

A
Project-I Report
on
**IMPLEMENTATION OF ENHANCED
ASSOCIATION RULE MINING ON
HORIZONTAL DISTRIBUTED
DATABASES**

Submitted in Partial Fulfillment of
the Requirements for the Degree
of

Bachelor of Engineering

in

Computer Engineering

to

North Maharashtra University, Jalgaon

Submitted by

**Puja Anil Naval
Ashwini Sudhir Patil
Pooja Vasant Sapkale
Punam Ashok Patil**

Under the Guidance of

Mr. Sandip S. Patil



DEPARTMENT OF COMPUTER ENGINEERING
SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY,
BAMBHORI, JALGAON - 425 001 (MS)
2016 - 2017

**SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY,
BAMBHORI, JALGAON - 425 001 (MS)
DEPARTMENT OF COMPUTER ENGINEERING**

CERTIFICATE

This is to certify that the Project-I entitled *Implementation of Enhanced Association Rule Mining on Horizontal Distributed Databases*, submitted by

**Puja Anil Naval
Ashwini Sudhir Patil
Pooja Vasant Sapkale
Punam Ashok Patil**

in partial fulfillment of the degree of *Bachelor of Engineering in Computer Engineering* has been satisfactorily carried out under my guidance as per the requirement of North Maharashtra University, Jalgaon.

Date: October 10, 2016

Place: Jalgaon

Mr. Sandip S. Patil
Guide

Prof. Dr. Girish K. Patnaik
Head

Prof. Dr. K. S. Wani
Principal

Acknowledgements

The project report on Implementation of Enhanced Association Rule Mining on Horizontal Distributed Databases is standing on the contribution of many people. First of all a hearty great full thanks to Principal, Prof. Dr. K. S. Wani & college authorities for providing facilities and excellent infrastructure & a very great full thanks to Prof. Dr. Girish K. Patnaik Head of Department, giving necessary guidance concerning project and help whenever required. The Coordination of and guidance of our guide Mr. Sandip S. Patil for guidelines and showing the right path to successfully reach to the destination. Also, special thanks to all the faculty and staff members for extending their helping hand directly or indirectly. Last but not the least, hearty express gratitude towards families, colleagues and friends for their kind co-operation and encouragement which help in the completion of the Project-I.

Puja Anil Naval

Ashwini Sudhir Patil

Pooja Vasant Sapkale

Punam Ashok Patil

Contents

Acknowledgements	ii
Abstract	1
1 Introduction	2
1.1 Background	2
1.2 Motivation	3
1.3 Problem Definition	3
1.4 Scope	4
1.5 Objective	4
1.6 Organization of Report	4
1.7 Summary	4
2 System Analysis	5
2.1 Literature Survey	5
2.2 Proposed System	7
2.3 Feasibility Study	7
2.3.1 Technical Feasibility	7
2.3.2 Operational Feasibility	7
2.3.3 Economical Feasibility	8
2.4 Risk Analysis	8
2.4.1 Risk	8
2.4.2 Need of Risk Analysis	8
2.4.3 Software Risk	8
2.4.4 Project Risk	9
2.4.5 Technical Risks	9
2.4.6 Business Risk	9
2.5 Project Scheduling	10
2.6 Effort Allocation	10
2.7 Summary	11

3	Requirement Analysis	12
3.1	Hardware Requirements	12
3.2	Software Requirements	12
3.3	Functional Requirements	13
3.4	Non-Functional Requirements	14
3.5	Summary	14
4	System Design	15
4.1	System Architecture	15
4.2	E-R diagram	16
4.3	Data Flow Diagram	16
4.4	UML Diagrams	18
4.4.1	Use Case Diagram	18
4.4.2	Class Diagram	19
4.4.3	Sequence Diagram	20
4.4.4	Activity Diagram	21
4.4.5	Component Diagram	22
4.4.6	Deployment Diagram	23
4.4.7	State Diagram	24
4.4.8	Collaboration Diagram	25
4.5	Summary	25
	Index	26
	Bibliography	26

List of Figures

2.1	Gantt Chart	10
2.2	Chart of Effort Allocation	11
4.1	System architecture for Horizontal Database With Enhances Secure Mining .	15
4.2	E-R Diagram for Proposed System.	16
4.3	Data Flow Diagram Level 0	17
4.4	Data Flow Diagram Level 1	17
4.5	Use Case Diagram	18
4.6	Class Diagram	19
4.7	Sequence Diagram	20
4.8	Activity Diagram	21
4.9	Component Diagram	22
4.10	Deployment Diagram	23
4.11	State Chart Diagram	24
4.12	Collaboration Diagram	25

Abstract

Data mining is the most fast growing area today which is used to extract important knowledge from large data collections but often these collections are divided among several parties. Privacy liability may prevent the parties from directly sharing the data and some types of information about the data. In this project propose a protocol for secure mining of association rules in horizontally distributed databases. The current integral protocol is that of Kantarcioglu and Clifton well known as K&C protocol. This protocol is based on an unsecured distributed version of the Apriori algorithm named as Fast Distributed Mining algorithm of Cheung et al. The main ingredients in our protocol are two novel secure multi-party algorithms one that computes the union of private subsets that each of the interacting players hold and another that tests the whether an element held by one player is included in a subset held by another. This protocol offers enhanced privacy with respect to the earlier protocols. In addition, it is not complicated and is importantly more effectual in terms of communication cost, communication rounds and computational cost.

Chapter 1

Introduction

Data mining aim to extract useful information from huge amount of data. Data mining is an emerging field which connects different major areas like databases, artificial intelligence and statistics. Data mining is a powerful tool so it can investigate and extract previously unknown patterns from large amounts of data.

Chapter is of 7 Sections, Section 1.1 describes Background of the project, Motivations behind the project describes in Section 1.2, Section 1.3 describes Problem Definition of the project, Scope of the project is describes in Section 1.4, Section 1.5 describes Objectives of the project, Organization of the whole project is describes in Section 1.6 and Section 1.7 gives Summary.

1.1 Background

Association Rule mining is one of the most important data mining tools used in many real life applications. It is used to reveal unexpected relationships in the data. In this paper, discuss the problem of computing association rules within a horizontally partitioned database, assume homogeneous databases. All sites have the same schema, but each site has information on different entities. The goal is to produce association rules it hold globally, while limiting the information shared about each site to preserve the privacy of data in each site.

Association Rule Mining:

Association rule mining is used to find interesting associations and/or correlation relationships among large sets of data items. Association rules show attributes value conditions it occur frequently together in a given data set.

Apriori Algorithm:

The Apriori Algorithm proposed to finds frequent items in a large amount of database. Apriori is an influential algorithm in market basket analysis for mining frequent item sets for Boolean association rules. The name of Apriori is based on the fact. The algorithm

uses a prior knowledge of frequent itemset properties. Apriori employs an iterative approach known as a level wise search, where k item sets are used to explore $(k+1)$ item sets. Apriori algorithm is an influential algorithm for mining frequent item sets for Boolean association rules. This algorithm contains a number of passes over the database. During pass k , the algorithm finds the set of frequent item sets L_k of length k that satisfy the minimum support requirement. Apriori is designed to operate on databases containing transactions. The purpose of the Apriori Algorithm is to find associations between different sets of data. It is sometimes referred to as “Market Basket Analysis”. Each set of data has a number of items and is called a transaction. The output of Apriori is sets of frequent item sets tell us how often items are contained in sets of data.

1.2 Motivation

Data mining is the most fast growing area today which is used to extract important knowledge from large data collections but often these collections are divided among several parties. The existing problem is important to solve because the excess leakage of information using the protocol is already available. In proposed method privacy of user increase so, if third party person try to access private data so it found in encrypted format means third party person cannot read their data. Only authorized person access their own private information.

1.3 Problem Definition

Proposed protocol is based on the algorithm, Fast Distributed Mining of association rules which is an unsecured distributed version of the Apriori algorithm used to generate a small number of candidate sets and the number of messages to be passed at mining association rules. The system includes a novel various protocol for providing secure computation with personal subsets in distributed information and it improves the potency and security of information mining. The planned and designed protocol provides full computing, parameterized computing and customized user computing, its tendency to decision user threshold functions, within which the two extreme cases correspond to the issues of computing the union and intersection of personal subsets. This mechanism provides associate degree extension to Apriori rule with quick distributed and Secure Multiparty computations. The proposed system uses two algorithms, specifically Apriori and S-FDM for locating frequent item sets from horizontally distributed databases. Association Rules square measure generated from the frequent things sets and classified whose confidence is larger than the minimum threshold confidence referred to as sturdy Association Rules. The sturdy associations rules square measure classified during this manner square measure presented the user. While extracting

data from distributed database system more number of irrelevant data occur. Irrelevant data is avoided by using the Apriori algorithm.

1.4 Scope

The proposed system is useful in medical system, banking Sector for increasing privacy of the data and secure their private information here. If third party person try to access private data then person becomes fails to access data and private data becomes private even after third party attack. If owner and user want to access the records in database so by using searching method it can easily access particular test record in less time and securely.

1.5 Objective

The proposed a protocol for secure mining of association rules in horizontally distributed databases it improves significantly upon the current leading protocol in terms of privacy and efficiency. The main ingredient in proposed protocol is a novel secure multiparty protocol for compute the union or intersection of private subsets and each of the interacting players holds.

1.6 Organization of Report

Chapter 2, describes the overall system analysis. It includes literature survey, proposed system, feasibility study, risk analysis, project scheduling and effort allocation. Chapter 3, gives system requirement specification such as hardware requirements, software requirements, functional requirements, non-functional requirements and other requirements and constraints. Chapter 4, shows system design by drawing system architecture, E-R diagram, database design, Data flow diagram, user interface design and all important UML diagrams.

1.7 Summary

In this chapter, basic introduction towards the system including Problem Statement, Problem Definition, Objective and Scope is described. Finally, Summary is presented in Last Section. In the next chapter, describes System Analysis of proposed system.

Chapter 2

System Analysis

The main purpose behind the development of proposed system is to overcome the drawbacks of existing system.

Section 2.1 describes Literature Survey, Proposed System describes in Section 2.2. In Section 2.3 describes Feasibility Study in which Economical Feasibility, Operational Feasibility and Technical Feasibility. Risk Analysis describes in Section 2.4. In Section 2.5 describes Project Scheduling. Effort Allocation describes in Section 2.6, Section 2.7 gives Summary.

2.1 Literature Survey

- Keying hash functions for message authentication

Bellare et al. in [3] presented the use of cryptographic hash functions like MD5 or SHA-1 for message authentication has become a standard approach in many applications, particularly Internet security protocols. Though very easy to implement, these mechanisms are usually based on ad hoc techniques and it lack a sound security analysis. In this paper present new, simple, and practical constructions of message authentication schemes based on a cryptographic hash function. Schemes, NMAC and HMAC, are proven to be secure as long as the underlying hash function has some reasonable cryptographic strengths. Moreover show, in a quantitative way, the schemes retain almost all the security of the underlying hash function. The performance of schemes is essentially of the underlying hash function. Moreover use the hash function as a black box, so that widely available library code or hardware can be used to implement them in a simple way and replaceability of the underlying hash function is easily supported.

- FairplayMP-A system for secure multi-party computation

Ben-David et al. in [4] presented “Fairplay Multi-Party”, a system for secure multi-party computation. Secure computation is one of the great achievements of modern cryptography, enabling a set of untrusting parties to compute any function of their

private inputs while revealing nothing but the result of the function. FairplayMP lets the parties run a joint computation, emulates a trusted party which receives the inputs from the parties, computes the function, and privately informs the parties of their outputs. FairplayMP operates by receiving a high-level language description of a function and a configuration file describing the participating parties. The system compiles the function into a description as a Boolean circuit, and perform a distributed evaluation of the circuit while revealing nothing else. FairplayMP supplements the Fairplay system, which supported secure computation between two parties. The underlying protocol of FairplayMP is the Beaver-Micali-Rogaway protocol which runs in a constant number of communication rounds. It modified the BMR protocol in a novel way and considerably improved its performance by using the Ben-Or-Goldwasser-Wigderson protocol for the purpose of constructing gate tables.

- Privacy-preserving graph algorithms in the semi-honest model

Brickell et al. in [5] presented scenarios in which two parties, each in possession of a graph, wish to compute some algorithm on their joint graph in a privacy-preserving manner i.e. without leaking any information about their inputs except and revealed by the algorithms output. Working in the standard secure multi-party computation paradigm, proposed system present new algorithms for privacy-preserving computation of all pairs shortest distance and single source shortest distance, as well as two new algorithms for privacy-preserving set union. Their algorithms are significantly more efficient than generic constructions. As in previous work on privacy-preserving data mining, it's prove the algorithms are secure provided the participants are honest, but curious

- Secret sharing homomorphisms: Keeping shares of a secret secret

Benaloh et al. in [6] presented independently proposed schemes by which a secret can be divided into many shares which can be distributed to mutually suspicious agents. This paper describes a homomorphism property attained by these and several other secret sharing schemes which allows multiple secrets to be combined by direct computation on shares. This property reduces the need for trust among agents and allows secret sharing to be applied to many new problems. One application described gives a method of verifiable secret sharing which is much simpler and more efficient than previous schemes. A second application is described which gives a fault-tolerant method of holding verifiable secret-ballot elections.

2.2 Proposed System

The proposed protocol improves in terms of simplicity and efficiency as well as privacy. The proposed protocol does not depend on commutative encryption and oblivious transfer. Proposed system computes a parameterized family of functions, which call doorstep functions, in which the two great cases communicate to the problems of computing the union and intersection of private subsets. The excess information that proposed protocol may leak is less sensitive than the excess information leaked by the protocol. Proposed two novel secure multiparty algorithms: one of them computes the union of private subsets that each of the interacting players hold and tests the inclusion of an element held by one player in a subset held by another. The problem of secure multiparty computation it solve here is the set inclusion problem.

2.3 Feasibility Study

The feasibility analysis shows the developers all the aspects of the project and know whether the project is practically possible to develop worth limited resources and time. There are few types of feasibility are exist so developer should take care of these feasibility or developer must aware about these feasibility.

- Economical Feasibility
- Operational Feasibility
- Technical Feasibility

2.3.1 Technical Feasibility

This study is carried out to check the technical feasibility ie. the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. If want to run proposed system then it is necessary JDK, Netbeans and Database is available on the system. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. So proposed system is technically feasible.

2.3.2 Operational Feasibility

Operational feasibility is beneficial if it can be turned into information system meet the organization operating requirement. The proposed system provide complete security to data owner. The system is user friendly in which only one machine require.

2.3.3 Economical Feasibility

The project involves the utilization of software which is freely available. Such as JDK 7 for coding purpose, MySQL Database is also required for storing the data. This all software are used freely and easily available. So proposed system is Economically feasible.

2.4 Risk Analysis

Risk analysis can be "broadly defined to include risk assessment, risk characterization, risk communication, risk management, and policy relating to risk, in the context of risks of concern to individuals, to public- and private-sector organizations, and to society at a local, regional, national, or global level."

2.4.1 Risk

Risk analysis and management are a series of steps, help a software team to understand and manage uncertainty. Many problem can plague a software project. A risk is a potential problem it might happen, it might not But regardless of the outcome, its a really good idea to identify it, assess its probability of occurrence, estimate its impact and establish a contingency plan should the problem actually occur. Everyone involved in the software process managers, software engineer and customer participate in the risk analysis and management. There will be following possible risk which can be related to the project:

- Some time system may fail when there is problem occur in hardware devices.
- When the computer system get hanged then the software become terminate.

2.4.2 Need of Risk Analysis

Think about the Boy Scout motto: Be prepared. Software is a difficult undertaking. Lots of things can go wrong, and frankly, many often do. Its for this reason to being prepare understanding the risks and taking proactive measures to avoid or manage them as well as key element of good software project management.

2.4.3 Software Risk

Although there has been considerable debate about the proper definition for software risk, there is general agreement the risk always involves two characteristics

- Uncertainty

The risk may or may not happen; ie. there are no 100% probable risks.

- Loss

If the risk becomes a reality, unwanted consequences or losses will occur.

When risks are analyzed, it is important to quantify the level of uncertainty and the degree of loss associated with each risk. To accomplish this, different categories of risks are considered.

2.4.4 Project Risk

Threaten the project plan ie. if project risks become real, and project schedule will slip and costs increase. Project risks identify potential budgetary, schedule, personnel (staffing and organization), resource, customer, and requirements problems and their impact on a software project. In project, project risk occurs if the requirement of technical member means technical team is unavailable according to project plan and estimation and if project is not completed within time in this situation project risk can occurs.

2.4.5 Technical Risks

Threaten the quality and timeliness of the software to be produced. If a technical risk becomes a reality, implementation may become difficult or impossible. Technical risks identify potential design, implementation, interface, verification, and maintenance problems. In addition, specification ambiguity, technical uncertainty, technical obsolescence, and “leading-edge” technology are also risk factors. Technical risks occur because the problem is harder to solve. In proposed system if any module of the website is not worked properly according to expectation then technical risk may occur.

2.4.6 Business Risk

Threaten the viability of the software to be built. Business risks often jeopardize the project or the product. Candidates for the top five business risks are:

1. Building a excellent product or system it no one really wants (market risk).
2. Building a product it no longer fits into the overall business strategy for the company (strategic risk).
3. Building a product, the sales force doesn't understand how to sell.
4. Losing the support of senior management due to a change in focus or a change in people (management risk).
5. Losing budgetary or personnel commitment (budget risks). It is extremely important to note the simple categorization won't always work. Some risks are simply unpredictable in advance.

2.5 Project Scheduling

Software project scheduling is an activity to distributes estimated effort across the planned project duration by allocating the effort to specific software engineering tasks. It is important to note, however, the schedule evolves over time. During early stages of project planning, a macroscopic schedule is developed. This type of schedule identifies all major software engineering activities and the product functions to which are applied. As the project gets under way, each entry on the macroscopic schedule is refined into a detailed schedule. Project scheduling can be done by Gantt chart as shown in Figure 2.1










PROJECT SCHEDULING			
Phases \ Duration	July	August	September
Introduction of Project	 25%	 100%	
Problem Defination		 35%	 100%
System Analysis		 35%	 100%
System requirement Specification		 50%	 100%
System Design			 100%

Figure 2.1: Gantt Chart

2.6 Effort Allocation

Software engineer or a team of engineers must incorporate a development strategy, encompasses the process, methods and tools layers described. This strategy is often referred to as a process model or a software engineering paradigm. A process model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used and the controls and deliverables are required. There are many process models in the software engineering, but it choose Water Fall Model because the project is totally dependent on previous modules. Another Reason for choosing this model is to provide better user satisfaction. The chart of effort allocation shown below in Figure 2.2

Implementation of Enhanced Association Rule Mining on Horizontal Distributed	Puja Naval	Pooja Sapkale	Punam Patil	Ashwini Patil
Introduction of Project	✓	✓		
System Analysis			✓	✓
System Requirement Specification	✓		✓	
System Design		✓		✓

Figure 2.2: Chart of Effort Allocation

Allocation of efforts to all project partners- Project means team work; Project is developed by combination of effort of team. So whole project is divided into modules and number of modules is allotted to team members. After completion of each module, it will be link from one module to another module to form a complete project.

This effort allocation should be used as a guideline only. The characteristics of each project must dictate the distribution of effort. Work expended on project planning rarely accounts for more than 2-3 percent of effort, unless the plan commits an organization to large expenditures with high risk. Requirements analysis may comprise 10 to 25 percent of project effort. Effort expended on analysis or prototyping should increase in direct proportion with project size and complexity. A range of 20 to 25 percent of effort is normally applied to software design. Time expended for design review and subsequent iteration must also be considered.

2.7 Summary

In this chapter, Section 2.1 is described Literature Survey, Proposed System is described in Section 2.2, Section 2.3 are described Feasibility Study such as Economical Feasibility, Operational Feasibility and Technical Feasibility. Risk Analysis are described in Section 2.4, Section 2.5 is described Project scheduling, Effort Allocation is described in Section 2.6. In the last Section 2.7 Summary is presented. In the next chapter, describes about Requirement Analysis of proposed system.

Chapter 3

Requirement Analysis

This chapter focuses on the various requirements of the system such as, Hardware Requirements, Software Requirements, Functional Requirements and Non-Functional Requirements of the proposed system.

Section 3.1 describes Hardware Requirements, Software Requirements describes in Section 3.2, Section 3.3 describes Functional Requirements, Non-Functional Requirements describes in Section 3.4, Section 3.5 gives Summary.

3.1 Hardware Requirements

Minimum Hardware Requirements includes:

- Hard Disk 40 GB.
- RAM 512 Mb.
- Floppy Drive
- Monitor
- Mouse
- Keyboard

3.2 Software Requirements

For these project, softwares are required to be installed on computer system. Proposed system used Java language for coding, need java runtime environment for execution. Also for storing data it require database server, LAMP/WAMP is used in project.

Software are required for project are as follows:

- JDK 7

- NetBeans 7.4
- Ubuntu/Windows
- LAMP/WAMP 2.0

3.3 Functional Requirements

- Privacy Preserving Data Mining:

One, in which the data owner and the data miner are two different entities, and another, in which the data is distributed among several parties who aim to jointly perform data mining on the unified corpus of data and hold. In the first setting, the goal is to protect the data records from the data miner. Hence, the data owner aims at anonymizing the data prior to its release. The main approach in this context is to apply data perturbation. Computation and communication costs versus the number of transactions N the perturbed data can be used to infer general trends in the data, without revealing original record information. In the second setting, the goal is to perform data mining while protecting the data records of each of the data owners from the other data owners. This is a problem of secure multiparty computation. The usual approach here is cryptographic rather than probabilistic.

- Distributed Computation:

Here compared the performance of two secure implementations of the FDM algorithm. In the first implementation (denoted FDM-KC), executed the unification step using protocol UNIFI-KC, where the commutative cipher was 1024-bit RSA in the second implementation (denoted FDM) proposed system used Protocol UNIFI, where the keyed-hash function was HMAC. The two implementations with respect to three measures:

1. Total computation time of the complete protocols (FDMKC and FDM) over all players. It includes the Apriori computation time and the time to identify the globally s -frequent item sets, as described in later.
2. Total computation time of the unification protocols only (UNIFI-KC and UNIFI) over all players.
3. Total message size. Ran three experiment sets, where each set tested the dependence of the above measures on a different parameter:
 N the number of transactions in the unified database.

- Frequent Itemsets:

The solution was proposed by Kantarcioglu and Clifton. It considered two possible settings. If the required output includes all globally s -frequent item sets, as well as the sizes of their supports, then the values of x can be revealed for all. In such a case, those values may be computed using a secure summation protocol, where the private addend of P_m is $\text{supp}_m(x) \cdot sNm$. The more interesting setting, however, is the one where the support sizes are not part of the required output.

- Association Rules:

Once the set F_s of all s -frequent item sets is found, it may proceed to look for all (s, c) -association rules (rules with support at least sN and confidence at least c). In order to derive from F_s all (s, c) association rules in an efficient manner it rely upon the straightforward lemma.

- Eclat algorithm:

Eclat algorithm proposed by ZAKI in 2000, is based on the breadth-first search strategy, which adopts the technologies of vertical data format, lattice theory, equivalence classes, intersection and so on. The main steps of Eclat are listed as follows: scan the database to get all frequent 1-item sets, generate candidate 2-item sets from frequent item sets, then get all frequent 2-item sets by clipping non frequent candidate item sets; generate candidate 3-item sets from frequent 2-item sets and then get all frequent item sets by clipping non-frequent candidate item sets; repeat the above steps, until no candidate item set can be generated. Same as Apriori, Eclat algorithm also adopts the join operation to generate candidate $(K+1)$ item set by taking the union of two k -item set.

3.4 Non-Functional Requirements

In Non-functional requirements of project implements the functions which does not effect on function and behavior of project for desired goal and objective of project. Non-functional requirement just provides user friendliness and notifications are not most necessary for project.

3.5 Summary

In this chapter, Section 3.1 are described Hardware Requirements, Software Requirements are described in Section 3.2, Section 3.3 are described Functional Requirements, Non-Functional Requirements are described in Section 3.4. In last Section 3.5 Summary is presented. In the next chapter, describes about overall System Design of of proposed system.

Chapter 4

System Design

This chapter describes System Architecture, E-R Diagram, Data Flow Diagram and UML Diagram of the proposed system.

Section 4.1 describes the System Architecture, E-R Diagram describes in Section 4.2. In Section 4.3 describes Data Flow Diagrams, UML Diagrams describes in Section 4.4. The Section 4.5 gives Summary.

4.1 System Architecture

System architecture is the conceptual model defines the structure, behaviour and more views of a system. System Architecture for horizontal database with enhances secure mining is shown in Figure 4.1.

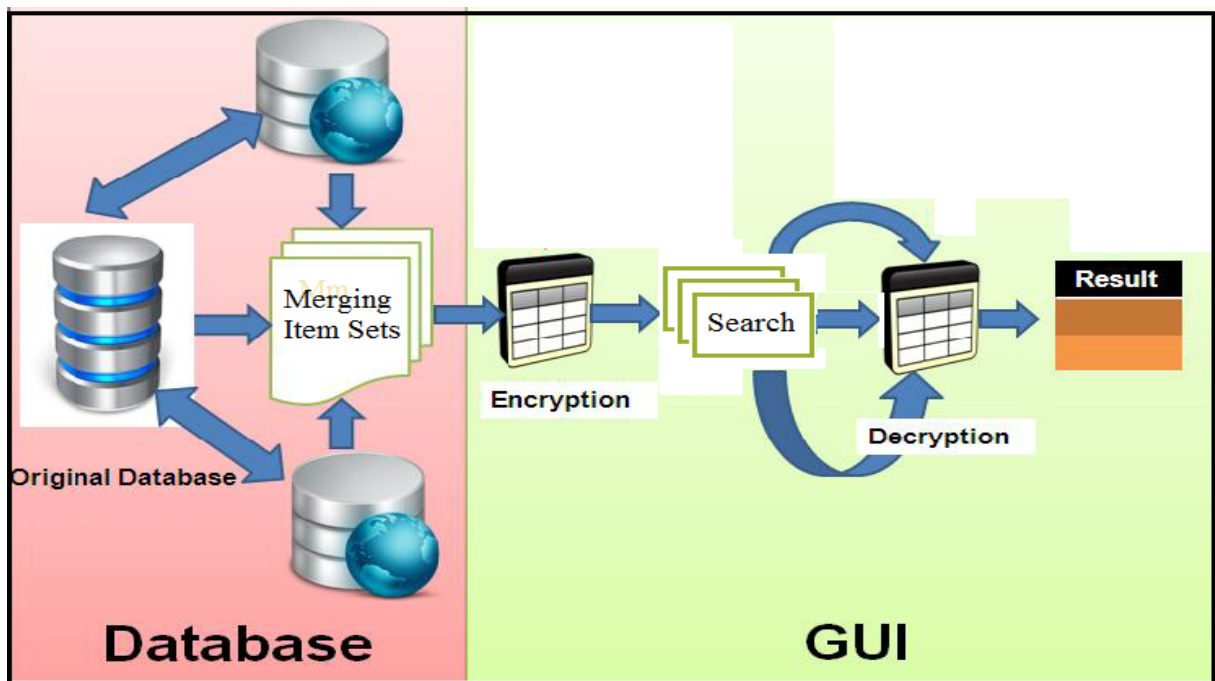


Figure 4.1: System architecture for Horizontal Database With Enhances Secure Mining

4.2 E-R diagram

The entity relationship data model is based on a perception of a real world consist of a collection of basic objects called entities and relation among these objects. E-R diagram is shown in following Figure 4.2.

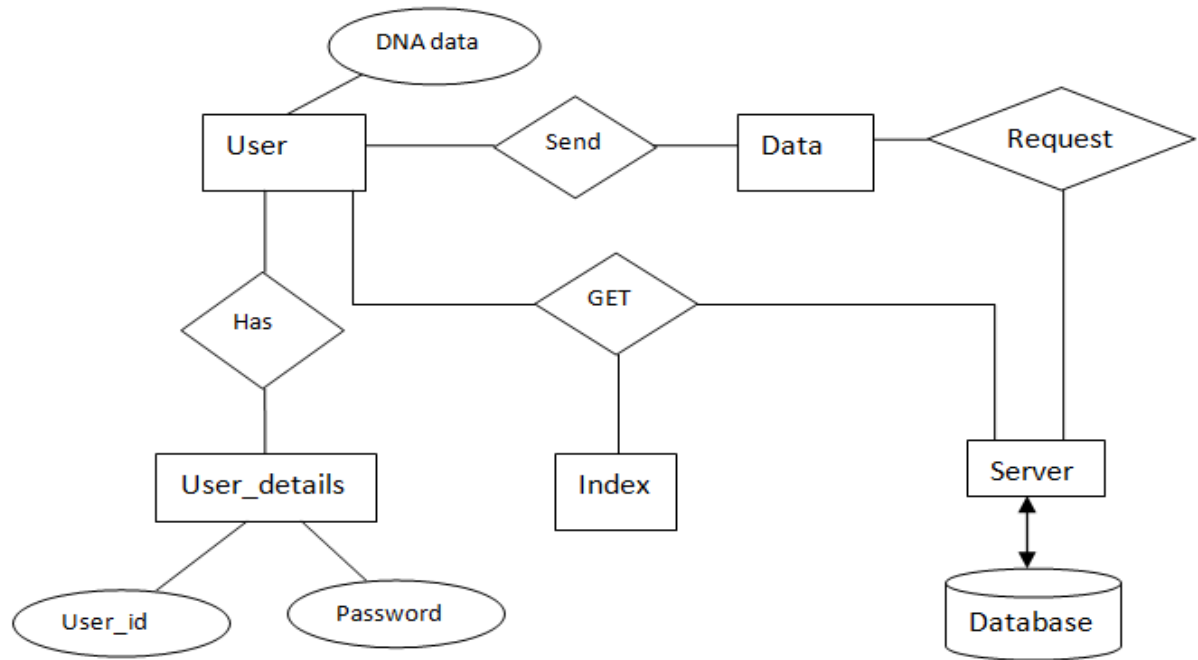


Figure 4.2: E-R Diagram for Proposed System.

4.3 Data Flow Diagram

- The DFD is also called as bubble chart. It is a simple graphical formalism it 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.
- 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 it interacts with the system and the information flows in the system.
- DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique it depicts information flow and the transformations are applied as data moves from input to output.
- The data flow diagram serves two purposes:

1. To provide an indication of how data are transform as the moves through the system.
2. To depict the function transforms the data flow.

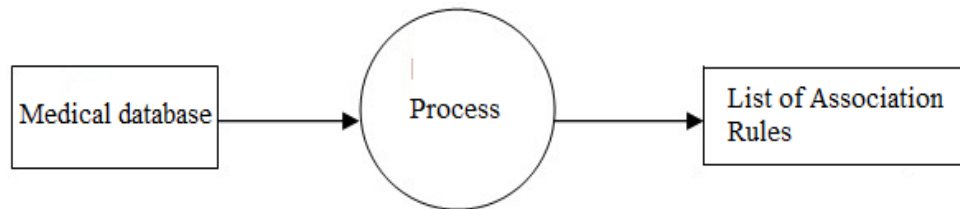


Figure 4.3: Data Flow Diagram Level 0

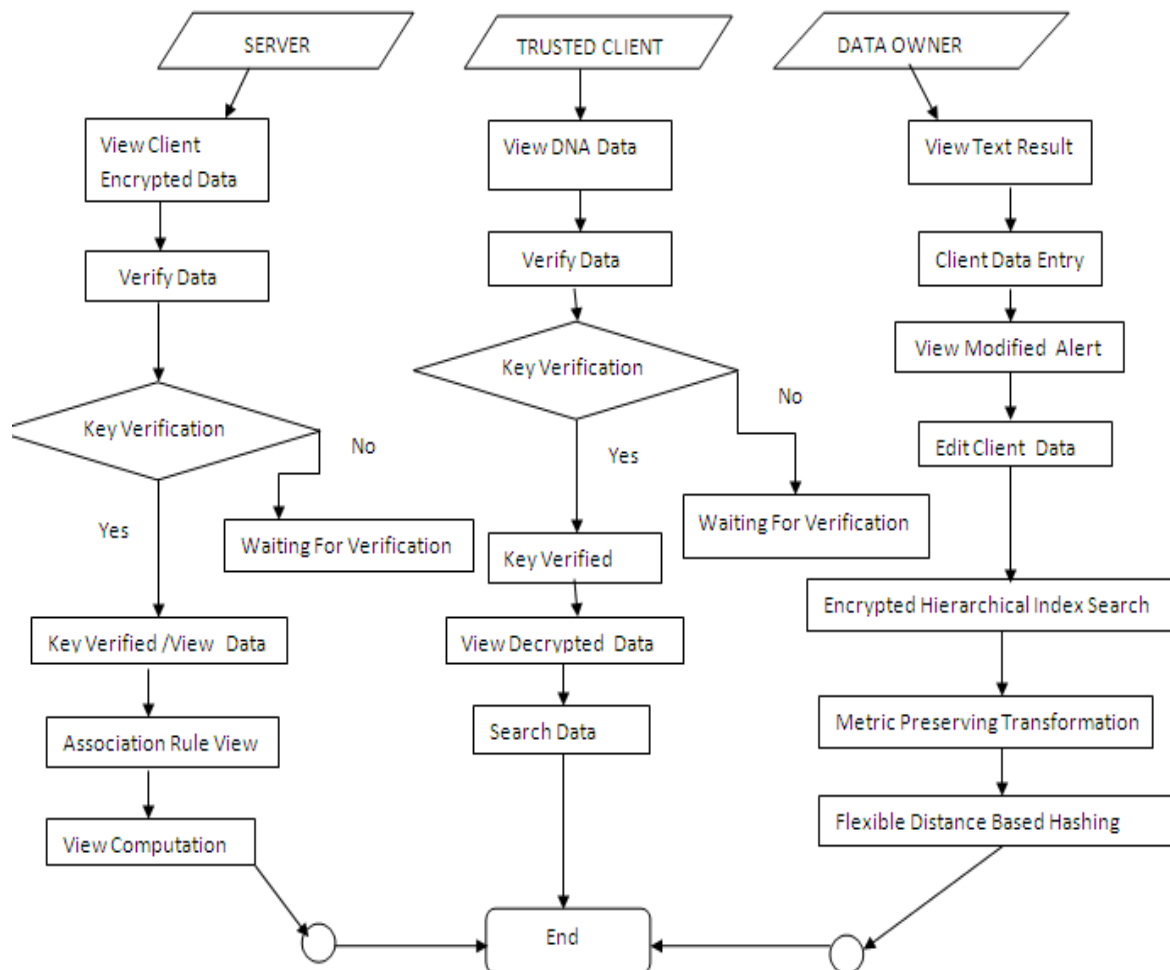


Figure 4.4: Data Flow Diagram Level 1

4.4 UML Diagrams

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. In the Unified Modeling Language contain two main types i.e. Structural Diagram and Behavior Diagram. In Structural Diagram, contains use case, collaboration, class, component, deployment, object, package, composite structure diagrams and Behavioral Diagram, in which sequence, communication, timing, state chart, activity diagrams.

4.4.1 Use Case Diagram

A use case diagram in the Unified Modeling Language 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 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. The Use Case Diagram is shown in given Figure 4.5.

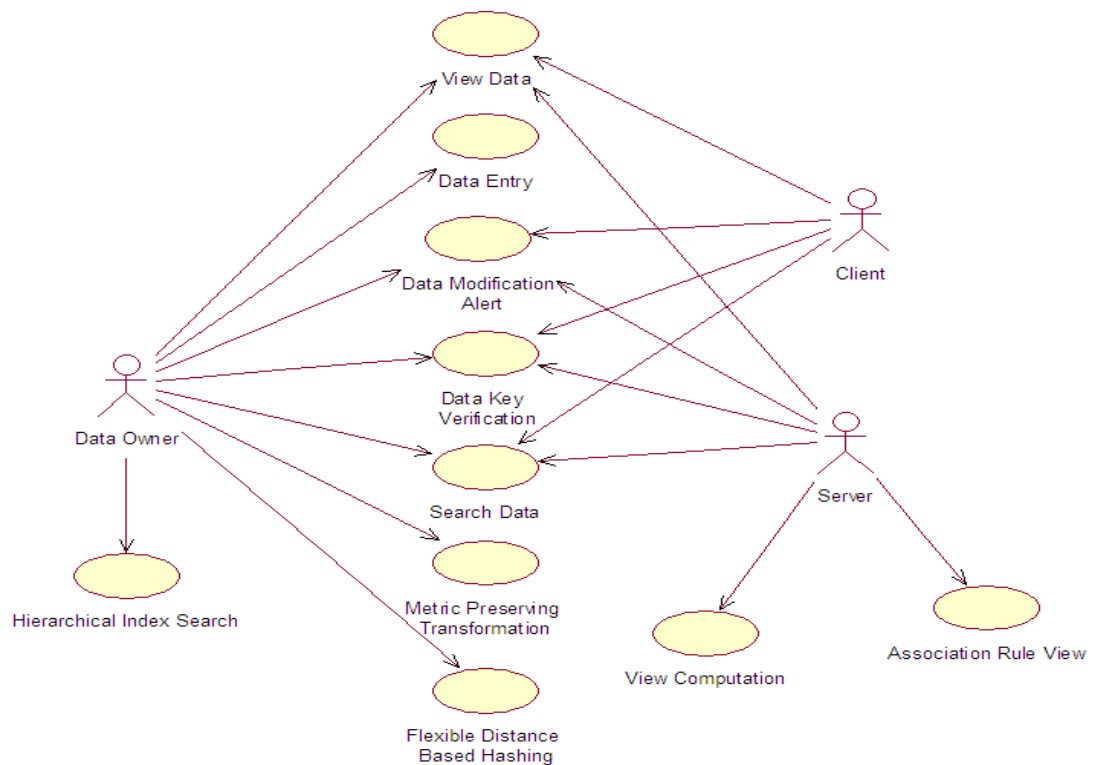


Figure 4.5: Use Case Diagram

4.4.2 Class Diagram

A Class Diagram in the Unified Modeling Language is a type of static structure diagram describes the structure of a system. Each box is divided into horizontal parts. The top part contains the name of the class. The middle section lists the attributes of the class and the third section of the class diagram contains the operations or behaviors of the class. The Class Diagram is shown in given Figure 4.6.

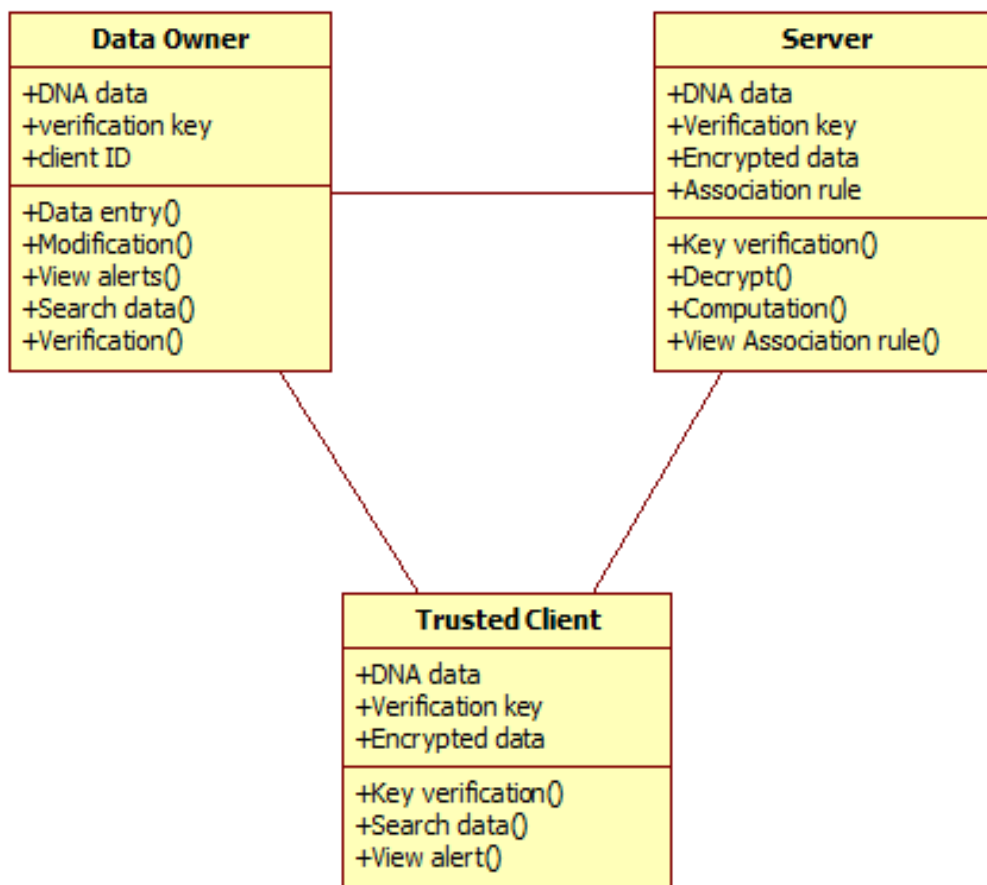


Figure 4.6: Class Diagram

4.4.3 Sequence Diagram

A Sequence Diagram in Unified Modeling Language is a kind of interaction diagram 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. The Sequence Diagram is shown in given Figure 4.7.

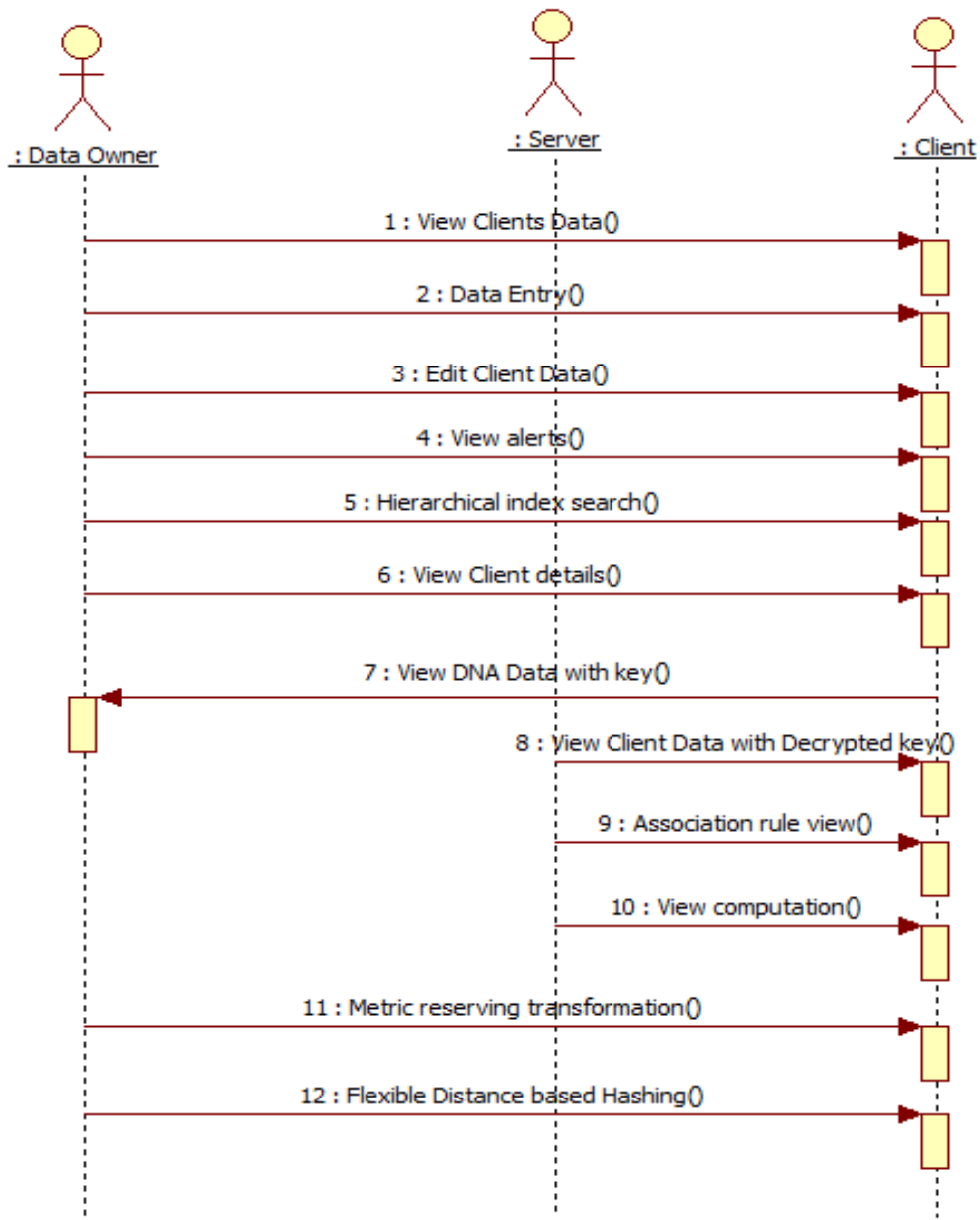


Figure 4.7: Sequence Diagram

4.4.4 Activity Diagram

A Activity Diagram depicts the dynamic behavior of a system or part of a system through the flow of control between actions of the system performs. It is similar to a flowchart except the activity diagram can show concurrent flows. The Activity Diagram is shown in given Figure 4.8.

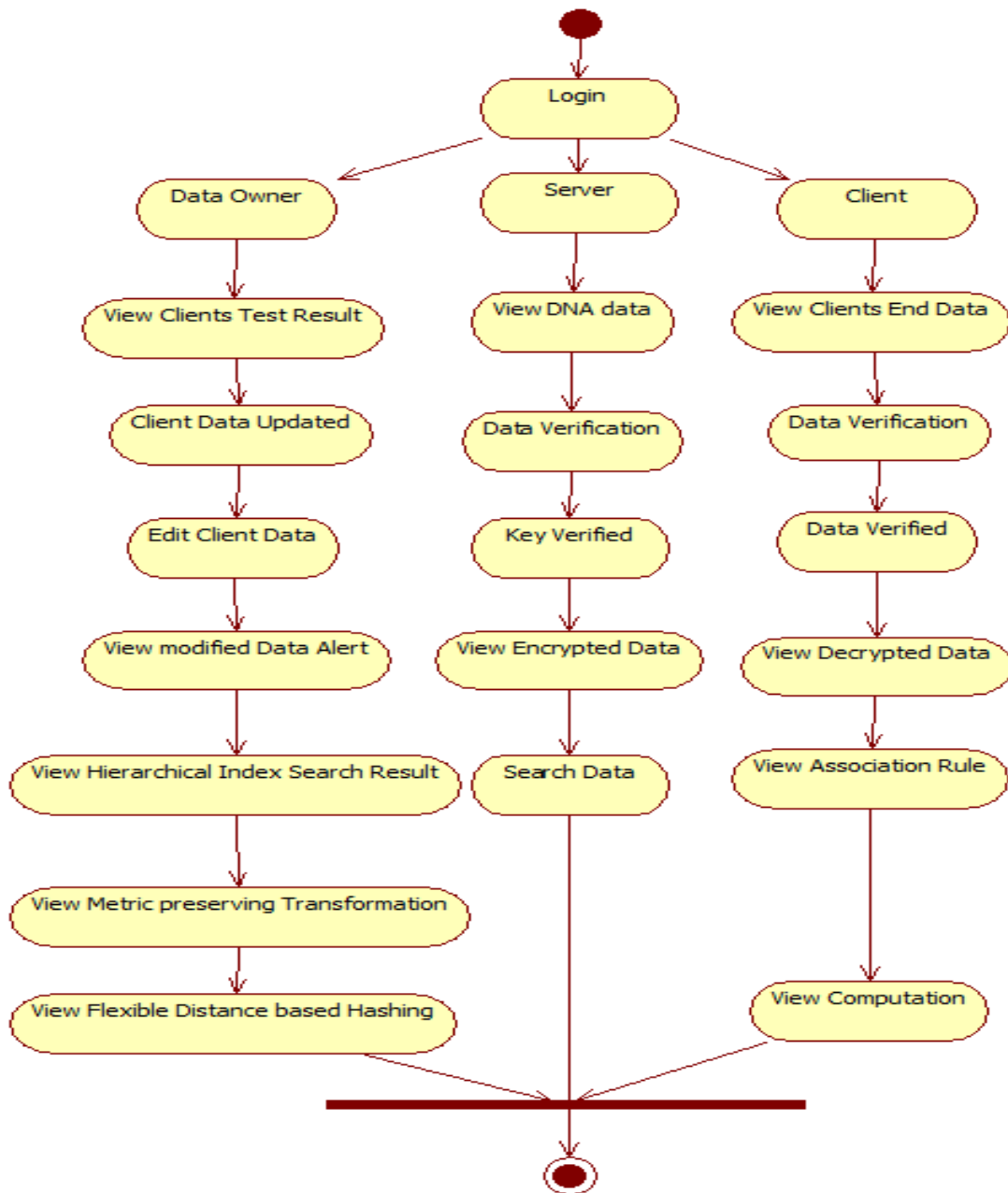


Figure 4.8: Activity Diagram

4.4.5 Component Diagram

A Component Diagram depicts how components are wired together to form larger components and or software systems. Component diagram is used to illustrate the structure of arbitrarily complex systems. The Component Diagram is shown in given Figure 4.9.

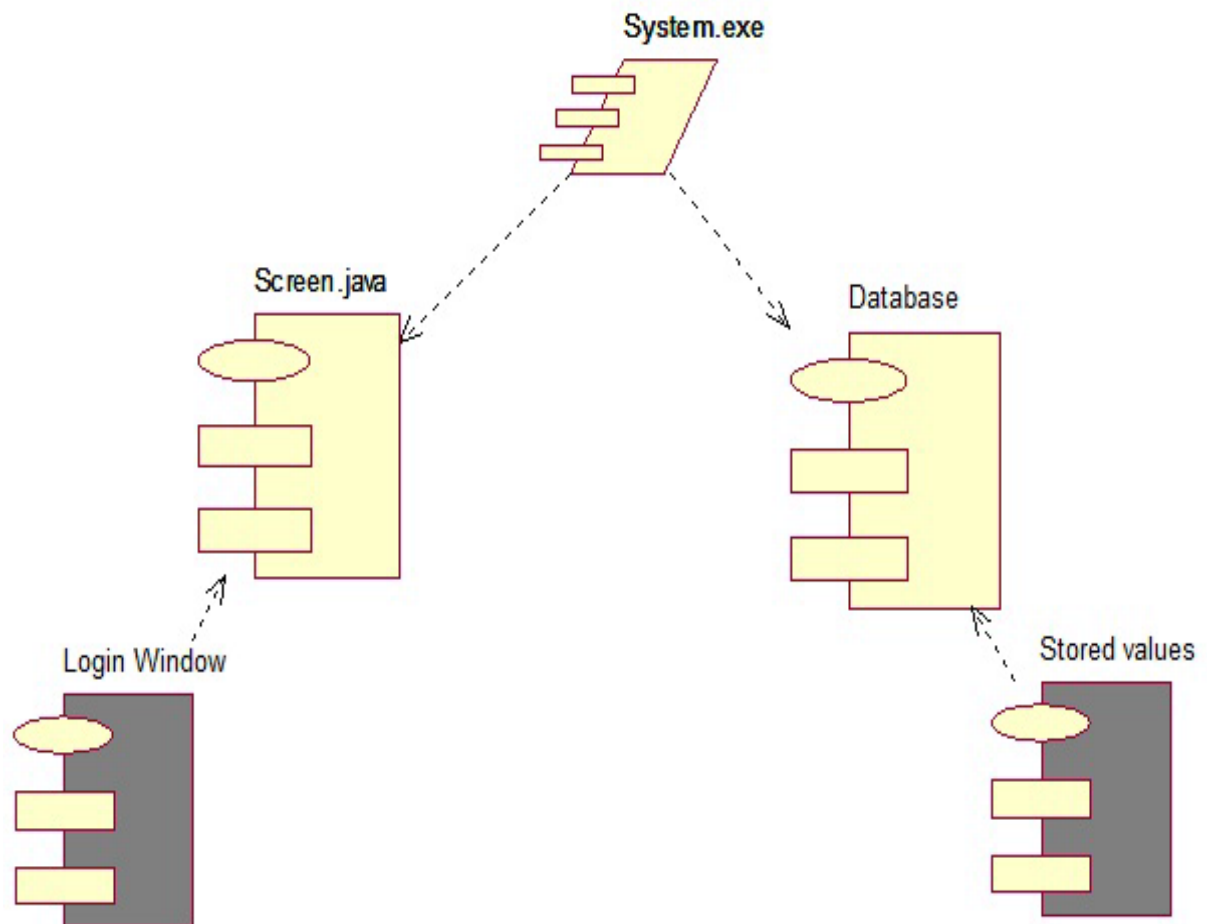


Figure 4.9: Component Diagram

4.4.6 Deployment Diagram

A Deployment Diagram focuses on the structure of a software system and is useful for showing the physical distribution of a software system among hardware platforms and execution environments. The Deployment Diagram is shown in given Figure 4.10.

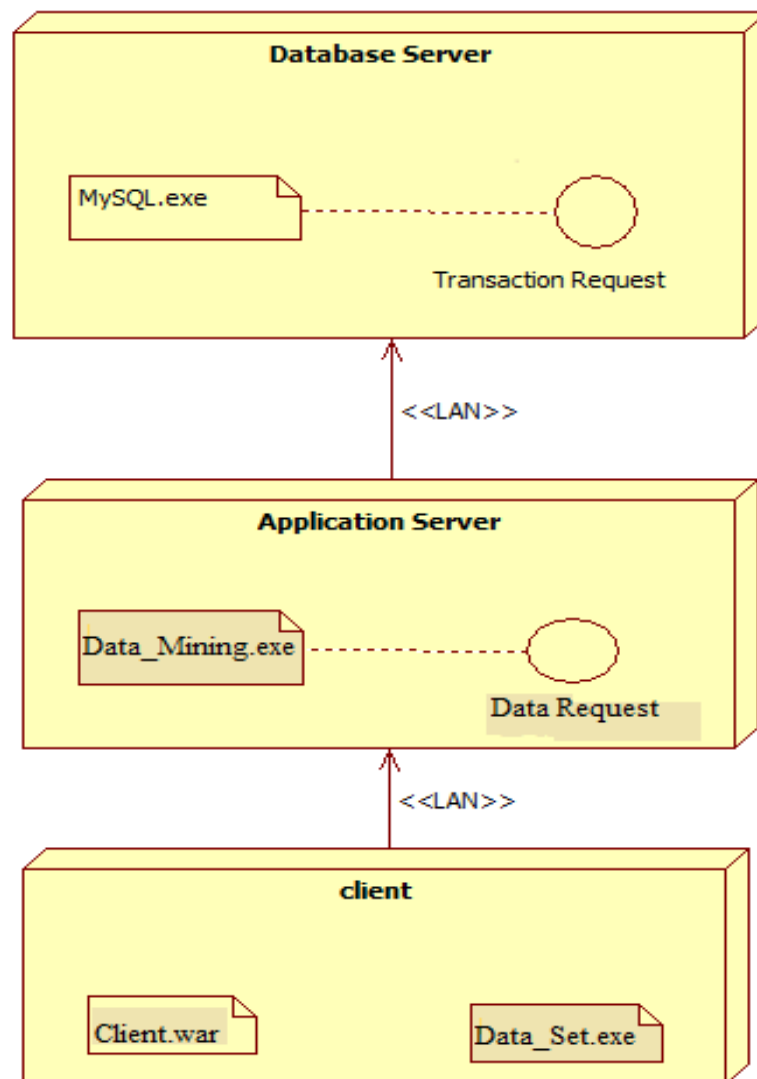


Figure 4.10: Deployment Diagram

4.4.7 State Diagram

A State Diagram models an objects states, the actions are performed depending on those states, and the transitions between the states of the object. The State Diagram is shown in given Figure 4.11.

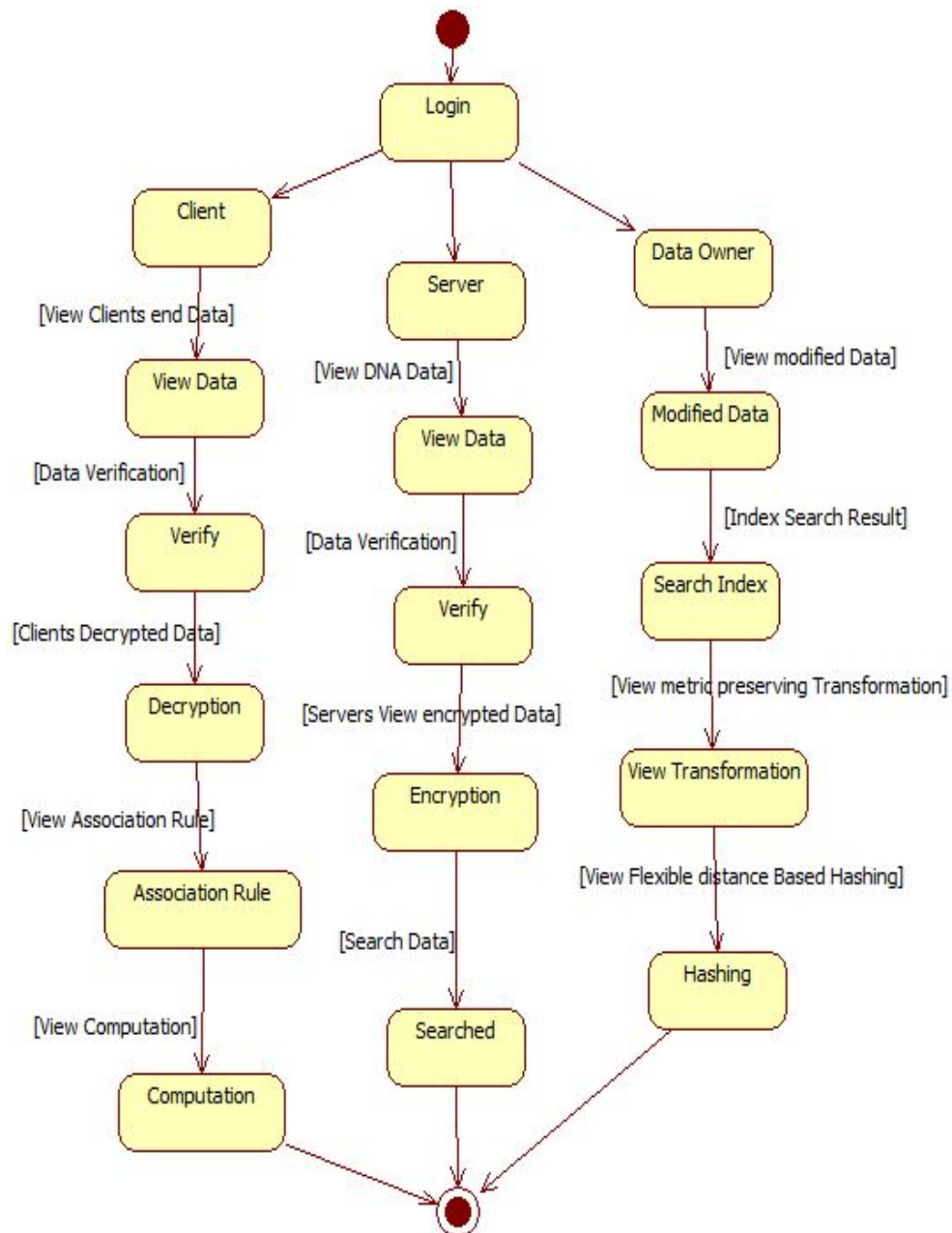


Figure 4.11: State Chart Diagram

4.4.8 Collaboration Diagram

The Collaboration Diagram also called a “communication diagram” provides another indication of the temporal order of the communications but emphasizes the relationships among the objects and classes instead of the temporal order. The Collaboration Diagram is shown in given Figure 4.12.

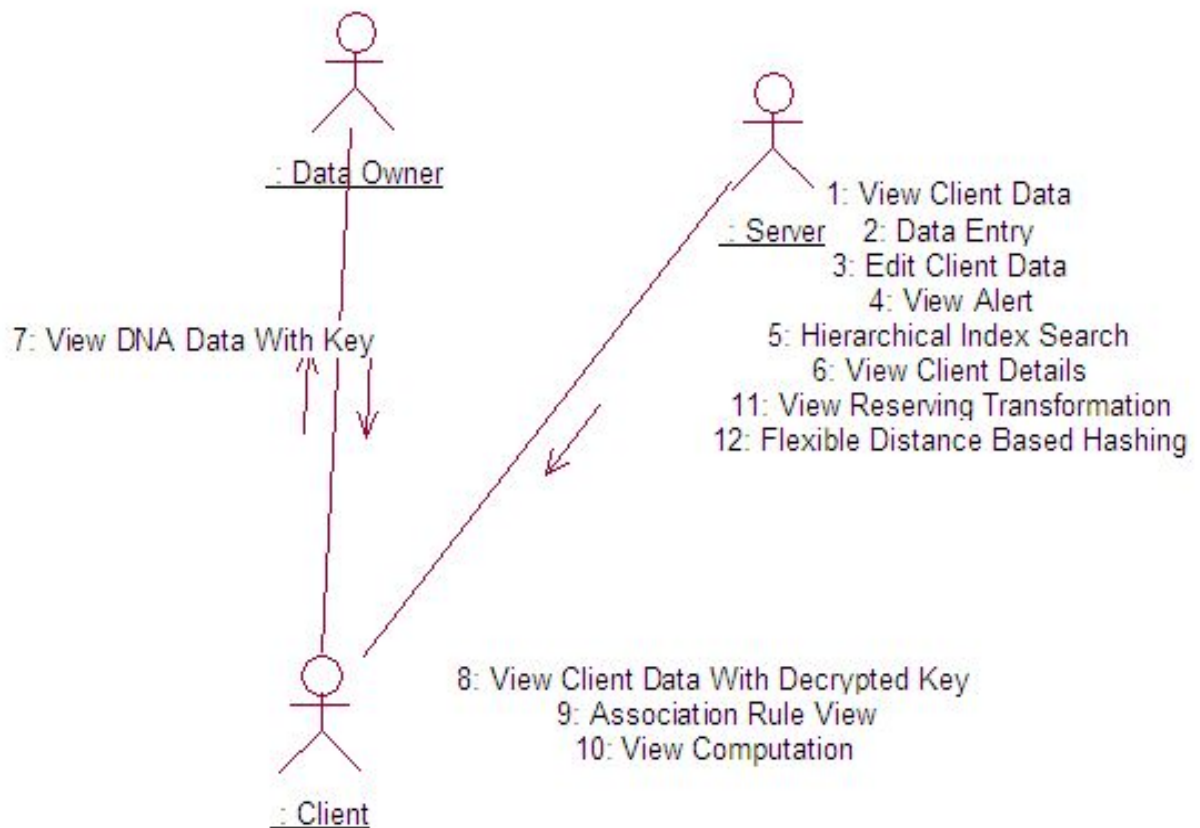


Figure 4.12: Collaboration Diagram

4.5 Summary

In this chapter, Section 4.1 is described the System Architecture, E-R Diagram is described in Section 4.2. In Section 4.3 are described Data Flow Diagrams, UML Diagrams are described in Section 4.4. In the last Section 4.5 Summary is presented.

Bibliography

- [1] Tamir Tassa, “Secure mining of association rule in horizontally distributed databases”, IEEE trans. Knowledge and Data Engg., Vol. 26, no.2, April 2014.
- [2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In VLDB, pages 487499.
- [3] M. Bellare, R. Canetti, and H. Krawczyk. Keying hash functions for message authentication. In Crypto, pages 115, 1996.
- [4] A. Ben-David, N. Nisan, and B. Pinkas. FairplayMP - A system for secure multi-party computation. In CCS, pages 257266, 2008.
- [5] J. Brickell and V. Shmatikov. Privacy-preserving graph algorithms in the semi-honest model. In ASIACRYPT, pages 236252, 2005.
- [6] J.C. Benaloh. Secret sharing homomorphisms: Keeping shares of a secret secret. In Crypto, pages 251260, 1986.