

BLOCKCHAIN BASED PERSONAL IDENTITY SECURITY SYSTEM

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

By

Hemanth Kumar K (40110563)

Katta Thanay (40110577)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

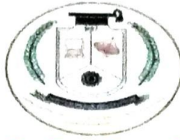
**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Category – 1 University by UGC

Accredited with Grade “A++” by NAAC| 12B Status by UGC| Approved by AICTE

**JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119**

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | 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 **KARWAN HEMANTH Kumar (40110563)** and **KATTA THANAY (40110577)** who carried the Project entitled **"BLOCKCHAIN BASED PERSONAL IDENTITY SECURITY SYSTEM"** under my Supervision from November 2023 to April 2024.


13/4/2024
Internal Guide

Ms. ANANDHA SREE R, 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. LAKSHMANAN M.E., Ph.D.,
Jeppiaar Nagar, Rajiv Gandhi Salai,
Chennai - 600 119.

Submitted for Viva voce Examination held on

15/4/24


Internal Examiner


External Examiner

DECLARATION

I, **Karwan Hemanth (Reg.No - 40110563)**, hereby declare that the Project Report entitled **“BLOCKCHAIN BASED PERSONAL IDENTITY SECURITY SYSTEM”** done by me under the guidance of **Ms. ANANDHA SREE R, M.E., (Ph.D.,)** is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE:

PLACE: Chennai

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, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms. ANANDHA SREE R, M.E.,(Ph.D.)** for her 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

Identity theft is the unauthorized acquisition of another person's confidential information in order to misuse it. Organizations and individuals should exercise caution when it comes to protecting their identities in order to avoid fraud as a result of identity theft. This information is freely available to attackers in user profiles. Attackers use this information to obtain additional information without raising suspicions about a final attack, identity theft, or fraud. Our block chain based Personal Identity Security System assists in securely storing personal identity data without fear of it being compromised or lost. In this system, the admin can access all the users and verify those accounts. The admin has access to all of the user's documents that have been uploaded to the system. They can view all of the user's activity logs. The admin can view the status and see if any fabrication is taking place. The admin can view the user's complaints. The user can view if logs have been updated or deleted, as well as the status of fabricated document details. If a user discovers a fabrication or an action that they did not perform, they can notify the admin. The user can view all of the details by scanning the QR Code. They can add and verify all the documents with the ledger.

Chapter No.	TITLE	Page No.
	ABSTRACT	V
	LIST OF FIGURES	VIII
	LIST OF ABBREVIATIONS	IX
1	INTRODUCTION	1
2	LITERATURE SURVEY	2
	2.1 Inferences in Literature system	2
	2.1.1 E-Voting System	2
	2.1.2 Online Voting System	2
	2.1.3 Online Voting System Using Face Recognition	3
	2.1.4 Dual Authentication Voting System	3
	2.1.5 Unique Id Based Voting System	3
3	REQUIREMENT ANALYSIS	4
	3.1 Function Requirements	4
	3.1.1 Usability	4
	3.1.2 Robustness	4
	3.1.3 Security	5
	3.1.4 Reliability	5
	3.1.5 Compatibility	5
	3.1.6 Flexibility	5
	3.1.7 Safety	5
	3.2 Non-Function Requirements	6
	3.3 Open Problems In Existing System	7
4	DESCRIPTION OF PROPOSED SYSTEM	9

4.1	Working	9
4.2	Architecture / Overall Design of Proposed System	10
4.3	Selecting Methodology	11
	4.3.1 Admin Module	12
	4.3.2 User Module	12
	4.3.3 Crud Modue	12
	4.3.4 Documentation Module	12
4.4	UML & URL Diagrams	13
	4.4.1 Data Flow Diagram	15
	4.4.2 Use Case Diagram	16
	4.4.3 Activity Diagram	17
	4.4.4 Sequence Diagram	18
5	IMPLEMENTATION DETAILS	19
	5.1 Test Plan	19
	5.2 Verification	19
	5.3 Validation	19
	5.4 Basics of Software Testing	19
	5.4.1 Black Box Testing	20
	5.4.2 White Box Testing	20
6	RESULTS AND DISCUSSION	21
7	CONCLUSION	23
	REFERENCES	24
	APPENDIX	26
	A. Source Code	26
	B. Screenshots	50
	C. Research Paper	61

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
4.1	System Architecture Diagram	9
4.2	Data Flow Diagram	13
4.3	Use Case Diagram	14
4.4	Activity Diagram	15
4.5	Sequence Diagram	16

LIST OF ABBREVIATIONS

ABBREVIATION

EXPANSION

IDMS	Identity Management System
P2P	Peer to Peer
BC	Blockchain
SSI	Self-sovereign Identity
SHA	Secure Hash Algorithm
AES	Advanced Encryption

CHAPTER-1

INTRODUCTION

Data leaks or identity fraud (e.g., stealing of credentials or credit card information) frequently affect profitable loss and drop of trust in the identity providers. Within this environment, new identity operation results that satisfy the factual requirements must be explored. The particular data is traditionally stored in a centralized system, which makes it possible for hackers or bushwhackers to achieve their vicious pretensions by robbery/misusing/manipulating these data in this centralized system. To gain access to the services handed by these online operations, the customers must validate their identity by giving a valid identifier. This identifier can take numerous forms similar to particular information contained on a social security card, automobile license, passport, birth certification, and academy or work badges. Identity Management is the process of authenticating, authorizing, and identifying a thing (individually or a group of people or the system) to access one or further many resources. The biggest point of the blockchain is its decentralization where the whole database is maintained by all the bumps on the network. An agreement medium ensures that the creation and modification of data are agreed upon by all the bumps or the maturity of the bumps. Using this fashion the blockchain has the features of high security, not easy to be tampered with, and so on. This also helps to give the user safe and secure platform to secure his files. Blockchain technology heralds a new era of personal identity security, offering decentralized, immutable and privacy-enhancing solution. This technology allows people to take back control of their personal information, making it safer and building trust when they interact online. And also through cryptographic hashing and consensus mechanisms, blockchain ensures the integrity and performance of identity and performance of identity information.

CHAPTER-2

LITERATURE SURVEY

2.1 INFERENCES IN LITERATURE SURVEY

2.1.1 E-VOTING SYSTEM

The Online electoral system may be a web based application. The system has a centralized database to keep records of all the Voters and Candidates and Final Results. This Online Voting System is based on e-voting. This web based system is time saving, workload reduced information available at time and it provides security for the info. The election commission will maintain this website. This is an easy, safe and secure method that takes a minimum of your time. The word VOTE means to settle on from an inventory, to elect or to work out.

2.1.2 ONLINE VOTING SYSTEM

The project is mainly aimed at providing a secured and user-friendly Online Voting System. The problem of voting is still critical in terms of safety and security. This system deals with the design and development of a web based voting system using fingerprint and Aadhaar card in order to provide a high performance with high security to the voting system.

2.1.3 ONLINE VOTING SYSTEM USING FACE RECOGNITION

The basic idea of this system is to create an Online Voting System that will help to suppress deceive of the manual voting system and also the prior versions of online voting by camera for Face Recognition and OTP generation. We are also implementing location free voting system to the voters for whom it is not possible to come at the voting location (hometown).

2.1.4 DUAL AUTHENTICATION VOTING SYSTEM

This project focuses on a voting system using RFID and Finger print technologies. Each user is provided a voter's ID in the form of RFID Tag. The hardware design has a Finger print scanning sensor which is used to compare the finger print of the user with the pre-stored finger print of the user.

2.1.5 UNIQUE ID BASED VOTING SYSTEM

Elections are an extremely important part of the growth process in every country. Elections provide people with the opportunity to choose the leader they feel best represents their values, which in turn contributes to the growth of individuals. However, those who aspire to be leaders but are overly ambitious sometimes make poor decisions that hinder their progress. The use of manual elections has been shown to increase the likelihood of electoral fraud.

CHAPTER-3

REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analyzing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

3.1 FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the all requirements of particular software systems including the all functions, performance and security requirements. The function of the system is depends mainly on the quality hardware used to run the software withgiven functionality.

3.1.1 USABILITY

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

3.1.2 ROBUSTNESS

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

3.1.3 SECURITY

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot be access the system there by providing high security.

3.1.4 RELIABILITY

It is the probability of how often the software fails. The measurement of is often express MTBF(Mean Time Between Failures). And the requirement is needed in ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

3.1.5 COMPATIBILITY

It is supported by version above all web browsers. Using any web server like local host makes the system real-time experience

3.1.6 FLEXIBILITY

These requirements determine the resources required, this is easy and the flexibility of the project is that it has the ability to run on different and various environment being executed by different users.

3.1.7 SAFETY

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

3.2 NON- FUNCTIONAL REQUIREMENTS

PORTABILITY

It is the usability of the same software in different environments. The project can be run in any operating system. By prioritizing the data compatibility and adhering to industry standards, the system can be effortlessly integrate with various operating systems and devices with simple setup and maintenance, ensuring smooth and hassle-free user experiencing.

PERFORMANCE

These requirements determine the resources required, time interval and throughout, it helps with everything that deals with the performance of the system. The system ensures swift authentication and identity verification process. Scalability features enable the system to accommodate growing user base and increasing transaction volumes without sacrifice response.

ACCURACY

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective. With accurate data validation techniques verify the integrity of identity theft. Ultimately, prioritizing accuracy instills trust in the system ability to securely manage personal identities.

MAINTAINABILITY

Maintainability basically defines that how easy it maintain the system. It means that how easy it is to maintaining the system, analyze, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

HARDWARE REQUIREMENTS

- System : Pentium i3 Processor
- Hard Disk : 500 GB.
- Monitor : 15" LED
- Input Devices : Keyboard, Mouse, Ram: 2 GB

SOFTWARE REQUIREMENTS

- Operating system : Windows 10
- Coding Language : Java

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

Identity Management System (IDMS) refers to how users or individuals are identified and authorized to use organizational systems and services. Since traditional identity management and authentication systems rely heavily on a trusted central authority, they cannot mitigate the effects of single points of failure. As a decentralized and distributed public ledger in a peer-to-peer (P2P) network, Blockchain (BC) technology has garnered a considerable amount of attention in the field of IDMS in recent years. Through Self-Sovereign Identity (SSI), users can implementation of a BC-based IDMS can significantly increase the degree of privacy and security of a user's SSI. However, the integration of BC-based IDMS to provide a user with SSI is still an unorganized area of research in its early stages of development. This article presents an extensive literature review of state-of-the-art academic publications as well as commercial market offerings regarding the applicability of BC-based SSI solutions. It also provides a detailed preliminary regarding the building blocks of blockchain technology and a progressive roadmap of IDMS solutions. In order to develop an effective BC-based IDMS solution that focuses on securing a user's SSI, this article outline has five essential components of a BC-based on the IDMS: authentication, integrity, privacy, trust, and simplicity. Furthermore, we perform a security analysis that outlines

several types of adversarial threats that can cause potential damage to the BC-based IDMS. We identify and discuss associated issues and challenges by analyzing several notable BC- based IDMS solutions in academic literature. We also highlight potential research gaps and provide future research scope.

PROBLEM STATEMENT :

The current systems possess number of problems such as proxies, vulnerable to data theft but blockchain proves to be the solution for this type of Identity related problem.

RISK AND ANALYSIS OF PROJECT:

The main objective is to maintain a decentralized system to store identity of a person and protect individual identities and massively reduce fraudulent activities.

DIS – ADVANTAGES OF EXISTING SYSTEM :

- Time consumption.
- Less security.
- Server issues happen often.

CHAPTER- 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 WORKING

In this system, the admin can all the users and verify those accounts. The admin has access to all of the user's documents that have been uploaded to the system. They can view all of the user's activity logs. The admin can view the status and see if any fabrication is taking place. The admin can view the user's complaints. To log in to the system, the user must first register; once the account has been verified by the admin, the user can log in with a username and password. The user can update the profile as needed. The user can manage their documents in the system by adding, updating, and deleting them. All documents uploaded by the user are also verified by the admin. The user can view the QR Code as well as the status of the documents that are being fabricated. The user can view if logs have been updated or deleted, as well as the status of fabricated document details. If a user discovers a fabrication or an action that they did not perform, they can notify the admin. The user can view all of the details by scanning the QR Code. They can add and verify all the documents with the ledger.

ADVANTAGES OF PROPOED SYSTEM

- User friendly.
- More Enhanced compared to previous.
- Always active.
- There is no server issues.

ALGORITHMS USED :

- **SHA – SECURE HASH ALGORITHM**

Secure Hash Algorithms also known as SHA are a family of cryptograph functions designed to keep data secured. It works by transforming the data using a hash function: an algorithm is of that consists of bitwise operations, modular additions, and is compression functions.

- **AES – ADVANCED ENCRYPTION STANDARD**

The AES Encryption algorithm (also known as the Rijndael algorithm) is a symmetric block cipher algorithm with a block/chunk size of 128 bits. It converts these individual blocks using keys of 128, 192, and 256 bits.

4.2 SYSTEM ARCHITECTURE

Describing the overall features of the software is concerned with defining the requirements and establishing the high level of the system. During architectural design, the various web pages and their interconnections are identified and designed. The major software components are identified and decomposed into processing modules and conceptual data structures and the interconnections among the modules are identified. The following modules are identified in the proposed system.

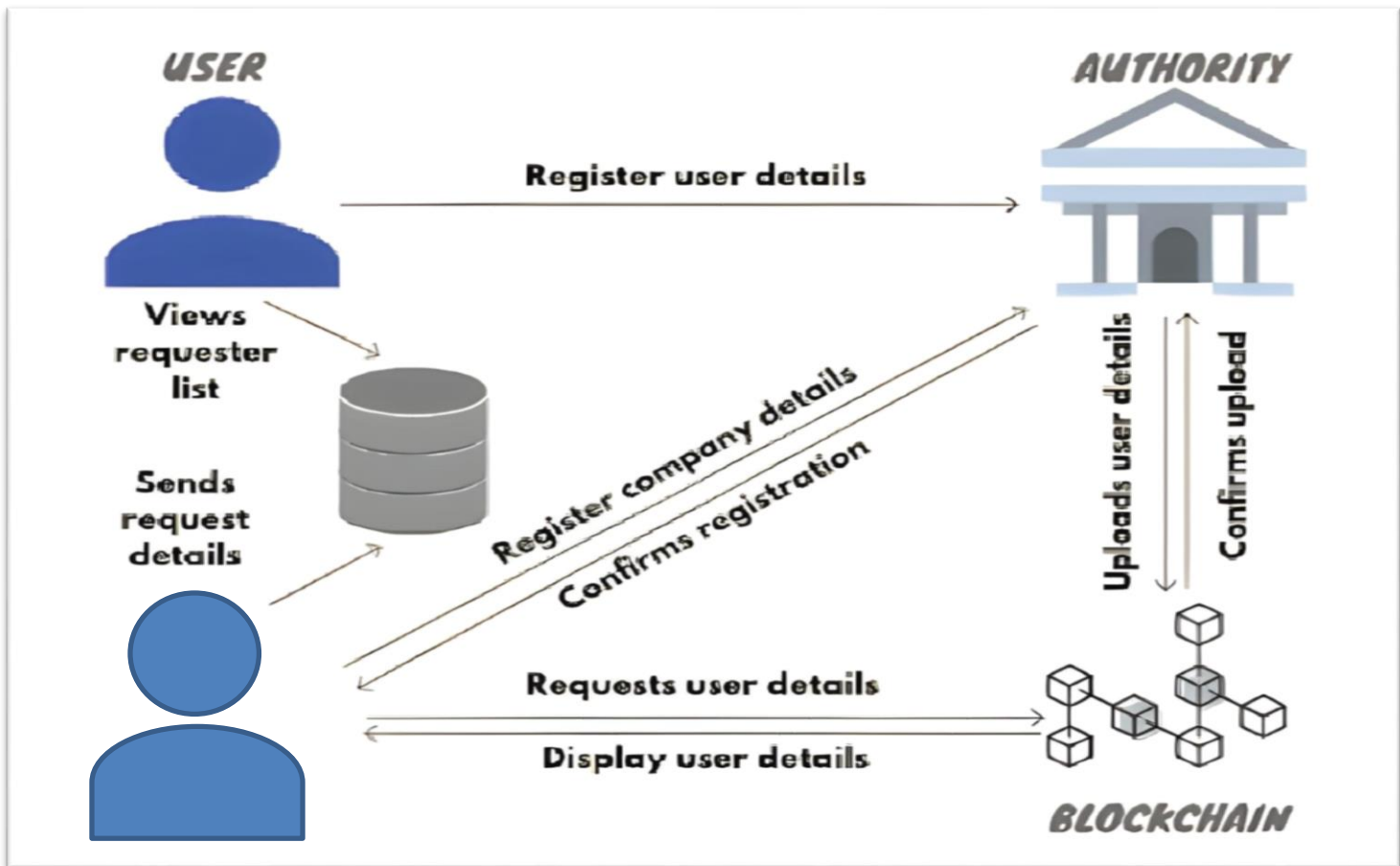


Figure 4.1 Architecture Diagram

4.3 SELECTING METHODOLOGY OR PROCESS MODEL:

- Admin Module
- User Module
- CRUD Module
- Documentation Module

4.3.1 ADMIN MODULE :

In this module, Admin having all of their credentials who all are given their details in their requirements, the admin manage all the users at the same time admin will view the either user added or not may some people did the fabrication document that admin find out and checking all the requirements of the performance and verifying the blocks using blockchain.

4.3.2 USER MODULE :

In this module, Users are having authentication and security to access the detail which is presented in the ontology system. Before accessing or searching the details user should have the account in that otherwise they should register first. At the very least, you need to provide an email address, username, password, display name, and whatever profile fields you have set to required. The display name is what will be used when the system needs to display the proper name of the user.

4.3.3 CRUD MODULE :

Full form of CRUD is create, read, update and delete. You can do the all operation here and make it your credentials proper.

4.3.4 DOCUMENTATION MODULE:

User can check whether the documents have been updated or deleted and the reputation of solid file info. If any person sees any faux or pastime which he has no longer done then he can tell the administrator. User can see all of the information by way of scanning the QR code. They can verify the books of accounts in addition to all the documents.

4.4 UML&URL DIAGRAMS:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling 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.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

We are going to achieve this project in block chain:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.

Provide extendibility and specialization mechanisms to extend the core concepts:

1. Be independent of particular programming languages and development process.
2. Provide a formal basis for understanding the modeling language.
3. Encourage the growth of OO tools market.
4. Support higher level development concepts such as collaborations, frameworks, patterns and components.
5. Integrate best practices.

4.4.1 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.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

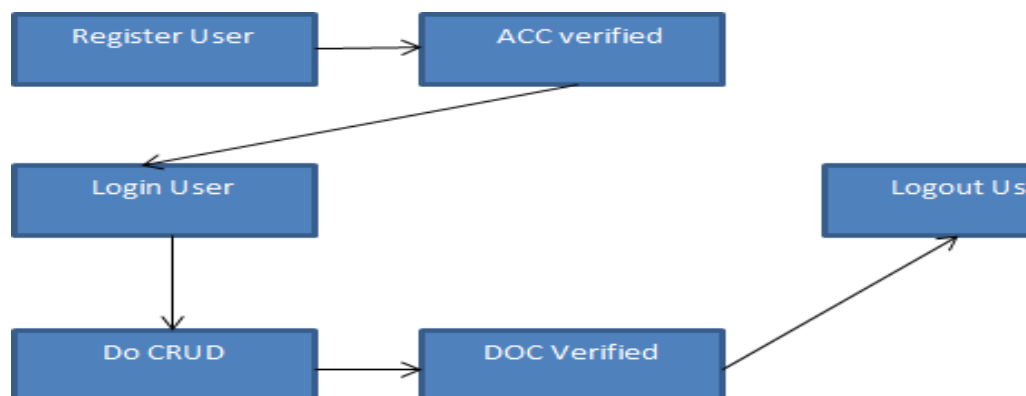


Fig 4.2 Data flow diagram of working

4.4.2 USE CASE:

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.

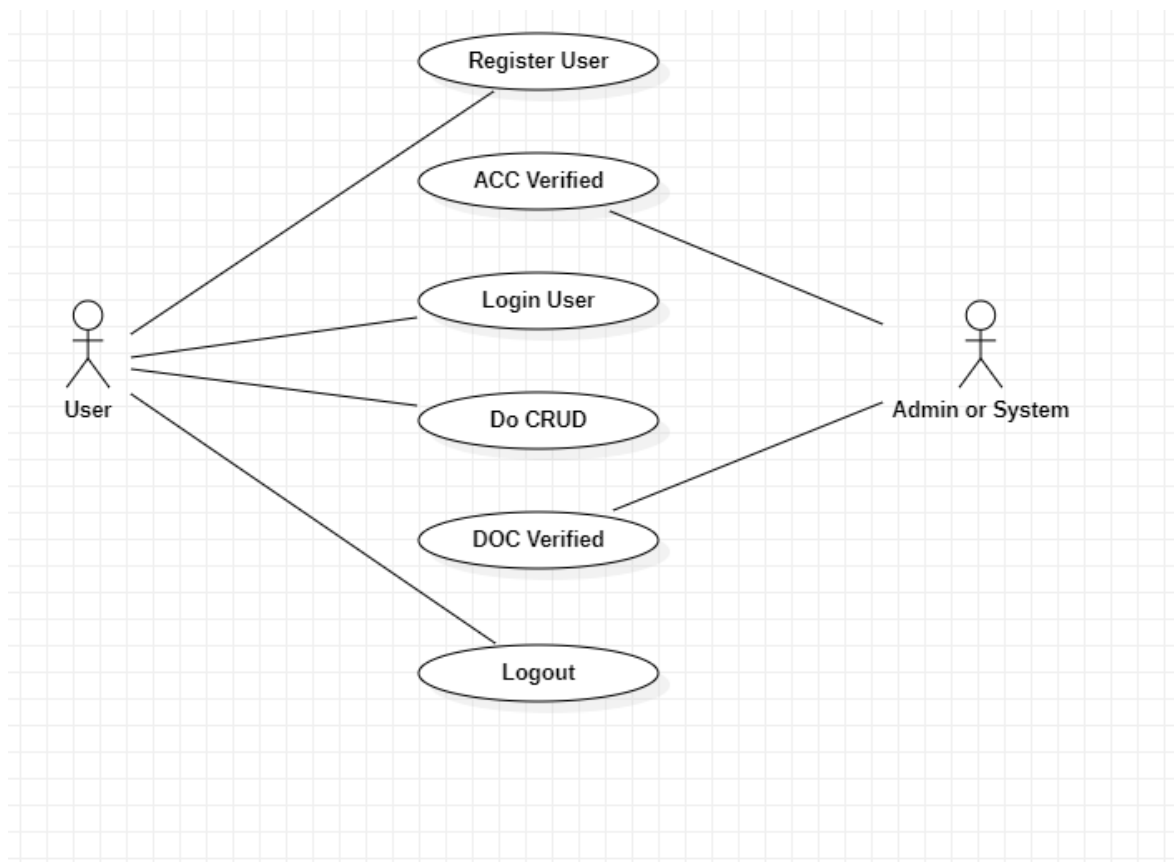


Figure 4.3 Use Case of working

4.4.3 ACTIVITY DIAGRAM:

This activity diagram contains three activities that are Server, TPA, Client and. This diagram of control shows the flow between these activities.

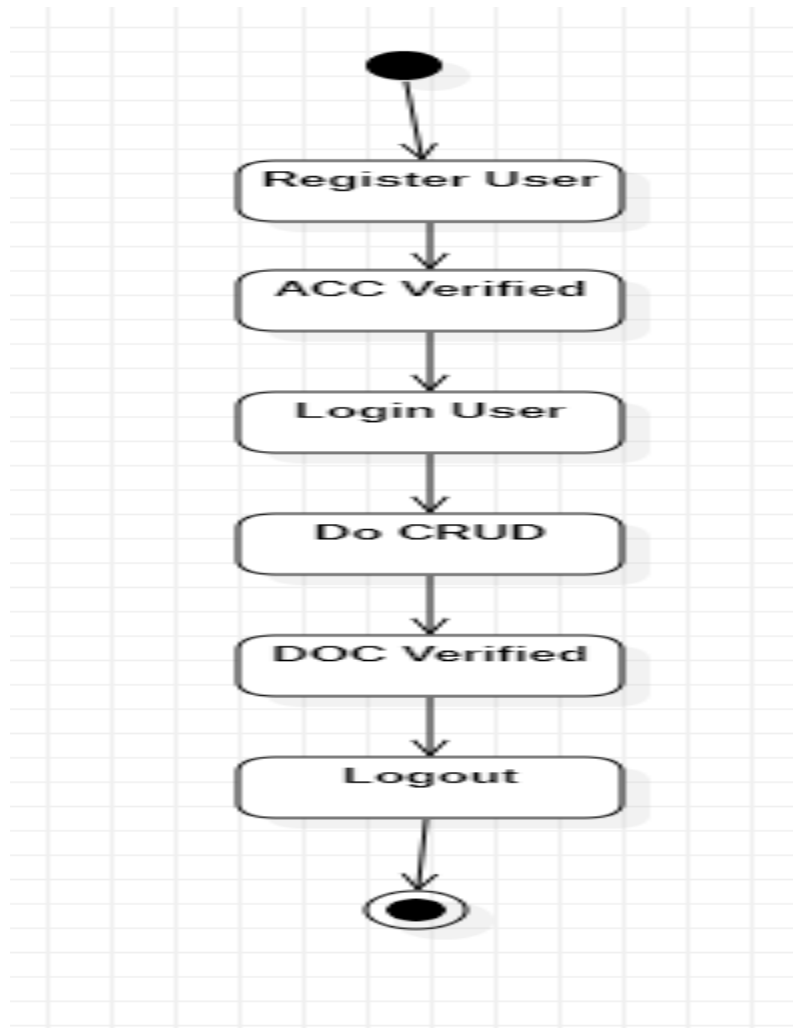


Figure 4.4 Activity diagram

4.4.4 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.

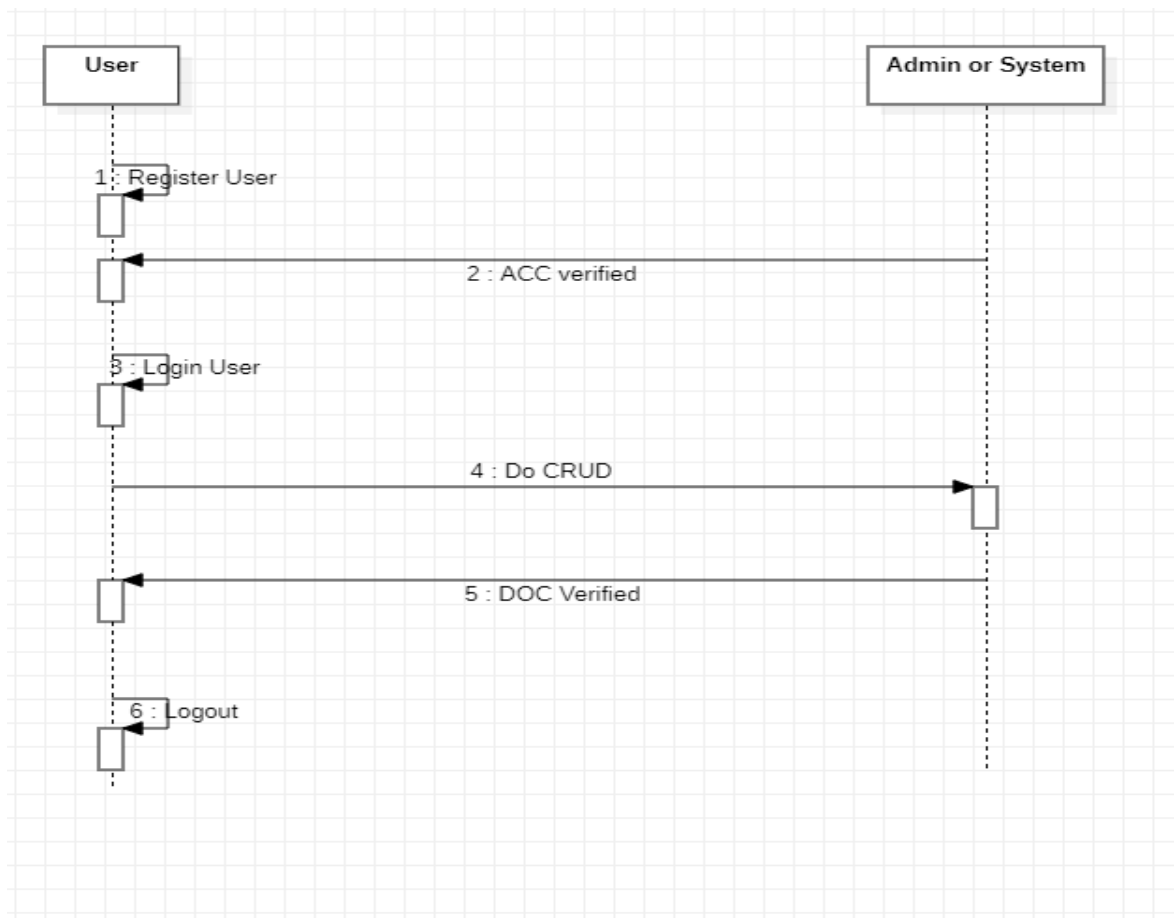


Figure 4.5 Sequence diagram of working process

CHAPTER 5

SYSTEM TESTING

5.1 TEST PLAN

Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of a software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process.

5.2 VERIFICATION

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves the way we want it to.

5.3 VALIDATION

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

5.4 BASICS OF SOFTWARE TESTING

There are two basics of software testing: black box testing and white box testing.

5.4.1 BLACK BOX TESTING

Black box testing is a testing technique that ignores the internal use of mechanism of the system and focuses on the output generated against any input and execution of the system. It is also called functional testing.

5.4.2 WHITE BOX TESTING

White box testing is a testing technique that takes into account the internal mechanism of a system. It is also called structural testing and glass box testing. Black box testing is often used for validation and whitebox testing is often used for verification.

CHAPTER-6

RESULTS AND DISCUSSION

There are extreme worries approximately the significant virtual media used to mediate communications among people, offerings and gadgets. A succession of centered solutions from service carriers is wasteful in duplication, has significant safety vulnerabilities and burdens users. The concept of private identity, which combines verified private traits and a unified digital identity, permits information users to discover their ownership and recognize how their records is getting used. A predominant application of generation is believed to be in finance. With the appearance of blockchain technology, self-sustaining identities can now obtain decentralization, in which each node participates independently from the others. Decentralized structures ought to circulate away from the supply of centralized authorities. Similar standards are brought to connect whilst retaining one's independence and internal intimacy.

So using blockchain for personal identity security system helps us to improve security in specific regions. It actually helps us to spread data across different network with security by some special codes. And people are finding it better for their privacy being protected because the access is managed carefully. Not only has that it kept record of any transactions and making it easier to change the mistakes. The system makes sure that the unauthorized people getting information with the help of decentralized structure and secret codes.

Smart contracts are like smart rules that controls and who could see the data of the user. This decentralized and cryptographic makes sure that the risks of centralized systems are avoided. In which it offers more secure environment which immune to hacking and auditable trail of identity related transactions. Making sure that everyone involved in the network is responsible for their action and simplifies any error detection and remediation.

The system using blockchain to secure personal identities showed it works

well, as seen in the results. It made personal information safer by using special technology that's hard for third parties to break into. People also had more control over their privacy, deciding who gets to see the information they needed. If there were any mistakes, the system made it easier to find and fix them. By emerging as a favorable solution to the hurdles that occurred by the old fashioned methods of the management system. Providing uninterrupted cooperation of different nodes or participants, the synchronization of transactions, or the overall balance and unity achieved in the decentralized and distributed ledger system. Which shows us its enhanced privacy and transparent accountability in the digital world.

We use AES in the Blockchain so that the data couldn't be exposed to unauthorized access or tampering. AES encryption is vital for safeguard sensitive information, preserving confidentiality and integrity. Its absence increases the risk of data breaches, potentially compromising the entire identity system. AES ensures that only authorized parties can access the data and manipulate it. Therefore leaving AES makes system vulnerable to exploitation and undermines its effectiveness in protecting users.

And also SHA is another important component in the working of project It ensures data integrity by generating a unique hash for each piece of data, making it virtually impossible to alter without detection. Without SHA there is a higher risk of data manipulation or corruption, compromising the trust worthiness of the identity system. Integrating SHA and AES builds strong trust in the system, mitigating the risk of data breaches, tampering and enhancing overall security posture. This is very reliable way to verify authenticity of the data stores on blockchain.

So by analyzing AES provides confidentiality by encrypting sensitive data while SHA ensures data integrity and authenticity by generating unique hash values for blockchain data.

CHAPTER-7

CONCLUSION

Our Project demonstrates how blockchain technology can be applied to identity management for improving security. The document gets stored on the hash of this document along with the other details entered by the user gets stored on the blockchain. Any third party can request the data from the user by mentioning the user's public address and the user has the right to approve, reject or partially approve the document request. These transactions also get stored on the blockchain. By leveraging its decentralized and immutable nature, blockchain gives access to safeguarding sensitive information from unauthorized tampering. In this several key components work unitedly to provide a robust and trustworthy framework for managing and protecting users' personal identity data.

To keep your information safe, the system uses special codes called encryption, which scramble your data so that only you or someone with the right key can access it. It's like giving it a unique sample. If someone tries to change anything with that sample the information in that changes too, showing that something is happening to the file.

This system also lets you control your own information. You decide who gets to see it and when. Plus it doesn't rely on one big company or organization to keep your data safe. Instead, it's spread out across many computers, making it harder for hackers to break in.

In addition to safekeeping your personal information, this system also makes things more convenient. You don't have to keep filling out forms with the same information again and again. Once you've stored your details in the blockchain, you can easily share them with trusted parties whenever needed.

Furthermore, because the blockchain is decentralized there's no single point of failure. Traditional systems often store data in one central location, making them vulnerable to cyber attacks or system failures.

Moreover, blockchain-based identity systems can help address issues of inclusivity and accessibility. In many parts of the world, people don't have access to official forms of identifying, which can hinder their ability to get participate fully in society. By providing a secure and verifiable digital of identity, blockchain technology can empower individuals to access to all essential services such as banking, healthcare, and education.

System puts you in control of your digital footprint. Instead of having your data scattered across various online platforms and databases, you can centralize and manage it all in one secure location. Since all transactions are recorded on the blockchain, it's easy to track and audit who has most accessed your information and when. This helps build trust between the individuals, businesses, and organizations, fostering a more transparent and accountable digital ecosystem.

REFERENCES

- Abhijeet R. Raipurkar, Shreyas Bobde, Anurag Tripahi, Mohit Sahu, "Digital Identity System Using Blockchain-based Self Sovereign Identity & Zero Knowledge Proof " in 2023 OITS International Conference on Information Technology (OCIT), 2023.
- Christidis. K and Devetsikiotis. M, "Blockchains and smart contracts for the internet of things," IEEE Access, vol. 4, pp. 2292–2303, 2016.
- Foundation O.K and the Open Rights Group, "Personal data and privacy working group," 2014.
- Komal Gilani, Emmanuel Bertin, Julien Hatin, Noel Crespi "A Survey on Blockchain-based Identity Management and Decentralized Privacy for Personal Data" 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), 2020.
- Kaaniche. N and Laurent. M, "A blockchain-based data usage auditing architecture with enhanced privacy and availability," in Network of Computing and Applications (NCA), 2017 IEEE 16th International Symposium on, pp. 1–5, IEEE, 2017.
- Ohme R.B, Christin .N, Edelman .B, and Moore .T, "Bitcoin: Economics, technology, and governance ,"The Journal of Economic Perspectives, vol. 29, no. 2, pp. 213–238, 2015.
- Oliver Hasan Padmanegara, Ratna Komala Putri, Reva Yuliani, Eryadi K.

Masli, “Blockchain and The Public Sector: Blockchain-Based Identity Management Systems For Public Services and The Impact on Privacy and Security Risks” International Conference on Digital Business and

Technology Management (ICONDBTM),2023.

- Poikola .A, Kuikkaniemi .K, and Honko .H, “My data anordic model for human-centered personal data management and processing,” Finnish Ministry of Transport and Communications, 2015.
- Regulation G. D.P, “Regulation (eu) 2016/679 - directive 95/46,” Official Journal of the European Union (OJ), vol. 59, pp. 1– 88, 2016.
- Tankard .C, “What the GDPR means for businesses,” Network Security, vol. 2016, no. 6, pp. 5–8, 2016.
- Yirui Bai, Hong Lei, Suozai Li, Haoyu Gao, Jun Li, Leixiao Li, “Decentralized and Self-Sovereign Identity in the Era of Blockchain: A Survey” IEEE International Conference on Blockchain (Blockchain),2022.

FUTURE SCOPE

It holds significant promise for transformative innovation. As advancing of decentralized identity management systems that can be used across different applications and services can provides users with greater pilot over their personal identity data.While integration with emerging tech as AI, biometrics, and IOT holds the potential to further enhance security. Additionally, ongoing research into scalability solutions and user-centric design principles will be central in ensuring that blockchain – based identity security systems can handle increasing transaction volumes and Intuitive for users of all backgrounds. Overall, the future outlook for this blockchain-based personal identity security systems is characterized by continued innovation, collaboration, and adaption to meet the evolving needs of society in an increasingly digital world.

APPENDIX

A. SOURCE CODE

```
/*
SQLyog Community v12.02 (32 bit)
MySQL - 5.5.29 : Database - policy
*****
*/

/*!40101 SET NAMES utf8 */;

/*!40101 SET SQL_MODE=""*/;

/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
    UNIQUE_CHECKS=0 */;
/*!40014 SET
    @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
    FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
    SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
CREATE DATABASE /*!32312 IF NOT EXISTS*/`policy` /*!40100 DEFAULT
    CHARACTER SET latin1 */;

USE `policy`;

/*!Table structure for table `oreg` */

DROP TABLE IF EXISTS `oreg`;
```

```

CREATE TABLE `oreg` (
  `sno` int(11) NOT NULL AUTO_INCREMENT,
  `id` varchar(200) DEFAULT NULL,
  `name` varchar(200) DEFAULT NULL,
  `password` varchar(200) DEFAULT NULL,
  `mail_id` varchar(200) DEFAULT NULL,
  `dob` varchar(200) DEFAULT NULL,
  `Gender` varchar(200) DEFAULT NULL,
  `profile` varchar(200) DEFAULT NULL,
  `cont` varchar(200) DEFAULT NULL,
  `address` varchar(200) DEFAULT NULL,
  `status` varchar(200) DEFAULT 'Waiting',
  KEY `sno` (`sno`)
) ENGINE=InnoDB AUTO_INCREMENT=19 DEFAULT CHARSET=latin1;

/*Data for the table `oreg` */

insert into
  `oreg`(`sno`,`id`,`name`,`password`,`mail_id`,`dob`,`Gender`,`profile`,`co
nt`,`address`,`status`) values
(12,'123','Abi','1234','techpowerotp@gmail.com','12/09/1999','Male','head
er.jpg','75643243442','Chennai','Activated'),(17,'1235','Gokul','1234','tech
powerotp@gmail.com','12/09/1999','Male','WIN_20231019_15_16_57_Pr
o.jpg','75643243442','Chennai','Activated'),(18,'123','Maran','1234','maran
12@gmail.com','12/09/1999','Male','health.gif','75643243442','Chennai','A
ctivated');

```

```

/*Table structure for table `reg` */

```

```

DROP TABLE IF EXISTS `reg`;

```

```

CREATE TABLE `reg` (
  `sno` int(11) NOT NULL AUTO_INCREMENT,
  `id` varchar(200) DEFAULT NULL,
  `name` varchar(200) DEFAULT NULL,

```

```

`password` varchar(200) DEFAULT NULL,
`mail_id` varchar(200) DEFAULT NULL,
`dob` varchar(200) DEFAULT NULL,
`profile` varchar(200) DEFAULT NULL,
`Gender` varchar(200) DEFAULT NULL,
`cont` varchar(200) DEFAULT NULL,
`address` varchar(200) DEFAULT NULL,
`status` varchar(200) DEFAULT 'Waiting',
PRIMARY KEY (`sno`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `reg` */

/*Table structure for table `reqreg` */

DROP TABLE IF EXISTS `reqreg`;

CREATE TABLE `reqreg` (
  `sno` int(11) NOT NULL AUTO_INCREMENT,
  `id` int(11) DEFAULT NULL,
  `name` varchar(200) DEFAULT NULL,
  `mail_id` varchar(200) DEFAULT NULL,
  `keyy` int(20) DEFAULT NULL,
  `statua` varchar(200) DEFAULT 'Waiting',
  KEY `sno` (`sno`)
) ENGINE=InnoDB AUTO_INCREMENT=24 DEFAULT CHARSET=latin1;

/*Data for the table `reqreg` */

insert into `reqreg`(`sno`,`id`,`name`,`mail_id`,`keyy`,`statua`) values
(15,123,'Abi','techpowerotp@gmail.com',8456,'Sent'),(19,1235,'Gokul','te
chpowerotp@gmail.com',3051,'Sent'),(20,1235,'Gokul','techpowerotp@g
mail.com',9030,'Sent'),(21,123,'Maran','maran12@gmail.com',NULL,'Wait
ing'),(22,123,'Maran','maran12@gmail.com',NULL,'Waiting'),(23,123,'Mar
an','maran12@gmail.com',5244,'Sent');

```

```
/*Table structure for table `request` */
```

```
DROP TABLE IF EXISTS `request`;
```

```
CREATE TABLE `request` (  
  `sno` int(11) NOT NULL AUTO_INCREMENT,  
  `id` varchar(200) DEFAULT NULL,  
  `name` varchar(200) DEFAULT NULL,  
  `reg` varchar(200) DEFAULT NULL,  
  `oname` varchar(200) DEFAULT NULL,  
  `trap` varchar(200) DEFAULT NULL,  
  `image` varchar(200) DEFAULT NULL,  
  `status` varchar(200) DEFAULT 'waiting',  
  `fkey` varchar(200) DEFAULT '-----',  
  `request` varchar(200) DEFAULT '0',  
  `downlad` varchar(200) DEFAULT '0',  
  KEY `sno` (`sno`)  
) ENGINE=InnoDB AUTO_INCREMENT=22 DEFAULT CHARSET=latin1;
```

```
/*Data for the table `request` */
```

```
insert into  
  `request`(`sno`,`id`,`name`,`reg`,`oname`,`trap`,`image`,`status`,`fkey`,`re  
  quest`,`downlad`) values  
  (17,'123','Abi','123','Abi','BD37245EF8E81B87','Sign.PNG','Send  
  Key','693','1','1'),(20,'1235','Gokul','1235','Gokul','2C1BA6432696CAD5','  
  AA_Home.jsp','Send  
  Key','5706','1','1'),(21,'123','Maran','123','Maran','E790A9564B6AF371','p  
  aracetamol.jpg','Send Key','4781','1','1');
```

```
/*Table structure for table `upload` */
```

```
DROP TABLE IF EXISTS `upload`;
```

```

CREATE TABLE `upload` (
  `reg` varchar(200) DEFAULT NULL,
  `name` varchar(200) DEFAULT NULL,
  `fkey` varchar(200) DEFAULT NULL,
  `dates` varchar(200) DEFAULT NULL,
  `trapdoor` varchar(200) DEFAULT NULL,
  `image` varchar(200) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/*Data for the table `upload` */

insert into `upload`(`reg`,`name`,`fkey`,`dates`,`trapdoor`,`image`) values
('123','Abi','693','01/11/23
18:27:33','BD37245EF8E81B87','Sign.PNG'),('1235','Gokul','5706','06/11/
23
17:54:28','2C1BA6432696CAD5','AA_Home.jsp'),('123','Maran','4781','27/
11/23 12:23:19','E790A9564B6AF371','paracetamol.jpg');

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET
  FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

/*Login Action*/

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;

```

```

import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author ADMIN
 */
@WebServlet(urlPatterns = {"/AA_Login_Action"})
public class AA_Login_Action extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and
     * <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();

        String name=request.getParameter("user");
        String pass=request.getParameter("pass");
        System.out.println("username.." +name+" "+pass);
        if(name.equalsIgnoreCase("Key")&&pass.equalsIgnoreCase("Key")){
            out.println("<script type='\"text/javascript\"'>");
            out.println("alert(\"Welcome Attribute Authority\")");
            out.println("</script>");
            RequestDispatcher
            rd=request.getRequestDispatcher("AA_Home.jsp");
            rd.include(request, response);

```

```

    }
    else{
        out.println("<script type=\"text/javascript\">");
        out.println("alert(\"Invalid Login\")");
        out.println("</script>");
        RequestDispatcher rd=request.getRequestDispatcher("aalogin.jsp");
        rd.include(request, response);
    }

}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs

```

```

    * @throws IOException if an I/O error occurs
    */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
        response)
        throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    } // </editor-fold>
}

```

/*Reg Login*/

```

import java.io.File;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.regex.Matcher;

```

```

import java.util.regex.Pattern;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletContext;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileItemFactory;
import org.apache.commons.fileupload.FileUploadException;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

```

```

/**
 *
 * @author ADMIN
 */
@WebServlet(urlPatterns = {"/Reg_Login_Action"})
public class Reg_Login_Action extends HttpServlet {

```

```

/**
 * Processes requests for both HTTP <code>GET</code> and
 * <code>POST</code>
 * methods.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
protected void processRequest(HttpServletRequest request,
    HttpServletResponse response)

```

```

        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {

        HttpSession session=request.getSession();

        ServletContext sc=request.getSession().getServletContext();

        List<String> ms = new ArrayList<String>();
        String finalimage = "";
        boolean isMultipart = ServletFileUpload.isMultipartContent(
            request);
        List<String> m = new ArrayList<String>();
        File savedFile;

        if (!isMultipart) {

            System.out.println("File Not Uploaded");
        }
        else {
            FileItemFactory factory = new DiskFileItemFactory();
            ServletFileUpload upload = new ServletFileUpload(factory);
            List items = null;

            try {
                items = upload.parseRequest(request);
            }
            catch (FileUploadException e)
            {
                e.printStackTrace();
            }
            Iterator itr = items.iterator();
            while (itr.hasNext()) {

```

```

List<String> al = new ArrayList<String>();

String sss = "";
FileItem item = (FileItem) itr.next();
String value = "";
String a[];
if (item.isFormField()) {
    String name = item.getFieldName();
    //System.out.println("name: "+name+"\n");
    value = item.getString();
    // System.out.println("value: "+value);
    al.add(value);
    for (int i = 0; i < al.size(); i++) {
        sss += al.get(i);
        System.out.println("is this image" + sss);

    }

    a = sss.split("-");
    for (int i = 0; i < a.length; i++) {

        String am = a[i];
        System.out.println("aaaaaaaaaaaaaaaaaaaa" + a[i]);
        m.add(a[i]);
    }
} else {

    String itemName = item.getName();

    String reg = "[.*]";
    String replacingtext = "";

    Pattern pattern = Pattern.compile(reg);

```

```

Matcher matcher = pattern.matcher(itemName);
StringBuffer buffer = new StringBuffer();

while (matcher.find())
{
    matcher.appendReplacement(buffer, replacingtext);
}
int IndexOf = itemName.indexOf(".");
int Indexf = itemName.indexOf(".");
String domainName = itemName.substring(IndexOf);

finalimage = buffer.toString() + domainName;
System.out.println("Final Image===" + finalimage);
ms.add(finalimage);
ms.get(0);
String fn=ms.get(0);
System.out.println("trying to put all in store");
savedFile = new
File(sc.getRealPath("profile")+"\\ "+finalimage);

```

```
String a0=request.getParameter(value);
```

```

try
{
    item.write(savedFile);
} catch (Exception e) {
    e.printStackTrace();
}

}
}
}

```

```
System.out.println("eeeeeeeeeeeeeeeeeeeeeeeeeejajkljffffffffffffffffffff");
```

```

Class.forName("com.mysql.jdbc.Driver");
Connection
con=DriverManager.getConnection("jdbc:mysql://localhost/policy","root","
root");
PreparedStatement ps=con.prepareStatement("insert into
reg(id,name,password,mail_id,dob,profile,Gender,cont,address)values('"+
+m.get(0)+"','"+m.get(1)+"','"+m.get(2)+"','"+m.get(3)+"','"+m.get(4)+"','"+
ms.get(0)+"','"+m.get(5)+"','"+m.get(6)+"','"+m.get(7)+"')");
        System.out.println(ps);
ps.executeUpdate();
System.out.println(ps);
out.println("<script type='text/javascript'>");
out.println("alert('Successfully Register DataUser')");
out.println("</script>");
        RequestDispatcher
rd=request.getRequestDispatcher("User_Login.jsp");
        rd.include(request, response);
    }
    catch(Exception ex){ out.println("<script type='text/javascript'>");

    out.println("alert('Try again')");
    out.println("</script>");
        RequestDispatcher rd=request.getRequestDispatcher("index.html");
        rd.include(request, response);

    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/**

* Handles the HTTP <code>GET</code> method.

*

* @param request servlet request

```

* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    processRequest(request, response);
}

/**
* Handles the HTTP <code>POST</code> method.
*
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    processRequest(request, response);
}

/**
* Returns a short description of the servlet.
*
* @return a String containing servlet description
*/
@Override
public String getServletInfo() {
    return "Short description";
}

```

```
}
```

```
/*User Login*/
```

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
```

```
/**
```

```
*
```

```
* @author ADMIN
```

```
*/
```

```
@WebServlet(urlPatterns = {"/User_Login_Action"})
public class User_Login_Action extends HttpServlet {
```

```
/**
```

```
* Processes requests for both HTTP <code>GET</code> and
   <code>POST</code>
```

```
* methods.
```

```
*
```

```
* @param request servlet request
```

```
* @param response servlet response
```

```
* @throws ServletException if a servlet-specific error occurs
```

```
* @throws IOException if an I/O error occurs
```

```

*/
protected void processRequest(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    HttpSession session=request.getSession();
    try{ String name=request.getParameter("user");
        String pass=request.getParameter("pass");

        Class.forName("com.mysql.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost/policy","root","
root");
        PreparedStatement ps=con.prepareStatement("select * from reg
where name='"+name+"' and password='"+pass+"' ");
        System.out.println(ps);
        ResultSet r=ps.executeQuery();
        if(r.next()){
            String id=r.getString("id");
            String mail=r.getString("mail_id");
            String pic=r.getString("profile");
            String sno=r.getString("sno");
            System.out.println(id+" "+mail+" "+pic+" "+name);
            session.setAttribute("name",name);
            session.setAttribute("id", id);
            session.setAttribute("pic", pic);
            session.setAttribute("sno", sno);

            System.out.println("pname"+name+"id"+id+"pic"+pic+"mail"+mail);
            out.println("<script type='\"text/javascript\"'>");
            out.println("alert(\"Welcome  "+name+"\")");
            out.println("</script>");
            RequestDispatcher
rd=request.getRequestDispatcher("User_Home.jsp");

```

```

        rd.include(request, response);
    }
    else{
        out.println("<script type=\"text/javascript\">");
        out.println("alert(\"Invalid Login\")");
        out.println("</script>");
        RequestDispatcher
rd=request.getRequestDispatcher("User_Login.jsp");
        rd.include(request, response);
    }

}
catch(Exception e){
    System.out.println(e);
}
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/**

* Handles the HTTP <code>GET</code> method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

```
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
```

```
    throws ServletException, IOException {
```

```
        processRequest(request, response);
```

```
    }
```

/**

```

    * Handles the HTTP <code>POST</code> method.
    *
    * @param request servlet request
    * @param response servlet response
    * @throws ServletException if a servlet-specific error occurs
    * @throws IOException if an I/O error occurs
    */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
        response)
        throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    } // </editor-fold>

}

/*UserReg login*/

import java.io.File;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.util.ArrayList;

```

```

import java.util.Iterator;
import java.util.List;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletContext;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileItemFactory;
import org.apache.commons.fileupload.FileUploadException;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;

```

```

/**
 *
 * @author ADMIN
 */
@WebServlet(urlPatterns = {"/Userreg_Lgoin_Action"})
public class Userreg_Login_Action extends HttpServlet {

```

```

/**
 * Processes requests for both HTTP <code>GET</code> and
 * <code>POST</code>
 * methods.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */

```

```

protected void processRequest(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {

        HttpSession session=request.getSession();

        ServletContext sc=request.getSession().getServletContext();

        List<String> ms = new ArrayList<String>();
        String finalimage = "";
        boolean isMultipart = ServletFileUpload.isMultipartContent(
            request);
        List<String> m = new ArrayList<String>();
        File savedFile;

        if (!isMultipart) {

            System.out.println("File Not Uploaded");
        }
        else {
            FileItemFactory factory = new DiskFileItemFactory();
            ServletFileUpload upload = new ServletFileUpload(factory);
            List items = null;

            try {
                items = upload.parseRequest(request);

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

```

```

Iterator itr = items.iterator();
while (itr.hasNext()) {
    List<String> al = new ArrayList<String>();

    String sss = "";
    FileItem item = (FileItem) itr.next();
    String value = "";
    String a[];
    if (item.isFormField()) {
        String name = item.getFieldName();
        //System.out.println("name: "+name+"\n");
        value = item.getString();
        // System.out.println("value: "+value);
        al.add(value);
        for (int i = 0; i < al.size(); i++) {
            sss += al.get(i);
            System.out.println("is this image" + sss);

        }

        a = sss.split("-");
        for (int i = 0; i < a.length; i++) {

            String am = a[i];
            System.out.println("aaaaaaaaaaaaaaaaaaaa" + a[i]);
            m.add(a[i]);
        }

    } else {

        String itemName = item.getName();

        String reg = "[.*]";

```

```

String replacingtext = "";

Pattern pattern = Pattern.compile(reg);
Matcher matcher = pattern.matcher(itemName);
StringBuffer buffer = new StringBuffer();

while (matcher.find())
{
    matcher.appendReplacement(buffer, replacingtext);
}

int IndexOf = itemName.indexOf(".");
int Indexf = itemName.indexOf(".");
String domainName = itemName.substring(IndexOf);

finalimage = buffer.toString() + domainName;
System.out.println("Final Image===" + finalimage);
ms.add(finalimage);
ms.get(0);
String fn=ms.get(0);
System.out.println("trying to put all in store");
savedFile = new
File(sc.getRealPath("profile")+"\\ "+finalimage);

String a0=request.getParameter(value);

try
{
    item.write(savedFile);
} catch (Exception e) {
    e.printStackTrace();
}

}

}

}

```

```

System.out.println("eeeeeeeeeeeeeeeeeeeeeeeeeejajkljffffffffffffffffffff");
Class.forName("com.mysql.jdbc.Driver");
Connection
    con=DriverManager.getConnection("jdbc:mysql://localhost/policy","root","
    root");
PreparedStatement ps=con.prepareStatement("insert into
    reg(id,name,password,mail_id,dob,profile,Gender,cont,address)values('"+
    +m.get(0)+"','"+m.get(1)+"','"+m.get(2)+"','"+m.get(3)+"','"+m.get(4)+"','"+
    +ms.get(0)+"','"+m.get(5)+"','"+m.get(6)+"','"+m.get(7)+"')");
        System.out.println(ps);
ps.executeUpdate();
System.out.println(ps);
out.println("<script type='text/javascript'>");
out.println("alert(\"Successfully Register Dataowner\")");
out.println("</script>");
        RequestDispatcher
        rd=request.getRequestDispatcher("User_Login.jsp");
        rd.include(request, response);
    }
    catch(Exception ex){ out.println("<script type='text/javascript'>");

        out.println("alert(\"Try angin\")");
        out.println("</script>");
        RequestDispatcher rd=request.getRequestDispatcher("index.html");
        rd.include(request, response);

    }
}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click
on the + sign on the left to edit the code.">
/**
 * Handles the HTTP <code>GET</code> method.

```

```

* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    processRequest(request, response);
}
/**
* Handles the HTTP <code>POST</code> method.
*
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    processRequest(request, response);
}

/**
* Returns a short description of the servlet.
*
* @return a String containing servlet description
*/
@Override
public String getServletInfo() {
    return "Short description";
}

```

B. SCREENSHOTS

Blockchain Based Personal Identity Security System

Home
Admin
User
Logout

User Registration Page

ID:

Username:

Password:

Mail ID:

DOB:

Profile: [krishnaradha.jpg](#)

Gender:

Contact:

Address:

[Click here to back](#)

localhost:8080 says
Successfully Register User

OK

Blockchain Based Personal Identity Security System

Home

Admin

User

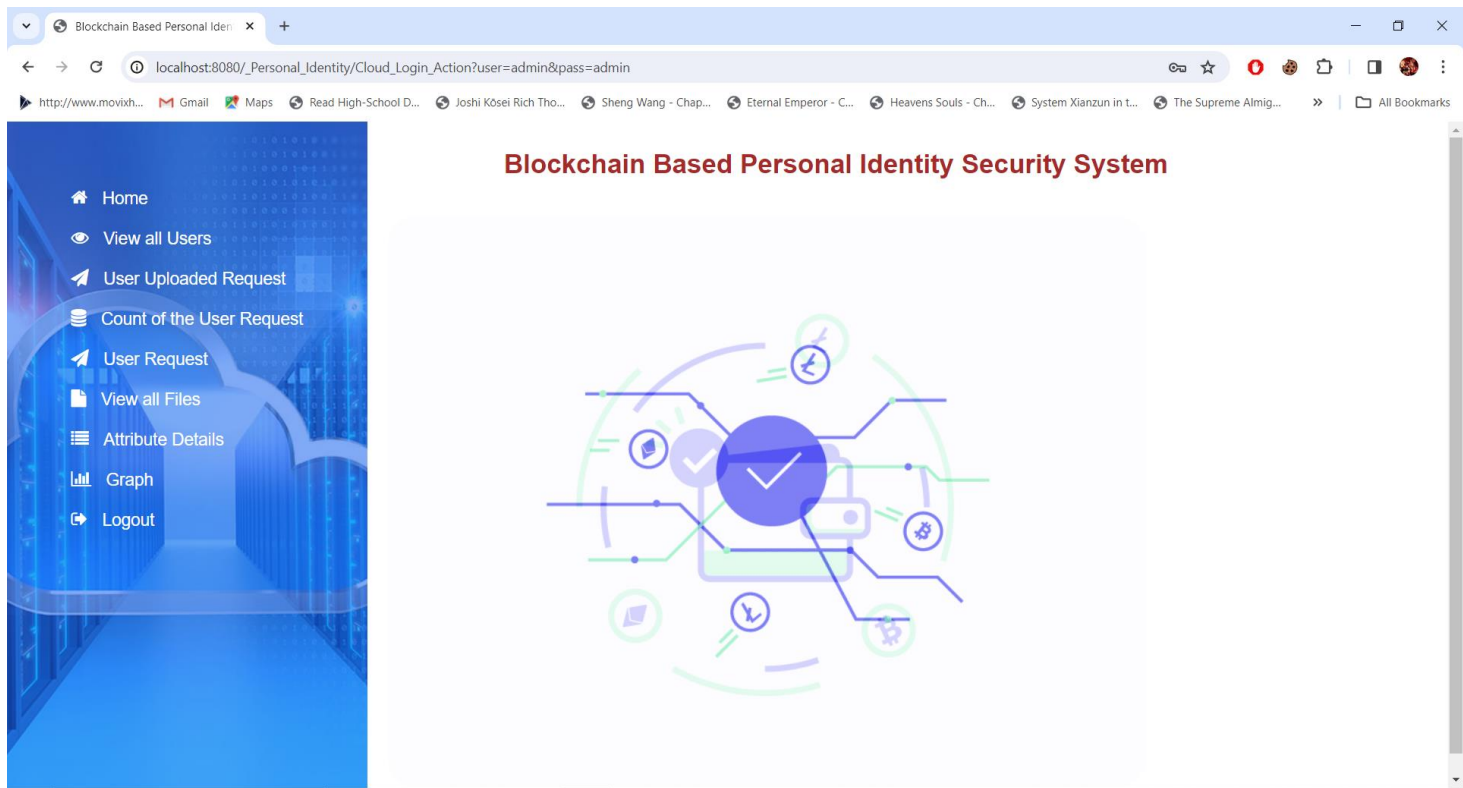
Logout

Admin Login

Username:admin

Password:.....

LOGIN



Blockchain Based Personal Identity Security System

View all Users

Id	Name	Mail	DOB	Gender	Contact	Address	Action
Thanay	Thanay04	nvhsq.chinnu@gmail.com	22/02/2003	Male	9648710928	nizampet	Waiting

Home

View all Users

User Uploaded Request

Count of the User Request

User Request

View all Files

Attribute Details

Graph

Logout

Blockchain Based Personal Identity Security System

localhost:8080/Personal_Identity/uauth.jsp?id=Thanay

localhost:8080 says
Authorize User Successfully

OK

Blockchain Based Personal Identity Security System

localhost:8080/Personal_Identity/owner?user=Thanay04&pass=Thanaylahari

Thanay04

Home

View Profile

Upload File

Manage File & Scan QR Code

View File & Make Request

Download File

Logout

Block chain Based Personal Identity

localhost:8080/_Personal_Identity/uploadrequest.jsp

[http://www.movixh...](#)
[Gmail](#)
[Maps](#)
[Read High-School D...](#)
[Joshi Kōsei Rich Tho...](#)
[Sheng Wang - Chap...](#)
[Eternal Emperor - C...](#)
[Heavens Souls - Chi...](#)
[System Xianzun in t...](#)
[The Supreme Almig...](#)
[All Bookmarks](#)

Thanay04

[Home](#)
[Upload File](#)
[Manage File & Scan QR Code](#)
[View File & Make Request](#)
[Download File](#)
[Logout](#)

Block Chain Based Personal Identity Security System

Send Request:

Request

Enter Key:

Check it !

localhost:8080/_Personal_Identity/uploadrequest.jsp

ChatGPT

WhatsApp

Home - Netflix

Block chain Based Personal Identity

Block Chain Based Personal Identity

localhost:8080/_Personal_Identity/ownerreq.jsp

[http://www.movixh...](#)
[Gmail](#)
[Maps](#)
[download utorrent f...](#)
[Read High-School D...](#)
[Joshi Kōsei Rich Tho...](#)
[Sheng Wang - Chap...](#)
[Eternal Emperor - C...](#)
[Heavens Souls - Chi...](#)
[System Xianzun in t...](#)

Home

View all Users

User Uploaded Request

Count of the User Request

User Request

View all Files

Attribute Details

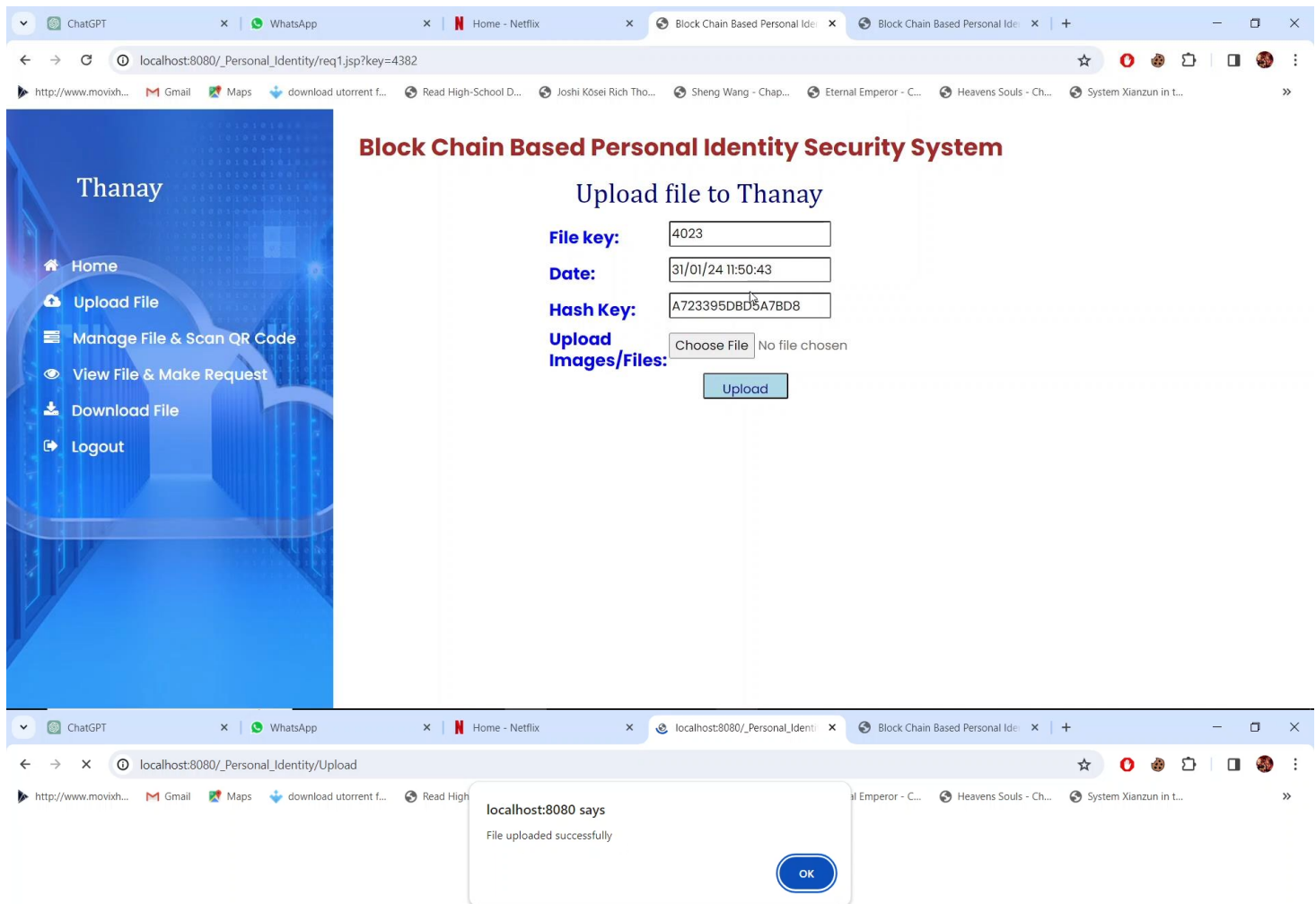
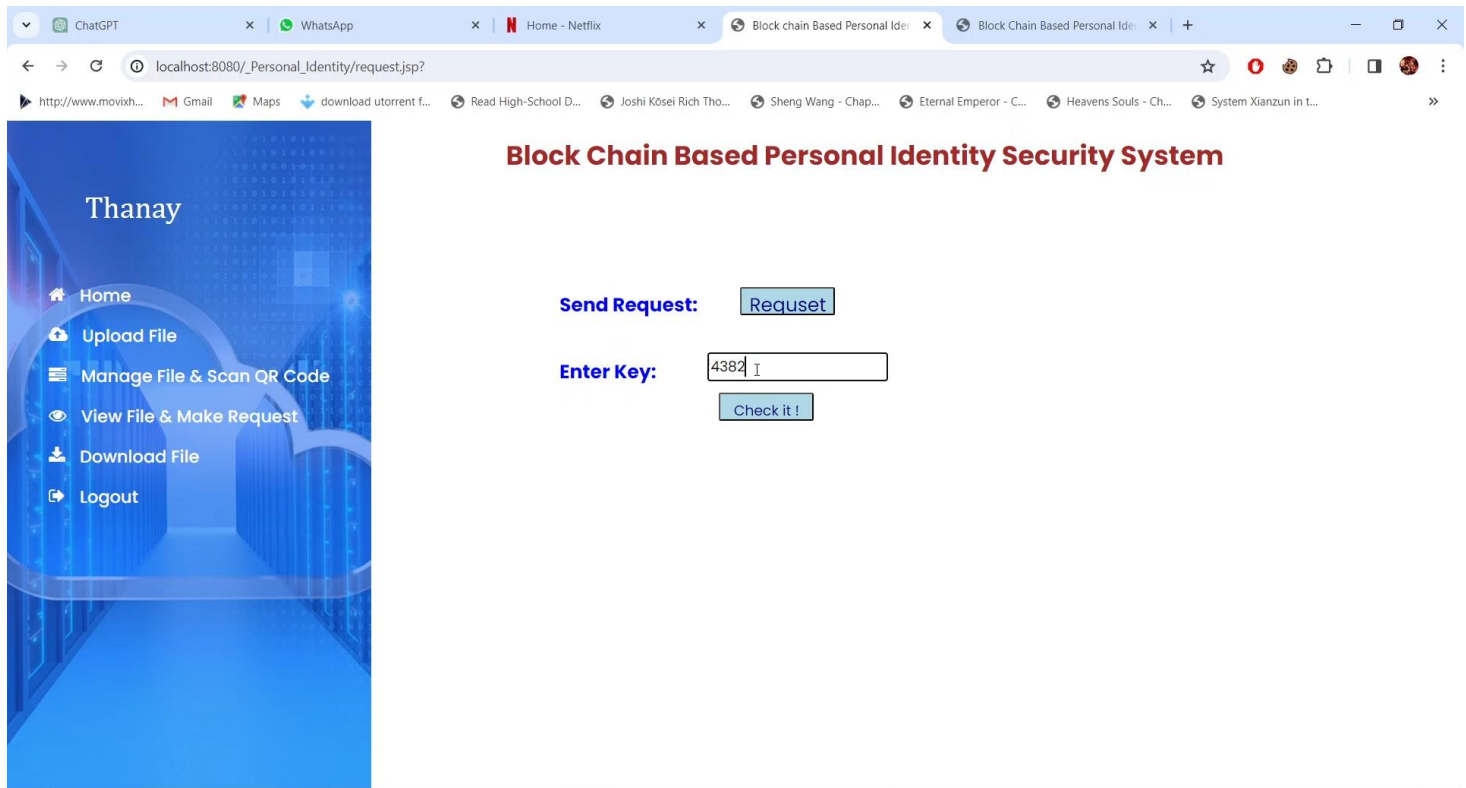
Graph

Logout

Block Chain Based Personal Identity Security System

View User Request

User ID	User Name	Email Id	Status	Action
10942	Thanay	nvhsq.chinnu@gmail.com	Waiting	Send Key
10942	Thanay	nvhsq.chinnu@gmail.com	Waiting	Send Key



Thanay

Home

Upload File

Manage File & QR Code

View File & Make Request

Download File

Logout

Block Chain Based Personal Identity Security System

Search File Search With K

ID	Name	Hash Key	File Name	Date	View	Action
10942	Thanay	A723395DBD5A7BD8	image_6.jpg	31/01/24 11:50:43	View	Request Send

Home

localhost:8080 says

OTP Sent to mail id

OK

Home

View all Users

User Uploaded Request

Count of the User Request

User Request

View all Files

Attribute Details

Graph

Logout

No.of.Request Count Attribute

Range (point-wise)

File Name

Thanay

Home

Upload File

Manage File & Scan QR Code

View File & Make Request

Download File

Logout

Block Chain Based Personal Identity Security System

Enter Your Private File

Private Key:	
Hash Key:	
Download	

Home

localhost:8080 says

Key Generated to User Mail Id.

OK

Thanay

Home

Upload File

Manage File & Scan QR Code

View File & Make Request

Download File

Logout

Block Chain Based Personal Identity Security System

Enter Your Private File

Private Key:	4023
Hash Key:	
Download	

SQLyog Enterprise - MySQL GUI - [New Connection - root@localhost]

Thanay

Home

Upload File

Manage File & Scan QR Code

View File & Make Request

Download File

Logout

Block Chain Based Personal Identity Security System

Enter Your Private File

Private Key:	4023
Hash Key:	A723395DBD5A7BD8
Download	

image_6.jpg
95.1 KB • Done

Blockchain Based Personal Identity Security System

Karwan Hemanth Kumar
Student
Department of Computer Science and Engineering
Sathyabama Institute of Science & Technology, Chennai, India
krwnhemu@gmail.com

Thanay Katta
Student
Department of Computer Science and Engineering
Sathyabama Institute of Science & Technology, Chennai, India
thanaykatta@gmail.com

Anandha Sree R
Professor
Department of Computer Science and Engineering
Sathyabama Institute of Science & Technology, Chennai, India
anandha.sree@gmail.com

ABSTRACT- *Identity theft is unauthorized acquisition of another confidential information of a person and misuse it. Organizations and individuals should exercise caution when it comes to protecting their identities to avoid fraud as a result of identity theft. This information is freely available to attackers in user profiles. Attackers use this information to obtain additional information without raising suspicions about a final attack, identity theft, or fraud. Our block chain based Personal Identity Security System assist in securely storing personal identity data without fear of it being compromised or lost. In this system, the admin can access all the users data and verify those accounts. The admin has access to all of the user's documents that have been uploaded to the system. They can view all of activity logs of the user. The admin can view the status and see if any fabrication is taking place. The admin can view the user's complaints. The user can view if logs have been updated or deleted, as well as the status of fabricated document details. If a user discovers a fabrication or an action that they did not perform, they can notify the admin. The user can view all of the details by scanning the QR Code. They can add and verify all the documents with the ledger.*

KEYWORDS: *Blockchain, Self - sovereign identity, Identity management, Decentralization, Privacy, Security, Authentication, Identity proofs, GDPR, Prototype.*

INTRODUCTION:

A blockchain is a decentralized garage machine that stores statistics in a sequence of cryptographic blocks related by hashing previous blocks. In this task, the front-end includes HTML, CSS and JavaScript and the back-end consists of ASP.NET. The IDE used is Visual Studio and MSSQL database. Our blockchain-based personal identity security device permits non-public identity records to be stored securely without fear of being compromised.

In this environment, the administrator can discover all user's accounts. Administrators have get right of entry to to all person documents uploaded to the gadget. Everything can be achieved by the user. Check the admin reputé and see if any production is going on. You can check as a user complaint.

To log into the machine, the person have to first sign in; Once the account is verified by the administrator, the consumer can log in with their username. User can update profile as according to his requirement. The consumer can manipulate their documents in the system by way of including, adding and deleting them. All files submitted through the person are confirmed by using the Administrator. User can take a look at the fame of organized files with QR code. This work aims to establish a decentralized gadget to save the identification of people, guard private identities and notably lessen fraudulent activities.

LITERATURE REVIEW

E-Voting System,

The on line balloting system can be used online. This device has a centralized database for all voter and candidate registrations and final results. This on line voting method is primarily based on digital vote casting. This internet-based totally device saves time, offers timely information, reduces workload and ensures facts security. This internet site is operated with the aid of the Election Commission. This is a easy, safe and at ease technique as a way to take you less time. The meaning of the phrase "VOTE" is to settle, examine, or act on a commodity.

Online Voting System,

This assignment in particular pursuits to offer a secure and user pleasant online balloting machine. The problem of vote casting is very critical in phrases of protection and safety. The corporation is designing and developing an internet-primarily based vote casting gadget the use of fingerprint and Aadhaar card to offer excessive efficiency and excessive security vote casting device.

Online voting System Using Face Recognition,

The primary idea of this method is to create an online vote casting system so that it will help dispose of fraud from manual vote casting methods and previous versions of on line digital camera vote casting for facial popularity and OTP era. We are enforcing a region-based totally vote casting system for citizens who're unable to vote (Home City).

Dual Authentication Voting System

The task uses RFID and fingerprint technology within the vote casting device. Each consumer gets a voter ID in the form of an RFID tag. The hardware layout consists of a scanning fingerprint sensor, which is used to examine the consumer's fingerprint with the person's formerly stored fingerprint.

Unique Id Based Voting System

The use of guide elections has been proven to growth the hazard of electoral fraud. It assign a evident identity to the voter, strengthens the security and gives precise result of election. When registering each people is verified and a unique identifier, which may link with biometric data, is given. This unique identifier make certain that only authorized voters can join. This makes voter privacy as main priority and takes strong security measures to safeguard from fraud making sure so the everyone can participate by following the rules. So, the unique based voting system works to simplify the voting process with highest standards of security.

EXISTING SYSTEM:

Identity Management System (IDMS) How customers or individuals are recognized and certified to use organizational structures and services. Traditional identification control and blockchain systems depend upon a depended on middle, for this reason reducing uncarried points of failure. Peer-to-Per (P2P), a vision centered at the black community. (SSI) has entire manage over your virtual identification. Successful implementation of IDMS BC-primarily based SSI increases the security and safety degree for the application. BC A arguable vicinity of studies within the early ranges of IMDS. This article affords an extensive literature evaluation from BC-based SSI solutions and business marketplace services combined with trendy academic courses. It additionally offers a proof of the Blockchain technology building in IDMS roadmap. As an efficient BC-based IDMS answer that supports SSIN's imaginative and prescient certification, this story describes the five key regions of a BC-based totally IDMS user: trendy, perfection, anger, consider and simplicity. In addition to the meta-evaluation, which describes the numerous styles of adversarial threats that may reason potential damage to BC-based totally IDMS, some of BC-based totally IDMS solutions are extensively explored and discussed inside the instructional literature. We also spotlight ability research possibilities and targets for destiny research.

Disadvantage:

- Time of intake.
- Low security.
- Server troubles occur frequently.

PROPOSED SYSTEM:

In this surroundings, the administrator can identify all users with those money owed. The administrator has get entry to all consumer files uploaded to the system. Everything can be achieved through the consumer. Check the admin status and notice if any manufacturing is going on. You can check as a user complaint.

To log into the system, the user have to first register; Once the account is confirmed by means of the administrator, you can log in along with your username and password. User can update profile as per his requirement. User can manage their files in the device via including, including and deleting. All files submitted by using the user are tested by using the Administrator. User can test QR code and standing of organized documents.

The consumer can test whether or not records were up to date or deleted and the fame of each file. If he discovers a fraud or an movement that the consumer did not do, he can inform the Administrator. The user can view all of the details by way of scanning the QR code. All documents can be added and established through the public file.

Advantages:

- Ease of use for the user.
- Better than before.
- Continually energetic.
- Known server problems.

SYSTEM ARCHITECTURE:



MODULES:

- Admin Module.
- User Module.
- CRUD Module.
- Documentation Module.

Admin Module:

In this module, the admin furnished details on all their record necessities, the admin can control all users immediately, the admin can view the delivered consumer or the admin can view the report manufacturing found and all of the necessities to view and validate. Blocks using blockchain.

User Module:

In this module, users are furnished with authentication and protection to get entry to the details shown within the ontology machine. Before having access to or browsing info, the consumer must have an account, in any other case sign up first. At a minimal, you should offer your e mail address, username, password, show call, and any other profile fields you have installation. The display call is used while the system desires to show the consumer's actual name..

Crud Module:

The full uncooked form of the identifier is to create, read, update and delete. You can do the whole lot right here and do it with your documents.

Documentation Module:

The person can test whether documents were updated or deleted and get precise report records about the record. If someone sees fraud or does no longer need to cope with it anymore, they are able to document it to the

administrator. QR code person can see all the data. They can verify all the documents and account of the books.

RESULT & DISCUSSION

There are extreme worries approximately the significant virtual media used to mediate communications among people, offerings and gadgets. A succession of centered solutions from service carriers is wasteful in duplication, has significant safety vulnerabilities and burdens users. The concept of private identity, which combines verified private traits and a unified digital identity, permits information users to discover their ownership and recognise how their records is getting used. A predominant application of generation is believed to be in finance. With the appearance of blockchain technology, self sustaining identities can now obtain decentralization, in which each node participates independently from the others. Decentralized structures ought to circulate away from the supply of centralized authorities. Similar standards are brought to connect whilst retaining one's independence and internal intimacy.

So using blockchain for personal identity security system helps us to improve security in specific regions. It actually helps us to spread data across different network with security by some special codes. And people are finding it better for their privacy being protected because the access is managed carefully. Not only that it keeps record of any transactions and making it easier to change the mistakes. The system makes sure that the unauthorized people getting information with the help of decentralized structure and secret codes.

Smart contracts are like smart rules that controls and who could see the data of the user. This decentralised and cryptographic makes sure that the risks of centralised systems are avoided. In which it offers more secure environment which immune to hacking and auditable trail of identity related transactions. Making sure that everyone involved in the network is responsible for their action and simplifies any error detection and remediation. The system using blockchain to secure personal identities showed it works well, as seen in the results. It made personal information safer by using special technology that's hard for third parties to break into. People also had more control over their privacy, deciding who gets to see the what information they needed. If there were any mistakes, the system made it easier to find and fix them. By emerging as a favourable solution to the hurdles that occurred by the old fashioned methods of the management system. Providing uninterrupted cooperation of different nodes or participants, the synchronization of transactions, or the overall balance and unity achieved in the decentralized and distributed ledger system. Which shows us its enhanced privacy and transparent accountability in the digital world.

CONCLUSION:

Ultimately, blockchain-primarily based private identity safety structures need to improve the security and privateness of personal identity statistics and prevent identification robbery and fraud. However, demanding situations nevertheless stay, which includes the need for large facts sets and ensuring interoperability with existing systems. It is important that the transparency and accountability of those agencies build customer self belief. Continued studies and improvement of these structures is essential to boom the advantages. Despite the demanding situations, blockchain-based private identity safety structures provide substantial capacity benefits by means of selling a safer and extra obvious digital society and giving human beings extra manipulate over their personal identification facts. More studies and development is wanted to maximise these benefits, together with exploring greater agile clever contracts, integrating other varieties of statistics, and making sure transparency and accountability. The use of blockchain-based private identification safety systems has the capacity to enhance the way we control and shield private identification information. By promoting a safer and more dependable digital society, blockchain-primarily based structures will help save you harm from identity theft and fraud and supply humans more control over their private identification information. It is consequently essential to continue research and development in this area and to work towards the efficient and ethical implementation of these structures.

FUTURE SCOPE:

The destiny surroundings of blockchain, based totally on personal identification security systems is sizeable, there are many opportunities for research and development. Here are a few capability regions of future improvement: Interoperability: the improvement of interoperability requirements and protocols among different blockchain-based totally private identity safety systems will improve the performance and scalability of these structures. Decentralized identity control: By growing decentralized identification management structures that can be used throughout a couple of programs and offerings, customers can benefit more manage over their private identity data.

REFERENCES:

1. G. D. P. Regulation, "Regulation (eu) 2016/679 - directive 95/46," Official Journal of the European Union (OJ), vol. 59, pp. 1–88, 2016.
2. C. Tankard, "What the gdpr means for businesses," Network Security, vol. 2016, no. 6, pp. 5–8, 2016.
3. O. K. Foundation and the Open Rights Group, "Personal data and privacy working group," 2014.
4. A. Poikola, K. Kuikkaniemi, and H. Honko, "Mydata a nordic model for human-centered personal data management and processing," Finnish Ministry of Transport and Communications, 2015.
5. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
6. Christopher Allen, Kim Hamilton Duffy, and Joe Andrieu, "Self-Sovereign Identity and Decentralized Identifiers," 2016.
7. Ali Dorri, Salil S.Kanhere, and Raja Jurdak, "Blockchain-Based Identity Management for Autonomous Vehicle," 2017.
8. Prateek Sachan and Omer Rana, "A Review of Blockchain-Based Identity Management Systems," 2018.
9. Atif Khan, Abid Khan, and Jaehun Bang, "Blockchain Identity Management: Features, Open Challenges, and Future Trends," 2019.

