**A BLOCKCHAIN-BASED DATA SHARING SYSTEM**

Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering degree in Computer Science and Engineering

By

## Student name: J K Srimathi Reg No.: 41613018



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

# SCHOOL OF COMPUTING

**SATHYABAMA**

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

**Accredited with Grade “A” by NAAC JEPPIAAR NAGAR, RAJIV GANDHISALAI, CHENNAI – 600119**

## November – 2022

**i**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

## (DEEMED TO BE UNIVERSITY)

Accredited with ―A‖ grade by NAAC Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

[**www.sathyabama.ac.in**](http://www.sathyabama.ac.in/)

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**BONAFIDE CERTIFICATE**

This is to certify that this Product Report is the bonafide work of **J K Srimathi (41613018)** who carried out the Design entitled **“A BLOCKCHAIN BASED DATA SHARING SYSTEM”** under my supervision from June 2022 to November 2022.

**Design Supervisor (**Ms.C.A.Daphine Desona Clemency M.E**)**

## Head of the Department Dr.Vigneswari

**Submitted for Viva-voce Examination held on**

## Internal Examiner External Examiner

**ii**

## DECLARATION

I**, J K Srimathi (41613018),** hereby declare that the Product Design Report entitled **“A BLOCKCHAIN BASED DATA SHARING SYSTEM”** done by me under the guidance of Ms.C.A Daphine Desona Clemency M.E**.,** 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**

**iii**

## ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management** of **SATHYABAMA** 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.Vigneswari.,** 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 Design Supervisor Ms.C.A.Daphine Desona Clemency M.E**.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-1 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.

**Iv**

## ABSTRACT

Every citizen has the right to secure their data and, thus, the right to control their particular information, deciding to where, when, and whom their information is available. This paper proposes a secure, nimble, and effective system for a distributed, automatic, and transparent data trading between disciplines using blockchain, smart contracts, trust, and character. We develop and apply a prototype of a trust and character system grounded on real- life relations. The results show that the proposed system provides security and sequestration in a quick and distributed way, performing hundreds of deals per second, and effectively chastising vicious behavior.

v

# Table of contents

|  |  |  |  |
| --- | --- | --- | --- |
| **Chapter No** | **TITLE** | | **Page No.** |
|  | **ABSTRACT** | | v |
|  | **LIST OF FIGURES** | | vii |
| 1 | **INTRODUCTION**  1.1 Overview | | 1 |
| 2 | **LITERATURE SURVEY-**  2.1 Product Availabilities | |  |
| 3 | **REQUIREMENTS ANALYSIS** | |  |
|  | 3.1 | Objective |  |
|  | 3.2 | * + 1. Hardware Requirements     2. Software Requirements (if needed) |  |
| 4 | **DESIGN DESCRIPTION OF THE PROPOSED PRODUCT** | |  |
|  | 4.1 | Proposed Product   * + 1. Design Diagram of the full product     2. Various stages |  |
|  | 4.2 | Product Features   * + 1. Novelty of the Product     2. Product Upgradation |  |
| 5 | **CONCLUSION** | |  |

|  |  |  |
| --- | --- | --- |
|  | **REFERENCES** |  |
|  |  |  |

**CHAPTER 1 INTRODUCTION**

## 1.1 OVERVIEW

Data sharing is one of the most crucial requirements in today’s world. Access control of personal data is a major threat. Downsides of centralized data sharing:

Easy access to third parties, lack of control, falling prey to high fees, and the threat from attackers. Decentralized data sharing solves this problem. Some of the features of a blockchain-based system: are more privacy, security, and control over data.

Our paper proposes:

-> A blockchain-based decentralized data-sharing system

-> Integration of a Trust and Reputation System into the system for enhancing transparency and authenticity.

**1**

# CHAPTER 2 LITERATURE REVIEW

**2.1 PRODUCT AVAILABILITIES**

Every individual has the right to privacy and, consequently, to decide how, when, and where their personal information is made available. This study suggests a blockchain-based, smart contract-based, trust-based, and efficient system for distributed, automatic, and transparent data exchange between domains. We create and put into practice a working prototype of a trust and reputation system. The outcomes demonstrate that the suggested system offers security and privacy in a rapid and dispersed manner, handling hundreds of transactions per second and successfully deterring hostile conduct.

# CHAPTER 3

## REQUIREMENTS ANALYSIS

* 1. **OBJECTIVE OF THE PRODUCT**

Our paper aims at providing a data sharing platform with the following features:

* + 1. Building a decentralized and distributed system-thus preventing data breaches and leaks.
    2. Incorporating smart contracts- thus making the system more automatic and transparent.

Integrating a trust and reputation system (TRS)-thus building the reputation of parties, providing legitimacy and authenticity.

* + 1. Decentralized nature of the system reduces the risk of malicious parties and attackers. Improves the security and integrity of the system thereby reducing foul play.

## REQUIREMENTS

### HARDWARE REQUIREMENTS FOR BLOCKCHAIN

1.High core 2.GPUs

3.Nodes/clients 4.clusters

### SOFTWARE REQUIREMENTS FOR BLOCKCHAIN

1. solidity - popular language used by blockchain developers which is designed by Ethereum Virtual machine
2. Geth - built using Go programming language. Used to host different tasks on blockchain
3. Mist - it is an official Ethereum wallet. Used to deploy smart contracts
4. remix – browser-based blockchain tool used to create and deploy smart contracts
5. testnet
6. Ganache 7.Truffle

8. meta mask

# CHAPTER 4

## DESIGN DESCRIPTION OF THE PROPOSED PRODUCT

* 1. **PROPOSED PRODUCT**
     1. The gadget uses the integrity and auditability properties of the blockchain to keep each user's entry to permission to personal and touchy facts.
     2. A popularity and faith machine that considers the history of interactions and the participants' opinions to construct a seller's reputation.
     3. The transactions consist of the evaluation of the vendor and the quality of the marketed data.
     4. Conversely, popularity represents a world view of the system regarding a particular vendor and consists of aggregating individual trusts of all shoppers in the system

### Design Diagram of full product

* + 1. ***Various Stages***

There are four modules in this project. We test each module separately.

1. Analysing the data
2. Creating a blockchain
3. Incorporating smart-contracts
4. Testing the integrated system

We analyze each part of the system separately and then finally release the final data- sharing system.

### Analysing data

Several works look into how trust and reputation can be used to provide security in computer networks. To develop trust, Velloso et al. offer a trust model based on human interactions.

Between nodes in an ad hoc network The Eigen trust algorithm, proposed by Kamvar et al., assigns a global trust value to each peer in a peer-to-peer (P2P) file- sharing system.

The technique is based on each peer's previous file uploads and employs indirect trust to calculate the system's overall trust. Sun et al. offer security methods in a mobile ad hoc network against attacks on trust and reputation (MANET). In trustless contexts, blockchain technology can provide audibility and traceability.

We take the ideas from the preceding publications and adapt them to build a trust and reputation system that takes into account the features of blockchains and distributed trustless environments. The other efforts use blockchain to build trust and reputation.

Oliveira and colleagues suggest a blockchain reputation-based consensus mechanism (BRBC). Dennis and Owen suggest a file transfer reputation system in which users' recommendations are openly stored on the blockchain. However, the authors do not provide a solution to the on-off attack and do not implement and examine their proposal Malik et al. propose a paradigm for managing participants' confidence in a blockchain that records supply chain transfers. Buyers, government agencies, and sensors vouch for the product's quality by providing assessments via blockchain transactions.

A smart contract determines a trader's reputation by averaging the ratings of the entities. When measuring reputation, the authors limit their idea to the scenario of a supply chain and ignore product pricing. Furthermore, the concept is vulnerable to on-off attacks, in which a user switches between lawful and malicious activities, causing network damage without being discovered.

To restrict access to IoT devices, Putra et al. offer a trust and reputation management system for blockchain-based networks. The system use smart contracts to evaluate each node's trust and reputation, finding and deleting rogue nodes from the network. The authors employ reputation as an attribute to regulate device access, specifying a minimum reputation that a participant must have to ensure data access. However, the concept is vulnerable to one-time assaults and does not reward data owners. Furthermore, the owners' marketing of data is a desirable quality.

Unlike the previously listed publications, this research presents a blockchain-based trust and reputation system that is successful in punishing harmful behaviour and is adaptable to a blockchain-based data marketplace. The entire marketplace system is secure, nimble, and automated, rewarding sellers who sell their personal information while punishing fraudulent sellers.

ATTACKER MODEL:

Attacks on the system of trust and reputation constitute an attempt by a malicious node to improve or harm the reputation of legitimate nodes. We discuss five attacks: I the bad-mouthing assault, (ii) the on-off attack, (iii) the Sybil attack, (iv) the newbie attack, and (v) the conflicting-behavior attack.

Table I lists each attack as well as the potential countermeasures.

A blockchain attacker, in our opinion, seeks to prevent a participant from adding a genuine transaction or block to the blockchain. The consensus protocol's fault- tolerance attribute requires the attacker to control the majority of organisations in order to effectively alter the consensus protocol, mitigating this form of attack. The blockchain structure's immutability and distribution properties enable transaction and block issuers to verify that their proposal was correctly added to the network.

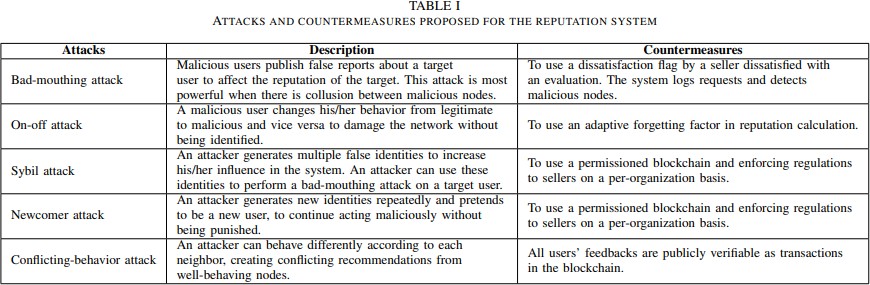
Attacks on vendors or purchasers attempt to obtain sensitive and private advertised data or mimic the victim. We believe that all advertising data in the blockchain is encrypted, which means that if an attacker obtains access to the data, he or she must obtain the key that decrypts it in order to obtain personal information about the seller. To deceive other parties, an attacker may attempt to imitate the target.

However, this technique is rendered ineffective once the system compels all issuers to sign their transactions.

Furthermore, the blockchain documents any attempt to modify its structure using a stolen pair of keys, allowing the victim to avoid future damage by replacing his or her stolen pair of keys.

In our approach, network attacks are viewed as an attacker attempting to isolate a participant in order to prevent him or her from issuing transactions. We minimise this form of assault by building redundant channels amongst blockchain network

participants.



### Creating a blockchain

BLOCKCHAIN:

A goal of blockchain is to allow digital information to be reached and distributed, but not edited. A blockchain is the foundation for immutable ledgers, records or transactions that cannot be destroyed. Bitcoin and Ethereum are popular examples of blockchain.

* + - * + The blockchain is shared, immutable ledger that facilitates the process of transactions and tracking assets in a business network.
        + The blockchain consists of a set of blocks maintained by a decentralized peer- to-peer network.
        + Distributed ledgers can be used to support financial transactions or smart contracts.

TYPES OF BLOCKCHAIN:

The basic application of the blockchain is to perform transactions in a secure network. That is why people use blockchain and ledger technology in different scenarios.

There are 4 types of blockchain they are

1. Public blockchain
2. Private blockchain
3. Hybrid blockchain
4. Consortium blockchain

TO BUILD A BLOCKCHAIN:

To build a new blockchain we have to follow some of the steps which are

1. To choose a consensus mechanism. A blockchain’s operating protocol is also known as its consensus mechanism.
2. Design your blockchain architecture.
3. Audit your new blockchain and its code.
4. Verify legal compliance.

To create a new blockchain we need to follow these basic steps. And we should know about the time taken and the cost to create the blockchain.

TIME TAKEN TO DEVELOP A BLOCKCHAIN:

Once the Poc (Proof of concept) is done, it takes 4 to 5 weeks to develop minimum features. Launching an application on the main net takes around 2 to 3 months based on the requirements of a client.

Blockchain Programming developers support the use of c ++ as it is directly abundant in terms of run time polymorphism, function overloading, and multithreading. It allows developers to mould the data according to their needs. But not only C ++ but many other languages are used to build or develop a blockchain.

COST OF BLOCKCHAIN:

The cost of a blockchain app development project will depend on all these factors. For instance, if you hire a small to midsize mobile app development company, the mobile app cost will range from $ 20,000- $ 70,000. The multinational mobile app developers will charge around $ 60,000 to $ 200,000 +.

HOW BLOCKCHAIN CAN HELP DATA SECURITY?

About Blockchain:

* Blockchain is a public ledger. It is distributed and decentralized network.
* Blockchain technology was described in the year 1991 by the research Scientists Stuart Haber and W. Scott Stornetta.
* Blockchain is a software Protocol that runs with internet.
* It is a chain of blocks which contain information. Each block records all the recent transactions and goes into the blockchain as a permanent database.
* Blockchain is a constantly growing ledger that keeps a permanent record of all the transactions that have taken place in a secure, chronological and immutable way.
* Each time a block gets completed a new block is generated. BLOCKCHAIN SECURITY ASSUMPTIONS:

Security Assumptions in Blockchain:

Algorithms:

→Byzantine Fault Tolerance

→ Well - designed and implemented algorithms. Cryptography:

→ Digital Signature

→ Hash Functions Infrastructure:

→ Reliable communications network.

→ Secure updated nodes.

BLOCKCHAIN SECURITY ISSUES:

The Blockchain is implemented as software running on distributed nodes.

Blockchain Security can be jeopardized by failing to identify and correctly fix bugs. The users find very difficult to identify the bugs and they find difficult to correct the bugs. users are failing to install updates.

Malicious actors can work inside or outside the blockchain ecosystem to compromise its security

→ Denial of service.

→Intercepting, modifying or dropping communications.

→ Node hijacking.

BLOCKCHAIN ISSUES AND CHALLENGES:

Decentralization:

Blockchain networks are designed to be decentralized

→ No central authority

→No single point of failure

They rely on a consensus algorithm to achieve synchronization across nodes.

→ These algorithms make security assumptions.

→Violating these assumptions can destroy the security of the distributed ledger. A decentralized, distributed network is difficult to disable if something goes wrong

→ Every node still has a complete copy of the ledger.

LIMATIONS:

Blockchain technology has the potential to disrupt and improve many industries and traditionally centralized systems. But the biggest limitation is scalability.

It is very hard for blockchains to grow and support increasing number of transactions.

### smart contracts

A smart contract is a self-executing contract with the terms of the agreement between buyer and seller being directly written into lines of code.

The code controls the execution, and transactions are trackable and irreversible. usually, smart contracts run on the Ethereum blockchain.

Smart contracts are based on blockchain technology. In short, it is a distributed register that looks like a decentralized system and is located on many computers connected to one network. Blockchain makes it possible for users to make transactions, transfer information and securities without banks and intermediaries.

Example:

To build the best customer experience

Smart contracts can cultivate a stronger B2C relationship in real-time. For example, a shoe brand partnering with a streaming music service offers complimentary subscription time if the consumer creates a playlist to listen to while jogging. A smart contract sends the customer an offer for a discount on new.

Shoes or suggests songs with a similar tempo to add to the playlist. It could boost customer expectations by supporting runners who enjoy listening to music and tracking their fitness.

Benefits of smart contracts:

1. speed efficiency and accuracy
2. trust and transparency
3. security

### Integrating trust and reputation system

The blockchain construction is complementary to the one second hand in camilo and others.

Accompanying the definition of three types of undertaking for dosser business. They are the three types:

1. Broadcast transaction.
2. purchase transaction.
3. reaction undertaking.

The propose of a response undertaking apart from the three transactions to present the trust and name structure. The data owners concerned in making accessible and commercialized transmit advertisement undertaking and the hidden submits the announced data to storage. Attendant fit advocating and processing big amounts of dossier and issues a marked advertisement under taking. The notice transaction must hold a brief writing of the announced data type example, dossier from healing sensors and the dossier price. Buyers follow dossier in the blockchain by qualifying advertisement undertaking. To those who concerned in getting data advertised in the blockchain issue purchase under taking updating the buyer's IP address to an SDN boss, the reads the undertaking and grants approach.

Data partner issue answer undertaking instinctively subsequently the dossier addition from a colleague.

The answer transaction sends to the client the key that decrypts the closer.

A Smart Contract kills the undertaking before accord colleagues order bureaucracy in a block.

### Testing the final data sharing system

1. Analysing the data: this step checks for the proper consistency in the data.
2. Creating a blockchain: we create a blockchain then we test it. There are several phases in testing a blockchain.

Phases of blockchain testing

The initiation phase is the first stage of testing a blockchain system. In the design phase, the key components of the system that must be tested are identified, and a well-detailed test strategy tailored to the blockchain system is developed. Result phase This is the final phase, which includes a report on the overall test performed in the system.

1. Testing smart contracts: there are several setups for testing a smart contract
   1. Setting up the development environment
   2. Creating a Truffle project
   3. Writing the smart contract
   4. Compiling the smart contract
   5. Migrating the smart contract
   6. Testing the smart contracts

In Truffle, there are two ways to test smart contracts. JavaScript is used for the second and Solidity for the first.

1. Testing integrated system: here we pass test cases to the integrated system and its working is analyzed.

# PRODUCT FEATURES

* + 1. ***Novelty of the product***
* Data sharing in blockchain can provide organizations with a secure way to store and distribute data.
* It is a decentralized network. And it has no owners in the blockchain database.
* the data travels through the peer-to-peer network secured by an immutable cryptographic signature.
* Smart contracts.

# CHAPTER 5 CONCLUSION

* Blockchain technology, combined with smart contracts, presents the needed transparency for impenetrable information trading, permitting holders to keep manage of their data.
* The effects exhibit that the proposed system mitigates usual attacks from popularity structures efficiently and information transactions rapidly in the blockchain.
* A fee of hundred and fifty transactions per second.
* Our proposed trust and reputation mannequin closely punish malicious sellers.
* In future work, we intend to put in force the proposed recognition system in clever contracts to warranty the popularity processing in a distributed and computerized way.

# REFERENCES

* E. Lodderstedt, M. McGloin, and P. Hunt, “OAuth 2.0 threat model and security considerations,” 2013, iETF. RFC 6819. Available at [http://www.rfc-editor.org/rfc/rfc6819.txt.](http://www.rfc-editor.org/rfc/rfc6819.txt) Last access: 15 July 2020.
* G. F. Camilo, G. A. F. Rebello, L. A. C. de Souza, and O. C. M. B. Duarte, “AutAvailChain: Automatic and secure data availability through blockchain,” in IEEE GLOBECOM, 2020, pp. 1–6, To be published.
* M. T. de Oliveira, L. H. Reis, D. S. Medeiros, R. C. Carrano, S. D. Olabarriaga, and D. M. Mattos, “Blockchain reputationbased consensus: A scalable and resilient mechanism for distributed mistrusting applications,” Computer Networks, vol. 179, p. 107367, Oct. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/ pii/S1389128620300360.
* G. D. Putra, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, “Trust Management in Decentralized IoT Access Control System,” in IEEE ICBC’2020. IEEE, 2020, p. 9.
* G. A. F. Rebello, I. D. Alvarenga, I. J. Sanz, and O. C. M. B. Duarte, “BSec-NFVO: A blockchain-based security for network function virtualization orchestration,” in IEEE International Conference on Communications (ICC), 2019, pp. 1–6.
* O. J. A. Pinno, A. R. A. Gregio, and L. C. De Bona, “ControlChain: ´ A new stage on the IoT access control authorization,” Concurrency and Computation: Practice and Experience, 2019, e5238 cpe.5238.
* H. T. T. Truong, M. Almeida, G. Karame, and C. Soriente, “Towards secure and decentralized sharing of IoT data,” in IEEE Blockchain’2019, 2019, pp. 176–183.
* T. S. L. Nguyen, G. Jourjon, M. Potop-Butucaru, and K. L. Thai, “Impact of network delays on Hyperledger Fabric,” in IEEE Conference on Computer Communications Workshops (INFOCOM), 2019, pp. 222– 227.
* C. Gorenflo, S. Lee, L. Golab, and S. Keshav, “FastFabric: Scaling hyperledger fabric to 20,000 transactions per second,” in IEEE ICBC’2019, 2019, pp. 455–463.
* R. A. Michelin, A. Dorri, R. C. Lunardi, M. Steger, S. S. Kanhere,

R. Jurdak, and A. F. Zorzo, “SpeedyChain: A framework for decoupling data from blockchain for smart cities,” in Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, 2018, pp. 145– 154.