

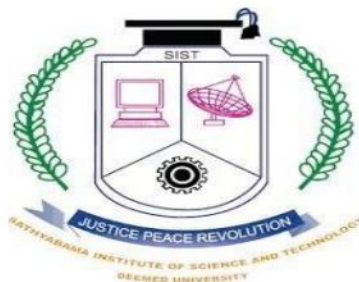
BLOCK CHAIN BASED TRANSACTION AND SETTLEMENT SYSTEM

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

Y. CHANDRA KIRAN (Reg. No - 40111556)

M. GANESH REDDY (Reg. No - 40111513)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Category – 1 University by UGC

**Accredited with Grade “A++” by NAAC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600119**

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

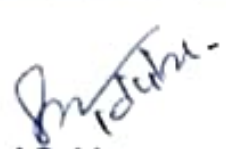
Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

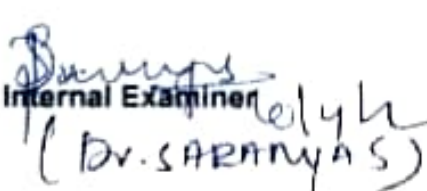
This is to certify that this Project Report is the bonafide work of **Y. Chandra Kiran(40111556)** and **M. Ganesh Reddy(40111513)** who carried out the Project work entitled "**BLOCK CHAIN BASED TRANSACTION AND SETTLEMENT SYSTEM**" under my supervision from November 2023 to April 2024.

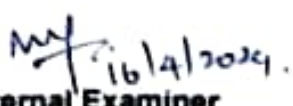

Internal Guide

Dr. J. JABEZ, M.E., Ph.D.,


HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Dr. L. LAKSHMINARASIMHAN, M.E., Ph.D.,
Chennai - 600 119.

Submitted for Viva voce Examination held on 16 -04 - 2024


Internal Examiner
(Dr. SARANYA S)


External Examiner
[Dr. Meery Paul Selvan]

DECLARATION

I, **M. Ganesh Reddy (40111513)**, hereby declare that the Project Report entitled **"BLOCK CHAIN BASED TRANSACTION AND SETTLEMENT SYSTEM"** done by me under guidance of **DR. J. JABEZ M.E., Ph.D.**, at Sathyabama institute of science and technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE: 12-04-2024

PLACE: Chennai

M. L. S. V. Ganesh Reddy

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of Sathyabama institute of science and technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D., Dean**, School of Computing, and **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. J. Jabez M.E., Ph.D.**, for his valuable guidance, suggestions, and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Blockchain is growing as a potentially Out of line force capable of changing the financial services industry by making the fund transfer immediate, cheaper, and more secure. Current existing system is not secure enough to give 100% fraud protection because of more manual work and lack of security of data. Blockchain is nothing but a chain made of blocks (nodes). These process node does the all-major work. These blocks are connected to each other using cryptography. This system will be more expeditious, more efficient, and has user affectional interfaces in the banking and has zero probability of losing data while processing of the user data. In integration to enabling trade, block chain is larceny-and tamper-resistant model, it eliminates errors and the duplication, block chain is ideal for reserving the data in blocks and using a tamper-proof hash format, so the data can be securely stored by bank and make the current existing system much more secure and faster. Which is already done in case of crypto currency.

The decentralized nature of the Bit coin blockchain, all transactions can be transparently viewed by either having a personal node or using block chain explorers that allow anyone to see transactions occurring live. Each node has its own copy of the chain that gets updated as fresh blocks are confirmed and added. This means that if you wanted to, you could track a bit coin wherever it goes.

For example, exchanges have been hacked in the past, resulting in the loss of large amounts of crypto currency. While the hackers may have been anonymous except for their wallet address the crypto they extracted are easily traceable because the wallet addresses are published on the block chain.

There is a need for a more secure, transparent, and efficient transaction system that can prevent fraud, reduce disputes, and promote sustainable land management. Blockchain technology has the potential to address these challenges by providing a decentralized, tampered transaction, and transparent system for transaction system.

CONTENTS

CHAPTER NO	TITLE	PAGE NUMBER
	ABSTRACT	i
	LIST OF FIGURES	iv
1	INTRODUCTION	1
	1.1 AIM	2
	1.2 MOTIVATION OF THE PROJECT	2
	1.3 SCOPE OF THE PROJECT	2
2	LITERATURE SURVEY	3
	2.1 INFERENCES FROM LITERATURE SURVEY	3
3	REQUIREMENT ANALYSIS	6
	3.1 SYSTEM REQUIREMENTS	6
	3.1.1 HARDWARE REQUIREMENTS	6
	3.1.2 SOFTWARE REQUIREMENTS	6
4	DESCRIPTION OF PROPOSED SYSTEM	7
	4.1 EXISTING SYSTEM	7
	4.2 PROPOSED SYSTEM	8
5	IMPLEMENTATION DETAILS	9
	5.1 SYSTEM ARCHITECTURE	9
	5.2 DATA FLOW DIAGRAM	10
	5.3 USE CASE DIAGRAM	11
	5.4 SEQUENCE DIAGRAM	12
	5.5 ACTIVITY DIAGRAM	13
	5.6 BLOCK CHAIN	14
	5.7 HASHING IN BLOCK CHAIN	14
	5.7.1 HASH FUNCTION PROPERTIRS	14
	5.7.2 COMMON HASHING ALGORITHMS	15
	5.7.3 ROLE OF HASHING IN BLOCKCHAIN	15
	5.8 JAVA	16

	5.9 HTML	16
	5.10 CSS	17
	5.11 JAVA SCRIPT	19
	5.12 MSSQL	21
	5.13 NETBEANS IDE	22
6	RESULTS & DISCUSSION	25
	6.1 MODULES GENERATED	25
	6.1.1 ACQUIRER SETTLEMENT BANK	25
	6.1.2 MERCHANT	25
	6.1.3 CUSTOMER	26
	6.2 RESULT	27
7	SUMMARY AND CONCLUSION	28
	7.1 CONCLUSION	28
	7.2 FUTURE SCOPE	28
	REFERENCES	29
	APPENDIX	
	A. SOURCE CODE	31
	B. SCREEN SHOTS	40
	C. RESEARCH PAPER	43
	D. CERTIFICATE	48

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
5.1	System architecture	9
5.2	Data Flow Diagram	10
5.3	Use Case Diagram	11
5.4	Activity Diagram	12
5.5	Sequence Diagram	13

CHAPTER-1

INTRODUCTION

Blockchain is digital, distributed, and public ledger. Blockchain technology was first used in now a days popular crypto currency called BITCOIN (virtual currency), but it is expected that its characteristics of accurate and guarded data transfer in distributed P2P network could make other applications possible. In block chain nodes are connected cryptographically and in chronological order. The block chain offers ability to distribute ledgers in decentralized way and that's a key concept. Unlike centralized system where the ledgers of records are stored in single specific entity like a bank or governmental institution, the block chain shares the ledgers in between its participants. That is what make block chain de centralized. This means that ledgers are written and stored in network and its members are responsible to update and monitor it. Every block of record is constantly synchronized by the different members of the open network, creating multiple copies of the data through a shared record-keeping system ensuring no single person or an organization holds ownership of data. When an incipient transaction or an edit to a subsisting transaction comes in to a block chain, generally a most of the nodes within a block chain implementation must apply steps (algorithm) to calculate and validate and verify the past of the individual block chain's block that is introduced. When all the node come to an agreement or consensus that history and signature of the transaction is valid then that new block is then added to chain of the transaction. If majority of nodes does not agree to the integration in ledger ingression, then block is not added to chain. This kind of working allows this block chain methodology to run without any need of central authority block chain each block contain number of transactions. It provides a decentralized, immutable data store that can be used across a network of users, engenders assets, and acts as a shared ebony book that records all transactions. So, a block chain establishes an auditable and indisputable open record of information that is cheaper, faster, and more secure than any other centralized system.

1.1 AIM:

There is a need for a more secure, transparent, and efficient transaction system that can prevent fraud, reduce disputes, and promote sustainable land management. Blockchain technology has the potential to address these challenges by providing a decentralized, tampered transaction, and transparent system for transaction system.

1.2 MOTIVATION OF THE PROJECT:

The ledger which holds the details of all transactions which happen on the Block chain, is open and completely accessible to everyone who is associated with the system. . Even though the complete ledger is publicly accessible, the details of the people involved in the transactions remains completely anonymous. Every single transaction is verified by cross-checking the ledger and the validation signal of the transaction is sent after a few minutes. Through the usage of several complex encryption and hashing algorithm, the issue of double spending is eliminated.

1.3 SCOPE OF THE PROJECT:

The decentralized nature of the Bit coin blockchain, all transactions can be transparently viewed by either having a personal node or using block chain explorers that allow anyone to see transactions occurring live. Each node has its own copy of the chain that gets updated as fresh blocks are confirmed and added. This means that if you wanted to, you could track a bit coin wherever it goes.

For example, exchanges have been hacked in the past, resulting in the loss of large amounts of crypto currency. While the hackers may have been anonymous except for their wallet address the crypto they extracted are easily traceable because the wallet addresses are published on the block chain.

Of course, the records stored in the Bit coin block chain are encrypted. This means that only the person assigned an address can reveal their identity. As a result, block chain users can remain anonymous while preserving transparency.

CHAPTER – 2

LITERATURE SURVEY

2.1 INFERENCE FROM LITERATURE SURVEY

Face Recognition Based Payment Processing System,2023

The use of many cards' transaction in various places like shopping, restaurants, lodges, and online transaction in regarding booking hotels, movies, flights, and train tickets etc., are on a rise each second. Hence the problem of people carrying payment card & cash along with themselves and keep the cards/cash secure to use it all the time; chances of being victim of theft

Issues and Challenges of Data Transaction Management in Cloud Environment, 2022

In this paper, we aim to improve phishing detection techniques using machine learning techniques. We propose a learning-based aggregation analysis mechanism to decide page layout similarity, which is used to detect phishing pages. Our experiment results shows that our approach is accurate and effective in detecting phishing pages.

A Literature Review of Block Chain-Based Applications in Supply Chain,2022

This paper addresses this research gap by examining, through qualitative analysis, the study finds that block chain applications are moving in the direction of efficient automation. Appropriate consensus algorithms in different supply chain contexts will improve efficiency while reducing costs, while smart contracts play a huge role in security. Therefore, this paper explored articles related to the topic and explored articles in recent years by keyword combination.

A Block chain Based Decentralized Transaction Settlement System in Banking Sector,2021

The Indian banking system is one of the complex bank payment systems in this world. The current infrastructure that is used by Indian bank system is real time gross settlement system based and it follows a centralized architecture. Due to this centralized architecture the processing of transactions is slow and cumbersome. It also causes large amount for security and recovery purposes. The real time gross settlement-based system demands high need for security, resilience, and performance. Moving from traditional system to block chain platform is not the prior concern but making a system that provide security, confidentiality, and decentralized money lending mechanism is the core idea. Here proposed a novel system that enable a decentralized Banking system and services based on Ethereum block chain platform.

A Systematic Literature Review of Block chain Technology for Smart Villages, 2022

Block chain Technology is the next step for innovation and development and it has far many applications in sustainable rural development that are yet to be discovered. The objective of this paper is to explicitly review research conducted in rural development to fill the undone work in the future with better research ideas, to make rural areas a liveable and advanced place while also maintaining their integrity leading to sustainable development. To conduct such a review, a systematic research methodology is applied following regulations in the conduction of standardized but explorative analysis. Within the timeline of 2010–2021, 112 papers are carefully selected to perform the systematic review. This review will provide a comprehensible as well as concise research compendium for all applications proposed, implemented, and possible in the future to realize the concept of smart villages for the development of rural areas using block chain technology.

Block chain-based Payment Systems: A Bibliometric & Network Analysis,2022

National Natural Science Foundation of China has sponsored most academic documents, with the US emerging as the most impactful country. This study has also found that block chain-based payments literature congregated in 5 disciplines. These areas are computer science, engineering, decision science, mathematics, and business. A co-authorship analysis also networks relations between nations, authors, and organizations globally, creating unique clusters that maximize research productivity. In summary, this study designs an analytical map of the research landscape, which can guide future research.

Keywords: Block chain; Payments; Bibliometric Analysis; Co-authorship Analysis; Co-occurrent.

CHAPTER-3

REQUIREMENT ANALYSIS

3.1 SYSTEM REQUIREMENTS

3.1.1 Hardware Requirement

- Laptop or PC
- Windows 7 or higher
- I3 processor system or higher
- 4 GB RAM or higher
- 100 GB ROM or higher

3.1.2 Software Requirement

- Java
- MSSQL
- SQLyog Enterprise
- Java script
- Net beans IDE

CHAPTER - 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 EXISTING SYSTEM:

Frontend supply chains involve the transfer of large quantities of products. Therefore, incomplete transfer and product loss are inevitable. This project presents a high-efficiency group ownership transfer protocol that incorporates block chains to enable the efficient transfer of large quantities of products and prevent disputes regarding transfer completion. This protocol collocates authenticated information of a manufacturer with its exclusive block chain address to generate an initial transaction block. The block chain is then used to trace the original product manufacturers to prevent counterfeiting, and a grouping proof is employed to prevent disputes regarding transfer comprehensiveness. The proposed protocol could resist common off-chain radio frequency identification ownership transfer attacks, indicating its security. A security testing tool was employed to prove whether the on-chain smart contracts could also resist the attacks on block chain. The analysis and experiment results revealed that the proposed protocol required fewer messages for ownership transfer and half the calculation time compared with the existing protocols. Thus, for the simultaneous transfer of ownership of a massive number of tags, the proposed protocol is the most efficient, with a reduced calculation time, thus making it most suitable for bulk cargo ownership transfer in the frontend supply chain environment.

DISADVANTAGES:

- They were transfer the products without knowing the person.
- It has no money transaction option.
- And it does not have security.
- Even it will not store the customer data as well

4.2 PROPOSED SYSTEM:

The settlement bank is critical in assisting brick-and-mortar and online retailers in conducting business and providing convenient payment options. If the card issuer authorizes the transaction, funds will be transferred from the issuing bank's account to the merchant's account with the assistance of the settlement bank. Our Blockchain-based Transaction and Settlement System connects merchants with the banks and their services for quick and easy transactions. In this project, the front-end involves Html, CSS, and JavaScript and the back-end involves ASP.NET. The IDE used is Visual Studio and the database is MS SQL .In this system, the bank has a list of all merchants with service period details. The list also includes all of the merchant's transactions, as well as the status of any modified or tempered transactions. The bank will send a notification to the merchant to renew services, they may renew the service agreement for the specified period. All customers are charged fees per transaction by the bank. The merchant connects to the bank and views the service period or expiry date. They can pay for services using a bank account. The status of any modified or tempered transactions made by the Merchant for services is also listed with the transactions. The merchant may send agreement approval. The customer must register to log in to the system with their basic details. They can view a list of all the merchant information. Although the customer can start the transaction process with a particular merchant, the actual transaction is handled by the Acquirers (admin) account. The customer will receive a list of all transactions as well as information about any modified or tempered transactions. Customers can provide feedback by sharing their experiences.

ADVANTAGES:

- It is easy to maintain.
- It is user-friendly.
- All transactions are monitored.
- If any transaction is suspected, the bank is notified immediately.

CHAPTER - 5

IMPLEMENTATION DETAILS

5.1 SYSTEM ARCHITECTURE:

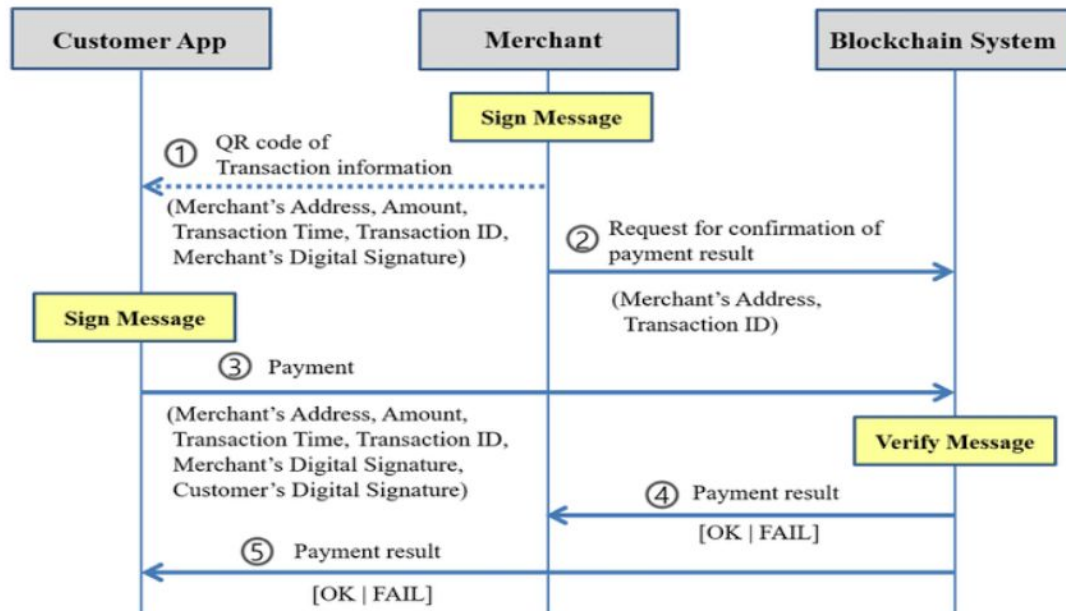


Fig 5.1: System architecture

5.2 DATA FLOW DIAGRAM

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

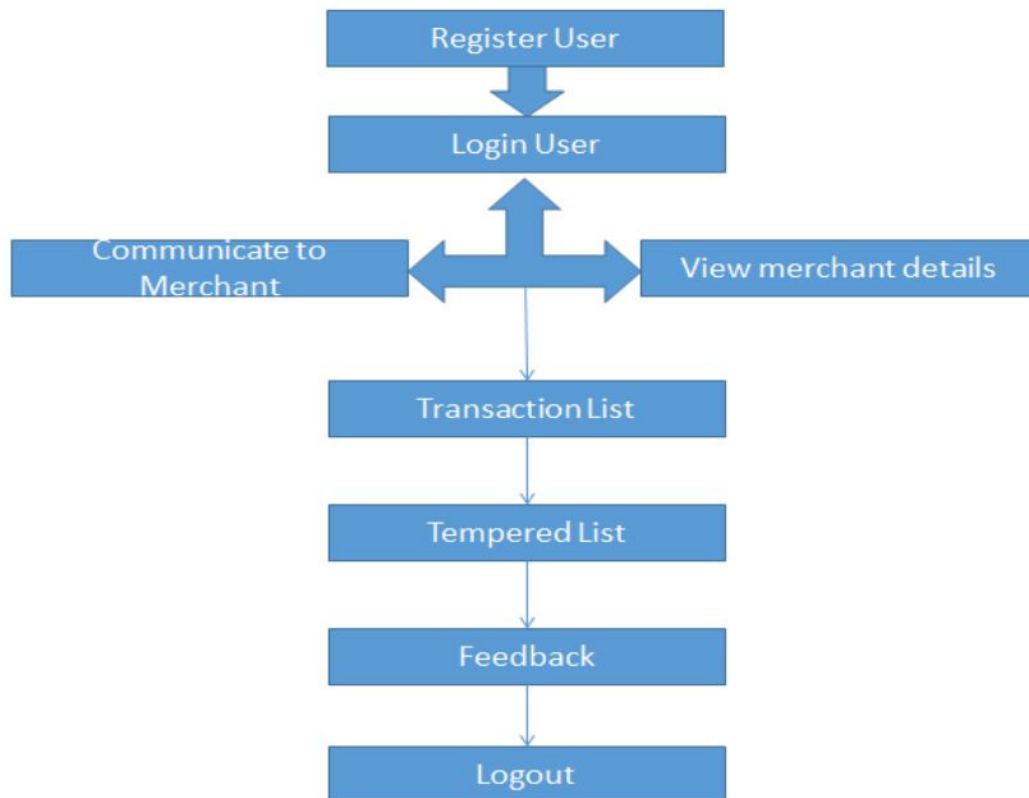


Fig 5.2: Data Flow Diagram

UML & URL DIAGRAMS

UML stands for Unified Modelling Language. UML is a standardized general purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

5.3 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

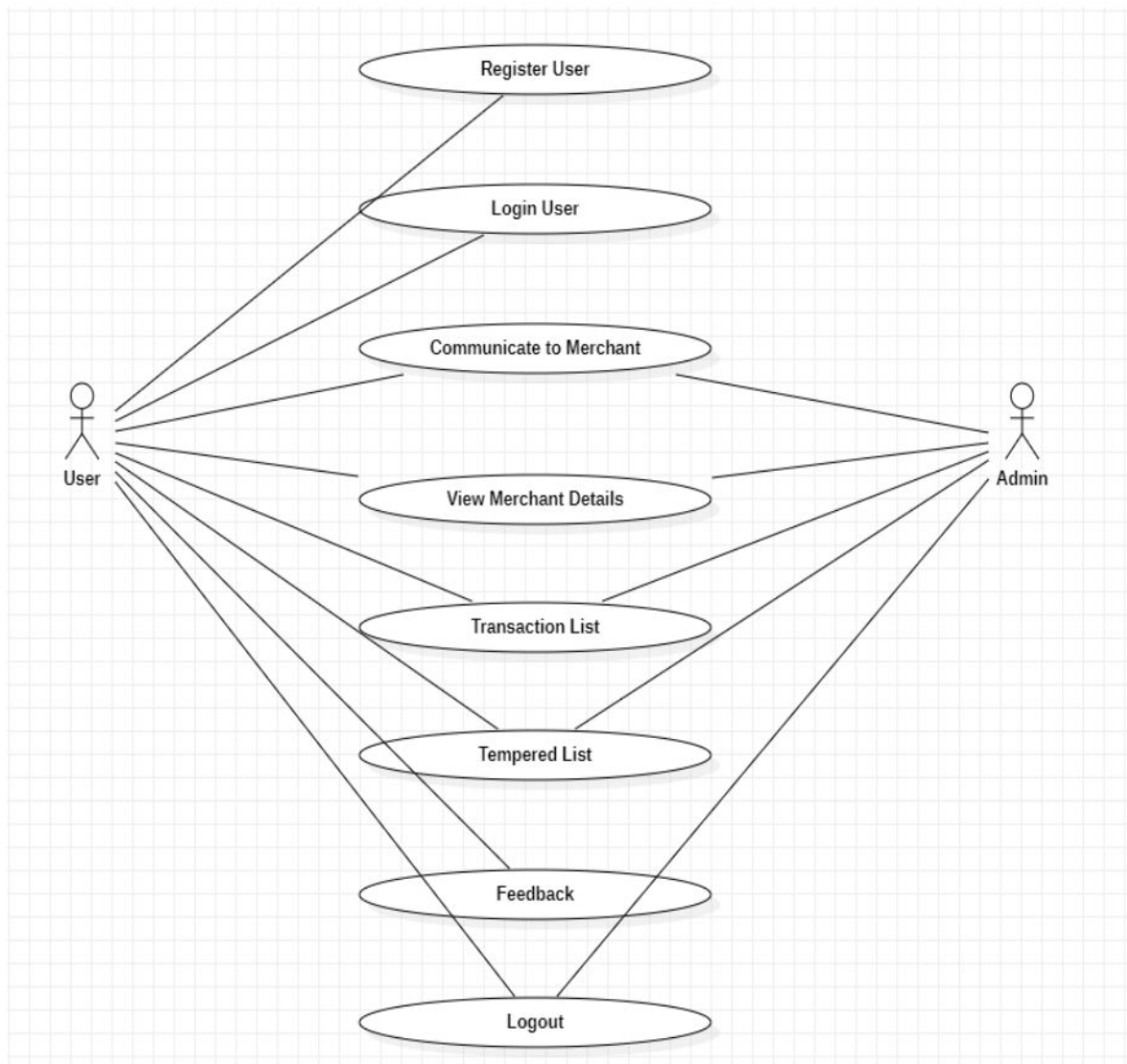


Fig 5.3: use case diagram

5.4 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

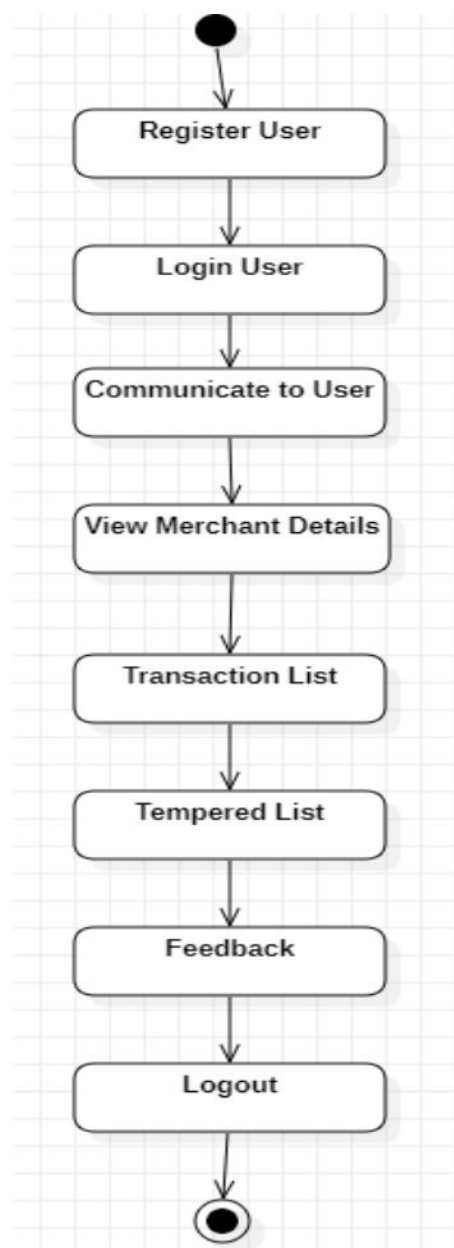


Fig 5.4: Activity Diagram

5.5 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

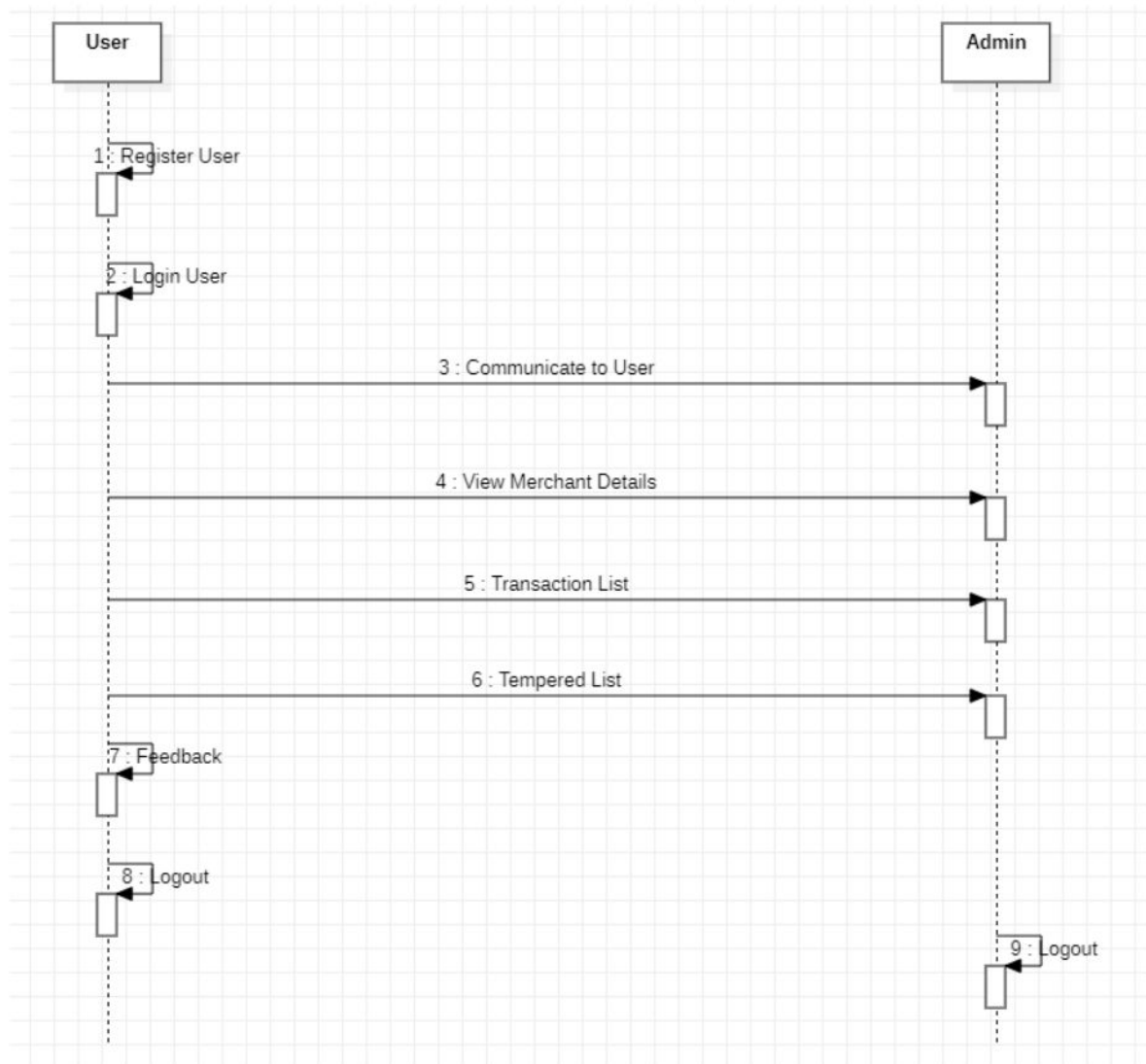


Fig 5.5: Sequence Diagram

5.6 BLOCK CHAIN

Blockchain is a decentralized digital ledger technology that records transactions across a network of computers. It operates on a distributed system, meaning no single entity has control over the entire network. Each transaction is verified by multiple participants (nodes) in the network, ensuring transparency and security. Once recorded, data on the blockchain is immutable, meaning it cannot be altered or deleted. Blockchain technology is the backbone of cryptocurrencies like Bitcoin, but its applications extend beyond finance to various industries such as supply chain management, healthcare, and more, offering secure and transparent record-keeping.

5.7 HASHING IN BLOCKCHAIN

Hashing is a fundamental concept in blockchain technology that plays a crucial role in ensuring the integrity and security of data stored on the blockchain. In simple terms, a hash function is a mathematical algorithm that takes an input (or 'message') and produces a fixed-size string of characters, which is typically a hexadecimal number.

Here's an explanation of hashing and some of the commonly used algorithms in blockchain:

5.7.1. Hash Function Properties:

- **Deterministic:** For the same input, a hash function will always produce the same output.
- **Quick Computation:** Hash functions are designed to compute quickly, even for large inputs.
- **Irreversibility:** It should be computationally infeasible to reverse the process and obtain the original input from the hash output.
- **Avalanche Effect:** A small change in the input should result in a significantly different hash output.

5.7.2. Common Hashing Algorithms:

a. SHA-256 (Secure Hash Algorithm 256):

- SHA-256 is one of the most used hash functions in blockchain, notably in Bitcoin.
- It produces a 256-bit (32-byte) hash value, which is typically represented as a 64-character hexadecimal string.
- SHA-256 is known for its cryptographic strength and resistance to collision attacks (two different inputs producing the same hash).

b. RIPEMD-160 (RACE Integrity Primitives Evaluation Message Digest):

- RIPEMD-160 is often used in conjunction with SHA-256 in blockchain systems, particularly for creating cryptocurrency addresses.
- It produces a 160-bit hash value, which is shorter than SHA-256 but still provides sufficient security for address generation.

5.7.3. Role of Hashing in Blockchain:

- **Data Integrity:** Each block in the blockchain contains a hash of the previous block's header, creating a chain of blocks linked by hashes. Any change to the data in a block would alter its hash, breaking the chain and indicating tampering.
- **Merkle Trees:** Hashes are used to create Merkle trees, a data structure that efficiently summarizes and verifies the integrity of large sets of data.
- **Mining:** In Proof of Work (PoW) consensus mechanisms, miners compete to find a hash value (nonce) that meets a certain criteria and results in a hash with a predefined number of leading zeros. This process is computationally intensive but crucial for securing the blockchain.

Overall, hashing and its associated algorithms are essential components of blockchain technology, providing data integrity, security, and consensus mechanisms crucial for maintaining a decentralized and trustless ledger.

5.8 JAVA

Java is a versatile, object-oriented programming language that was developed by Sun Microsystems in the mid-1990s and later acquired by Oracle Corporation. It was designed to be platform-independent, allowing developers to write code once and run it on any device that supports Java without needing to recompile.

Key features of Java include its simplicity, robustness, security, and portability. It provides a large standard library, which makes it easier for developers to accomplish common tasks without having to write their own code from scratch.

Java is widely used in various domains, including web development, mobile application development (Android), enterprise software, scientific computing, and more. Its popularity is due to its versatility, performance, and community support.

The Java Virtual Machine (JVM) plays a crucial role in executing Java bytecode on different platforms. Java programs are compiled into bytecode, which is then interpreted by the JVM, making Java applications platform-independent.

5.9 HTML

HTML, short for HyperText Markup Language, is the standard markup language used to create web pages. It provides a structure for content on the World Wide Web, defining elements and their arrangement within a document.

Key points about HTML:

1. Markup Language: HTML uses markup tags to define elements within a web page. These tags surround content to indicate how it should be displayed or function.
2. Structure: HTML documents consist of a hierarchy of elements, including headings, paragraphs, lists, links, images, and more. These elements define the structure and organization of content.
3. Semantic Markup: HTML5 introduced semantic elements such as `<header>`, `<footer>`, `<nav>`, `<article>`, `<section>`, and `<aside>`, which provide meaning to the content and improve accessibility and search engine optimization.

4. Attributes: HTML elements can have attributes that provide additional information or modify the behavior of the element. Attributes are included within the opening tag and provide attributes like ``id``, ``class``, ``href``, ``src``, etc.

5. Hyperlinks: HTML allows the creation of hyperlinks using the ``<a>`` tag, enabling users to navigate between different web pages or sections within a page.

6. Images and Multimedia: HTML supports embedding images, audio, video, and other multimedia content using tags like ``<img>``, ``<audio>``, and ``<video>``.

7. Forms: HTML provides form elements such as ``<input>``, ``<textarea>``, ``<select>``, and ``<button>`` for collecting user input. Forms are essential for interactions such as user registration, login, and data submission.

8. Accessibility: HTML supports accessibility features such as alt attributes for images, semantic markup for better screen reader interpretation, and ARIA roles and attributes for enhancing accessibility for users with disabilities.

9. Cross-Browser Compatibility: HTML is supported by all modern web browsers, ensuring consistent rendering and functionality across different platforms and devices.

10. Integration with CSS and JavaScript: HTML is often combined with Cascading Style Sheets (CSS) for styling and JavaScript for dynamic behavior, enabling the creation of interactive and visually appealing web pages.

In summary, HTML serves as the foundation for creating structured and interactive web pages, providing the basic structure and elements necessary for presenting content on the World Wide Web.

5.10 CSS

CSS, short for Cascading Style Sheets, is a style sheet language used to define the presentation and appearance of HTML documents. It enables web developers to control the layout, design, and formatting of web pages across different devices and screen sizes.

Key points about CSS:

1. **Styling HTML Elements:** CSS allows developers to apply styles to HTML elements, such as fonts, colors, margins, padding, borders, and backgrounds, enhancing the visual presentation of web content.
2. **Selectors and Declarations:** CSS uses selectors to target specific HTML elements and declarations to specify the styles applied to those elements. Selectors can target elements based on their type, class, ID, attributes, or relationship with other elements.
3. **Cascading Style Sheets:** CSS follows the cascade principle, where multiple styles can be applied to an element, and conflicts are resolved based on specificity and inheritance rules.
4. **External, Internal, and Inline Styles:** CSS can be applied to HTML documents externally using separate style sheets linked via the `<link>` element, internally within the `<style>` element in the `<head>` section of an HTML document, or inline directly within HTML tags using the `style` attribute.
5. **Box Model:** CSS utilizes the box model to describe the layout and spacing of elements on a web page. Each HTML element is treated as a rectangular box, comprising content, padding, borders, and margins, which can be styled using CSS properties.
6. **Responsive Web Design:** CSS supports responsive web design techniques, allowing developers to create layouts that adapt to different screen sizes and orientations. Media queries enable the customization of styles based on factors such as screen width, height, and resolution.
7. **Flexbox and Grid Layout:** CSS provides layout modules such as Flexbox and CSS Grid Layout, which offer powerful tools for creating complex and responsive page layouts with ease, facilitating the development of modern web interfaces.
8. **Vendor Prefixes and Browser Compatibility:** CSS vendor prefixes are used to implement experimental or proprietary CSS features in web browsers before they

are standardized. However, with the advancement of CSS standards, the need for vendor prefixes has reduced, improving browser compatibility and reducing code redundancy.

9. Preprocessors and Postprocessors: CSS preprocessors like Sass, Less, and Stylus extend the capabilities of CSS by introducing features such as variables, nesting, mixins, and functions, enhancing code maintainability and productivity. Postprocessors like Autoprefixer automatically add vendor prefixes to CSS rules, streamlining cross-browser compatibility.

10. Animation and Transitions: CSS allows the creation of animations and transitions to add interactivity and visual effects to web pages, enabling elements to move, fade, rotate, and transform in response to user actions or predefined triggers.

In summary, CSS plays a crucial role in web development by providing the means to control the visual presentation of HTML documents, ensuring consistency, flexibility, and responsiveness across different browsers and devices.

5.11 JAVA SCRIPT

JavaScript is a high-level, dynamic, and interpreted programming language primarily used for adding interactivity to web pages. Here are some key points about JavaScript:

1. Introduction: Developed by Brendan Eich at Netscape in 1995, JavaScript is now one of the core technologies for building web applications alongside HTML and CSS.

2. Client-Side Scripting: JavaScript is predominantly used as a client-side scripting language, meaning it runs on the user's web browser rather than on the web server. This allows for dynamic updates and interactive elements on web pages.

3. Syntax: JavaScript syntax is similar to that of C programming language. It supports imperative, functional, and object-oriented programming paradigms.

4. **Variables and Data Types:** JavaScript has dynamic typing, meaning variables can hold values of any data type without declaration (`var`, `let`, `const`). Data types include numbers, strings, booleans, objects, arrays, and more.
5. **Functions:** Functions in JavaScript are first-class citizens, meaning they can be assigned to variables, passed as arguments, and returned from other functions. Anonymous functions, arrow functions, and function expressions are common.
6. **DOM Manipulation:** JavaScript interacts with the Document Object Model (DOM), which represents the structure of HTML documents. This allows developers to dynamically manipulate the content and structure of web pages.
7. **Event Handling:** JavaScript enables event-driven programming, where functions (event handlers) are executed in response to user actions like clicks, mouse movements, or keyboard inputs.
8. **Asynchronous Programming:** JavaScript supports asynchronous programming through callbacks, promises, and `async/await` syntax. This allows non-blocking execution, crucial for handling tasks such as AJAX requests or timeouts.
9. **Libraries and Frameworks:** JavaScript has a vast ecosystem of libraries and frameworks such as jQuery, React, Angular, and Vue.js, which provide tools and abstractions for building complex web applications efficiently.
10. **Server-Side Development:** With the introduction of Node.js, JavaScript can also be used for server-side development. Node.js allows developers to build scalable network applications using JavaScript on the server-side.
11. **Security:** JavaScript execution can pose security risks such as cross-site scripting (XSS) attacks if not properly handled. Developers need to be mindful of security best practices like input validation and sanitization.
12. **Browser Compatibility:** While modern browsers have good support for JavaScript, inconsistencies in implementation across different browsers can still pose challenges for developers. Libraries like Babel help in transpiling modern JavaScript code to older versions for broader compatibility.

5.12 MSSQL

Microsoft SQL Server (MSSQL) is a relational database management system (RDBMS) developed by Microsoft. Here are some key points about MSSQL:

1. Introduction: Microsoft SQL Server is a powerful and widely used database management system, particularly in enterprise environments, for storing, querying, and managing relational data.
2. Editions: MSSQL comes in various editions tailored for different use cases and workloads, such as SQL Server Express (free), SQL Server Standard, SQL Server Enterprise, and SQL Server Developer Edition (for non-production use).
3. Features: MSSQL offers a wide range of features including support for SQL (Structured Query Language), ACID transactions, stored procedures, triggers, views, indexing, and more. It also provides advanced features like data encryption, high availability, and business intelligence capabilities.
4. T-SQL: Transact-SQL (T-SQL) is the proprietary extension of SQL used in Microsoft SQL Server. It includes additional features and functions specific to SQL Server, enabling developers to write powerful and efficient queries and scripts.
5. Management Tools: Microsoft provides several tools for managing MSSQL databases, including SQL Server Management Studio (SSMS), SQL Server Data Tools (SSDT), and SQL Server Configuration Manager. These tools offer GUI-based interfaces for database administration, development, and configuration.
6. Integration Services: SQL Server Integration Services (SSIS) is a powerful ETL (Extract, Transform, Load) tool provided by Microsoft for building data integration and data warehousing solutions. It enables the creation of workflows to extract data from various sources, transform it, and load it into target databases or data warehouses.
7. Analysis Services: SQL Server Analysis Services (SSAS) is an analytical data engine used for business intelligence and data analysis. It supports

multidimensional and tabular models, enabling users to create OLAP (Online Analytical Processing) cubes and data mining models for advanced analytics.

8. Reporting Services: SQL Server Reporting Services (SSRS) is a server-based reporting platform that allows users to create, publish, and manage interactive reports. It provides tools for designing reports and dashboards, scheduling report execution, and distributing reports to users.

9. Security: MSSQL offers robust security features including authentication, authorization, encryption, and auditing to protect sensitive data and ensure compliance with regulatory requirements.

10. Scalability and High Availability: SQL Server supports scalability and high availability through features like database replication, clustering, AlwaysOn Availability Groups, and Azure SQL Database, allowing organizations to build resilient and scalable database solutions.

11. Cloud Integration: Microsoft offers cloud-based versions of SQL Server such as Azure SQL Database and Azure SQL Managed Instance, providing fully managed database services in the cloud with scalability, reliability, and built-in security features.

Microsoft SQL Server remains a popular choice for organizations of all sizes due to its reliability, performance, and comprehensive feature set for managing relational data and supporting business-critical applications.

5.13 NET BEANS IDE

NetBeans IDE is a powerful integrated development environment primarily used for Java development, but it supports other languages as well. Here are some key uses and features of NetBeans IDE:

1. Java Development: NetBeans provides comprehensive support for Java development, including features like code editing, debugging, refactoring, and GUI design tools through its Swing GUI Builder.

2. **Project Management:** It offers robust project management features, allowing developers to organize their projects efficiently. NetBeans supports various project types, including Java SE, Java EE, Java ME, web applications, and more.
3. **Code Editor:** NetBeans comes with a feature-rich code editor with syntax highlighting, code completion, code templates, and automatic formatting. It also supports version control systems like Git, Subversion, and Mercurial, enabling seamless collaboration.
4. **Debugging:** NetBeans includes powerful debugging tools that help developers identify and fix bugs in their code quickly. It supports breakpoints, step-by-step debugging, variable inspection, and expression evaluation.
5. **Profiling:** NetBeans IDE provides built-in profiling tools for analyzing the performance of Java applications. Developers can identify performance bottlenecks, memory leaks, and optimize their code for better performance.
6. **GUI Design:** NetBeans includes a drag-and-drop GUI builder for creating Java Swing applications visually. This tool allows developers to design user interfaces easily and generate the corresponding Java code automatically.
7. **Support for Other Languages:** While NetBeans is primarily focused on Java development, it also offers support for other programming languages like HTML, CSS, JavaScript, PHP, and C/C++, with features such as code completion, syntax highlighting, and project templates.
8. **Plugins and Extensibility:** NetBeans has a rich ecosystem of plugins and extensions that extend its functionality. Developers can customize NetBeans to suit their specific needs by installing plugins from the NetBeans Plugin Portal or creating their own.
9. **Integration with Java EE and Web Technologies:** NetBeans provides seamless integration with Java EE technologies such as Servlets, JSP, EJB, and JSF, making it an excellent choice for developing enterprise Java applications. It also offers support for web development with frameworks like Spring and Struts.

10. Cross-Platform: NetBeans is a cross-platform IDE, meaning it runs on multiple operating systems including Windows, macOS, and Linux, providing flexibility for developers to work on their preferred platform.

Overall, NetBeans IDE offers a comprehensive set of tools and features that streamline the development process for Java and other programming languages, making it a popular choice among developers for building a wide range of applications.

CHAPTER – 6

RESULTS AND DISCUSSIONS

6.1 MODULES GENERATED

6.1.1 ACQUIRER SETTLEMENT BANK (MERCHANT'S BANK)

- **Login**

Login to the system

- **Merchants List**

All Merchants listed with the Service period details.

- **Merchant's Transactions**

All Merchant's transactions (for bank services) list, along with the status if a transaction is modified or tempered.

- **Notification**

Bank will send a notification to the merchant for renewing services.

- **Renew Agreement**

Bank can renew the agreement for services for the respective period.

- **Fee Transactions**

Charges per transaction fees from all customer transactions will be listed

6.1.2 MERCHANT

- **Login**

Login to the system.

- **My Profile**

- **Change Password**

- **Home Page**

Connect to Bank with Account Details and can see service period (Expiry date).

- **Service Transaction**

Merchant can pay for services to a bank account.

- **View Transactions**

Transactions made by Merchant for services are listed in this module, along with the status if a transaction is modified or tempered.

- **Customers Transactions**

All transactions of customers will be listed here, along with the status if a transaction is modified or tempered.

- **Customers Details**

All Customers list.

- **Notification**

Notification sent by Bank will be listed.

- **Renew Agreement**

Merchant can send approval of the agreement.

- **View Feedback**

Customers Feedback list.

6.1.3 CUSTOMER:

- **Register**

Register by filling in details.

- **Login**

Log in to the system.

- **My Profile**

Customer's registered details.

- **Change Password**

- **Merchant List**

All Merchants Listed in this module.

- **New Transaction**

Start to process the transaction to a specific merchant. (But the actual transaction is done with Acquirers (admin) account).

- **View Transaction**

All transactions of the respective customer will be listed here, along with the status if a transaction is modified or tempered.

- **Feedback**

Customer can share their experience of using the system.

6.2 RESULT

With the transaction history, the forementioned transaction project offers us a dependable, safe, and effective transaction. Our websites bottom section displays a gif video of every transaction that has ever been completed. Another option is to go to the Etherscan website, which keeps track of every transaction a user has made on the Ethereum network. Here, we conduct tests on the Goerli Test Ethereum network. The primary Ethereum that is shown in Metamask is activated when we launch our website onto a live network. Numerous networks display our genuine financial situation.

A repository containing the information of all transactions in an open block chain and completely on hand to all associated with the device. . Although the entire record is publicly available, the information of these involved inside the transaction are anonymous. Both transactions are checked towards the broking and a transaction affirmation token is sent after a couple of minutes. Using a couple of encryptions and having algorithms, the hassle of double spending is removed.

CHAPTER – 7

CONCLUSION

7.1 CONCLUSION

In addition to saving significant problems with centralized control of the system and involving agreement of the overall derived system, the paper concludes the operation of the transaction system over the antiquated that has been developed over the years by a system that has been established and changed by the given the foundations of block chain, which was formed in the earliest date of time but is capable of serving as an unconditional tool for creating an environment free of corruption. A great solution to the present problem is the block chain-based transaction system. Every new alteration to the block chain is made simpler, fundamental issues are solved, and peer-to-peer system sharing is made possible.

7.2 FUTURE SCOPE

Blockchain technology's primary future direction is cybersecurity. Data remains secure and verifiable even though the Blockchain system is open and distributed. Cryptography is used to encrypt data, eliminating security flaws like unauthorized data tampering. Currently, one of the most discussed techniques among businesses, blockchain employment is driving the IT sector's growth. Therefore, for individuals with the requisite skills and knowledge, blockchain technology presents enormous opportunities.

It can be useful for the validation and traceability of multicomponent transactions that require verification and traceability. It also provides safe transactions, reduces the cost of compliance, and speeds up the processing of data transfers. Additionally, it aids with contract management and product origin audits. It offers a variety of work descriptions to suit your interests and qualifications thanks to the wide range of applications available.

REFERENCES

- [1] Adarsh. P, Ashish Narvekar, Mayuresh. R, Romano. A, Sumedha. M, Salil. S, Yogini. L, Face Recognition Based Payment Processing System, pp. 12 – 30, 2022.
- [2] AiliLi, JianMao, JingdongBian, ShishiZhu , TaoWei, WenqianTian, ZhenkaiLiang, 2018 Issues and Challenges of Data Transaction Management in Cloud Environment, Support Vector Machine (SVM), DecisionTree (DT).
- [3] Bhabendu, Debasish, Jena, Mohanta, Panda, Soumyashree, Sobhanayak, Srichandan. Blockchain Technology: A Survey on Applications and Security Privacy Challenges, Doi: 2019100107. 10.1016/j.iot.2019.100107.
- [4] Böhme. R, Möser. M, An investigation into fraudulent tools in the Bitcoin ecosystem, APWG at Crime Researchers Summit, San Francisco, CA, NY, 2013, pp. 1-14, Doi: 10.1109 / CRS. 2013.6805780.
- [5] Deters. R and Rouhani. S, "Performance analysis of ethereum transactions in private blockchain", 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), pp. 70-74, 2017.
- [6] Gao. L, Wang. J, "Block Chain based Data Security Sharing Network Architecture Research", Journal of Computer Research and Development, vol. 54, no. 4, pp. 742-749, 2017.
- [7] Goranovic. A, Marcus. M, "Blockchain Applications in Microgrids: An Overview of Current Projects and Concepts", IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society, pp. 6153-6158, 2017.
- [8] Haapola. J, Pouttu. A, "P2P Model for Distributed Energy Trading Grid Control and ICT for Local Smart Grids", IEEE European Conference on Networks and Communications (EuCNC), pp. 1-5, 2017.
- [9] Hurlburt. G, "Could Blockchain Outlive Bitcoin ?," in IT Professional, vol. 18, no 2, pp. 12-16, Mar – Apr, 2016, i - Doi: 10.1109 / MITP.2016.21.

- [10] Lei Xu, Lin Chen, Nolan Shah, Nour Diallo, Weidong Shi, Yang Lu, Zhimin Gao, 2017. Enabling the Sharing Economy: Privacy Respecting Contract based on Public Blockchain. In Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts (BCC '17). Association for Computing Machinery, New York, USA.
- [11] Liang. Y, Wu. Z, "Secure Data Storage and Sharing System Based on Consortium Blockchain in Smart Grid", Journal of Computer Applications, vol. 37, no. 10, pp. 2742-2747, 2017.
- [12] Mubashir Husain. H, Martin. R et al., "Guest Editorial Special Section on Smart Grid and Renewable Energy Resources: Information and Communication Technologies with Industry Perspective", IEEE Transactions on Industrial Informatics, vol. 13, pp. 3119-3123, 2017.
- [13] Muneeswari. G, A Survey on Distributed Transactional Memory System with a Proposed Design for Parallel Processors, Decision Tree, Random Forest, Gradient Boosting (GBM), Generalized Linear Model and K-Nearest-Neighbors, 2022.
- [14] Pinna. A and Ruttenberg. W, "Distributed ledger technologies in securites post-trading", European Central Bank: Occasional Paper Series, no. 172, pp. 1-35, 2016.
- [15] She. W, Yang. X et al., "Transaction Certification Model of Distributed Energy based on Consortium Blockchain", Journal of University of Science and Technology of China, vol. 48, pp. 307-313, 2018.

APPENDIX

A. SOURCE CODE

```
package Algorithm;

import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;

public class AESDecryption {
    byte[] keyBytes = new byte[] {
        0x73, 0x2f, 0x2d, 0x33, (byte) 0xc8, 0x01, 0x73, 0x2b, 0x72,
        0x06, 0x75, 0x6c, (byte) 0xbd, 0x44, (byte) 0xf9, (byte) 0xc1
    };

    SecretKeySpec skeySpec;
    Cipher cipher;

    public AESDecryption() {
        skeySpec = new SecretKeySpec(keyBytes, "AES");
        try {
            cipher = Cipher.getInstance("AES");
            cipher.init(Cipher.DECRYPT_MODE, skeySpec);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public String toDeycrypt(String encrypted) {
        try {
            byte[] dec = new
sun.misc.BASE64Decoder().decodeBuffer(encrypted);
            byte[] utf8 = cipher.doFinal(dec);
            return new String(utf8, "UTF8");
        } catch (Exception e) {
            e.printStackTrace();
            return null;
        }
    }
}
```

```

    }
}
package Algorithm;

import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;

import sun.misc.BASE64Encoder;

public class AesEncryption {

    byte[] keyBytes = new byte[] {
        0x73, 0x2f, 0x2d, 0x33, (byte) 0xc8, 0x01, 0x73, 0x2b, 0x72,
        0x06, 0x75, 0x6c, (byte) 0xbd, 0x44, (byte) 0xf9, (byte) 0xc1
    };

    SecretKeySpec skeySpec;
    Cipher cipher;
    public AesEncryption() {
        skeySpec = new SecretKeySpec(keyBytes, "AES");
        try {
            cipher = Cipher.getInstance("AES");
            cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    public String toEncrypt(byte[] content) {
        try {

            byte[] utf8 = cipher.doFinal(content);
            return (new BASE64Encoder()).encode(utf8);

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

```

        return null;
    }
}

package Algorithm;

import java.util.List;

public class Block {

    private int previousHash;
    private List<Transaction> transactions;

    public Block(int previousHash, List<Transaction> transactions) {
        this.previousHash = previousHash;
        this.transactions = transactions;
    }

    public int getPreviousHash() {
        return previousHash;
    }

    public void setPreviousHash(int previousHash) {
        this.previousHash = previousHash;
    }

    public List<Transaction> getTransactions() {
        return transactions;
    }

    public void setTransactions(List<Transaction> transactions) {
        this.transactions = transactions;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        Block block = (Block) o;
        if (previousHash != block.previousHash) return false;

```

```

        return transactions != null ? transactions.equals(block.transactions) :
block.transactions == null;
    }
    @Override
    public int hashCode() {
        int result = previousHash;
        result = 31 * result + (transactions != null ? transactions.hashCode() : 0);
        return result;
    }
}
package Algorithm;

import java.util.ArrayList;
import java.util.List;

public class Blockchain {
    //instantiating the list of blocks
    private List<Block> blockChain;
    public Blockchain() {
        //further, instantiating the list as an ArrayList
        this.blockChain = new ArrayList<>();
    }
    public void addBlock(Block block) {
        this.blockChain.add(block);
    }
    public List<Block> getBlockChain() {
        return this.blockChain;
    }
    public int size() {
        return this.blockChain.size();
    }
    @Override
    public String toString() {
        String blockChain = "";

```

```

    for(Block block : this.blockChain)
    blockChain+=block.toString()+"\n";
    return blockChain;
}
}

```

```

package Algorithm;

```

```

import java.security.MessageDigest;
public class SHA256Helper {
    public static String generateHash(String data) {
        try {
            // we will get the instance of SHA256
            MessageDigest digest = MessageDigest.getInstance("SHA-256");
            //by using digest() function, we will get has as one dimensional byte array
            byte[] hash = digest.digest(data.getBytes("UTF-8"));
            //we wanna use hexadecimal values not bytes in our program
            // will covert byte into hexadecimal
            StringBuffer hexadecimalString = new StringBuffer();
            for (int i = 0; i < hash.length; i++) {
                String hexadecimal = Integer.toHexString(0xff & hash[i]);
                if (hexadecimal.length() == 1) hexadecimalString.append('0');
                hexadecimalString.append(hexadecimal);
            }
            return hexadecimalString.toString();
        }
        catch (Exception e) {
            throw new RuntimeException(e);
        }
    }
}

```

```

package Algorithm;

```

```

public class Transaction {
    private String sourceName;
    private String destinationName;
    private Long sum;

    public String getSourceName() {
        return sourceName;
    }
    public void setSourceName(String sourceName) {
        this.sourceName = sourceName;
    }
    public String getDestinationName() {
        return destinationName;
    }
    public void setDestinationName(String destinationName) {
        this.destinationName = destinationName;
    }
    public Long getSum() {
        return sum;
    }
    public void setSum(Long sum) {
        this.sum = sum;
    }

    public Transaction(String sourceName, String destinationName, Long sum) {
        this.sourceName = sourceName;
        this.destinationName = destinationName;
        this.sum = sum;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        Transaction that = (Transaction) o;

```

```

    if (sourceName != null ? !sourceName.equals(that.sourceName) :
that.sourceName != null) return false;
    if (destinationName != null ? !destinationName.equals(that.destinationName) :
that.destinationName != null)
return false;
    return sum != null ? sum.equals(that.sum) : that.sum == null;
}
@Override
public int hashCode() {
    int result = sourceName != null ? sourceName.hashCode() : 0;
    result = 31 * result + (destinationName != null ? destinationName.hashCode() : 0);
    result = 31 * result + (sum != null ? sum.hashCode() : 0);
    return result;
}
}

```

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Stylized by NodeThirtyThree + Free CSS Templates</title>
<meta name="keywords" content="" />
<meta name="description" content="" />
<meta name="robots" content="all" />
<link href="screen.css" rel="stylesheet" type="text/css" media="screen" />
<style type="text/css">
<!--
.style6 {
    color: #000000;
    font-size: 36px;
    font-weight: bold;
}
.style7 {color: #993300}
.style8 {
    color: #FFFFFF;
    font-size: 24px;

```

```

}
-->
</style>
</head>
<body><form action="" method="post">
<div class="style8" id="logo">
    Block chain based transaction and settlement system</div>
<div id="menu" align="center">
    <ul>
        <li><a href="admin.jsp" accesskey="1" title="">Home</a></li>
        <li><a href="accountcreation.jsp" accesskey="2"
title="">Account</a></li>
        <li><a href="customer.jsp" accesskey="3" title="">Customer</a></li>
        <li><a href="monthlyreport.jsp" accesskey="5" title="">report</a></li>
        <li><a href="deposit.jsp" accesskey="6" title="">Deposit</a></li>
        <li><a href="reportgeneration.jsp" accesskey="7" title="">AML
Report</a></li>
        <li><a href="block.jsp" accesskey="8" title="">Block</a></li>

        <li><a href="login.jsp" accesskey="9" title="">Log OUT</a></li>
    </ul>
    <hr />
</div>
<div id="bg">
<table width="1251" height="319" border="0" bgcolor="#FFFFFF">
<tr>
<td height="295"><table width="1259" border="0">
<tr>
<td width="209">&nbsp;</td>
<td width="637"><table width="700" border="0">
<tr>
<td width="697" height="158" align="center"><span class="style10
style6">WELCOME TO <span class="style7">ADMIN LOGIN</span> FORM
</span></td>

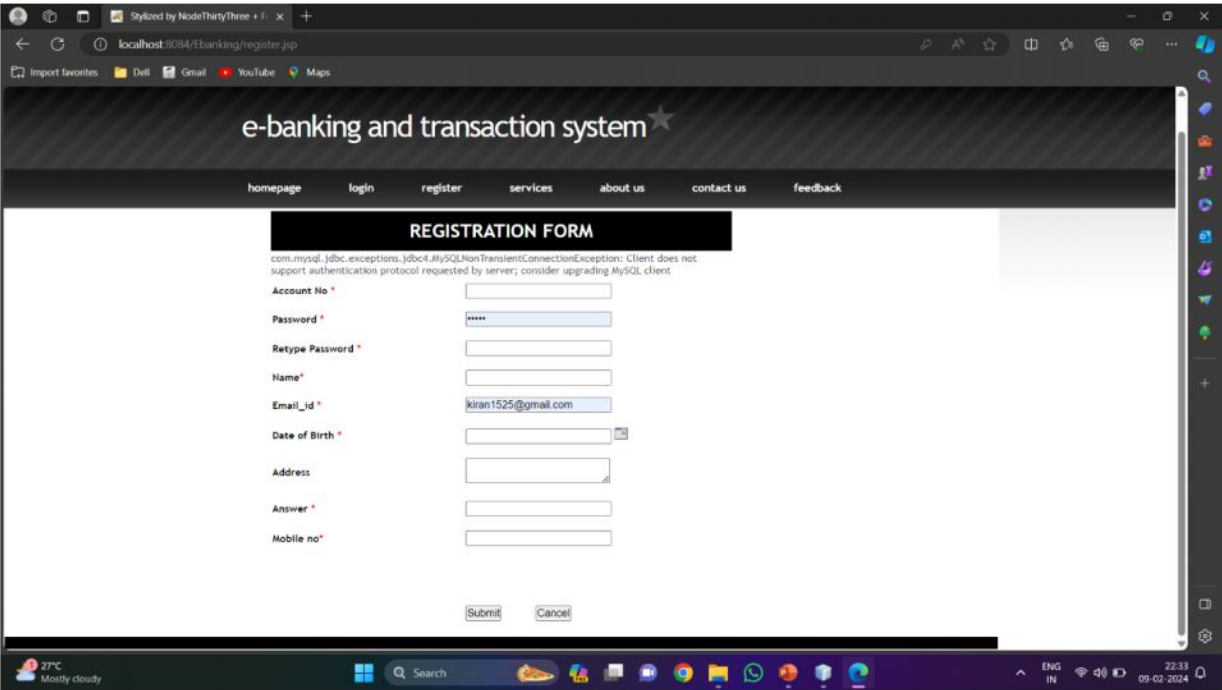
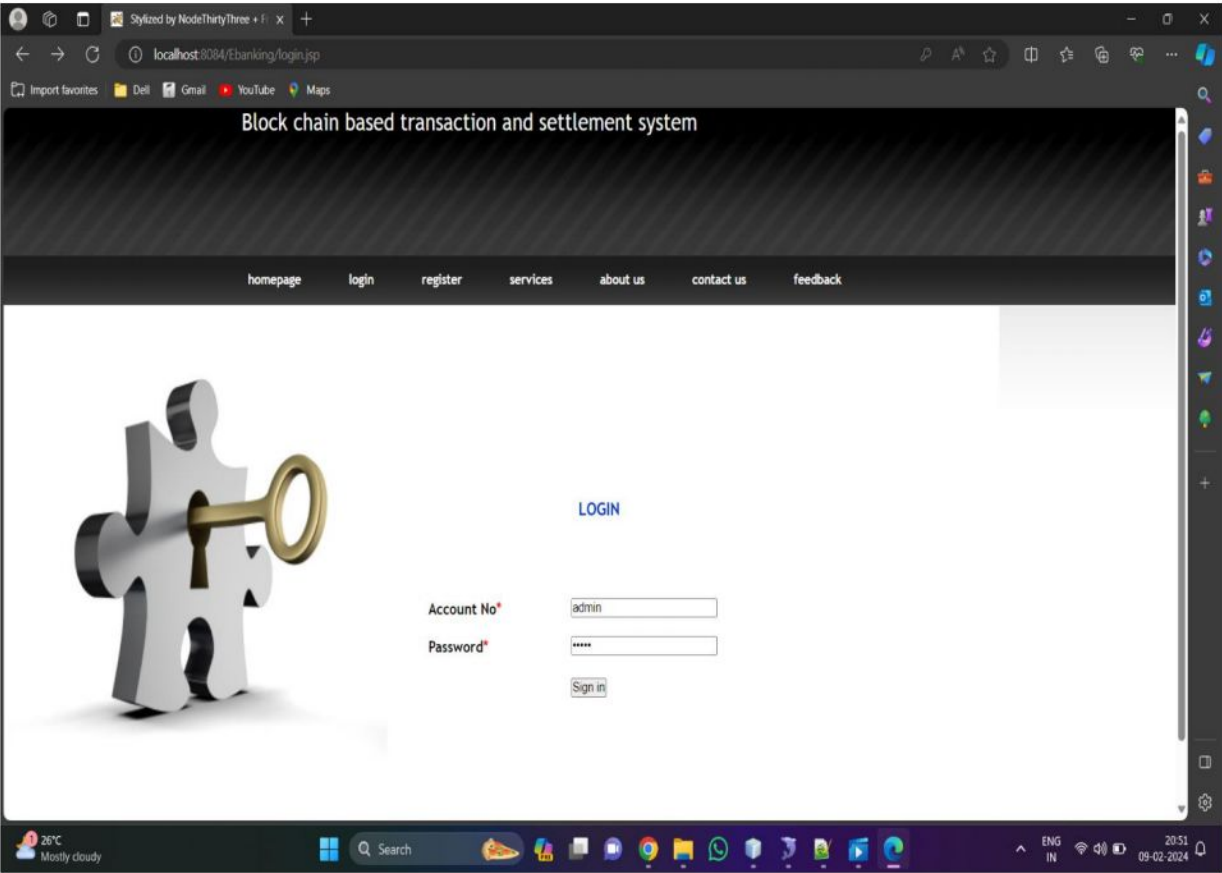
```

```

</tr>
<tr>
  <td><p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <p>&nbsp;</p></td>
</tr>
<tr>
  <td>&nbsp;</td>
</tr>
</table></td>
<td width="379">&nbsp;</td>
</tr>
</table></td>
</tr>
  <tr>
    <td background="images/button3.png" bordercolordark="#000000">&nbsp;</td>
  </tr>
</table>
<!-- end #page -->
</div>
<!-- end #bg -->
</form>
</body>
</html>

```

B. SCREENSHOTS



localhost:8080/Ebanking/accountcreation.jsp

home account customer report deposit aml report block log out

ACCOUNT CREATION

Account No *	12345603	Annual Income *	
Name *		Occupation *	Select Occupation
Date of Birth *		Mobile no *	
Age *		Account Type	Select Account Type
Current Address *		Pancard *	
Permanent Address		Smart Card *	
City *		Expire Date	
Country *		Gender	Select the Gender
State *		CreditCard Pin *	
Bank Name *		Branch Name *	
Email -Id *			

localhost:8080/Ebanking/aloan.jsp

Hai 12345603 WELCOME TO city BANK

NAME	chandu	INSURANCE(If ANY)	☐ YES ☐ NO
FATHER NAME		INCOME TAX ASSESSMENT	
DATE OF BIRTH		DEPOSIT AMOUNT	
AGE		NOMINEE	
NATIONALITY	SELECT NATIONALITY	RATION CARD NO	
RELIGION	SELECT RELIGION	PROPERTY DETAILS	SELECT THE PROPERTY
COMMUNITY		ANY PREVIOUS LOAN?	☐ YES ☐ NO
EDUCATIONAL QUALIFICATION	SELECT EDUCATIONAL	LOAN APPLIED FOR	SELECT THE LOAN TYPE
DESIGNATION		BANK ACCOUNT NO	12345603
MONTHLY INCOME		PANCARD NO	
EXPERIENCE		PREFERRABLE AMOUNT	

localhost8080/Ebanking/fundtransfer.jsp

Import favorites Dell Gmail YouTube Maps

homepage change password fundtransfer apply loan loan status report log out



FUND TRANSFER

Account no	
Name	
Amount	
Bank Name	
Branch Name	
Creditcard Pin	

B. RESEARCH PAPER

BLOCK CHAIN BASED TRANSACTION AND SETTLEMENT SYSTEM

Y.CHANDRA KIRAN[1], M. GANESH REDDY[2], Mr. JABEZ.J[3]

[1]Under Graduate student, Department of CSE, Sathyabama Institute of Science and Technology, Chennai, India. [2]Assistant Professor, Department of CSE, Sathyabama Institute of Science and Technology, Chennai, India. Chandra2k99@gmail.com, jabez.it@sathyabama.ac.in

ABSTRACT: Fund transfers could become instantaneous, more affordable, and more secure thanks to the growing influence of block chains, which have the potential to disrupt the financial services sector. The present approach is not sufficiently safe to provide 100% protection against fraud due to increased manual labor and inadequate data security. All that blockchain is a chain composed of nodes, or blocks. The main work is completed by these process nodes. Cryptography is used to link these blocks together. This system will handle customer data more quickly, efficiently, and with zero chance of data loss. It also features friendly user interfaces for banking. In addition to facilitating trade, block chain is a tamper- and theft-resistant concept that gets rid of mistakes and duplication.

Keywords: Block chain; Payments; Bibliometric Analysis; Co-authorship Analysis; Keyword Co-concurrent

INTRODUCTION:

It is a digital, disbursed and public ledger block chain. Block chain generation changed into first used today below the name of BITCOIN (digital foreign money), however it ought to offer dependable and cozy information switch properties in distributed P2P networks for different programs. Block nodes are connected cryptographically and chronologically. The block affords an possibility to distribute newspapers in a decently based manner and is a key concept. Unlike a centralized machine wherein information are stored in a particular enterprise along with a banking agency or government, a block chain shares records amongst its individuals. This is what makes the critical block. This approach that data are written and stored on the network, and its individuals are chargeable for updating and monitoring them. Individual data are continuously synchronized

across different members of the open network, multiplying statistics samples via a common retention machine, which ensures that no person or entity owns the facts. When an preliminary transaction or amendment of a stay transaction enters a block chain, a couple of nodes inside the block chain usually use steps (algorithms) to calculate, confirm, and affirm the block chain's beyond history. Introduction. When all the nodes attain agreement or consensus about the validity of the transaction history and signature, this new block is brought to the particular transaction chain. If numerous nodes in the integration round do now not receive integration, the block will now not be added to the chain. This type of paintings permits this block chain gadget to be implemented with out the want for a significant block chain authority where each block has a certain range of transactions. It gives decentralized, immutable data storage to be used via a community of customers, creating property and operating as a not unusual provider that records all transactions. Thus, block chain creates a verifiable and irrefutable open report of records that is inexpensive, quicker and extra at ease than some other centralized device..

OBJECTIVE:

Therefore, it's far important to establish a extra comfy, obvious and green transaction gadget to prevent fraud, lessen disputes and sell sustainable land control. Block chain technology has the strength to deal with those demanding situations even as offering a decentralized, transaction evidence and obvious machine for transactions.

LIERATURE REVIEW:

Face Recognition Based Payment Processing System,2023

The stream of multiple playing cards increases each second in diverse areas together with shopping, eating places, inns and cinemas, and on line transactions associated with the acquisition of tickets. Hence the question of the human beings sporting their foreign playing cards and cash and keeping the cards/coins secure for non-workout use; A case of theft

Issues and Challenges of Data Transaction Management in Cloud Environment,

In this article, we goal to enhance the detection strategies of phishing using gadget mastering strategies. Specifically, we endorse a gaining knowledge of-based analysis mechanism to detect format similarities that may be used to come across phishing pages. Our experimental consequences show that our technique is correct and powerful in detecting hacked pages.

A Literature Review of Block chain-Based Applications in Supply Chain,2022

This paper explores this studies hole and thru a qualitative evaluation, the examine shows a barrier to transferring programs inside the path of green automation. Appropriate consensus mechanisms enhance performance in numerous deliver chain environments, at the same time as clever contracts play an essential role in protection. Therefore, this text has researched articles and articles on the topic in latest years thru a combination of keywords.

A Block chain Based Decentralized Transaction Settlement System in Banking Sector,2021

The Indian banking device is one of the most complex banking structures within the world. The cutting-edge infrastructure used by the Indian banking gadget is based totally on a actual-time complicated settlement system and follows a centralized shape. The transaction manner is slower and incredibly greater bulky because of this centralized structure. It spends a lot of cash on safety and recuperation purposes. A real-time total device-based solution calls for excessive requirements for safety, resilience and overall performance. The transition from a traditional device to a blockchain platform isn't always the primary difficulty, but the fundamental concept is to create a gadget that gives safety, privacy, and a decentralized money trade mechanism. A new system for banking and decentralized garage offerings based totally on the Ethereum price platform is proposed right here.

A Systematic Literature Review of Block chain Technology for Smart Villages, 2022

Blockchain generation is the next step in innovation and improvement and there are nevertheless many programs to be determined in sustainable rural development. The motive of this

newsletter is to offer a clear assessment of studies completed inside the subject of rural improvement to fill the gaps in future work with higher studies ideas to make rural areas livable and sustainable. Places To perform this kind of review, a systematic studies technique is used based totally on the standards of engaging in exploratory analysis. During the length 2010-2021, 112 articles had been carefully decided on to behavior a systematic assessment. This review presents a comprehensive and concise series of research proposals, outcomes and potential destiny applications to apprehend the concept of captive villages for rural development using technological boundaries..

Block chain-based Payment Systems: A Bibliometric & Network Analysis, 2022

China's countrywide herbal technology fund presents investment for lots instructional papers, and America is becoming a completely influential country. The observe also located that the literature on block chain-based totally payments is grouped into five classes. These departments are laptop technology, engineering, choice technological know-how, arithmetic and commercial enterprise. Co-opted analytical networks of connections among nations, authors, and establishments round the arena create specific clusters that increase studies productiveness. In précis, this study creates an analytical map of the studies panorama that may guide future studies.

EXISTING SYSTEM:

Front-give up deliver chains involve the trade of massive volumes of products. Therefore, incomplete transfer and loss of work is inevitable. This paper affords a excessive-capability group property transfer protocol that integrates blocks to switch massive-scale merchandise efficiently and keep away from conflicts for switch of completion. This protocol combines the tool's authentication records with its precise block cope with to shape the first block transaction. The block become used to become aware of the manufacturers of the authentic merchandise in order no longer to be faked, and the group test to keep away from disputes approximately the integrity of the alternate. The proposed protocol has the potential to face up to common radio frequency identity off-web site transmission attacks, which suggests its protection. A safety test device is used to demonstrate whether or not an on-chain clever agreement can resist attacks at the block chain. The analysis and experimental effects found out that the proposed protocol requires much less message transmission and half the computation time in comparison to present protocols. Therefore, the proposed protocol is extra green for moving the

ownership of numerous tags on the equal time, with much less computation time, the extra suitable for transferring the possession of a quantity of goods in a commercial transaction.

Disadvantages:

They changed things without the man or woman's knowledge.

- The item does no longer have a coin's choice.
- And it has no security.
- Also, now not a lake developer.

PROPOSED SYSTEM:

Banking plays an critical role for brick-and-mortar and on-line shops to conduct their enterprise and provide handy fee alternatives. If the card issuer authorizes the transaction, the charge is transferred from the issuing financial institution to the merchant with the help of a financial institution reconciliation. Our blockchain-primarily based transaction and agreement machine connects merchants to their banks and offerings, facilitating business transactions. In this mission, the the front-cease consists of HTML, CSS and JavaScript and the returned-end includes ASP.NET. The IDE used is Visual Studio and MS SQL database. In this system, the financial institution has a listing of all the traders with every time reserve. This listing consists of all service provider transactions and the popularity of change or stability transactions. The bank sends a notification to the merchant to renew the services, that can renew the carrier contract within a certain period. All clients are charged for all banking transactions. The service provider maintains the logs within the bank and prevents the carrier from passing the time or date. They assume they are able to pay for his or her services the usage of the bank. The fame of any revised or adjusted transaction made by means of the service provider for the Services is indexed with the transaction. The merchant may additionally send a agreement confirmation. The consumer has to register to log into the machine with their simple information. They can see a list of all of the merchant information. Although the buyer can initiate the transaction process with a certain service provider, the actual transaction is controlled by way of the client's account (administrator). You will acquire a list of all transactions and records about corrected or modified transactions. Customers can deliver remarks with the aid of sharing their reviews.

Advantages:

- Easy to put.
- It is user pleasant.
- All activities are monitored.
- If the transaction is suspicious, the bank is without delay knowledgeable.

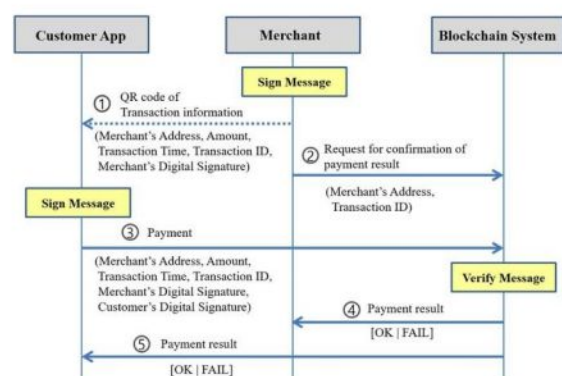
Goals:

We are going to enforce this assignment in the block:

Basic dreams of UML layout:

1. Provide customers with clean, clean-to-use visible phrase fashions in order to create and trade meaningful fashions.
2. Provide extension and commands for extending unique key principles. Provide extension and unique commands to increase key ideas
1. Be unbiased from programming languages and improvement techniques.
2. Provide a formal basis for expertise language formation.
3. Promote the boom of the OO equipment market place. Four. Support high-stage development ideas, together with collaborations, frameworks, models, and additives five. Integrate first-rate practices

SYSTEM ARCHITECTURE



SYSTEM REQUIREMENTS:

Hardware Requirement

- Laptop or PC
- Windows 7 or higher

- I3 processor system or higher
- 4 GB RAM or higher
- 100 GB ROM or higher

Software Requirement

- Laptop or PC
- Java

MODULES:

Acquirer settlement bank (merchant's bank)

Login Login to the computer

- Merchants list all merchants with every running time.
- Merchant Transactions statistics all transactions (for banking offerings) along with their popularity if the transaction is changed or cleared.
- **Login** Login to the computer
- Merchants list all merchants with every running time.
- **Login** Login to the computer
- Merchants list all merchants with every running time.
- Merchant Transactions statistics all transactions (for banking offerings) along with their popularity if the transaction is changed or cleared.
- Notification Bank sends a notification to the merchant to renew the offerings.
- The financial institution may additionally renew the carrier settlement on the time of renewal of the settlement.
- Transaction Fees Transaction expenses are indexed for all commercial enterprise clients..
- Notification Bank sends a notification to the merchant to renew the offerings.
- The financial institution may additionally renew the carrier settlement on the time of renewal of the settlement.
- Transaction Fees Transaction expenses are indexed for all commercial enterprise clients..
- The financial institution may additionally renew the carrier settlement on the time of renewal of the settlement.

- Transaction Fees Transaction expenses are indexed for all commercial enterprise clients. Merchant

- **Login** Login to the system.
- **My Profile**
- **Change Password**
- **Home Page** Login to bank with account info and think about service period (difficulty date)).
- **Service Transaction** Merchant can pay for services to a bank account.
- **View Transactions** Transactions by way of the service provider for services are indexed in this block, alongside the reputation of the transaction that has modified or declined.
- **Customers Transactions**
All client transactions are listed here with the repute if the transaction has modified or is pending.
- **Customers Details**
- All Customers list.
- **Notification** Notification sent by Bank will be listed.
- **Renew Agreement** Merchant can send approval of the agreement.
- **View Feedback** Customers Feedback list.

Customer:

Register Register by filling in details.

- **Login** Log in to the system.
- **My Profile** Customer's registered details.
- **Change Password**
- **Merchant List** All Merchants Listed in this module.
- **New Transaction** Start through processing the transaction for particular traders. (however the real element is executed because of the "administrator)).
- **View Transaction** All the activities of the respective client are listed right here along with their repute, if the object is changed or back.
- **Feedback** Customer can share their experience of using the system.

CONCLUSION:

A repository containing the information of all transactions in an open block chain and completely on hand to all associated with the device. . Although the entire record is publicly available, the information of these involved inside the transaction are absolutely anonymous. Both transactions are

checked towards the broking and a transaction affirmation token is sent after a couple of minutes. By the use of a couple of encryption and having algorithms, the hassle of double spending is removed.

FUTURE TRENDS:

Due to the decentralized nature of the Bit coin block chain, all operations may be seamlessly monitored on a non-public node or via using block chain explorers. Each node has its very own chain model, which is updated as new blocks are confirmed and introduced. This way you may track Bit coin anywhere you go if you need.

For example, exchanges had been hacked within the past, ensuing in a big loss of crypto currency. Hackers can stay nameless (besides for his or her wallet cope with) and may effortlessly mine the wallet due to the fact pockets addresses are posted in the block chain.

In truth, information saved on the Bit coin block chain are (nicely, broadly speaking) encrypted. That is, only the speaker can display his identification. As a result, block chain customers can stay anonymous whilst maintaining transparency.

REFERENCES:

- [1] M. Möser, R. Böhme noD. Breuker, "An investigation into fraudulent tools in the Bitcoin ecosystem," 2013 APWG at Crime Researchers Summit, San Francisco, CA, 2013, pages 1-14, doi: 10.1109 / CRS. 2013.6805780.
- [2] Mohanta, Bhabendu & Jena, Debasish & Panda, Soumyashree & Sobhanayak, Srichandan. (2019). Blockchain Technology: A Survey on Applications and Security Privacy Challenges. 8. 100107. 10.1016/j.iot.2019.100107.
- [3] D. A. Wijaya, "Extending asset management system functionality in bitcoin platform," 2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA), Tangerang, 2016, pp. 97-101, doi: 10.1109/IC3INA.2016.7863031.

[4] K. Saito and H. Yamada, "What's So Different about Blockchain? — Blockchain is a Probabilistic State Machine," 2016 IEEE 36th International Conference on Distributed Computing Systems Workshops (ICDCSW), Nara, 2016, pp. 168-175, doi: 10.1109/ICDCSW.2016.28.

[5] G. Hurlburt, "Could Blockchain Outlive Bitcoin ?," in IT Professional, vol. 18, no. 2, pages 12-16, Mar.-Apr. 2016, i- doi: 10.1109 / MITP.2016.21.

[6] Lei Xu, Nolan Shah, Lin Chen, Nour Diallo, Zhimin Gao, Yang Lu, and Weidong Shi. 2017. Enabling the Sharing Economy: Privacy Respecting Contract based on Public Blockchain. In Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts (BCC '17). Association for Computing Machinery, New York, NY, USA, 15–21.

D. CERTIFICATE

