities [11].

Organizations have to decide whether to embrace the IoT's role in industries in order to guarantee a reliable or potent industrial system. To address the constantly expanding demands of contemporary Industrial IoT requirements, a variety of IoT applications are being developed and deployed. Manufacturing organizations are using IoT to manage workforce issues, assure cyber security, comply with regulations, and to meet their desire for increased revenue from industrial activities [11].

The fourth industrial revolution's most significant component, blockchain, is intrinsically valuable and diverse. The World Economic Forum (WEF) 2016 chose it as one of the "Top Ten Rising Technologies of 2016." A widely used technology serves as a public record for transactions using digital currencies like Bitcoin [12].

The fourth industrial revolution, which is extremely relevant, encompasses numerous initiatives to use various next-generation technologies, including the Internet of Things (IoT), artificial intelligence (AI), big data, and block chain technology, in a variety of industries. Blockchain technology has gained prominence as the most significant new technology among these [12].

Industry 4.0 operations include the provenance of goods, raw materials, and data, financial transactions between industries, consumer and stakeholder interactions, and regulatory compliance. The immutability of the Blockchain is a key characteristic to elevate the reliability of a given item's provenance because the involved peers are trustworthy [17].

The Internet of Things, cyber-physical Systems, and other technologies have given rise to a new economic branch known as the Economy of Things, which is changing software functionality and the computer industry as a whole. With this idea, any contractor who uses a device to carry out a certain service will be paid. The Block chain is one of the most potential enablers in this situation [17].

The concept of a "economy of things" in which Internet of Things (IoT) devices are aware of their duties, have enough autonomy to form contracts with peers, and are capable of making payments upon task completion can be torn apart and strengthened by the use of a private blockchain. Finally, it considers how the entire Supply Chain will improve once the blockchain is implemented in the enterprise environment given these benefits [17].

2.5 Conclusion

An overview was given that included the factors that drove the industrial model to consider adopting a disruptive paradigm like that. Incidentally, we attempted to offer a solid background in each trend, analyzing the effects and evolution of Blockchain, in the Industrial Revolution, and what happened in the context of each not only with regard to the production lines environment, but also in the social sphere. We also conducted research on smart contracts. Through this, a comprehensive picture was built, and a connection could be established that showed not only the advantages of what might occur, but also its economic aspects in the society of the future in terms of Industry 4.0's widespread adoption. This all makes an effort to convey ideas that are important for understanding the entire chapter.

CHAPTER 3

SYSTEM DESIGN

3.1 Introduction

In this chapter, we have described the fundamentals of firmware and software used in our project to get a deeper insight into the working flow of the system. The chapter is broken down into five parts, first and last part is introduction and conclusion to give a quick insight of the things discussed in chapter and what is achieved respectively, then one part is dedicated to hardware in which the IoT side of our project i.e. all hardware used are discussed with their specs and advantages and other two parts are based on software and languages used with comparisons. The overview of system along with the flow diagram is given below:

Figure 3: Flow diagram

In the flow diagram a basic flow of the system can be shown i.e. the real time data fetched with the help of DHT11 humidity and temperature sensor operated by ESP8266 module is send to Firebase cloud server which is then used by smart contract via API to call the triggering functions and display the data and functions on frontend based on React.js simultaneously which can be monitor by both the parties involved in the shipment contract.

3.2 Hardware

Hardware components study helped us a lot; we found its specifications and major role for our project. It helped us to select appropriate components for our project. There is a huge variety of components in the field of electronics, by studying all those we came to the conclusion about the requirement of our project.

The components we used are:

1. NodeMCU
2. DHT11 sensor
3. IoT enabled Container

3.2.1 NodeMCU ESP8266 Microcontroller

Microcontroller is a compact integrated circuit used to perform specific operations in an embedded system. It includes a memory, processor and input/ output (I/O) peripherals on a single chip. It is sometimes called a microcontroller unit (MCU). To make it easy for microcontrollers to interface with sensors and other components, it is designed to be immediately accessible without using additional components as they have enough on board memory as well as providing pins for general input output operations. We are using NodeMCU ESP8266 in our project.

The Node Microcontroller Unit is an open-source software and firmware development environment with an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266 is Espress if Systems designed and manufactured and contains the important elements of a computer: CPU, GPIO, PWM, ADC, RAM, networking (Wi-Fi), and even a modern OS and SDK. All these features make it an excellent choice for all kinds of Internet of Things (IoT) projects and can solve the needs alone.

NodeMCU is based on ESP8266 which can connect different objects together and make the data transferrable using the Wi-Fi protocol. The NodeMCU offers quite a range of development environments and is compatible to use with the Arduino IDE (Integrated Development Environment).

Figure 4: Node MCU

3.2.2 DHT11 Humidity and Temperature Sensor

DHT11 is an inexpensive digital sensor for sensing temperature and humidity of the deployed environment. This sensor is easy to interface with any micro-controller (Arduino, Raspberry Pi or any) to measure the data instantaneously. It is used in various applications such as to gather and analyze humidity and temperature values in heating, ventilation and air conditioning systems.

Specifications:

- 3 to 5V power and I/O
- 2.5mA max current use during conversion
- Good for up to 20-80% humidity readings with 5% accuracy
- Upto 0-50 °C temperature readings +-2 °C accuracy
- Less than 1 Hz sampling rate (once every second)

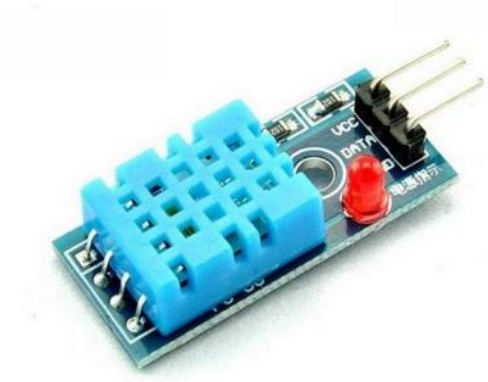


Figure 5:DHT11 Sensor

3.2.3 IoT Enabled Container

Supply chain management has a tremendous impact on the global economy in today's market. The transfer of goods from producer to consumer is a common definition of SCM.

Cold chain and logistics systems for vaccines, medicines and chemicals are critical in overcoming some of these issues.

Logistics requires that certain foods, chemicals, and pharmaceuticals be delivered without breaking the cold chain since, if the conditions aren't satisfied, bacteria might proliferate and cause the product to degrade. Because of this, accurate system control and monitoring are essential to chain management [18].

In this project, a system that supports the Internet of Things is created by creating hardware, software, and equipment that enables remote monitoring of temperature and humidity requirements.

In our project which is a vaccine shipment scenario, an IoT enabled container is created using IoT module deployed in thermacole box used for medicine transit. The cold temperature is maintained using ice packs inside the container.

3.3 Softwares

All the software used to build a working mechanism of the system are described below:

3.3.1 MetaMask

MetaMask is a popular cryptocurrency wallet and very well known for its availability on desktops and mobile phones both, ease of use, the buy, send, and receive ability of cryptocurrency from within the wallet. It is a browser plugin that serves as an Ethereum wallet and installs like any other plugin. Once the MetaMask plugin is installed it allows the users to transact the stored Ether and other tokens with any Ethereum address [9].

With the MetaMask connection to Ethereum-based applications, users can spend their amounts or coins in games, stake tokens etc.

3.3.2 Etherscan

It is a block explorer and block analytics medium that allows users to access details of any pending or confirmed Ethereum blockchain transactions. Etherscan let the users view their assets held on any public wallet address. Using Etherscan, you can see the current state of balance and transaction history by searching with the Ethereum address of the wallet. It also displays gas fees if any and smart contracts involved in that address [17].

Users can use Etherscan to:

Ethereum gas fees calculation with the Ether scan gas tracker

Smart contract lookup and verification

View the crypto assets having a public wallet address

Live transactions observation takes place on the Ethereum block chain

Single transactions lookup made from any Ethereum wallet

Identification of smart contracts having a verified source code and security audit

Rinkeby Testnet:

The Ethereum-based Rinkeby test net enables builders to experiment with protocol upgrades and decentralized websites and applications before the actual deployment I.e. on the Ethereum mainnet. Due to its proof-of-work consensus model the testnet is similar to the mainnet.

Why is TestNnet used?

The Rinkeby testnet is used by developers due to it presents the production environment of the mainnet of Ethereum which is based on a Proof of work mechanism, and to prepare for the upcoming Ethereum mainnet merge.

3.3.3 FireBase

Firebase is a platform for development known originally for its real-time database. The applications based on FireBase allow secure and direct access to the database by using the client-side codes.

The changes made by the user on their device are retained and synchronized automatically while being offline with the remote database. The local device is also updated and informed with changes in the database all in real-time and any conflicts which might occur are merged automatically [16].

The Firebase service also provides the Firebase Real-time Database Security Rules. This is an expression-based and flexible rules language. It is used by developers to structure the data in their application i.e. how it can be read or written. Firebase Authentication helps in defining who can access the data and how it can be accessed. Another advantage is Notifications can be sent with no additional coding with Firebase.

3.3.4 Arduino IDE

A text editor for writing code, a message area, a text console, a toolbar with buttons for basic operations, and a number of menus are all included in the Arduino Software Integrated Development Environment (IDE). The hardware is then connected so that the software may be uploaded and used to communicate with them. Sketches are the name for the programmes created with the Arduino IDE. It offers a variety of functions to programme various boards and track results.

The microcontrollers using Arduino IDE are programmed with a c/c++ syntax. Most c/c++ works but many standard libraries might not work. Many restrictions are made because of the RAM of little storage on the Arduino hardware.

3.3.5 Remix IDE

It allows developing and deployment of smart contracts for Ethereum blockchain. Solidity language is used to write and implement smart contracts. It gives developers a fast developing environment and has a useful set of plugins and GUIs. It compiles, executes and debug smart contracts and users can interact with it.

3.3.6 Figma Design Tool

Figma is a cloud-based designing tool used to design User Interface for the project. This tool also provides a lot of useful resources, plugins, and techniques that make the work smooth. It is a collaborative platform that helps with working, plan and design with team for better results.

3.4 Languages and Libraries

The programming languages and libraries used in the building the system are discussed below:

3.4.1 Web3.js

Web3.js is an open-source library and it is for the JavaScript programming language. It serves projects based on Ethereum in such a way that it enables builders to interact with the Ethereum block chain.

The Ethereum Foundation built this library and hence there's a large community behind it. Web.js includes functions required to communicate with an Ethereum node done particularly by a protocol known as the JavaScript Object Notation – Remote Procedure Call (JSON-RPC). Simply put, It is a connection between the Ethereum block chain and smart contracts. It is used to achieve a stable, secure and decentralized network.

3.4.2 Solidity Programming Language

Solidity programming languages are also called curly bracket language designed to create/write smart contracts that run on Ethereum and automate transactions on block

chains. It is similar to C++, Python and Java programming languages. Many block chain solutions are created using Solidity.

It makes it easy for the user to not worry about the risks of fraud and inability to similar currency usage. Solidity creates low-level code that is executed on the Ethereum Virtual Machine. A compiler like Remix IDE is used to break down human-readable code, which it turns into instructions read by the processor. We can say that solidity is more than just a programming language and crucial for the future of block chain technology.

3.4.3 React. Js

React is an UI development library based on java script programming language. It is widely used in web development. React.js offers a range of extensions for application architectural support for instance Flux and React Native.

WHY React? [20]

- React makes it easier to create dynamic web Apps due to its less coding requirements and offers more functionality contrary to JavaScript.
- Components are considered as the building blocks of any application on react. These components can be reused which reduces the development time of applications that have their logic and controls.
- It follows a unidirectional data flow. This means that when a React app is designed, developers often use hierarchical manner while nesting components which keep the flow of data in one direction. It makes it easy to identify the problem and debug the error.
- React is user friendly and easy to learn, as mostly it is a combination of basic HTML and JavaScript concepts with some additions.
- It can be used to create both mobile and web apps with the help of another library called React Native.

3.4.4 Node.Js

Node.js is a java script based programming language used to create a server-side code. Node.js offer number of useful libraries and open source packages to use freely. one of the advantage of Node.js is when an I/O operation (reading from the network, database or the file system access) is performed, it do operations when a response comes back instead of wasting CPU cycles waiting and blocking the threads. Here a system is created to access database (I.e FireBase) and manipulate data to fetch realtime data on frontward for display.

3.5 Conclusion

In the above chapter all the firm wares, software, languages and libraries used in the project are described. The hardware of the project i.e. NodeMCU ESP8266 microcontroller and DHT11 sensor is used to fetch humidity and temperature data, whereas the softwares of the project includes Arduino IDE to program MCU and DHT11, Remix IDE for deployment and debugging of smart contract, Metamask as crypto wallet, Etherscan to keep a track of balance and transactions, Rinkeby testnet to test the network before actual deployment, FireBase cloud server for realtime database are used, the languages and libraries used are such as Web3.js library to integrate smart contract, hardware and frontend, React Js is used to develop frontend and Solidity programming language to write the smart contract of the project. The necessary details and description along with an overview of the project flow is discussed.

CHAPTER 4

RESULTS

4.1 IoT module:

In this IoT module we have connected a DHT11 sensor to NodeMCU in which we have connected pin 1 of sensor to the ground pin, pin 2 to pin D5 and pin 3 to 3V pin of NodeMCU. This is done to get the data of temperature and humidity from shipment container.

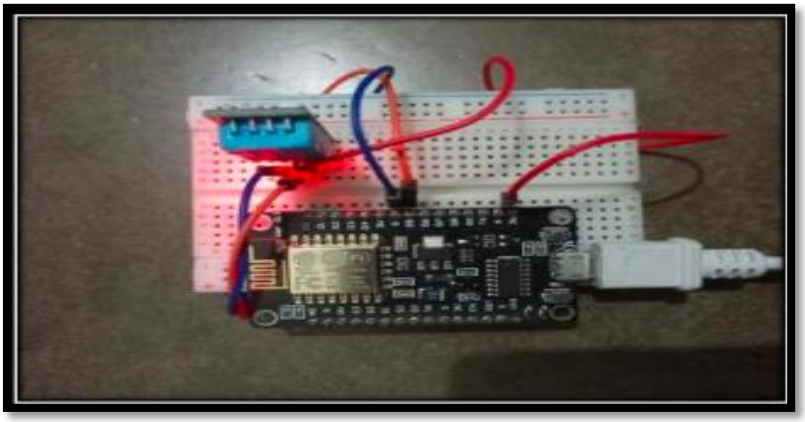


Figure 6: IOT MODULE

This IoT module is then placed in a shipment container where we have used icepacks to maintain the temperature.

Figure 7: NodeMCU and Temperature sensor connected (IOT MODULE)

Now temperature and humidity data is fetched from the NodeMCU to Arduino IDE. This data is then visualized on the serial monitor of Arduino IDE.

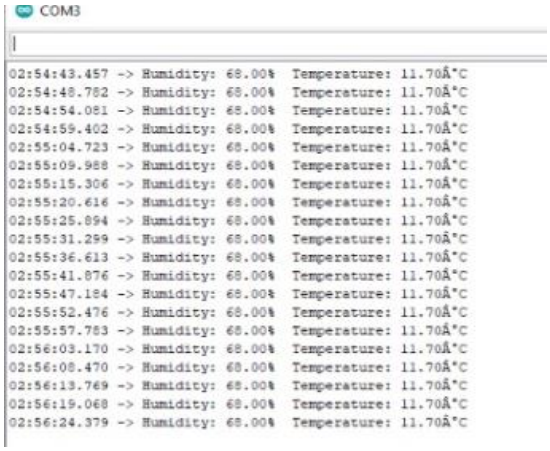


Figure 8: Output on Serial monitor

4.2 Firebase Data

In order to fetch the data and integrate it with smart contract, we needed the data to be stored in a database. For this purpose, we have used firebase. We created a project on firebase and then by using project credentials we have shared the data via wifi to firebase. This data is then visualized on Firebase Database.

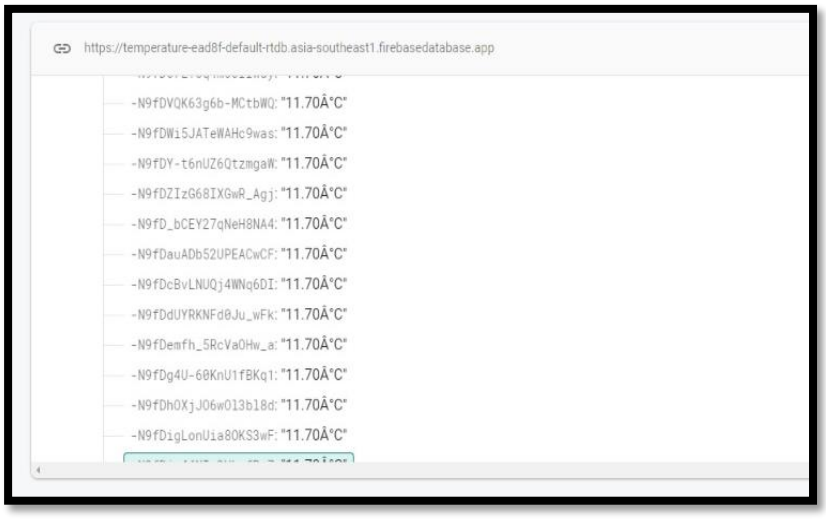


Figure 9: Data base showing humidity and temperature

4.3 Smart contract Output:

Smart contract output functions on Remix IDE are shown below. CreateOrder function creates the order, deposits the money which is 0.0025 ethers and generates an orderID. As soon as the order ID generates the shipment is started and the time for shipment in our case is 2 minutes. Once two minutes pass, then the shipment is completed and we can use the claim and refund function. If the conditions are met and the buyer approves the order then the seller will claim the deposited ethers by using withdraw function. Else if the conditions are not met and buyer requests refund then the seller will refund the ethers using refund function. TransferOwner function simply transfers the ownership of the contract from one seller to another, renounceOwner function renounces the ownership of the contract so that no one is the owner and no one may use the contract. Owner function gets the detail of the owner. OrdersRecord will take the order number and give the details of order. Total order shows the total number of orders. SaveData function saves the temperature data in blockchain in realtime, placeOrderPrice is defining the value of transaction and tempData shows the temperature data of the given order ID.

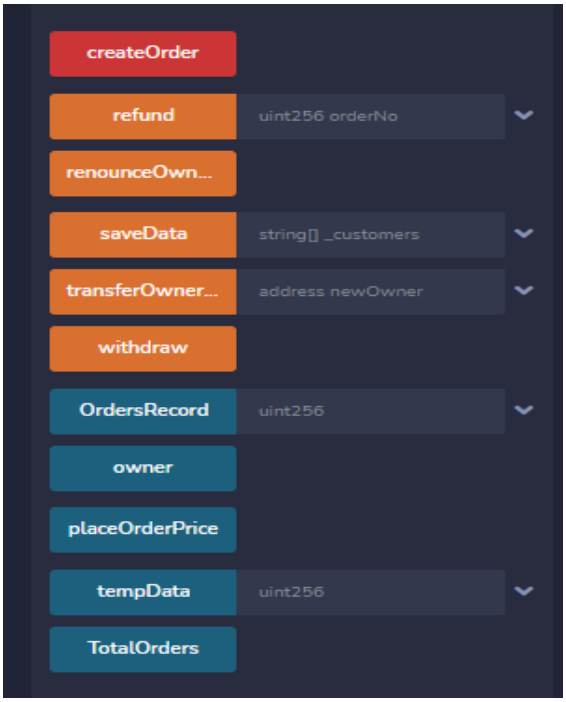


Figure 10: Display of deployed smart contract in remix IDE

4.4 Integration of Smart Contract and IoT Module Via Frontend

The connection between Smart Contract, Metamask, Rinkeby, Database and Frontend is build using web3 for their simultaneous inter-working.

The temperature send on database is displayed on frontend against the respective order Ids. The createOrder function in smart contract will create an order ID everytime the order is created by the user. TotalOrder function will save the count of orders being created and will show on frontend to user. Metamask is connected through Ethereum Rinkeby testnet to handle transactions. When the user will deposit an amount the wallet will pop-up to show the transaction details. The wallet address is also saved with the respective order ID.

If the temperature will go out of range (i.e. > 15℃) then the user can claim for refund amount, the seller will get the notification of user's claim, it will check the temperature record details and if the claim is right the amount will be refunded by seller otherwise if the temperature is maintained (I.e. < 15℃) throughout the designated shipment time the order will be approved by both parties. This is achieved by setting Boolean value against the refund and approve button to be enabled such that if claim value is 0, the shipment is considered to be successful, satisfying the temperature requirement hence approve is enabled on seller side and if the claim value is 1, the refund button will be enabled and seller will revert the transaction.

In a snapshot below a landing page of our frontend is shown

Figure 11: Frontend Landing page

For seller a log-in button is created (figure 11) to land on seller log-in page. The seller portal can be access only by seller verified by inserting the correct credentials (figure 12).

Figure 12: Seller's Log-in

The user will click on "User" button on the page (figure 11) to land on user page to create/monitor order. In figure 13, the order Id has been assign to the created order which can be seen in a pop-up notification which says "your Order number is 5"

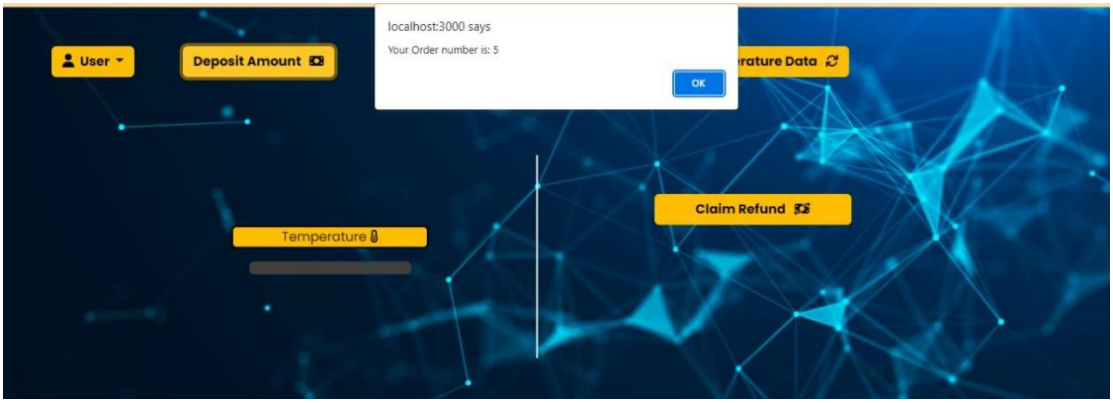


Figure 13: Order is created

On the seller's side, the list of active order Id will be available in a drop-down from which the seller can select any active order to monitor the shipment (figure 14).

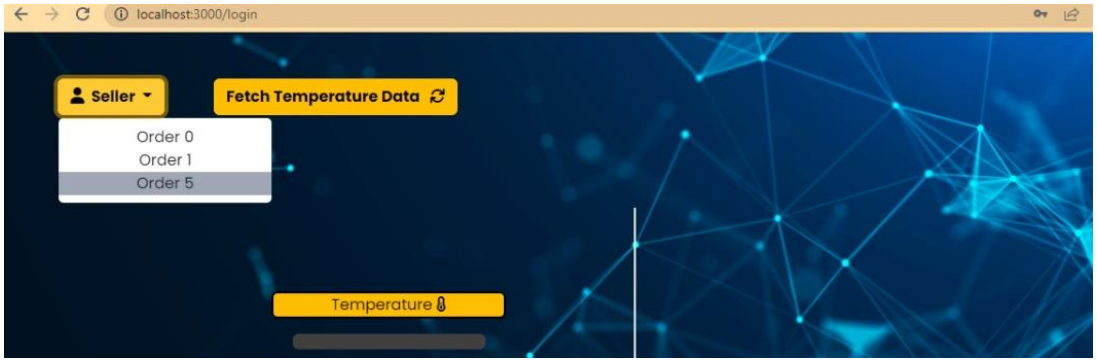


Figure 14: Seller's active Order Id list

By selecting the order Id the seller will check the temperature and claim value (figure 16), if the claim is made by the user then on the basis data record the seller will refund back the deposited amount to user (figure 17).

Case: Temperature out of range

The user will fetch data to monitor the shipment and temperature requirement (figure 15). Here you can see, the temperature is >15℃ hence the claim button is enabled.

Figure 15: User side with refund claim

Figure 16: Seller side with refund

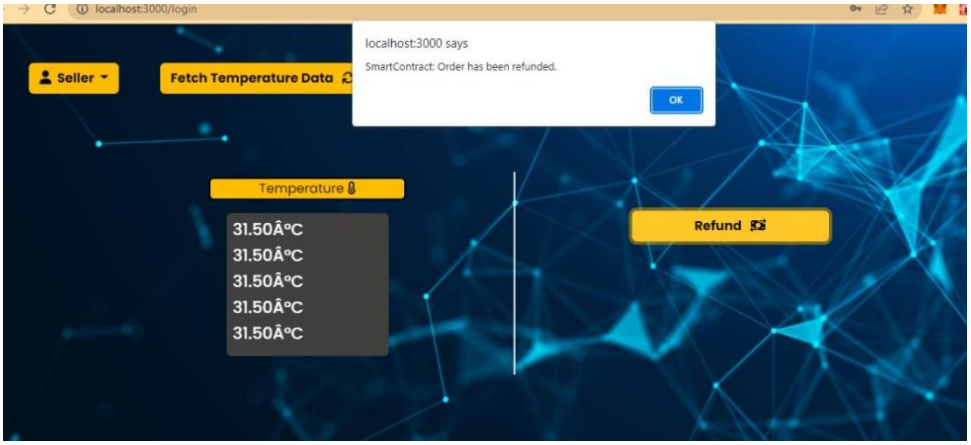


Figure 17: Amount refunded

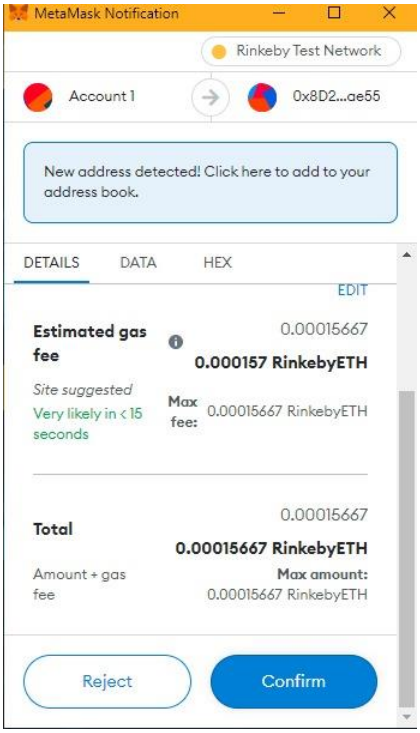


Figure 18: MetaMask wallet after refund

Case: Temperature is maintained throughout the shipment.

If temperature is maintained throughout the shipment i.e. temperature < 15℃, only then the order can be approve by seller and buyer (figure 19, 20, 21).

Figure 19: User side with successful case

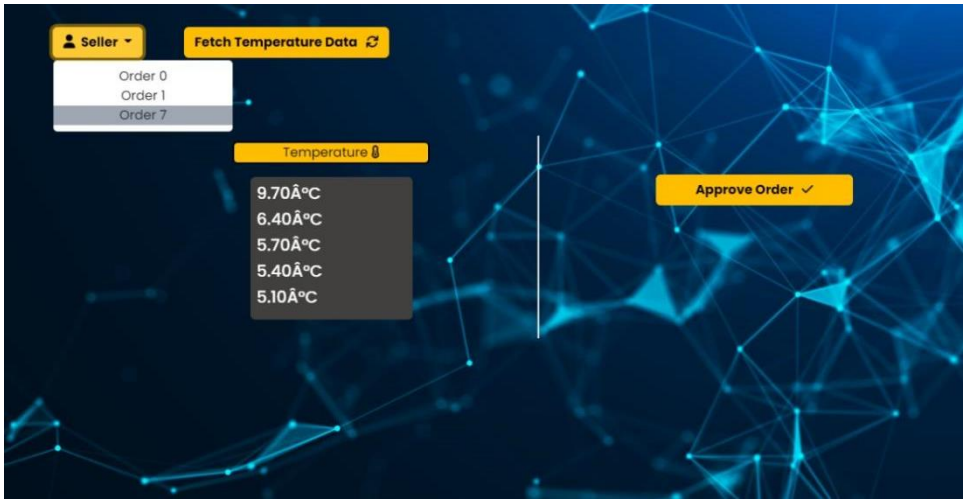


Figure 20: Seller side with successful case

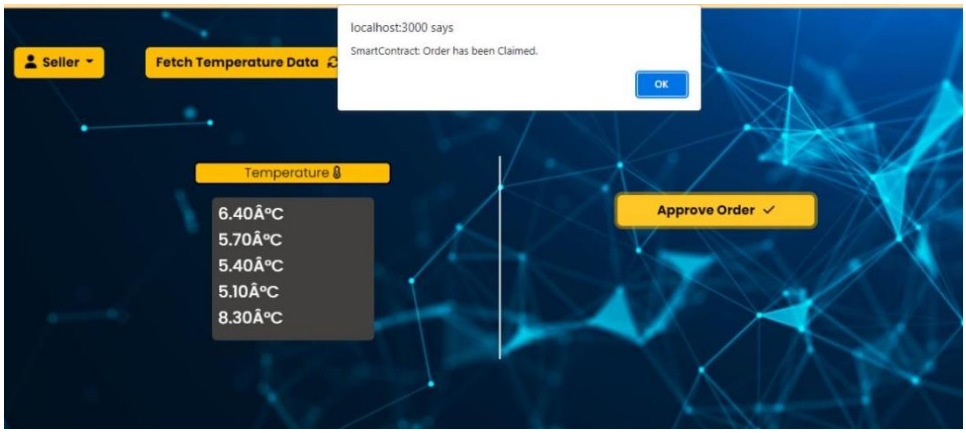


Figure 21: Order Approved

Shipment data stored in Decentralized Blockchain Storage:

Real time data of shipment is saved on decentralized storage by use of block chain Smart contract which can be seen in the figure 22 in data column.

Figure 22: Data saved in decentralized storage

CHAPTER 5

SUMMARY

5.1 CONCLUSION

A block chain-enabled infrastructure has been presented in this study to promote openness throughout the supply and distribution of temperature sensitive vaccines, medicines and chemicals. Smart contracts are being developed to track vaccine production and delivery. Self-reporting, self-monitoring, immutability, temper-proofing, accuracy, and transparency are among the features offered by the proposed solutions. The producer will use smart contracts to define some rules for the distribution of product. After a verification process by a healthcare provider, only registered recipients can receive the vaccine. The suggested solution's simulation results show that it is feasible in terms of GAS computation and transaction throughput. After analyzing the simulation results, the consumption of gas, mining process, difficulty, and transaction cost are all dependent on the input type or block size for block chain deployment. The solution provides:

- A framework for data transparency, immutability, and efficiency of registration for the vaccine campaign to avoid counterfeit and identity theft,
- A smart contract enabled framework for self-administering the vaccine distribution constraints in the cold chain about the fulfillment of vaccine
- A framework for vaccine supply chain management that will enable the features of tamper-proof, person identification, and avoid counterfeit.

In the future, an analysis of the acquired data on feedback and data storage can be performed to determine the efficiency of vaccination vials as well as to determine the best environment in which to store the vaccine vials. The suggested model keeps all transaction data in the block chain.

5.2 RECOMMENDATIONS FOR FUTURE WORK

Our project is a small scale model of automated shipment process using dual technology solution i.e. Internet of things and Block chain which can be further improve by incorporating some future work recommendation listed below to increase product quality and scalability for secure and transparent management system.

- Different tracking parameters like GPS, Air Quality, motion sensor etc can be added using different modules and sensors in the shipment container.
- The whole shipment mechanism can further be automated by adding more backend functionalities.
- Further research efforts should also be made for scalability and security that effects both technologies and their integration.
- The adoption of this dual technology in government infrastructure can effects the import/export quality and can speed up the interaction between companies and government.

REFERENCES

[1] Ashok, Ashvin & Brison, Michael & LeTallec, Yann. (2016). Improving cold chain systems: Challenges and solutions. Vaccine. 35. 10.1016/j.vaccine.2016.08.045.

[2] Bueno, Thiago & Morais, Everton & Fernandes, Luiz & Righi, Rodrigo & Alberti, Antonio. (2020). Blockchain and Industry 4.0: Overview, Convergence, and Analysis. 10.1007/978-981-15-1137-0_2.

[3] Chen, Jiachi & Xia, Xin & Lo, David & Grundy, John & Yang, Xiaohu. (2021). Maintenance-related concerns for post-deployed Ethereum smart contract development: issues, techniques, and future challenges. Empirical Software Engineering. 26. 10.1007/s10664-021-10018-0.

[4] Cil, A. Y. (2022, May 5). Internet of Things enabled real time cold chain monitoring in a container port - Journal of Shipping and Trade. SpringerOpen. https://jshippingandtrade.springeropen.com/articles/10.1186/s41072-022-00110-z

[5] Fakih, Umer. (2022). Blockchain based Smart Contracts: Technical Aspects and Future Research Blockchain based Smart.

[6] Glotova, E. (2021, June 26). Temperature monitoring in logistic 4.0. IOT Factory. https://iotfactory.eu/temperature-monitoring-in-logistic-4-0/

[7] Hewa, Tharaka & Ylianttila, Mika & Liyanage, Madhusanka. (2020). Survey on Blockchain based Smart Contracts: Applications, Opportunities and Challenges. Journal of Network and Computer Applications. 177. 10.1016/j.jnca.2020.102857.

[8] Hewa, T. M., Hu, Y., Liyanage, M., Kanhare, S. S., & Ylianttila, M. (2021). Survey on Blockchain-Based Smart Contracts: Technical Aspects and Future Research. IEEE Access, 9, 87643–87662. https://doi.org/10.1109/access.2021.3068178

[9] Hussey, M. D. P. (2022, June 4). What is MetaMask? How to Use the Top Ethereum Wallet. Decrypt. https://decrypt.co/resources/metamask

[10] Lou, Menghui & Dong, Xiaolei & Cao, Zhenfu & Shen, Jiachen. (2021). SESCF: A Secure and Efficient Supply Chain Framework via Blockchain-Based Smart Contracts. Security and Communication Networks. 2021. 1-18. 10.1155/2021/8884478.

[11] Nartey, Clement & Tchao, E. T. & Gadze, Dzisi & Keelson, Eliel & Klogo, Griffith & Kommey, Benjamin & Diawuo, Kwasi. (2021). On Blockchain and IoT Integration Platforms: Current Implementation Challenges and Future Perspectives. Wireless Communications and Mobile Computing. 2021. 1-25. 10.1155/2021/6672482.

[12] Pais, Mr & K, Fayiz & Raghavendra, Gagan & M, Gowthami & Suvarna, Jeevitha. (2022). Adoption of Blockchain in IoT: Challenges and Solutions. International Journal of Advanced Research in Science, Communication and Technology. 294-302. 10.48175/IJARSCT-5822.

[13] Reyna, Ana & Martín, Cristian & Chen, Jaime & Soler, Enrique & Díaz, Manuel. (2018). On blockchain and its integration with IoT. Challenges and opportunities. Future Generation Computer Systems. 88. 173-190. 10.1016/j.future.2018.05.046.

[14] Seunghwan Myeong, Seunghwan & Jung, Yuseok. (2019). Administrative Reforms in the Fourth Industrial Revolution: The Case of Blockchain Use. Sustainability. 11. 3971. 10.3390/su11143971.

[15] Saputhanthri, Amila & de Alwis, Chamitha & Liyanage, Madhusanka. (2021). Emergence of Blockchain based IoT Marketplaces.

[16] Team, E. A. (2022, July 31). What is Firebase? Educative: Interactive Courses for Software Developers. https://www.educative.io/answers/what-is-firebase.

[17] Wang, Zeli & Jin, Hai & Dai, Weiqi & Choo, Kim-Kwang Raymond & Zou, Deqing. (2021). Ethereum smart contract security research: survey and future research opportunities. Frontiers of Computer Science. 15. 10.1007/s11704-020-9284-9.

[18] Nwakanma, Cosmas & Nwadiugwu, Williams-Paul & Lee, Jae Min & Kim, Dong-Seong. (2019). Real-Time Validation Scheme using Blockchain Technology for Industrial IoT..

[19] Deshpande, C. (2022, July 18). The Best Guide to Know What Is React. Simplilearn.Com. https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs

[20] Zeli Wang, Hai Jin, Weiqi Dai, Kim-Kwang Raymond Choo, Deqing. (2021). Ethereum smart contract security research: survey and future research opportunities.