

Sensitive Data Transactions Between Users And Authority

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology degree in Information Technology

by

Gnana Sounthariya S (Register no 38120025)

Keerthiga N (Register no 38120042)



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600
119**

MARCH - 2022



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 INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of Gnana Sounthariya S (38120025) and Keerthiga N (38120042) who carried out the project entitled "SENSITIVE DATA TRANSACTIONS BETWEEN USERS AND AUTHORITY" under my supervision from October 2021 to April 2022.

Internal Guide

DR. JEBERSON RETNA RAJ M.E., Ph.D.,

Head of the Department

DR. R. SUBHASHINI M.E., PH.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **Gnana Sounthariya S (Reg no:38120025)** and **Keerthiga N (Reg no:38120042)** hereby declare that the Project Report entitled **SENSITIVE DATA TRANSACTIONS BETWEEN USERS AND AUTHORITY** done by me under the guidance of **Dr. R. Jeberson Retna Raj M.E., Ph.D.**, Department of Information Technology at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in INFORMATION TECHNOLOGY.

DATE:

PLACE: CHENNAI.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **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 and **Dr. R. Subhashini M.E., Ph.D., Head** of the Department of Information Technology 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 **Dr. R. Jeberson Retna Raj 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 Information Technology** who were helpful in many ways for the completion of the project

ABSTRACT

A block chain-based processing framework for sensitive data is proposed. The underlying block chain module provides technical support, such as virtual machines, consensus algorithms, transaction verification mechanisms, and accounting mechanisms. The E-contract layer module provides a distributed application service and uses the block chain technology to support it. The proposed smart system is used by each party get involved in the production of sensitive data. The final sensitive data are produced by the final data generator, and other modules involved in the process of data production are unaware of the final data. This approach prevents the leakage of sensitive data into the circulation.

TABLE OF CONTENT

CHAPTER NO	TITLE NAME	PAGE NO
	ABSTRACT	I
	TABLE OF CONTENTS	II
	LIST OF ABBREVIATION	IV
	LIST OF FIGURES	V
1	INTRODUCTION	
	1.1 GENERAL	1
	1.2 MOTIVATION	1
	1.3 OUTLINE OF THE PROJECT	2
	1.4 PROBLEM STATEMENT	2
	1.5 OBJECTIVE OF THE PROJECT	2
2	AIM AND SCOPE	
	2.1 AIM	3
	2.2 PROBLEM DEFINITATION	3
	2.3 LITERATURE REVIEW	3
	2.4 SCOPE	5
3	ALGORITHM AND METHODS	
	3.1 ALGORITHM	6
	3.2 EXISTING SYSTEM	9
	3.3 PROPOSED SYSTEM	11
	3.3.1 ADVANTAGES OF PROPOSED SYSTEM	11
	3.4 MODULES	11
	3.4.1 STAFF REGISTER	12
	3.4.2 STAFF LOGIN	12
	3.4.3 STAFF FILE VIEW	12
	3.4.4. STAFF FILE REQUEST	12
	3.4.5 STAFF DOWNLOAD FILE	12

	3.4.6 TEAM LEADER LOGIN	12
	3.4.7 TEAM LEADER FILE UPLOAD	13
	3.4.8 TEAM LEADER FILE VIEW	13
	3.4.9 MANAGEMENT LOGIN	13
	3.4.10 MANAGEMENT TEAMLEADER LOGIN	13
	3.4.11 MANAGEMENT GENERATE KEY	13
	3.4.12 MANAGEMENT RESPONSE	13
4	SYSTEM REQUIREMENTS AND DESIGN	
	4.1 REQUIREMENTS	14
	4.1.1 HARDWARE REQUIREMENTS	14
	4.1.2 SOFTWARE REQUIREMENTS	14
	4.2 SOFTWARE DESCRIPTION	14
	JAVA FRANEWORK	15
	JSP	15
	JAVA SERVLETS	15
	4.3 SOFTWARE TESTING	16
	4.3.1 FEASIBILITY STUDY	16
	4.3.2 ECONOMICAL FEASIBILITY	16
	4.3.3 TECHNICAL FEASIBILITY	16
	4.4 SYSTEM ARCHITECTURE	20
	4.4.1 USE CASE DIAGRAM	21
5	RESULTS	
	5.1 SOFTWARE SETUP	22
	5.2 EXPERIMENTAL RESULTS	22
	5.3 LIMITATION	23
6	CONCLUSION	
	6.1 CONCLUSION	24
	6.2 FUTURE WORK	24
	REFERENCE	25
	APPENDIX	26
	SOURCE CODE	26

LIST OF ABBREVIATION

1.	DB	Data Base
2.	DBC	Data Base Confidentiality
3.	JVM	Java Virtual Machine
4.	JSP	Java Server Page
5.	SHA	Secure Hash Algorithm
6.	AES	Advanced Encryption Standard

LIST OF FIGURES

FIGURE NO	TITLE OF THE FIGURE	PAGE NO
3.1	AES architecture	8
3.2	Byte substitution	9
3.3	Row shifting	9
3.4	Column mixing	10
3.5	XOR operation	10
4.1	Architecture diagram	20
4.2	Use case diagram	21

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Sensitive data transaction is confidential records that have kept protected and with no reach by any outsiders until there's a permission which must get right to enter. Admittance to reactive data must be shortened through sufficient records protection and records security which promotes designed to prevent statistics leaks with statistics violation. Sensitive statistics can be any sort of records that wishes to be protected from unauthorized get right of entry to shield the privacy or security of a man or woman or enterprise. It can include any data pertaining to passwords. we propose an exchange structure for delicate information and a blockchain-based framework. The sharp organization comprises of a blockchain module, which is an electronic agreement layer hub, and a Software-as-a-Service module. A dispensed module transporter is outfitted by contract layer. Also, this runs the touchy measurements exchange system code and that further created at the Electronic-contract layer by utilizing the calculation to encrypt the age. Thus blockchain concept presents mechanical help, like computerized machines, agreement calculations, and components for exchange confirmation. The software layer concept presents a stage supplier using cloud that empowers every birthday celebration to participate in big business correspondence through online interfaces. The brilliant gadget utilizes the concept to notice every move and cooperation of every birthday festivity in producing data

1.2 MOTIVATION

- Every company has sensitive information that hackers are after. Almost any kind of information regarding a business, their clients or customers and business dealings can be sold on the black market.
- A lot of business owners believe that they don't have to worry too much now that security has gotten an upgrade.
- What most people fail to realize is that as technology improves so do hacking

attempts. Hackers improve their skills and they can use the same technology to breach security.

- The above are therefore reasons to design and implement an sensitive data transactions using block chain.

1.3 OUTLINE OF THE PROJECT

The project is all about transferring the sensitive data. This project helps the company to avoid data theft. We are using SHA and AES algorithm. SHA is used for generating the hash code and AES is used for encrypt sensitive data. A company employee should login in database and then if staff wants a file means staff send a request to team lead once team lead accept the request the request will send to management once management accept it then staff can download it. There are two file key send it in staff one from team lead another one is from management.

1.4 PROBLEM STATEMENT

In this project we have fostered my the electronic application. We are utilizing the various modules from assigned the information starting with one module then onto the next modules. Also, every information move will screen in the principle information dealing with modules. The group chief will be share the date structure pdf configuration of information will be share(store the encryption information in data set) AES calculation. AES is the semantic key encryption calculation and SHA is the hashing calculation we interface the information base.

1.5 OBJECTIVE OF THE PROJECT

Collecting staff information and maps their relationships for a complete picture of user account organization. We'll help you with privacy design when creating users, groups, and role-based permissions Encryption is a very generic term and there are many ways to encrypt data. Companies need to implement and manage encryption correctly. The key to a good encryption strategy is using strong encryption and proper key management. Encrypt sensitive data before it is shared over untrusted networks (ex. Encrypted file storage).

CHAPTER 2

AIM AND SCOPE

2.1 AIM

The aim of our project is to design a system that will be transfer the data into the organization and secure way transfer the data from particular staff to management. In this project management handle the every data. Include staff and team leader information.

2.2 PROBLEM DEFINITION

The fundamental goal of the undertaking is Sensitive realities are private realities that should be keep up with safeguarded and far away from all pariahs except if they have authorization to get admission to it. Admittance to tricky measurements must be bound through adequate information security and realities insurance rehearses intended to forestall records breaks and information breaks. Security is consequently provided through the idea of a one way that has an element that is ordinarily the vital component of the SHA calculation. The pre-picture obstruction is essential to clean up beast force assaults from a bunch of colossal and strong machines.

2.3 LITERATURE REVIEW

According to the Survey on Sensitive data transaction between user and authority system, the data transaction system includes the processes of file transformation, Safe and secured, administrator or authority person, etc.

T. Dargahi print et.al "A deliberate writing audit of blockchain network safety," in 2020 - - This study investigates peer-explored distributions endeavouring to utilize blockchain for digital protection and gives a far-reaching examination of the most generally utilised blockchain security applications. [1]. J. Lee and D. Liu, "CNN based Malicious Website Detection by Invalidating Multiple Web Spams," in 2020 - - The objective of this exploration is to make another profound learning-based vindictive URL discovery calculation. We made a person-based word inserting approach, and the vector portrayal of a URL is advanced consequently by consolidating character and

word installing [2]. W. Martin,

Friedhelm, and K. Axel, "Following assembling processes utilizing blockchain-based symbolic structures" in 2019 - - We have proposed a blockchain-based inventory network the executive's framework that empowers following and the following merchandise, remembering their change for the creation cycle utilising brilliant agreements [3]. N. Malik, D. Puthal, S. P. Mohanty, E. Koulianos and G. Das, "All that You Wanted to Know About the Blockchain: Its Promise, Components, Processes, and Problems," in 2018 - - The blockchain is used from one side of the planet to the other to get decentralised shared framework. This article gave a total outline of the blockchain by exhibiting the blockchain's working worldview and afterwards laying out the framework's qualities [4]. L. Peng, Z. Yan and W. Feng, "Protection conservation in the permissionless blockchain: A review. Advanced Communications and Networks", in 2020 - - Existing security assurance advances are examined and positioned utilising our proposed appraisal models. From the point of view of protection issues, we in the end sort out open exploration issues as well as future examination ways [5]. N. Kakadu and U. Patel, "Secure Secret Sharing Using Homomorphic Encryption", 2020 - - Secret sharing is a crucial instrument for keeping up with information protection and mystery. Secret sharing is the most common way of imparting a mystery snippet of data to a gathering. The reason for secret sharing is to give secret security, protection, and data covering [6].

M. Ananth and S. Sundari, "Secure multi-party calculation in differential private information with Data Integrity Protection," in 2015 - Secure multi-party registering and differential protection for in an upward direction divided information between parties are consolidated in this examination. For the outstanding instrument, a multi-party convention has been created to accomplish this [7]. T. Lei, Y. Gao, S. Jiao, Z. Xia, "Known-Plaintext Attack and Ciphertext-Only Attack for Encrypted Single-Pixel Imaging", in 2019 -- Under explicit circumstances, the ciphertext-just assault is executed considering factual parts of single-pixel power values. The two methodologies are fit for breaking encoded SPI frameworks and recuperating the key containing the right lighting designs [8]. S. Kaushik, and S. Puri, "Online exchange handling utilising upgraded delicate information move security model", in 2012 - - A methodology is introduced in this paper that plans to protect significant data given during on the web exchange handling (OLTP) [9]. W. Zheng, K. Dai, P. Li and R. Chen,

"Nutbar: A Blockchain-as-a-Service Platform", in 2019 - - Developers might zero in on the business code as opposed to keeping up with and observing the framework, permitting them to examine how to utilize blockchain innovation more appropriately to their business cases [10].

F. Club and C. Pataki's, "An Efficient Blockchain-Based Privacy-Preserving Collaborative Filtering Architecture", in 2020 - - Extensive exploratory discoveries show that our system outflanks best in class approaches as far as execution and adequacy. Moreover, a survey of its benefits and prospects offers an establishment for future review [11]. D. Chilies, J. Homan, "Mechanical check of exchange handling frameworks", in 2000 - - State machines lay out the locking and fix/retry components at scattered destinations in our framework, while affirmations depict the cooperation between locales as indicated by the two-stage submit convention. Utilising PVS' intuitive evidence checker, we showed that our framework meets the atomicity, strength, and serializability necessities [12].

2.4 SCOPE

- To collect staff information and maps their relationships for a complete picture of user account organization.
- Encryption is a very generic term and there are many ways to encrypt data.
- To companies need to implement and manage encryption correctly.
- The key to a good encryption strategy is using strong encryption and proper key management.
 - Every staff request permission will be connected in hash values in database

CHAPTER 3

ALGORITHM AND METHODS

3.1 ALGORITHM

In the field of cryptography and crypt analytics, the SHA-1 algorithm is a crypt-formatted hash function that is used to take a smaller input and produces a string that is 160 bits, also known as 20-byte hash value long. The hash value therefore generated, is known as a message digest which is typically rendered and produced as a hexadecimal number which is specifically 40 digits long.

Use of SHA algorithm

These SHA algorithms are widely used in security protocols and applications, including the ones such as TLS, PGP, SSL, IPsec, and S/MIME. These also find their place in all the majority of cryptanalytic techniques and coding standards which is mainly aimed to see the functioning and working of majorly all governmental as well as private organizations and institutions. Major giants today such as Google, Microsoft, or Mozilla have started to recommend the use of SHA-3 and stop the usage of the SHA-1 algorithm.

How dose AES work?

The AES algorithm uses a substitution-permutation, or SP network, with multiple rounds to produce cipher text. The number of rounds depends on the key size being used. A 128-bit key size dictates ten rounds, a 192-bit key size dictates 12 rounds, and a 256-bit key size has 14 rounds. Each of these rounds requires a round key, but since only one key is inputted into the algorithm, this key needs to be expanded to get keys for each round, including round 0.

- AES is a block cipher.
- The key size can be 128/192/256 bits.
- Encrypts data in blocks of 128 bits each.

That means it takes 128 bits as input and outputs 128 bits of encrypted cipher text as output. AES relies on substitution-permutation network principle which means it is

performed using a series of linked operations which involves replacing and shuffling of the input data.

Working of the cipher:

AES performs operations on bytes of data rather than in bits. Since the block size is 128 bits, the cipher processes 128 bits (or 16 bytes) of the input data at a time.

The number of rounds depends on the key length as follows:

- 128 bits key – 10 rounds
- 192 bits key – 12 rounds
- 256 bits key – 14 rounds

Creation of Round keys

A Key Schedule algorithm is used to calculate all the round keys from the key. So the initial key is used to create many different round keys which will be used in the corresponding round of the encryption.

Encryption

AES considers each block as a 16-byte (4 byte x 4 bytes = 128) grid in a column major arrangement

Each round comprises of 4 steps:

- SubBytes
- ShiftRows
- Mixcolumns
- Add Round Key

The last round doesn't have the Mixcolumns round.

The SubBytes does the substitution and ShiftRows and Mixcolumns performs the permutation in the algorithm.

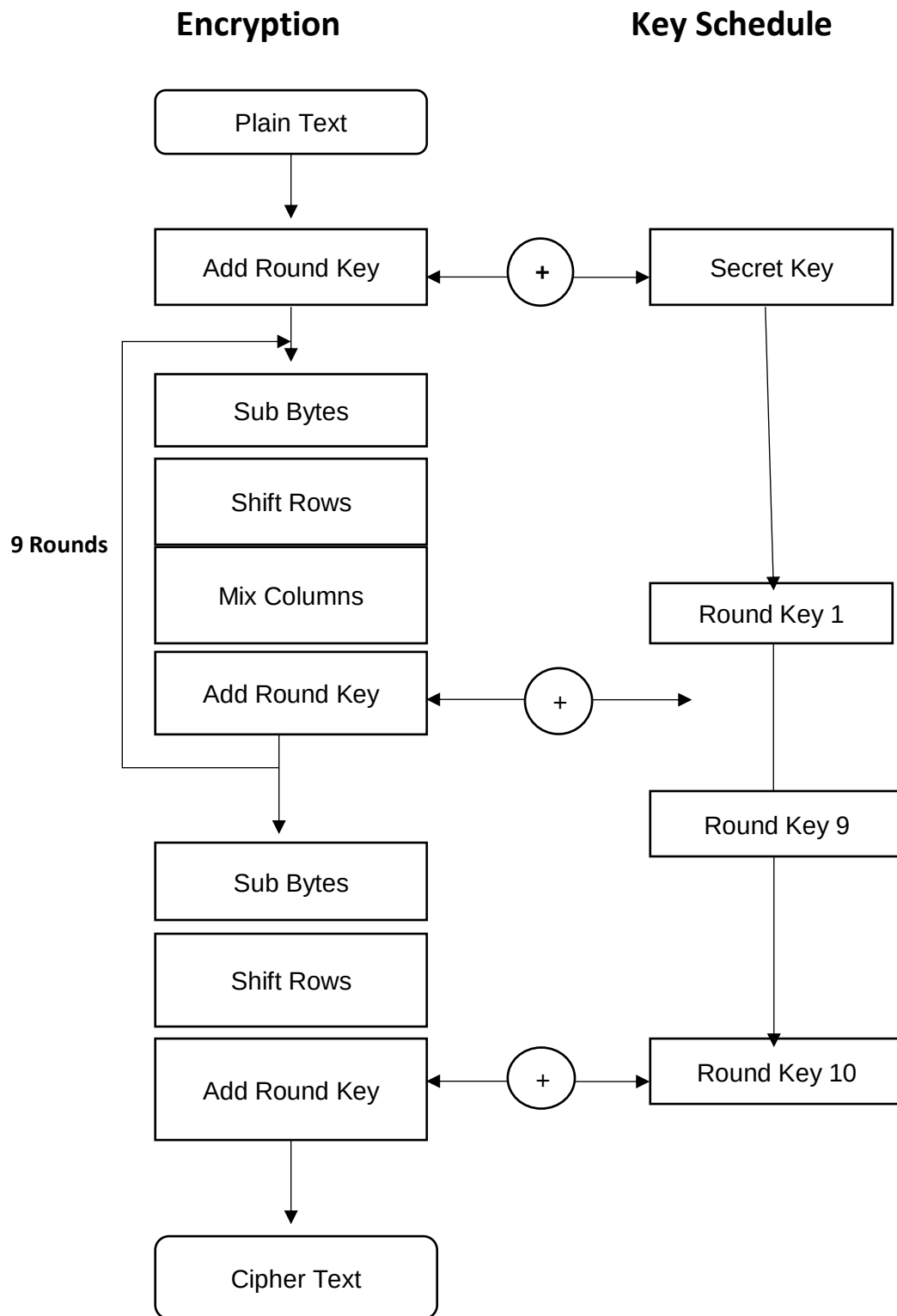


Figure 3.1 AES architecture

In this step each byte is substituted by another byte. It performed using a lookup table also called the S-box. This substitution is done in a way that a byte is never substituted by itself and also not substituted by another byte which is a compliment of the current byte. The result of this step is a 16-byte (4 x 4) matrix like before.

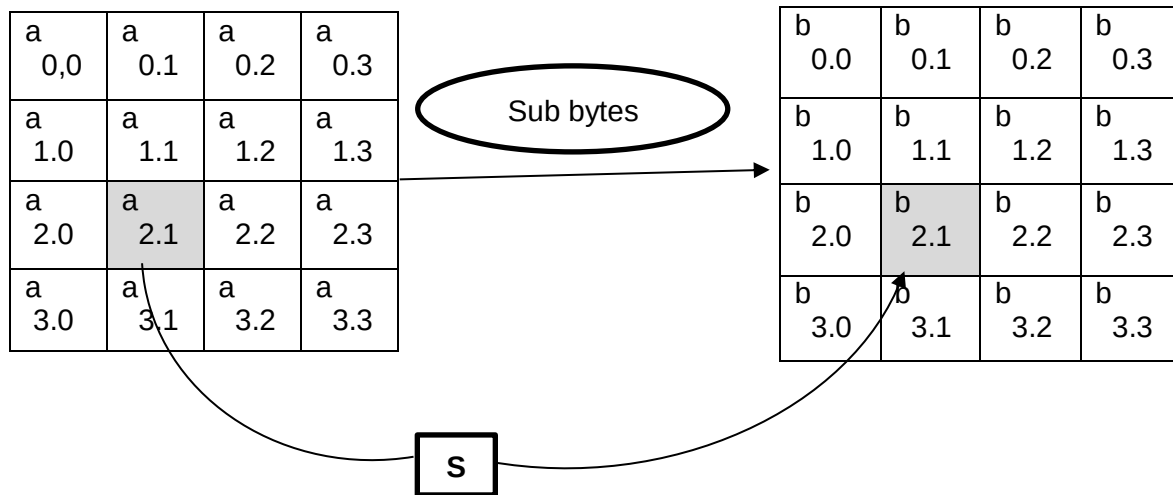


Figure 3.2 Bytes substitution

This step is just as it sounds. Each row is shifted a particular number of times.

- The first row is not shifted
- The second row is shifted once to the left.
- The third row is shifted twice to the left.
- The fourth row is shifted thrice to the left.

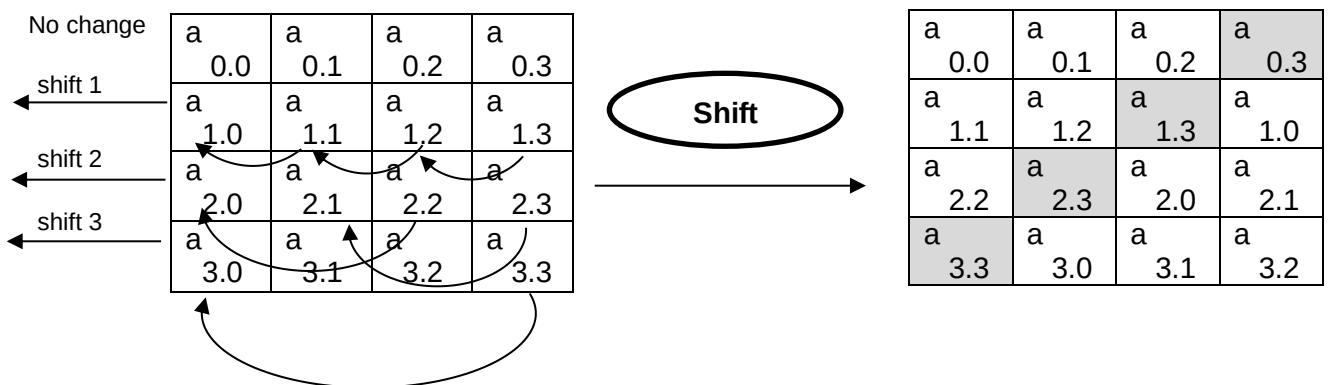


Figure 3.3 Row shifting

. This step is basically a matrix multiplication. Each column is multiplied with a specific matrix and thus the position of each byte in the column is changed as a result.

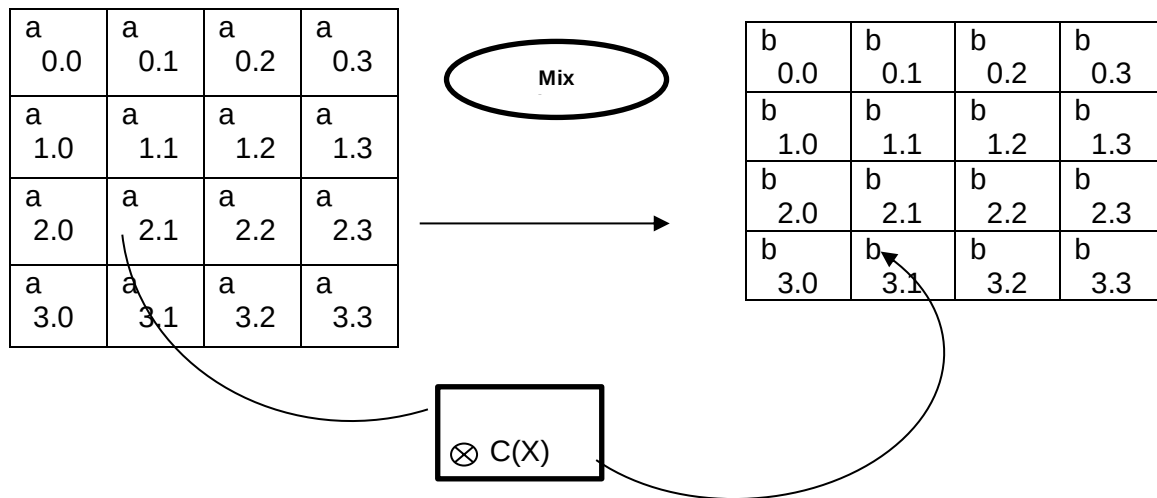


Figure 3.4 Column mixing

Now the resultant output of the previous stage is XOR-ed with the corresponding round key. Here, the 16 bytes is not considered as a grid but just as 128 bits of data.

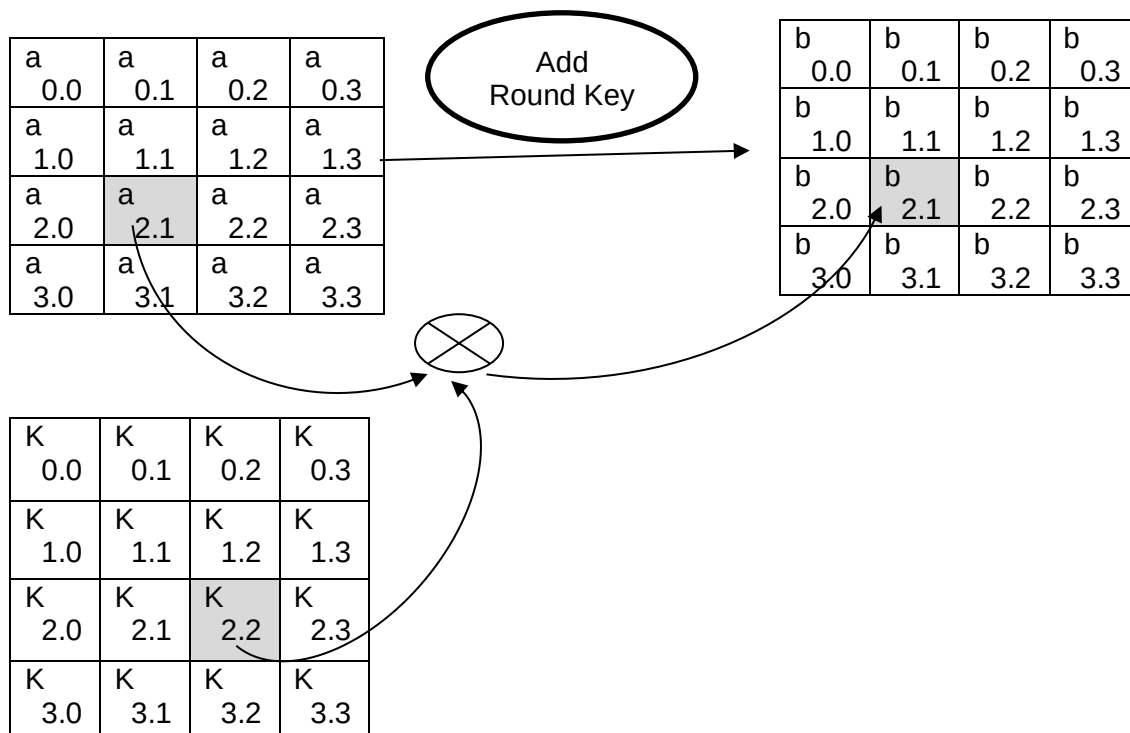


Figure 3.5 XOR operation

3.2 EXISTING SYSTEM

In existing homomorphic secret sharing and secure multi-party computing technologies have the problems of massive communication rounds and too much traffic load. It takes long time to process various hash function methods.

3.3 PROPOSED SYSTEM

The proposed smart system is used by admin or authority get involved in the production of sensitive data. The final sensitive data are produced by the final data administrator or authority person, and other modules involved in the process of data aggregation are unaware of the final data.

3.3.1 ADVANTAGES OF PROPOSED SYSTEM

- Efficiency way to transfer the data.
- Data stroage is secure.
- Huge amount of data will be store in secure (AES Algorithm).
- Using session in every login field.
- Every permission access data connecting hash (chain).
- Monitoring the each and every data transaction.
- It gives standard and valid solution to process the data with hash function.

3.4 MODULES

- Staff register
- Staff login
- Staff file view
- Staff file request
- Staff file download
- Team leader login
- Team leader file upload
- Team leader fie view
- Management login
- Management Team leader Registration

- Management generate key
- Management response

3.4.1 Staff register

The register module provides a conceptual framework for entering data on those staff in a way that: eases data entry & accuracy by matching the staff entry to the data source (usually paper files created at point of care), ties easily back to individual staff records to connect registers to staff data, and collects data elements to enable better supervision of donation programs.

3.4.2 Staff login

In this module in our project, here symbolizes a unit of work performed within a database management system (or similar system) against a database, and treated in a coherent and reliable way independent of other transactions. A transaction generally represents any change in database user will transfer the amount to provider.

3.4.3 Staff file view

In this module the staff will also view the team leader added file. And analysis the details will be responsible for your file stored in database.

3.4.4 Staff file request

In this module is used to help to the staff to Request for download file with the land longitude and the user will update the report along with their opinion and the will be stored the database.

3.4.5 Staff file download

In this module the staff download the file after management accept the request. It will be stored on local storage.

3.4.6 Team leader login

In this module in our project, here symbolizes a unit of work performed within a database management system (or similar system) against a database, and treated in a coherent and reliable way independent of other transactions. A transaction generally represents any change in database user will transfer the amount to provider.

3.4.7 Team leader file upload

The team leader can then select a file from their computer and click the **Upload** button to submit the file to the server. The Java file upload Servlet will then capture that file and persist. It will be stored in database.

3.4.8 Team leader file view

This module to help us the staff add the file to the staffs. The data directly stored in database. Then staff will view the uploaded file.

3.4.9 Management login

In this module in our project, here symbolizes a unit of work performed within a database management system (or similar system) against a database, and treated in a coherent and reliable way independent of other transactions. A transaction generally represents any change in database user will transfer the amount to provider.

3.4.10 Management Team leader registration

The register module provides a conceptual framework for entering data on those team leader in a way that: eases data entry & accuracy by matching the team leader entry to the data source (usually paper files created at point of care), ties easily back to individual team leader records to connect registers to team leader data, and collects data elements to enable better supervision of team programs.

3.4.11 Management generate key

In this module the management generate key for the staff request. Because the key for the security purpose. After get the key from management the staff will download the file with key.

3.4.12 Management response

In this module the bank will response the data file fully analyzed data in category wise view Bank will be responsible for your file stored in database.

CHAPTER 4

SYSTEM REQUIREMENT AND DESIGN

4.1 REQUIREMENT

4.1.1 HARDWARE REQUIREMENT

- A system with minimum 4GB DD RAM
- A system processor for PENTIUM IV 2.6 GHz, Intel Core 2 Duo
- A system monitor for 15" COLOR
- A system hard disk minimum 40GB

4.1.2 SOFTWARE REQUIREMENT

- IDE Eclipse installed in the system

Front End	: J2EE (JSP, SERVLETS) JAVASCRIPT
Back End	:MY SQL 5.5
Operating System	: Windows 07
IDE	: Eclipse

4.2 SOFTWARE SPECIFICATION

Java has been tested, refined, extended, and proven by a dedicated community. And numbering more than 6.5 million developers, it's the largest and most active on the planet. With its versatilty, efficiency, and portability, Java has become invaluable to developers by enabling them to:

- Write software on one platform and run it on virtually any other platform
- Create programs to run within a Web browser and Web services
- Develop server-side applications for online forums, stores, polls, HTML forms processing, and more,

JAVA FRAMEWORK

Java is considered by many as one of the most influential programming languages of the 20th century, and is widely used from application software to web applications. The Java framework is a new platform independent that simplifies application development on the Internet. Java technology's versatility, efficiency, platform portability, and security make it the ideal technology for network computing. From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

JSP

One of the main reasons why the JavaServer Pages technology has evolved into what it is today and it is still evolving is the overwhelming technical need to simplify application design by separating dynamic content from static template display data. Another benefit of utilizing JSP is that it allows to more cleanly separate the roles of web application/HTML designer from a software developer.

JAVA SERVLETS

The web server is defined as the package of a large number of programs installed on a computer connected to the Internet or intranet for downloading the requested files using File Transfer Protocol, serving e-mail and building and publishing web pages. A web server works on a client server model.

4.3 SOFTWARE TESTING

4.3.1 FEASIBILITY STUDY

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest term, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide a historical background of the business or project, description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations.

4.3.2 ECONOMICAL FEASIBILITY

The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating, etc. in order to estimate whether the new system will perform adequately or not.

4.3.3 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources.

4.3.4 OPERATIONAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

4.3.5 SYSTEM TESTING

The software, which has been developed, has to be tested to prove its validity. Testing is considered to be the least creative phase of the whole cycle of system design. In the real sense it is the phase, which helps to bring out the creativity of the other phases makes it shine.

White box testing

White-box testing, sometimes called glass-box, is a test case design method that uses the control structure of the procedural design to derive test cases. Using White Box testing methods, we can derive test cases that

- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.

Black box testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. It is an unsupervised clustering algorithm which signifies a distribution of data.

Unit testing

Unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure.

Functional testing

Functional testing is a quality assurance (QA) process and a type of black box testing that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (not like in white-box testing).

Performance testing

In general testing performed to determine how a system performs in terms of responsiveness and stability under a particular workload. It can also serve to investigate, measure, validate or verify other quality attributes of the system, such as scalability, reliability and resource usage. Performance testing is a subset of performance engineering, an emerging computer science practice which strives to build performance into the implementation, design and architecture of a system.

Integration testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with. Individual modules, which are highly prone to interface errors, should not be assumed to work instantly when put together. The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface.

Validation testing

Verification and Validation are independent procedures that are used together for checking that a product, service, or system meets requirements and specifications and that it full fills its intended purpose.

System testing

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called *assemblages*) or between any of the *assemblages* and the hardware.

Output testing

After performing the validation testing, next step is output testing of the proposed system since no system could be useful if it does not produce the required output generated or considered in to two ways. One is on screen and another is printed format. The output comes as the specified requirements by the user. Hence output testing does not result in any correction in the system.

User acceptance testing

User acceptance of a system is the factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required.

- Input screen design.
- Output screen design.
- Online message to guide user.

4.4 SYSTEM ARCHITECTURE

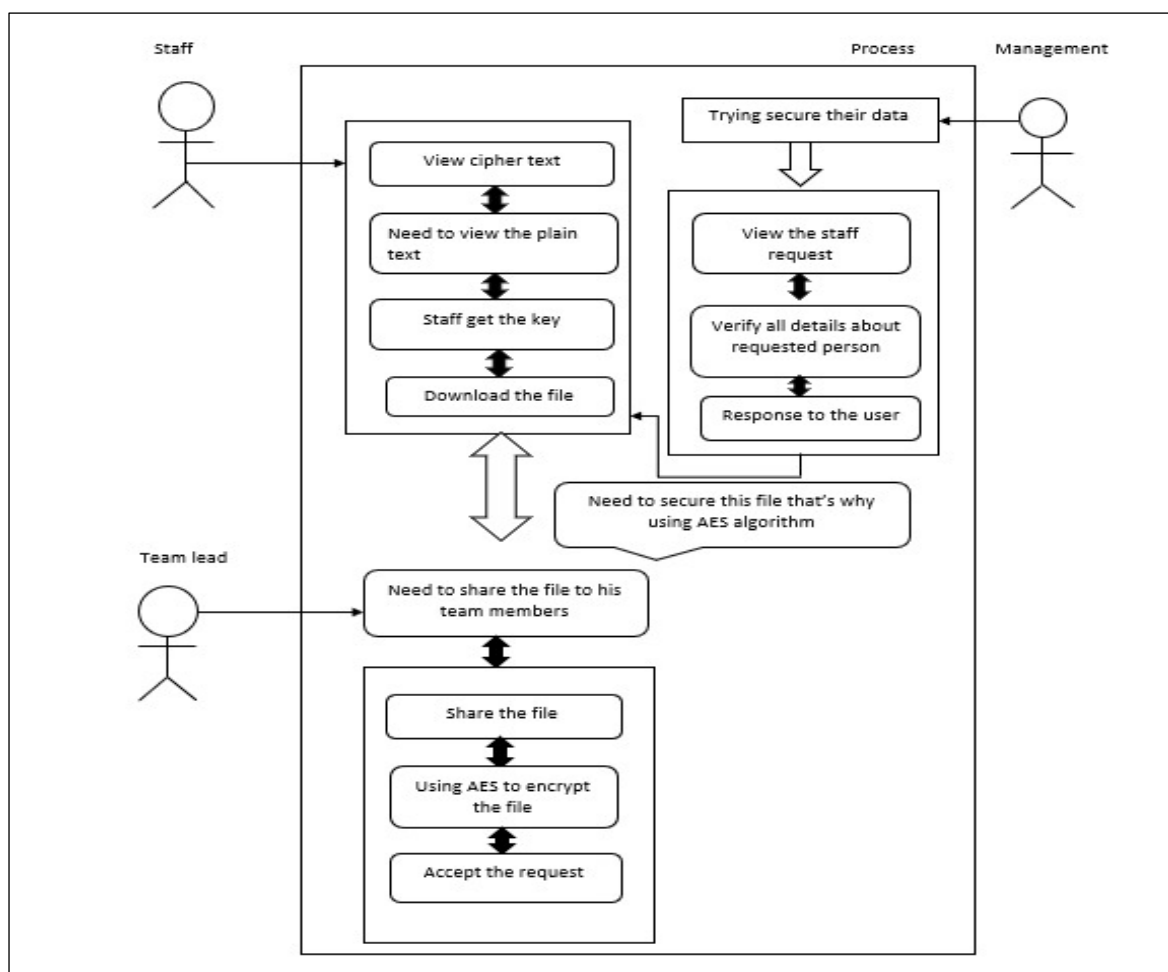


Fig 4.4 Architecture diagram

A data float chart is a graphical portrayal of the "stream" of measurements through a reality's framework. It varies from the flowchart as it shows the records float as an option of the control float of the program. An information float organization can likewise be utilized for the representation of records handling. Hence, The DFD is intended to show how a gadget is partitioned into more modest segments and to feature the coast of data between them. Below structured design planner lays out the straightforward construction of the framework, we propose a Hash code Solomon calculation, and we can position piece of measurements in area processing gadget and mist producer to protect the system. In addition, considering computational insight, this calculation can figure the appropriation share saved in cloud, haze, and close by machine, separately. Through the hypothetical wellbeing assessment and test assessment, the possibility of our plan has been approved, which is really a compelling enhancement to existing distributed storage conspire.

4.4.1 USE CASE DIAGRAM

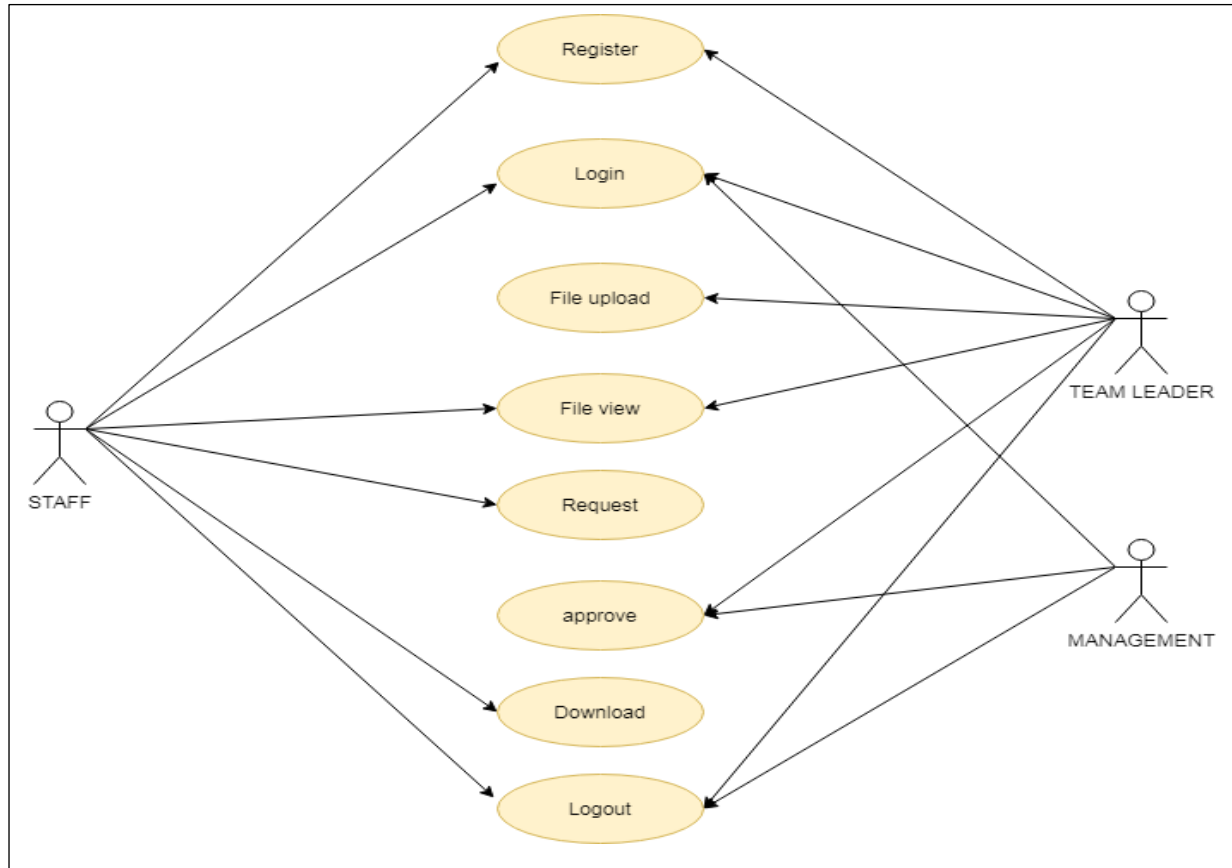


Fig 4.2 use case diagram

The use case diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. For this in our component diagram first propose a data In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 SOFTWARE SETUP

For the execution of the project we need to have a software called Eclipse with version 3.8. After the installation of the software we need to have certain library function such as java jdk, mysql.

5.2 EXPERIMENTAL RESULTS

After doing literature survey, different techniques have been found for sensitive data transaction and they use different types of data input. First, JSP and Servlets are gaining rapid acceptance as means to provide dynamic content on the Internet. With full access to the Java platform, running from the server in a secure manner, the application possibilities are almost limitless. When JSPs are used with Enterprise JavaBeans technology, e-commerce and database resources can be further enhanced to meet an enterprise's needs for web applications providing secure transactions in an open platform. J2EE technology as a whole makes it easy to develop, deploy and use web server applications instead of mingling with other technologies such as CGI and ASP. There are many tools for facilitating quick web software development and to easily convert existing server-side technologies to JSP and Servlets.

5.3 LIMITATION

1.Session

The Session Layer is the 5th layer in the Open System Interconnection (OSI) model. This layer allows users on different machines to establish active communications sessions between them. It is responsible for establishing, maintaining, synchronizing, terminating sessions between end-user applications. In Session Layer, streams of data are received and further marked, which is then resynchronized properly, so that the ends of the messages are not cut initially and further data loss is avoided. This layer basically establishes a connection between the session entities. This layer handles and manipulates data which it receives from the Session Layer as well as from the Presentation Layer.

2.Hardware requirements

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the system does and not how it should be implemented.

3.MVC

MVC Pattern stands for Model-View-Controller Pattern. This pattern is used to separate application's concerns. Model - Model represents an object or JAVA servlet carrying data. It can also have logic to update controller if its data changes. View - View represents the visualization of the data that model contains. Problem with multiple faces

CHAPTER 6

CONCLUSION

6.1 CONCLUSION

Block chain-based resource efficient and secure data sharing mechanism. Using cryptography technique to secure the data when share the team leader from staffs. Every data transaction should secure AES algorithm to encrypt the data. Whenever management response to the staff using SHA algorithm to generate the hash code connect the users into decentralized process. In every Login process using session tracking to provide the data only the particular user. Session Tracking is a way to maintain state (data) of an user. It is also known as session management in servlet. Http protocol is a stateless so we need to maintain state using session tracking techniques. Each time user requests to the server, server treats the request as the new request.

JSP and Servlets are quickly gaining popularity as tools for delivering dynamic content over the Internet. The application options are nearly unlimited when you have full access to the Java platform and can run it securely from the server. E-commerce and database resources can be further enhanced when JSPs are combined with Enterprise JavaBeans technology to suit an enterprise's needs for web applications that provide safe transactions on an open platform. Instead of mingling with other technologies like CGI and ASP, J2EE technology makes it simple to design, deploy, and consume web server applications. There are numerous tools available to help you develop web software quickly and efficiently convert existing server-side technologies to JSP and Servlets.

6.2 FUTURE ENHANCEMENT

1. Implementing a real-world anonymous database system.
2. Improving the efficiency of protocols, in terms of number of messages exchanged and in terms of their sizes, as well.
3. Implement using two are more algorithms.

REFERENCES

- [1] P. J. Taylor, T. Dargahi, A. Dehghantanha, R. M. Parizi, and K. K. R. Choo, "A systematic literature review of blockchain cyber security," *Digital Communications and Networks*, vol. 6, no. 2, pp. 147-156, 2020.
- [2] D. Liu and J. Lee, "CNN based Malicious Website Detection by Invalidating Multiple Web Spams," *IEEE Access*, vol. 8, no. 1, pp. 97258-97266, 2020.
- [3] W. Martin, V. Friedhelm, and K. Axel, "Tracing manufacturing processes using blockchain-based token compositions," *Digital Communications and Networks*, vol. 6, no. 2, pp. 167-176, 2019.
- [4] D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos and G. Das, "Everything You Wanted to Know About the Blockchain: Its Promise, Components, Processes, and Problems," *IEEE Consumer Electronics Magazine*, vol. 7, no. 4, pp. 6-14, 2018.
- [5] L. Peng, W. Feng, and Z. Yan. (2020). Privacy preservation in permissionless blockchain: A survey. *Digital Communications and Networks*. [Online]. Available: <https://doi.org/10.1016/j.dcan.2020.05.008>.
- [6] N. Kakade and U. Patel, "Secure Secret Sharing Using Homomorphic Encryption," in *Proc. 2020 11th International Conference on Computing, Communication and Networking Technologies*, 2020, pp. 1-7.
- [7] S. Sundari and M. Ananthi, "Secure multi-party computation in differential private data with Data Integrity Protection," in *Proc. 2015 International Conference on Computing and Communications Technologies*, 2015, pp. 180-184.
- [8] S. Jiao, T. Lei, Y. Gao, Z. Xie and X. Yuan, "Known-Plaintext Attack and Ciphertext-Only Attack for Encrypted Single-Pixel Imaging," *IEEE Access*, vol. 7, no.2, pp. 119557-119565, 2019.
- [9] S. Kaushik, and S. Puri, "Online transaction processing using enhanced sensitive data transfer security model," in *Proc. 2012 Students Conference on Engineering and Systems*, 2012, pp. 1-4.
- [10] W. Zheng, Z. Zheng, X. Chen, K. Dai, P. Li and R. Chen, "NutBaaS: A Blockchain-as-a-Service Platform," *IEEE Access*, vol. 7, pp. 134422-134433, 2019.
- [11] F. Casino and C. Patsakis, "An Efficient Blockchain-Based Privacy-Preserving Collaborative Filtering Architecture," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1501-1513, Nov. 2020.
- [12] D. Chkhaev, J. Hooman and P. van der Stok, "Mechanical verification of transaction processing systems," in *Proc. ICFEM 2000. Third IEEE International Conference on Formal Engineering Methods*, 2000, pp. 89-97.

APPENDIX

A. SOURCE CODE

Home page

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<style>

* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: "Abel", sans-serif;
    font-size: 10px;
    scroll-behavior: smooth;
}
.wrapper {
    width: auto;
    height: 100vh;

    background-image: linear-gradient( rgba(0, 0, 0, 0.5), rgba(0, 0, 0, 0.5)),
url("https://gdurl.com/gJJk");
    background-position: center;
    background-size: cover;
    background-repeat: no-repeat;
    backdrop-filter: opacity(80%);
}
.Container {
    width: 100%;
    height: 100%;
    display: flex;
    justify-content: center;
    align-items: center;
}
.nav {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 80px;
    border-bottom: 1px solid rgba(255, 255, 255, 0.521);
    display: flex;
    justify-content: space-between;
    align-items: center;
    padding: 0 50px;
```

```

}
.logo {
  font-family: "Abel", sans-serif;
  font-size: 2.5rem;
  font-weight: 600;
  letter-spacing: 0.7rem;
  color: white;
  margin: 4%;
}
.menu {
  display: inline-block;
  line-height: 80px;
}
.menu ul {
  list-style: none;
  /* display: flex;
  flex-direction: row;
  justify-content: center;
  align-items: center; */
}
.menu ul li {
  display: inline-block;
}
.menu ul li a {
  text-decoration: none;
  font-family: "Raleway", sans-serif;
  font-size: 1.2rem;
  font-weight: 600;
  letter-spacing: 0.1rem;
  color: white;
  border: 1px solid transparent;
  border-radius: 4px;
  padding: 10px 15px;
  margin: 0 5px;
  transition: 0.5s ease;
}
.menu ul li a:hover {
  border-color: white;
}
.menu ul li:nth-child(5) a {
  color: #fff200;
  border: 1px solid #fff200;
}
.menu ul li:nth-child(5) a:hover {
  color: black;
  background-color: #fff200;
}
.header {
  text-align: center;
}
.header h1 {
  font-family: "Raleway", sans-serif;
  font-size: 4rem;
  font-weight: 600;
  letter-spacing: 0.2rem;
  color: white;
}

```

```

padding: 45% 20px 8px;
}
.header p {
font-family: "Raleway", sans-serif;
font-size: 1.5rem;
font-weight: 600;
letter-spacing: 0.2rem;
color: white;
padding: 10px 15px;
}
button {
font-size: 1.5rem;
font-weight: 600;
letter-spacing: 0.15rem;
color: black;
background-color: #fff200;
padding: 20px 30px;
margin: 50px 5px 0;
border: none;
cursor: pointer;
}

```

```
</style>
```

```
<body>
```

```

<div class="wrapper">
  <div class="Container">
    <div class="nav">
      <div class="logo">

      </div>
      <div class="menu">
        <ul class="navMenu">
          <li><a href="#">Home</a></li>
          <li><a href="Stafflogin.jsp">Staff</a></li>
          <li><a href="tlog.jsp">Team Leader</a></li>
          <li><a href="managementlog.jsp">Management</a></li>

        </ul>
      </div>
    </div>
    <div class="header">
      <h1>Welcome</h1>

    </div>
  </div>
</div>
</body>
</html>

```

Staff download page

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ page import="java.io.FileInputStream" %>
<%@ page import="java.io.IOException" %>
<%@ page import="java.io.PrintWriter" %>
<%@ page import="javax.swing.JDialog" %>
<%@ page import="javax.swing.JOptionPane" %>
<%@ page import="java.sql.ResultSet"%>
<%@ page import="java.sql.PreparedStatement" %>
<%@ page import="Dbconn.dbconn" %>
<%@ page import="java.sql.*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<% String status="Downloaded";
String id=request.getParameter("id");%>
<%
String key1=request.getParameter("fkeyl");

System.out.println("download00000000000000000000====="+key1);

String key2=request.getParameter("headkey");

System.out.println("download00000000000000000000====="+key2);
JOptionPane joptionpane =new JOptionPane("CLICK OK");

JDialog jdialog= joptionpane.createDialog("Alert");

jdialog.setAlwaysOnTop(true);

jdialog.show();

String newkey =JOptionPane.showInputDialog("Enter Your filekey");

Connection con;
con=dbconn.create();
if(newkey.equals(key1))

{
    String newkey1 =JOptionPane.showInputDialog("Enter Your filekey");
    if(newkey1.equals(key2))
try
{
String realpath = getServletContext().getRealPath("/");
String filepath = realpath.substring(0, realpath.indexOf("."))
```

```

+ "ITJBC04\\WebContent\\Locals\\";

String filename=request.getParameter("fname1");
System.out.println(filename);

/*      String filepath="E:\\2016 Web\\ITJDM15\\WebContent\\compress\\"+compress+".zip";*/

response.setContentType("APPLICATION/OCTET-STREAM");
response.setHeader("Content-Disposition","attachment; filename=\"\" + filename + "\"");

FileInputStream fileInputStream=new FileInputStream(filepath+filename);

int i;
while ((i=fileInputStream.read()) != -1) {
    out.write(i);
}

fileInputStream.close();
}

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

}

else
{
System.out.println("failed");

        JOptionPane.showMessageDialog(null, "Sorry Your key is wrong");

        response.sendRedirect("Error.jsp");
}

%>

</body>
</html>

```

B. SCEERN SHOTS

